Fuzzy Identity-Based Key-Insulated Cryptosystem

Jin Li[1]* and Kwangjo Kim[1]

International Research Center for Information Security (IRIS)
Information and Communications University(ICU)
103-6 Munji-Dong, Yuseong-Gu, Daejeon, 305-732, Korea
{jjl,kkj}@icu.ac.kr

Abstract. Fuzzy identity-based encryption (FIBE) has found many applications, such as biometric-based encryption, since its notion was firstly proposed by Sahai and Waters [15]. In this paper, we show how to minimize the damage of secret key exposure in FIBE. We introduce a new notion which we call fuzzy identity-based key-insulated encryption (FIB-KIE). In FIB-KIE, the secret key associated with an identity is shared between the user and a tamper-proof device. The master key is stored on a tamper-proof device and a temporary secret key used to perform cryptographic operations is stored in an insecure device and updated regularly with the help of a tamper-proof device that stores a master key. We first present the definition and security model of FIB-KIE. Then, we suggest an FIB-KIE scheme, which is provably secure under the proposed security model.

Keywords: Fuzzy Identity-Based, Encryption, Key-Insulated, biometric-Based, Bilinear groups

1 Introduction

In order to simplify key management procedure of the certificate-based public key infrastructure, Shamir [17] introduced the concept of identity-based cryptosystem in 1984. Identity-based cryptosystem is a public key cryptosystem where the public key can be an arbitrary string such as an email address, *etc.* A private key generator uses a master secret key to issue private keys to users that request them. For an identity-based encryption (IBE), it allows a sender to encrypt a message to an identity without access to a public key certificate.

As a related notion to IBE, fuzzy identity-based encryption (FIBE) [15] was proposed by Sahai and Waters at Eurocrypt 2005. In FIBE, the identity is viewed as a set of descriptive attributes. The user with secret key for identity ω is able to decrypt a ciphertext encrypted with identity ω' if and only if ω and ω' are within a certain distance of each other as judged by some metric. FIBE has many

* This work was partially supported by the 2nd stage of Brain Korea 21 Project sponsored by the Ministry of Education and Human Resources Development, Korea

important applications. For example, FIBE can be applied to enable encryption using biometric inputs as identities: the error-tolerance property of an FIBE scheme is used for biometric identities that have some noise. Furthermore, FIBE can be used to an attribute based encryption. In this application a party will wish to encrypt a document to all users that have a certain set of attributes.

However, because of the risk of key exposure in the device where secret key is stored for cryptographic operations, there are a lot of risks such as key leakage while using the secret keys. So, how to protect the security of secret key and how to minimize the damage of key exposures have been paid great attention. In order to minimize the damage of key exposure, many security notions were proposed, such as threshold cryptosystem [8] and forward secure cryptosystem [2,6]. The notion of key-insulated public key cryptosystem was first introduced by Dodis *et al.* [9,10] to minimize the damage of key exposure. It uses a combination of key splitting and key evolution to protect against key exposure. In a certificate-based key-insulated public key cryptosystem, a user begins by registering a single public key pk which remains for the lifetime of the scheme. The secret key associated with a public key is here shared between the user and a tamper-proof device. The master key is stored on a tamper-proof device and a temporary secret key used to perform cryptographic operations is stored in an insecure device and updated regularly with the help of a tamper-proof device which stores a master key. For example, the lifetime of the protocol is divided into distinct periods $1, 2, \cdots, N$. At the beginning of each period, the user interacts with the tamper-proof device to derive a temporary secret key which will be used to perform cryptographic operation during that period. On the other hand, the public key does not change at each period. Later, Li *et al.* [13] proposed the notion of identity-based key-insulated cryptosystem to mitigate the damage caused by the private key exposures in identity-based cryptosystem. In identity-based key-insulated cryptosystem, the user's master key is stored on a tamper-proof device and a temporary secret key used to perform cryptographic operations is stored in an insecure device and updated regularly with the help of a tamper-proof device. And, the user's identity is kept unchanged at all time periods.

1.1 Our Contributions

We describe how to minimize the damage of secret key exposure in FIBE. We introduce the notion of FIB-KIE. In FIB-KIE scheme, the secret key associated with an identity is shared between the user and a tamper-proof device. The master key is stored on a tamper-proof device and a temporary secret key used to perform cryptographic operations is stored in an insecure device and updated regularly with the help of a tamper-proof device that stores a master key.

We first propose the notion and security model of FIB-KIE. Then, we present our construction of FIB-KIE. In fact, from a given FIB-KIE scheme, a new FIBE could be derived. To the best of our knowledge, this is the first contribution to address the key exposure problem in FIBE.

1.2 Related Work

After the notion of FIBE was proposed, many improvements [1,5,7,12] were proposed. In [1], they showed how to shorten the public parameters, however, it could only be proved to be secure in the random oracle model.

Later, there are many extensions for FIBE. Chase [7] proposed a multi-authority attribute-based encryption scheme. In this protocol, each authority controls some of the attributes. If one user wants to decrypt any ciphertext, he/she has to get enough attributes from every attribute authority. So, it is not so flexible in practice. The author suggested a solution to this problem. In the new protocol, the user could decrypt the ciphertext if it has sufficient attributes from some of the authorities. Goyal *et al.* [12] proposed another kind attribute-based encryption. In this system, each private key is associated with an access structure that specifies which type of ciphertexts the key can decrypt, and each ciphertext is labeled by the encryptor with a set of descriptive attribute. Such a scheme is called "key-policy attribute-based encryption". In this kind of encryption scheme, the access structure is specified in the private key, while the ciphertexts are simply labeled with a set of descriptive attributes.

Later, in [5], instead of determining the decrypting policy in private key, it allows the encryption device to specify an associated access structure. In this work, they provided a ciphertext-policy attribute-based encryption. When a party encrypts a message, they specify an associated access structure over attributes. A user will only be able to decrypt a ciphertext if that user's attributes pass through the ciphertext's access structure.

Threshold cryptography [8] is used to reduce the damage of secret key exposure. In these models, each entity, belonging to a predetermined subset of the entities sharing the corresponding secret key, computes a partial value with the help of its share. The secret key is shared in a distributed manner and the attacker should compromise more than a predetermined number of shareholders. However, distributed computation is required to generate a valid signature or to decrypt a ciphertext, which is undesirable in many circumstances. Bellare and Palacio [3] combined the advantages of key-insulated cryptosystem and threshold cryptography. They proposed another key-insulated encryption scheme, with an additional property of optimal threshold.

Forward-secure public key cryptosystem [2,6] is also considered to be a method to limit the damages arising when secret keys are exposed. A forward-secure public key cryptosystem prevents an adversary with a secret key for one time period from breaking the scheme for the previous time periods and standard cryptographic computation is performed by only a single device. The first identity-based forward secure encryption is proposed in [18], which is based on the hierarchical identity-based encryption scheme [4] and forward-secure public-key encryption scheme [6].

2 Preliminaries

We first introduce some preliminaries on bilinear maps.

Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be cyclic groups of prime order p with the multiplicative group action. And, g is a generator of $\mathbb{G}_1$. Let $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$ be a map with the following properties:

1. Bilinearity: $\hat{e}(g_1^a, g_2^b) = \hat{e}(g_1, g_2)^{ab}$ for all $g_1, g_2 \in \mathbb{G}_1$, and $a, b \in_R \mathbb{Z}_p$;
2. Non-degeneracy: There exists $g_1, g_2 \in \mathbb{G}_1$ such that $\hat{e}(g_1, g_2) \neq 1$, in other words, the map does not send all pairs in $\mathbb{G}_1 \times \mathbb{G}_1$ to the identity in $\mathbb{G}_2$;
3. Computability: There is an efficient algorithm to compute $\hat{e}(g_1, g_2)$ for all $g_1, g_2 \in \mathbb{G}_1$.

As shown in [4], such non-degenerate bilinear maps over cyclic groups can be obtained from the Weil or the Tate pairing over algebraic curves.

Definition 1. *(***DBDH problem***) The Decision Bilinear Diffie-Hellman (DBDH) problem is that, given g, g^a, g^b, $g^c \in \mathbb{G}_1$ for unknown random $a, b, c \in \mathbb{Z}_p^*$, $T \in \mathbb{G}_2$, to decide if $T = e(g, g)^{abc}$.*

We say that the (t, ϵ) DBDH assumption holds in $\mathbb{G}_1$ if no t-time algorithm has the probability at least $\frac{1}{2} + \epsilon$ in solving the DBDH problem for non-negligible ϵ.

3 Definitions and Security Model

3.1 Definition

Definition 2. [**FIB-KIE**] *An FIB-KIE consists of 7-tuple of poly-time algorithms (Setup, Extract, Gen, Upd*, Upd, Enc, Dec) defined as follows:*

- *Setup(N, d): The set up algorithm, that takes as input a security parameter 1^λ, returns public parameter params and a master key sk.*
- *Extract: The private key extraction algorithm, that takes as input identity ω and master key sk, returns the secret key s_ω for ω.*
- *Gen: The user key generation algorithm, that takes as input the private key s_ω and the total number of time periods N, outputs user's master private key s_ω^* and user's initial secret key s_ω^0.*
- *Upd*: The device key-update algorithm, that takes as input indices j, k for time periods $(1 \leq j, k \leq N)$ and the master private key s_ω^*, returns a partial secret key $s_\omega^{j \to k}$.*
- *Upd: The user key-update algorithm, that takes as input indices j, k, a secret key s_ω^j, and a partial secret key $s_\omega^{j \to k}$, returns the secret key s_ω^k for time period k.*
- *Enc: The encryption algorithm, which takes as input public parameter params, a time period k, identity ω', and a message M, returns a ciphertext (k, ω', C).*
- *Dec: The decryption algorithm, which takes as input a ciphertext (ω, k, C) and a secret key s_ω^k, it checks if $|\omega \cap \omega'| \geq d$. If yes, it returns a message M or the special symbol $\perp$.*

We define the following oracles:

- EO: The Extraction Oracle, on input ω, a master key sk, outputs the corresponding secret key s_ω by running algorithm $\mathsf{Extract}$.
- KEO: The Key Exposure Oracle, on input signer ω and k, returns and stores the value s_ω^k.
- DO: The Decryption Oracle, on input (ω, k, C), returns $\mathsf{Dec}_{s_\omega^k}(k, C)$.

We say that an $\mathsf{FIB\text{-}KIE}$ $\mathcal{E}$ is semantically secure against an selective identity and adaptive chosen ciphertext attack (IND-sFID-CCA) if no polynomially bounded adversary $\mathcal{A}$ has a non-negligible advantage against the challenger $\mathcal{C}$ in the following IND-sID-CCA game.

- *Initial*: First, the adversary $\mathcal{A}$ outputs its challenge identity ω^*.
- $\mathcal{C}$ runs Setup of the scheme. The resulting system parameters *params* are given to $\mathcal{F}$. $\mathcal{A}$ issues the following queries as he wants:

  Phase 1: $\mathcal{A}$ queries $\mathsf{EO}(\omega), \mathsf{KEO}(\omega, j)$ and $\mathsf{DO}(\omega, j, C)$ in an arbitrary way.

  Challenge: Once the adversary decides that Phase 1 is over, it outputs two equal length plaintexts M_0^*, M_1^*, period k and on which it wishes to be challenged with respect to the identity ω^*. The challenger picks a random bit $b \in \{0, 1\}$ and sets $C^* = \mathsf{Enc}(\omega^*, k, M_b^*)$. It sends (ω^*, k, C^*) as the challenge to the adversary.

  Phase 2: $\mathcal{A}$ queries more $\mathsf{EO}(\omega)$, $\mathsf{KEO}(\omega, j)$ and $\mathsf{DO}(\omega, j, C)$ in arbitrary interleave.

  Guess: Finally, $\mathcal{A}$ outputs a guess $b' \in \{0, 1\}$.

The adversary wins the game if $b' = b$ and, for any $\mid \omega' \cap \omega^* \mid \geq d$, ω', (ω', k), and (ω', k, C^*) have never been queried to $\mathsf{EO}, \mathsf{KEO}$ and DO, respectively.

We refer to such an adversary $\mathcal{A}$ as an IND-sFID-CCA adversary. We define adversary $\mathcal{A}$'s advantage in attacking the scheme $\mathcal{E}$ as the following function of the security parameter λ: $Adv_{\mathcal{E},\mathcal{A}}^{ind-sfid-cca}(\lambda) = |Pr[b = b'] - \frac{1}{2}|$.

In this paper, the proof of security for our $\mathsf{FIB\text{-}KIE}$ makes use of a weaker notion of security called selective identity semantic security (IND-sFID-CPA). Semantic security is similar to chosen ciphertext security except that the adversary is more limited: it cannot issue decryption queries while attacking the challenge identity.

Definition 3. *We say that an FIB-KIE $\mathcal{E}$ is IND-sFID-CPA if for any polynomial time IND-sFID-CPA adversary $\mathcal{A}$, the function $Adv_{\mathcal{E},\mathcal{A}}^{ind-sfid-cca}(\lambda)$ is negligible.*

Definition 4. *An FIB-KIE has secure key updates if the view of any adversary $\mathcal{F}$ making a key-update exposure at (j, k) can be perfectly simulated by an adversary $\mathcal{F}'$ making key exposure requests at periods j and k.*

Definition 5. *An FIB-KIE is called (t, N)-key-insulated if the scheme remains secure for the remaining $N - t$ time periods against any adversary $\mathcal{F}$ who compromises only the insecure device for t time periods.*

Similar to [9], we also define a strong security notion that is called strong identity-based key insulated encryption. It ensures the security for the temporary secret keys in case that the tamper-proof device is broken.

Definition 6. *An FIB-KIE scheme is called a strong identity-based key-insulated scheme if an adversary who compromises only the physically-secure device cannot break the scheme at any time periods.*

3.2 Our Scheme

We also define the Lagrange coefficient $\Delta_{i,S}$ for $i \in Z_p$ and a set, S, of elements in Z_p:

$$\Delta_{i,S} = \prod_{j \in S, j \neq i} \frac{x-j}{i-j}$$

In this scheme, we assume that there are n attributes. Identities will be element subsets of some universe, U, of size $|U|$. Associate each element with a unique integer in Z_p. The attributes set is denoted by ω. Our construction is as follows:

1. **Setup**(N, n, d): First, define the universe, U of elements. For simplicity, let n=$|U|$ and we can take the first n elements of $\mathbb{Z}_p$, to be the universe. Namely, the integers $1, 2, \cdots, n \pmod p$. To generate parameters for the system of time periods N, select a random generator $g \in \mathbb{G}_1$, a random $x \in \mathbb{Z}_p^*$, and set $g_1 = g^x$. Next, pick random elements $g_2, h_1, \cdots, h_n, T_1, \cdots, T_N \in \mathbb{G}_1$, compute $Z = e(g_1, g_2)$. The public parameters are $params = (g, g_1, g_2, h_1, \cdots, h_n, T_1, \cdots, T_N, Z)$, master key is x.
2. **Extract**: On input a private key x, to generate the private key $(D_i)_{i \in \omega}$ for an identity ω, the following steps are taken:
   - A $d-1$ degree polynomial q is randomly chosen such that $q(0) = x$.
   - For each $i \in \omega$, choose $r_i \in_R \mathbb{Z}_p$. Then, compute $d_{i0} = g_2^{q(i)} \cdot (g_1 h_i)^{r_i}$ and d_{i1}=g^{r_i}.
   - Finally, output the private key $D_i = (d_{i0}, d_{i1})$ for each $i \in \omega$.
3. **Gen**: On input $(D_i)_{i \in \omega} = (d_{i0}, d_{i1})$, for each $i \in \omega$, compute the following values:
   - Choose a random element $\eta_i \in \mathbb{G}_1$;
   - Compute $d'_{i0} = d_{i0}/\eta_i$ and $d'_{i1} = d_{i1}$. Let $a_{i00} = \eta_i$, $a_{i01} = \phi$, $a_{i02} = \phi$, $a_{i03} = \phi$;
   - Finally, output $D'_i = (d'_{i0}, d'_{i1})$ and $D_i^0 = (a_{i00}, a_{i01}, a_{i02}, a_{i03})$.

   In fact, $D'_i = (g_2^{q(i)}(g_1 h_i)^{r_i}/\eta_i, g^{r_i})$, which is stored in tamper-proof device. $D_i^0 = (\eta_i, \phi, \phi, \phi)$, which is the initial temporary secret key stored in insecure device for user to do cryptographic operations.

4. **Upd***: On input indices $j, k \in [0, N]$ and $(D'_i)_{i \in \omega} = (d'_{i0}, d'_{i1})$, proceed as follows:

- Choose $t_i \in_R \mathbb{Z}_p^*$
- Compute $b_{ijk,0} = d'_{i0} \cdot T_k^{t_i}$, and $b_{ijk,2} = g^{t_i}$;
- Let $b_{ijk,1} = d'_{i1}$;
- Finally, return the partial secret key $(D_i^{j\to k})_{i\in\omega} = (b_{ijk,0}, b_{ijk,1}, b_{ijk,2})$.

5. **Upd**: On input indices $j, k \in [0, N]$, secret key $(D_i^j)_{i\in\omega} = (a_{ij0}, a_{ij1}, a_{ij2}, a_{ij3})$ and a partial secret key $(D_i^{j\to k})_{i\in\omega} = (b_{ijk,0}, b_{ijk,1}, b_{ijk,2})$, do the following steps:
   - First, let $a_{ik0} = a_{ij0}$ (in fact $a_{ik0} = \eta_i$ for all k), $a_{ik2} = b_{ijk,1}$, $a_{ik3} = b_{ijk,2}$.
   - Compute $a_{ik1} = b_{ijk,0} \cdot a_{ij0}$;
   - Erase $(D_i^j, D_i^{j\to k})_{i\in\omega}$.
   - Finally, output the secret key $(D_i^k)_{i\in\omega} = (a_{ik0}, a_{ik1}, a_{ik2}, a_{ik3})$.

   In fact, $(D_i^k)_{i\in\omega} = (\eta_i, g_2^{q(i)}(g_1h_i)^{r_i}T_k^{t_i}, g^{r_i}, g^{t_i})$. It is temporary secret key for user to do cryptographic operations at time period k.

6. **Enc**: On input an index k of a time period, a message $M \in G_2$, and an identity ω', the ciphertext is computed as follows:
   - Pick a $s \in_R \mathbb{Z}_p^*$;
   - Compute $A{=}Z^sM$, $B = g^s$, $(C_i)_{i\in\omega'} = (g_1h_i)^s$, $D = T_k^s$.
   - Output the ciphertext $\mathcal{C}{=}(A, B, (C_i)_{i\in\omega'}, D)$.
7. **Dec**: Take as input private key $(D_i^k)_{i\in\omega} = (a_{ik0}, a_{ik1}, a_{ik2}, a_{ik3})$ for identity ω and ciphertext $\mathcal{C} = (A, B, (C_i)_{i\in\omega}, D)$ for an identity ω', at time period k. Choose a d-element subset S of $\omega \cap \omega'$ if $|\omega \cap \omega'| \geq d$.

   Finally, output the plaintext $M = A/\prod_{i\in S}(\frac{e(a_{ik1},B)}{e(a_{ik2},C_i)e(a_{ik3},D)})^{\triangle_{i,S}(0)}$.

3.3 Correctness and Performance

The correctness of decryption is justified by the following equations:

$$
\begin{aligned}
&A/\prod_{i\in S}(\tfrac{e(a_{ik1},B)}{e(a_{ik2},C_i)e(a_{ik3},D)})^{\triangle_{i,S}(0)}\\
&= MZ^s/\prod_{i\in S}(\tfrac{e(g_2^{q(i)}((g_1h_i)^{r_i}T_k^{t_i},g^s)}{e((g_1h_i)^s,g^{r_i})e(T_k^s,g^{t_i})})^{\triangle_{i,S}(0)}\\
&= MZ^s/\prod_{i\in S}(e(g_2,g)^{sq(i)})^{\triangle_{i,S}(0)}\\
&= M
\end{aligned}
$$

In this scheme, algorithm Enc requires $3+ \mid \omega' \mid$ exponentiations computations in group G_1. It does not need to compute the pairing because that $e(g_1, g_2)$ is public parameter. In decryption algorithm, $3d$ paring and $3d$ exponentiations computations in group G_2 are required.

3.4 Security Analysis

Theorem 1. *The FIB-KIE has secure key updates and supports random key updates.*

Proof. Let $\mathcal{F}$ be an adversary who makes a key-update exposure at (j, k). This adversary can be perfectly simulated by an adversary $\mathcal{F}'$ who makes key exposure requests at periods j and k. Since $\mathcal{F}'$ can get $(D_i^j)_{i\in\omega} = (a_{ij0}, a_{ij1}, a_{ij2}, a_{ij3})$ and $(D_i^k)_{i\in\omega} = (a_{ik0}, a_{ik1}, a_{ik2}, a_{ik3})$, it could compute $D_i^{j\to k} = (b_{ijk,0}, b_{ijk,1}, b_{ijk,2})$, where $b_{ijk,0} = a_{ik1}/a_{ij0}$, $b_{ijk,1} = a_{ik2}$, and $b_{ijk,2} = a_{ik3}$. The proof that the scheme supports random key updates is trivial because the values j, k could be selected randomly in the above proof. □

Theorem 2. *Suppose the (t', ϵ')-DBDH assumption holds in $\mathbb{G}_1$ and the adversary makes at most q_k and q_e times queries to private key extraction and key exposure, respectively, then this FIB-KIE scheme is (t, q_k, q_e, ϵ)-IND-sFID-CPA, where $t' < t+4(q_k+4q_e)t_{exp}$ and t_{exp} is the maximum time for an exponentiation in $\mathbb{G}_1$, $\epsilon' \approx \frac{1}{N} \cdot \epsilon$.*

Proof. See Appendix A. □

Theorem 3. *The FIB-KIE is a strong fuzzy identity-based $(N-1, N)$-key-insulated encryption scheme.*

Proof. Assume an adversary $\mathcal{F}$ succeeds to attack the FIB-KIE with access to the tamper-proof device, we will construct an algorithm $\mathcal{C}$ described below solves DBDH problem in $\mathbb{G}_1$ for a randomly given instance $\{g, X = g^x, Y = g^y, Z = g^z, T\}$ and asked to tell if $T = e(g, g)^{xyz}$.

$\mathcal{C}$ sets $g_1 = X$ as the public key and sends it to $\mathcal{F}$. In order to simulate private key extraction and key exposure queries for identity ω, $\mathcal{C}$ randomly selects elements $d'_{i0}, d'_{i1} \in \mathbb{G}_1$ and sends $(D'_i)_{i\in\omega} = (d'_{i0}, d'_{i1})$ to $\mathcal{F}$. From the viewpoint of $\mathcal{F}$, it is indistinguishable from the real transcripts. $\mathcal{C}$ will answer private key extraction and key exposure queries as the proof for Theorem 2. If $\mathcal{F}$ could break the scheme, from the simulation we can infer that $\mathcal{C}$ can solve the DBDH problem as the proof in Theorem 2. Meanwhile, in the proof of Theorem 2, the adversary can query key exposure oracle up to $N-1$ (*i.e.*, $q_e = N-1$) time periods for an identity ω. So, it is obvious that the FIB-KIE scheme satisfies $(N-1, N)$-key-insulated. □

In order to achieve IND-sFID-CCA security in the standard model, we can use the technique of simulation-sound NIZK proofs [14]. However, it is not efficient because of NIZK proofs.

In order to get more efficient construction, by using the technique suggested in [11], the scheme can be converted into a IND-sFID-CCA FIB-KIE scheme, only in the random oracle model.

3.5 FIB-KIE with Flexible *d*

Similar to [15], we have two methods to have flexible value d. First, we can create multiple systems with different values of d and one can encrypt message by choosing the appropriate one.

The second method is that the attribute authority will reserve some root attributes that it will issue to everyone. So, the party encrypting the message can decrease d by increasing the number of these 'default' attributes it includes in the encryption identity.

4 Conclusion

To minimize the damage of secret key exposure in FIBE, we introduced a new notion of FIB-KIE. In FIB-KIE, the secret key associated with an identity is shared between the user and a tamper-proof device. The master key is stored on a tamper-proof device and a temporary secret key used to perform cryptographic operations is stored in an insecure device. For each time period, the temporary secret key is updated regularly with the help of a tamper-proof device that stores a master key. We presented the definition and security model of FIB-KIE. Then, a provably secure FIB-KIE scheme was proposed. The scheme is secure in the remaining time periods against an adversary who compromises the insecure device and obtains secret keys for the periods of its choice. Furthermore, the scheme remains secure for all time periods against an adversary who compromises only the tamper-proof device.

References

1. J. Baek, W. Suslio, and J. Zhou. *New Constructions of Fuzzy Identity-Based Encryption*. ASIACCS'07, pp. 368-370, ACM, 2007.
2. M. Bellare and S.K. Miner. *A Forward-Secure Digital Signature Scheme*. CRYPTO'99, LNCS 1666, pp. 431-448, Springer, 1999.
3. M. Bellare and A. Palacio. *Protecting against Key Exposure: Strongly Key-Insulated Encryption with Optimal Threshold*. Applicable Algebra in Engineering, Communication and Computing, vol. 16(6), pp. 379-396, 2006.
4. D. Boneh and X. Boyen. *Efficient Selective-ID Secure Identity based Encryption without Random Oracles*. EUROCRYPT'04, LNCS 3027, pp. 223-238, Springer, 2004.
5. J. Bethencourt, C. Mellon, A. Sahai, and B. Waters. *Ciphertext-Policy Attribute-based Encryption*. IEEE Symposium on Security and Privacy 2007, pp. 321-334, 2007.
6. R. Canetti, S. Halevi, J. Katz. *A Forward-Secure Public-Key Encryption Scheme*. EUROCRYPT'03, LNCS 2656, pp. 255-271, Springer, 2003.
7. M. Chase. *Multi-Authority Attribute based Encryption*. TCC'07, LNCS 4392, pp. 515-534, Springer, 2007.
8. Y. Desmedt and Y. Frankel. *Threshold Cryptosystems*. CRYPTO'89, LNCS 435, pp. 307-315, Springer, 1989.
9. Y. Dodis, J. Katz, S. Xu, and M. Yung. *Key-Insulated Public-Key Cryptosystems*. EUROCRYPT'02, LNCS 2332, pp. 65-82, Springer, 2002.
10. Y. Dodis, J. Katz, S. Xu, and M. Yung. *Strong Key-Insulated Signature Schemes*. PKC'03, LNCS 2567, pp. 130-144, Springer, 2003.
11. E.Fujisaki and T.Okamoto. *Secure Integration of Asymmetric and Symmetric Encryption Schemes*. CRYPTO'99, LNCS 1666, pp. 537-554, Springer, 1999.

12. V. Goyal, O. Pandey, A. Sahai, and B. Waters. *Attribute-Based Encryption for Fine-Grained Access Control of Encrypted Data*. CCS'06, pp. 89-98, ACM, 2006.
13. J. Li, F. Zhang, and Y. Wang. *A Strong Identity Based Key-Insulated Cryptosystem*. EUC'06, LNCS 4097, pp. 352-361, Springer, 2006.
14. A. Sahai. *Non-Malleable Non-Interactive Zero Knowledge and Adaptive Chosen Ciphertext Security*. IEEE Symp. on Foundations of Computer Science, 1999.
15. A. Sahai and B. Waters. *Fuzzy Identity-Based Encryption*. EUROCRYPT'05, LNCS 3494, pp. 457-473, Springer, 2005.
16. A. Shamir. *How to Share a Secret*. Communications of the ACM, vol. 22, pp. 612-613, ACM, 1979.
17. A.Shamir. *Identity-Based Cryptosystems and Signature Schemes*. CRYPTO'84, LNCS 196, pp. 47-53, Springer, 1984.
18. D.Yao, N.Fazio, Y.Dodis, and A.Lysyanskaya. *ID Based Encryption for Complex Hierarchies with Applications to Forward Security and Broadcast Encryption*. CCS'04, pp. 354-363, ACM, 2004.

Appendix A: Proof of Theorem 2

Proof.

Suppose that an adversary $\mathcal{A}$ has an advantage ϵ in attacking the scheme, we build an algorithm $\mathcal{C}$ that uses $\mathcal{A}$ to solve the DBDH problem. Algorithm $\mathcal{C}$ is given a random $(g, X = g^x, Y = g^y, Z = g^z, T)$ and asked to tell if $T{=}e(g,g)^{xyz}$ or not.

First, $\mathcal{A}$ outputs the challenge identity ω^*.

Simulation of Setup $\mathcal{C}$ sets $g_1 = X$ and $g_2 = Y$. For all $i \in \omega^*$, it chooses random $\beta_i \in \mathbb{Z}_p$ and sets $h_i = g_1^{-1} g^{\beta_i}$. For all $i \notin \omega^*$, it chooses random $\beta_i \in \mathbb{Z}_p$ and sets $h_i = g^{\beta_i}$. It also chooses a random value $\eta \in [1, N]$. For $1 \leq i \leq N$, and $i \neq \eta$, let $T_i = g_1^{\tau_i}$, where $\tau_i \in \mathbb{Z}_p$. For $i = \eta$, choose $\tau_i \in \mathbb{Z}_p$ and let $h_i = g^{\tau_i}$. It gives $\mathcal{A}$ the public parameters $params{=}(g, g_1, g_2, h_1, \cdots, h_n, T_1, \cdots, T_n)$. Notice that from the view of $\mathcal{A}$, all parameters are chosen at random as in the construction.

Assume $\mathcal{A}$ makes at most q_k private key extraction queries and q_e key exposure queries. Denote the winning probability and running time of $\mathcal{C}$ by ϵ' and t', respectively. Algorithm $\mathcal{C}$ interacts with $\mathcal{A}$ as follows:

Simulation of EO Oracle $\mathcal{A}$ makes requests for private keys where the identity set overlap between the identities for each requested key and ω^* is less than d. Suppose $\mathcal{A}$ requests a private key ω where $\mid \omega \cap \omega^* \mid < d$. We first define three sets Γ, Γ', S in the following manner: $\Gamma{=}\omega \cap \omega^*$, and Γ' such that $\Gamma \subseteq \Gamma' \subseteq \omega'$ and $\mid \Gamma' \mid{=}d-1$. Let $S = \Gamma' \cup \{0\}$. Next, we define the private key components D_i:

For $i \in \Gamma'$: Choose $s_i, r_i \in \mathbb{Z}_p$ and let $q(i) = s_i$. Then output $D_i = (g_2^{s_i}(g_1 h_i)^{r_i}, g^{r_i})$.

We have chosen a random $d-1$ degree polynomial $q(x)$ by choosing its value for the $d-1$ points randomly in addition to having $q(0) = x$. $\mathcal{C}$ is able to calculate the simulated private key for $i \notin \Gamma'$ as $D_i = (g_2^{\sum_{j \in \Gamma'} \Delta_{j,S}(i) q(j)} g_2^{\beta_i \Delta_{0,S}(i)} (g_1 h_i)^{r'_i}, g_2^{\Delta_{0,S}(i)} g^{r'_i})$ to $\mathcal{A}$. It is easy to verify this is a valid: *i.e.*, it is required to show that

$D_i = (g_2^{q(i)}(g_1h_i)^{r_i},\ g^{r_i}) = (g_2^{\Sigma_{j\in\Gamma'}\Delta_{j,S}(i)q(j)} g_2^{\beta_i\Delta_{0,S}(i)}(g_1h_i)^{r'_i},\ g_2^{\Delta_{0,S}(i)}g^{r'_i})$. Using interpolation, for $i \notin \Gamma'$, $q(i) = \Sigma_{j\in\Gamma'}\Delta_{j,S}(i)q(j) + \Delta_{0,S}(i)q(0)$ and $q(x)$ was implicitly defined by the random assignment of the other $d-1$ variables and the variable g_1. Let $r_i = -\Delta_{0,S}(i)y + r'_i$ (In fact, $\mathcal{C}$ doesn't know the value of r_i), then $g_2^{q(i)}(g_1h_i)^{r_i} = g_2^{\Sigma_{j\in\Gamma'}\Delta_{j,S}(i)q(j)} g_2^{\beta_i\Delta_{0,S}(i)}(g_1h_i)^{r'_i}$ and $g^{r_i} = g_2^{\Delta_{0,S}(i)}g^{r'_i}$. Therefore, the simulator is able to construct a private key for the identity ω. Furthermore, the distribution of the private key for ω is identical to that of the original scheme.

Simulation of KEO Oracle Assume $\mathcal{A}$ issues key exposure queries (ω, k). When $|\ \omega \cap \omega^*\ | < d$, $\mathcal{C}$ could compute $(D_i)_{i\in\omega}$ as in the private key simulation and then returns the secret key for any time period normally.

When $|\ \omega \cap \omega^*\ | \geq d$ and $k \neq \eta$, $\mathcal{C}$ choose $r_i, t'_i \in \mathbb{Z}_p$ and outputs the simulated private key as $(g_2^{\Sigma_{j\in\Gamma'}\Delta_{j,S}(i)q(j)}(g_1h_i)^{r_i} g_1^{\tau_k t'_i}, g_2^{-\Delta_{0,S}(i)/\tau_k})$. The private key is correct because that just let $t_i = -\frac{\Delta_{0,S}(i)}{\tau_k}y + t'_i$.

Otherwise, when $|\ \omega \cap \omega^*\ | \geq d$ and $k = \eta$, $\mathcal{C}$ fails and exits.

Simulation of Challenge Ciphertext After these interactions, $\mathcal{F}$ outputs two messages M_0, M_1 and ω^* at time period η'. If $\eta' \neq \eta$, $\mathcal{C}$ fails and aborts. Otherwise, when $\eta' = \eta$, $\mathcal{C}$ picks a random bit $b \in \{0,1\}$ and responds with the ciphertext as $C = (TM_b,\ Z,\ (Z^{\beta_i})_{i\in\omega^*},\ Z^{\tau_\eta})$. The ciphertext is simulated correctly if $T = e(g,g)^{xyz}$ because let $s = z$, the ciphertext could be written as $C = (TM_b, Z, (Z^{\beta_i})_{i\in\omega^*}, Z^{\tau_\eta}) = (TM_b,\ g^s,\ ((g_1h_i)^s)_{i\in\omega^*},\ T_\eta^s)$.

$\mathcal{A}$ issues more private key queries ω and key exposure queries $(\omega,\ k)$, restriction is that $\omega \neq \omega^*$ and $k \neq \eta$. $\mathcal{C}$ responds as before.

This completes the description of algorithm $\mathcal{C}$. Finally, $\mathcal{A}$ outputs guess b' with advantage ϵ'. If $\mathcal{C}$ does not abort, then, $\mathcal{C}$ outputs b' as the result to the DBDH problem. For $\mathcal{A}$ has an advantage ϵ in attacking the scheme, from the simulation we can infer that $\mathcal{C}$ can solve the DBDH problem with advantage $\epsilon' \approx \frac{1}{N}\epsilon$, which is the success probability of the event that $\eta' = \eta$. □