ON DELEGATING PRIVATE KEY DERIVATION IN HIERARCHICAL IDENTITY

Khoo Terh Jing[1*], Radzi bin Ismail[2], Mohd Wira Mohd Shafiei[3]

[1] *Institute of Electrical Engineering, Yanshan University, Qinhuangdao 066004, China*

[2] *School of Transportation, Shijiazhuang Tiedao University, Shijiazhuang 050043, China*

[3] *Collaborative Innovation Center, Shijiazhuang Tiedao University, Shijiazhuang 050043, China*

Abstract

As of Hierarchical Identity Based Encryption (HIBE) systems, there are two important tasks should be accomplished properly, the first one is to establish logically hierarchical relationship between entities in the hierarchy tree, which is essentially accomplished through private key derivation by delegating responsibilities to lower-level PKGs, and the other task is to achieve encryption privacy of ciphertext targeting an intended recipient. In this paper we classify the mechanisms of private key derivation in HIBE systems, which explicitly define how and to what extent an entity in the hierarchy takes its level PKG's role of generating valid private keys for its descendants in the hierarchy. Moreover, a new delegation mechanism - Authorized Delegation is introduced, which can prevent any entity from deriving private keys for its descendants with use of its private key, and delegate the responsibility of generating private keys for a specified entity through authorization by distributing a specific secret to an entity as an ancestor of the specified entity by the root PKG (primitive authorization) or some other authorized entities (chained authorization). As for encryption privacy of ciphertext in a HIBE system, which measures the possibility that ciphertexts targeting an entity are successfully decrypted by its ancestors or descendants, we study encryption privacy from two distinct perspectives, i.e., private key derivation perspective and private key legitimacy perspective. Furthermore, Dominated Encryption Privacy and Dedicated Encryption Privacy are defined and discussed from private key legitimacy perspective.

Keywords: Identity-Based Encryption, HIBE, Private Key Derivation, Authorized Delegation, Encryption Privacy.

1. INTRODUCTION

An Identity Based Encryption (IBE) system is a public key system that an entity's public key can be any identifier of the entity (arbitrary string that is public, and can identify the entity), and private key for the entity can be calculated from its identifier with use of a master key by an authority, called private key generator (PKG). Since the introduction of the concept of Identity-based Encryption (IBE) by Shamir in 1984 [13], there are no usable IBE constructions until the works by Boneh and Franklin [6], Cocks [8], and Sakai et al. [12]. IBE schemes proposed by Boneh and Franklin [6][5] are based on bilinear pairings on elliptic curves [14], and security of these schemes can be reduced to the computational intractability of Bilinear Diffie-Hellman (BDH). These systems [6][8] utilize cryptographic hash functions that are modeled as random oracles when proving security of the schemes.

The concept of hierarchical identity based encryption (HIBE) was first introduced by Horwitz and Lynn [11]. Gentry and Silverberg [10] then presented the first HIBE construction, of which the security is based on the Bilinear Diffie-Hellman (BDH) assumption [5] in the random oracle model. Canetti et al. [7], Boneh et al. [1], Gentry [9] and Waters [15][16] constructed their schemes under the Decisional Bilinear Diffie-Hellman assumption (or variants of BDH assumption) and proved the security of their systems in standard model. Boneh and Boyen [1][3] introduced one selective identity, chosen-plaintext (IND-sID-CPA) secure HIBE system BB_1 without using random oracles under Decisional Bilinear Diffie-Hellman (DBDH) assumption.

Later, works by Boneh et al. [2][4], Gentry [9] and Waters [15][16] provided some fully secure schemes without random oracles. In [4], Boneh, Boyen and Goh presented a HIBE scheme that is selective identity secure in standard model and fully secure in the random oracle model; notably, the scheme can achieve limited delegation and short ciphertexts regardless of the hierarchy depth.

The remainder of this article is organized as follows. In the next section, we describe the motivations on studying mechanisms of private key derivation, particularly aiming to preventing the key escrow problem from being exaggerated. We classify mechanisms of private key derivation in HIBE systems with reference to some criteria in Section 3, and present definitions for all of them. Moreover, a new mechanism – authorized delegation is introduced, where only authorized entities are capable of generating private keys, and only those specified entities' private keys can be generated. In Section 4, we discuss the encryption privacy from private key legitimacy perspective instead of private key derivation perspective, and compare two types of encryption privacy, i.e. dominated encryption privacy and dedicated encryption privacy. Finally, we conclude in Section 5.

2. MOTIVATIONS

Since the first practical Identity Based Encryption (IBE) construction based on intractability of Computational Bilinear Diffie-Hellman (Calculation-BDH) problem by Boneh and Franklin, a wide variety of IBE and Hierarchical IBE (HIBE) systems from pairings have been proposed, and find their bright future in potential application scenarios, such as secure e-mail, domain-based security, Vehicular Ad Hoc Networks and so on. However, it is taken for granted that in a HIBE system entities in the hierarchy have the power of generating valid private keys for their descendants; specifically, an entity can not only directly use its own private key and some public parameters to derive valid private keys along the hierarchy tree (usually derivation is accomplished by randomizing an ancestor's private key), but also unrestrictedly generate valid private keys for all descendants of the entity if the entity wants to.

Although key escrow problem is inherent in IBE public cryptography, resulting from the mechanism of generating private keys for entities in the system, but we do not let the problem be exaggerated in HIBE systems where lower level PKGs are unrestrictedly delegated to be capable of generating private keys for their descendants.Particularly, it is irrational and undesirable that a lower level PKG (as an ancestor) can generate valid private keys for a descendant with direct use of its private key, and a lower level PKG can generate private keys for all of its descendants.

3. DELEGATING PRIVATE KEY GENERATION

Different from the One-PKG characterized IBE, Hierarchical Identity Based Encryption (HIBE) accommodates level-oriented PKG configuration. Namely, the top level PKG is the root PKG (at level zero), who maintains a hierarchy tree of which non-leaf nodes are viewed as level PKGs. HIBE allows the root PKG to balance the workload by delegating identity authentication, private key generation and private key distribution to lower level PKGs. Usually, delegation of private key generation for entities in the hierarchy is main job of responsibility delegation (from root PKG to lower level PKGs). The mechanisms of delegating private key generation can be classified into three classes, i.e. unlimited delegation, limited delegation and authorized delegation, with reference to criteria listed below. For ease of representation, it is assumed that $Entity_i$ with identity $ID_i = (I_1, \ldots, I_i)$ is an ancestor of $Entity_j$ with identity $ID_j = (I_1, \ldots, I_j)$, that is ID_i is a prefix of ID_j such that $ID_i[k] = ID_j[k]$ for all $k \in \{1, \ldots, i\}$.

(1) Only an entity's private key or some specially crafted content other than private key should be utilize to generate private keys for the entity's descendants, besides some public parameters, such as system parameters, identities (public keys) of the descendants whose private keys are derived, and so on.

(2) whether the private key generation can be hierarchically derived along the identity hierarchy tree; specifically, whether private key for $Entity_{j+1}$ can be derived given a private key for $Entity_j$ is generated.

(3) whether $Entity_i$ should first generate private keys for all entities which are descendants of $Entity_i$ and ancestors of $Entity_j$ prior to deriving a private key for $Entity_j$.

3.1. Unlimited Delegation

Definition 3.1 "Unlimited Delegation" means that an entity in the hierarchy can directly and unrestrictedly derive private keys for its descendants. Directly here means an entity can derive private keys for its descendants with only use of its private key, or the private key for the entity is the only needed secret for generating the descendants' private keys. Unrestrictedly means that the private key derivation can be accomplished hierarchically by an entity for all of its descendants.

As of private key derivation in [1][3], an $Entity_j$'s private key, denoted d_{ID_j}, can be hierarchically randomized to generate private keys for all of its descendants level by level. We exemplifies the private key derivation of $d_{ID_{j+1}}$ by $Entity_j$ with use of its private key d_{ID_j}, public system parameters, Identity of its child $Entity_{j+1}$ and random values.

Both private keys for $Entity_j$ and $Entity_{j+1}$ with identities $ID_j = (I_1, \ldots, I_j) \in \left(\mathrm{Z}_q^*\right)^j$ of depth $j \le \ell - 1$ and $ID_{j+1} = (ID_j, I_{j+1}) \in \left(\mathrm{Z}_q^*\right)^{j+1}$, denoted $d_{ID_j} \in \hat{\mathrm{G}}^{j+1}$ and $d_{ID_{j+1}} \in \hat{\mathrm{G}}^{j+2}$ respectively, can be extracted as

$$d_{ID_j} = \left(\hat{g}_0 + \sum_{k=1}^{j} r_k^{(ID_j)} (I_k \hat{g}_1 + \hat{h}_k), r_1^{(ID_j)} \hat{g}, \ldots, r_j^{(ID_j)} \hat{g} \right),$$

$$d_{ID_{j+1}} = \left(\hat{g}_0 + \sum_{k=1}^{j+1} r_k^{(ID_{j+1})} (I_k \hat{g}_1 + \hat{h}_k), r_1^{(ID_{j+1})} \hat{g}, \ldots, r_{j+1}^{(ID_{j+1})} \hat{g} \right).$$

Let Δd_0 be the result of $d_{ID_{j+1}}[0] - d_{ID_j}[0]$, Δd_0 is calculated as,

$$\Delta d_0 = \sum_{k=1}^{j} (r_k^{(ID_{j+1})} - r_k^{(ID_j)})(I_k \hat{g}_1 + \hat{h}_k) + r_{j+1}^{ID_{j+1}} (I_{j+1} \hat{g}_1 + h_{j+1}).$$

Correspondingly, other components can as well be calculated as $(r_1^{(ID_{j+1})} - r_1^{(ID_j)})\hat{g}$, …, $(r_j^{(ID_{j+1})} - r_j^{(ID_j)})\hat{g}$ and $(r_{j+1}^{(ID_{j+1})} - 0)\hat{g}$. Let $\left(d_0^{(ID_j)}, RD_1^{(ID_j)}, \ldots, RD_j^{(ID_j)} \right)$ denote the private key for $Entity_j$ (i.e. d_{ID_j}), and r_1, $\cdots$, r_{j+1} be $j+1$ random numbers from Z_q, the private key for $Entity_{j+1}$ can be derived as (other than extracting by $\mathrm{Extract}(mk, ID_{j+1})$),

$$d_{ID_{j+1}} = (d_0^{(ID_j)} + \sum_{k=1}^{j+1} r_k (I_k \hat{g}_1 + \hat{h}_k),$$
$$RD_1^{(ID_j)} + r_1 \hat{g}, \ldots, RD_j^{(ID_j)} + r_j \hat{g}, r_{j+1} \hat{g}).$$

By repeating the derivation process above, the private key of $Entity_{j+1}$ can be derived by any of its ancestors along the hierarchy. Consequently, not only ciphertexts intended for an entity can be decrypted by any of its ancestors, but also the ancestor, being with knowledge of a private key of the descendant, can do anything that the entity can do.

3.2 Limited Delegation

Definition 3.2 "Limited Delegation" means that an entity at depth k with public identity ID_k, denoted $Entity_k$, is given a restricted private key with t instead of $\ell - k$ ($t < \ell - k$, where ℓ is the maximum hierarchy depth) extra secrets that only authorizes the ancestor (i.e. $Entity_k$) to be able to derive private keys for its descendants of limited depth, beginning from $Entity_k$'s child to its descendant at depth $k+t$ (the deepest depth). If all $\ell - k$ extra secrets are provided, then the entity at depth k can generate private keys for all of its descendant. Particularly, the HIBE system fails to only generate valid private keys for a descendant at a specified depth ξ for $k+2 \le \xi \le \ell$ without deriving private keys for descendants at depth $k+1, \ldots, \xi$.

The HIBE system presented in [4] by Boneh, Boyen and Goh considers "limited delegation" and related encryption privacy of preventing an ancestor from successfully decrypting ciphertexts targeting its descendants. Private keys for $Entity_j$ and $Entity_{j+1}$ with identities $ID_j = (I_1, \ldots, I_j) \in (\mathbb{Z}_q^*)^j$ and $ID_{j+1} = (ID_j, I_{j+1})$ respectively are extracted by the root PKG as

$$d_{ID_j} = (\alpha g_2 + r^{(ID_j)}(g_3 + \sum_{k=1}^{j} I_k h_k),$$
$$r^{(ID_j)} g, \quad r^{(ID_j)} h_{j+1}, \quad \ldots, \quad r^{(ID_j)} h_\ell)$$
$$d_{ID_{j+1}} = (\alpha g_2 + r^{(ID_{j+1})}(g_3 + \sum_{k=1}^{j+1} I_k h_k),$$
$$r^{(ID_{j+1})} g, \quad r^{(ID_{j+1})} h_{j+2}, \quad \ldots, \quad r^{(ID_{j+1})} h_\ell)$$

where $r^{(ID_j)}$ and $r^{(ID_{j+1})}$ are two random numbers picked from $\mathbb{Z}_q$ by the root PKG. For clarity of representation, d_{ID_j} is denoted as ($d_0^{(ID_j)}$, $d_1^{(ID_j)}$, $RH_{j+1}^{(ID_j)}$, …, $RH_\ell^{(ID_j)}$). By picking a random number r from $\mathbb{Z}_q$, a private key for $Entity_{j+1}$ is derived with use of d_{ID_j} as

$$d_{ID_{j+1}} = (d_0^{(ID_j)} + r(g_3 + \sum_{k=1}^{j+1} I_k h_k) + I_{j+1} RH_{j+1}^{(ID_j)},$$
$$d_1^{(ID_j)} + rg, RH_{j+2}^{(ID_j)} + rh_{j+2}, \ldots, RH_\ell^{(ID_j)} + rh_\ell).$$

By repeating the process above, private keys for all descendants of $Entity_j$ can be derived with use of the $Entity_j$'s private key and needed historical content. Different from private key derivation in BB$_1$ system, where private keys for a child can be derived by only randomizing its parent's private keys, this HIBE system however does need some extra historical content in deriving a private key for a child, in addition to randomizing the parent's private key. For example, in deriving a private key for $Entity_{j+1}$ with use of $Entity_j$'s private key, $RH_{j+1}^{(ID_j)}$ ($= r^{(ID_j)} h_{j+1}$) as an important historical argument, is needed for calculating $I_{j+1} RH_{j+1}^{(ID_j)}$ as one important share of the resulted private key for $Entity_{j+1}$, as well as randomizing $Entity_j$'s private key to get the other share of the private key for $Entity_{j+1}$.

Then, if the root PKG does not provide $\ell - j$ components $RH_{j+1}^{(ID_j)}$, …, $RH_\ell^{(ID_j)}$ when distributing a private key for $Entity_j$, where $(d_0^{(ID_j)}, d_1^{(ID_j)})$ is still a valid private key for $Entity_j$ from perspective of cryptographic operation without considering private key derivation, there is no means of generating valid private keys for any descendant of $Entity_j$ with using $Entity_j$'s private key $(d_0^{(ID_j)}, d_1^{(ID_j)})$, because of lack of the needed historical argument $RH_{j+1}^{(ID_j)}$.

Actually, only the component $RH_{j+1}^{(ID_j)}$ instead of all those $\ell - j$ components is not provided, it is impossible to derive a private key for $Entity_{j+1}$ with use of $Entity_j$ ' private key, and thus disabling the hierarchical private key derivation along the hierarchy tree. That is, by providing a restricted private key with only t components $r^{(ID_j)} h_{j+k}$ for $k = 1, \ldots, t$ to $Entity_j$, $Entity_j$ can be capable of only generating private keys for its descendants of bounded depth t, i.e. from descendant at depth $j+1$ to descendant at depth $j+t$ along hierarchy tree.

3.3 Authorized Delegation

Limited delegation does prevent private keys for those descendants at depth beyond the limited depth from being derived. Nevertheless, there is no means to only derive a private key for $Entity_{j+t}$ with use of $Entity_j$'s private key without revealing private keys for those entities which are at level between $j+1$ and $j+t-1$. This undesirable breach in privacy is resulted from the need of use of a parent's private key when deriving a child's private key.

Definition 3.3 "Authorized Delegation" means that private keys for an entity cannot be derived directly from its ancestors' private keys. However, by distributing a secret specifically crafted for an entity to its ancestor by the root PKG, the ancestor is thus authorized to generate private keys for the specific descendant, while failing to generate private keys for any entity other than the specified descendant.

In the sequel, we detail how a HIBE system with authorized delegation mechanism of generating private keys for entities (always descendants of the generator) can be constructed. Particularly, what should be accomplished is as follows,

(1) Planning private key composition, such as including share of randomizing the master secret and some components that make direct and unrestricted delegation of generating private keys for descendants impossible. Contrasted to private key derivation in [10][1][3], where both S_j and d_{ID_j} can be hierarchically randomized to generate private keys for $ID_{j+1}=(ID_j, I_{j+1})$, ID_{j+2}, …, and so on.

(2) Providing means for an entity, such as through authorization, to be capable of generating private keys for a specified descendant entity. Authorization here can be achieved through secret distribution by equipping the generator with specially crafted content for deriving private keys for the specified entity, somewhat analogous to HIBE system presented in [4] where by equipping $Entity_j$ with needed secrets $r^{(ID_j)}h_{j+1}$, … , $r^{(ID_j)}h_{j+t}$ for deriving a private key for $Entity_j$'s descendant at depth $j+t$.

The main idea of our construction is to differentiate between identifiers I_1, …, I_j of the identity ID_j when extracting a private key for $Entity_j$. Specifically, in addition to randomizing the master secret of the root PKG with each identifier of $\{I_1,\ldots,I_j\}$ independently and uniformly for extracting a private key for $Entity_j$, which is the only mechanism of private key extraction presented in [3] (as exemplified in Section 3.1), an extra share ζ resulted from the combination of $Entity_j$'s local identifier I_j and those random values picked correspondingly for identifiers I_1, …, I_{j-1}, is introduced into the $Entity_j$'s private key. The extra share ζ anchors the generated private key only to the identity $ID_j=(I_1,\ldots,I_j)$. Anchor here means that the private key generated for $Entity_j$ can neither be used to derive private keys for its descendants, nor to decrypt ciphertexts intended for its ancestors or descendants. Namely, it is infeasible for an entity to derive private keys for its descendants with its private key, because there is no means for the $Entity_j$ to wipe off the share defined on its local identifier I_j from its private key and to introduce needed share related to the local identifier of each of its descendants for private key derivation in order to generate the corresponding descendant's private key.

Moreover, our HIBE system does provide a mechanism for authorized private key derivation, i.e. deriving a valid private key for and only for a specified descendant. Assume that $Entity_i$ as an ancestor of $Entity_j$ is authorized to derive private keys for $Entity_j$ (the specified entity). $Entity_j$ can be authorized by distributing to it two copies of information, the first information is the result of randomizing the master secret along the identity hierarchy $I_1 \to \ldots \to I_i$, and the other copy is the result of combination of local identifier I_j of $Entity_j$ (not $Entity_i$) with those random numbers picked for identifiers $I_1,\ldots,I_i$ in randomizing the master secret. Then with these two copies of information, $Entity_i$ can further hierarchically randomized these two copies with identical random number series along the identity hierarchies $I_{i+1} \to \ldots \to I_j$ and $I_j \to \ldots \to I_j$ respectively, and at last add these two copies of information. The summation is a private key for $Entity_j$. It is worth noting that two copies of information during the derivation process, i.e. along the identity hierarchy $I_{j+1} \to \ldots \to I_{j-1}$, neither can be used to derive private keys for entities other than $Entity_j$, nor can be added to get a private key for any ancestor of $Entity_j$.

4 ENSURING ENCRYPTION PRIVACY

Different from public key cryptography implemented in Public Key Infrastructure (PKI), where an entity's public and private key pair can be selected by either the trusted authority or the entity itself, private keys for an entity in HIBE system can only be generated by the root PKG or some domain (lower-level) PKGs. That is key escrow problem is inherent in (H)IBE public cryptography, and ciphertexts for an entity can be decrypted by those entities that are capable of generating valid private keys for the entity. What we detailed in Section 3 is on deriving valid private keys by some lower-level PKGs for some of their descendants, then there is no encryption privacy of ciphertexts targeting those descendants as far as those lower-level PKGs are concerned. However, other than from private key derivation perspective, it is necessary to consider encryption privacy from private key legitimacy perspective, i.e. whether an entity's private key is legitimate for ciphertexts encrypted on identity of the entity's descendants.

4.1 Dominated Encryption Privacy

"Dominated Encryption Privacy" means that ciphertexts targeting an entity can be decrypted by all or some of its ancestors without burden of generating a private key for the entity but with direct use of these ancestors' private keys.

As for BB$_1$ system in [1][3], it is not necessary that any ancestor of $Entity_j$ should derive a private key for $Entity_j$ in order to decrypt ciphertexts intended for $Entity_j$. When encrypting a given message $M \in G_t$ intended for $Entity_j$ with identity $ID_j = (I_1, \ldots, I_j) \in \left(Z_q^*\right)^j$, the encryptor picks a random value $s \in Z_q^*$ and outputs the ciphertext as

$$C = \left(Mv^s, sg, s(I_1 g + h_1), \ldots, s(I_j g + h_j)\right) \in G_t \times G^{j+1}.$$

Let $Entity_k$ with identity $ID_K = (I_1, \ldots, I_k)$ be an ancestor of $Entity_j$, the private key for $Entity_k$, denoted $\left(d_0^{(ID_k)}, RD_1^{(ID_k)}, \ldots, RD_K^{(ID_k)}\right)$, is extracted as

$$d_{ID_K} = \left(\hat{g}_0 + \sum_{i=1}^{k} r_i \left(I_i \hat{g}_1 + \hat{h}_i\right), r_1 \hat{g}, \ldots, r_k \hat{g}\right).$$

$Entity_k$ can decrypt ciphertext C, denoted $\left(C_0, C_1, RE_1^{(ID_j)}, \ldots, RE_j^{(ID_j)}\right)$, intended for $Entity_j$, as

$$\begin{aligned} M &= C_0 \cdot \frac{\prod_{i=1}^{k} e(RE_i^{(ID_j)}, RD_i^{(ID_k)})}{e\left(C_1, d_0^{(ID_k)}\right)} \\ &= M \cdot v^s \cdot \frac{\prod_{i=1}^{k} e\left(s(I_i g_1 + h_i), r_i \hat{g}\right)}{e\left(sg, \hat{g}_0 + \sum_{i=1}^{k} r_i \left(I_i \hat{g}_1 + \hat{h}_i\right)\right)} \\ &= M \end{aligned}$$

That is, any ancestor of an entity can decrypt ciphertexts encrypted on the public key (i.e. identity) of the entity with only use of its own private key without need of deriving a valid private key for the entity.

4.2 Dedicated Encryption Privacy

“Dedicated Encryption Privacy” means that all entities other than the intended recipient of a ciphertext cannot decrypt the ciphertext, thus achieving encryption privacy of ciphertext dedicated only to the intended recipient.

As for encryption privacy of HIBE system presented in [4], assume that $Entity_i$ is an ancestor of $Entity_j$ (without respect to whether $Entity_i$ is capable of deriving private keys for $Entity_j$ or not), and an encryptor encrypts a given message $M \in G_1$ on $Entity_j$ ’s identity $ID_j = (I_1, \ldots, I_j)$ as,

$$C = \left(M \cdot e(g_1, g_2)^s, \quad sg, \quad s\left(g_3 + \sum_{k=1}^{j} I_k h_k\right)\right).$$

then $Entity_i$ can decrypt the ciphertext with its private key (see Section 3.2) as,

$$\begin{aligned} M' &= M \cdot e(g_1, g_2)^s \times \frac{e\left(rg, s\left(g_3 + \sum_{k=1}^{j} I_k h_k\right)\right)}{e\left(sg, g_2^{\alpha} + r\left(g_3 + \sum_{k=1}^{i} I_k h_k\right)\right)} \\ &= M \cdot \left(e(g,g)^{\sum_{k=i+1}^{j} I_k \alpha_k}\right)^{rs}. \end{aligned}$$

For a successful decryption, it is required that $\sum_{k=i+1}^{j} I_k \alpha_k$ is congruent to zero with modulus prime q, where α_k for $k = 1, \ldots, \ell$ are logarithms of $\log_g^{h_k}$. We have

$$\alpha_j \equiv -I_j^{-1} \sum_{k=i+1}^{j-1} I_k \alpha_k \pmod{q},$$

where I_j^{-1} is multiplicative inverse of I_j in Z_q. Because α_1, …, and α_ℓ are all uniformly and independently selected from Z_q, then the probability of event that α_j is of value $-I_j^{-1} \sum_{k=i+1}^{j-1} I_k \alpha_k \bmod q \in Z_q$ is $1/q$, which means the probability of success of decrypting a ciphertext intended for $Entity_j$ by $Entity_i$ as an ancestor equals the probability that a specific value $-I_j^{-1} \sum_{k=i+1}^{j-1} I_k \alpha_k \bmod q \in Z_q$ is selected as α_j. Similarly, if $Entity_i$ is a descendant of $Entity_j$, it is required that $\sum_{k=j+1}^{i} I_k \alpha_k$ is congruent to zero with modulus prime q for a successful description.

5 CONCLUDING REMARKS

We emphasize the crucial role of mechanisms of delegating private key generation in establishing logically hierarchical relationship between entities along hierarchy tree in HIBE systems, which should reflect the true institutional structures in real world, and classify the delegation mechanisms into three classes, with reference to how an entity’s private key can be generated by lower-level PKGs other than the root PKG. Moreover, a framework of achieving authorized delegation is proposed, which hierarchically derive secrets along identity hierarchies $I_1 \rightarrow \ldots \rightarrow I_j$ and $I_j \rightarrow \ldots \rightarrow I_j$, i.e. by randomizing the master secret of the root PKG along the former, and privacy specifically pertained to local identifier I_j (dedicated privacy) along the later, and at last get a private key for $Entity_j$. Contrasted to direct and unrestricted private key derivation in unlimited delegation HIBE systems, and restricted private key derivation of limited depth in limited delegation HIBE systems, authorized delegation can explicitly authorize some entity to generate valid private keys for some specified entity of which ancestors’ private keys are not needed or generated at all.

At last, two types of encryption privacy, i.e. dominated encryption privacy and dedicated encryption privacy, are discussed and compared from private key legitimacy perspective. It is unquestionably necessary to achieve dedicated encryption privacy when constructing a HIBE system.

6 CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

7 REFERENCES

[1] Dan Boneh and Xavier Boyen, "Efficient selective-id secure identity-based encryption without random oracles", In Christian Cachin and JanL. Camenisch, editors, *Advances in Cryptology - EUROCRYPT 2004*, volume 3027 of Lecture Notes in Computer Science, pages 223–238. Springer Berlin Heidelberg, 2004.

[2] Dan Boneh and Xavier Boyen, "Secure identity based encryption without random oracles", In Matt Franklin, editor, *Advances in Cryptology – CRYPTO 2004*, volume 3152 of Lecture Notes in Computer Science, pages 443–459. Springer Berlin Heidelberg, 2004.

[3] Dan Boneh and Xavier Boyen, "Efficient selective identity-based encryption without random oracles", *Journal of Cryptology*, 24(4):659–693, 2011.

[4] Dan Boneh, Xavier Boyen, and Eu-Jin Goh, "Hierarchical identity based encryption with constant size ciphertext", In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of Lecture Notes in Computer Science, pages 440–456. Springer Berlin Heidelberg, 2005.

[5] Dan Boneh and Matthew Franklin, "Identity-based encryption from the weil pairing", *SIAM J. Comput.*, 32(3):586–615, March 2003.

[6] Dan Boneh and Matthew K. Franklin, "Identity-based encryption from the weil pairing", In: *Proceedings of the 21st Annual International Cryptology Conference on Advances in Cryptology, CRYPTO '01*, pages 213–229, London, UK, UK, 2001. Springer-Verlag.

[7] Ran Canetti, Shai Halevi, and Jonathan Katz, "A forward-secure public-key encryption scheme", In Eli Biham, editor, *Advances in Cryptology - EUROCRYPT 2003*, volume 2656 of Lecture Notes in Computer Science, pages 255–271. Springer Berlin Heidelberg, 2003.

[8] Clifford Cocks, "An identity based encryption scheme based on quadratic residues", In Bahram Honary, editor, *Cryptography and Coding, volume 2260 of Lecture Notes in Computer Science*, pages 360–363. Springer Berlin Heidelberg, 2001.

[9] Craig Gentry, "Practical identity-based encryption without random oracles", In Serge Vaudenay, editor, *Advances in Cryptology - EUROCRYPT 2006*, volume 4004 of Lecture Notes in Computer Science, pages 445–464. Springer Berlin Heidelberg, 2006.

[10] Craig Gentry and Alice Silverberg, "Hierarchical id-based cryptography", In: *Proceedings of the 8th International Conference on the Theory and Application of Cryptology and Information Security: Advances in Cryptology, ASIACRYPT '02*, pages 548–566, London, UK, UK, 2002. Springer-Verlag.

[11] Jeremy Horwitz and Ben Lynn, "Toward hierarchical identity-based encryption", In LarsR. Knudsen, editor, *Advances in Cryptology - EUROCRYPT 2002*, volume 2332 of Lecture Notes in Computer Science, pages 466–481. Springer Berlin Heidelberg, 2002.

[12] Kiyoshi Ohgishi Ryuichi Sakai and Masao Kasahara, "Cryptosystems based on pairings", *Symposium on Cryptography and Information Security 2000* - SCIS2000, 2000.

[13] Adi Shamir, "Identity-based cryptosystems and signature schemes", In GeorgeRobert Blakley and David Chaum, editors, *Advances in Cryptology, volume 196 of Lecture Notes in Computer Science*, pages 47–53. Springer Berlin Heidelberg, 1985.

[14] J. Silverman, "The arithmetic of elliptic curve", Sprinter-Verlag, 1983.

[15] Brent Waters, "Efficient identity-based encryption without random oracles", In Ronald Cramer, editor, *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of Lecture Notes in Computer Science, pages 114–127. Springer Berlin Heidelberg, 2005.

[16] Brent Waters, "Dual system encryption: Realizing fully secure ibe and hibe under simple assumptions", In Shai Halevi, editor, *Advances in Cryptology - CRYPTO 2009*, volume 5677 of Lecture Notes in Computer Science, pages 619–636. Springer Berlin Heidelberg, 2009.