Short Hierarchical Identity-based Encryption in the Selective-ID Model

Xiaoming Hu
School of Computer & Information,Shanghai Second Polytechnic University, Shanghai, China
Email: xmhu@sspu.cn

Huajie Xu[1], Jian Wang[2], Yinchun Yang[2], Xiaolin Xu[2]
[1]School of Computer and Electronic Information, Guangxi University, Nanning, China
[2]School of Computer & Information,Shanghai Second Polytechnic University, Shanghai, China
Email: hjxu@gxu.edu.cn, {wangjian, ycyang, xlxu}@sspu.cn

***Abstract*—Recently Zhang et al. proposed a hierarchical identity-based encryption scheme which is the first efficient scheme where both ciphertexts and private keys achieve O(1)size, and is the best trade-off between private key size and ciphertext size at present. However, in this paper, it will be pointed out that their scheme exists an ambiguity or shortcoming which makes their scheme be insecure or non identity-based. Then, in order to overcome this problem, an improved hierarchical identity based encryption scheme is proposed with the same efficiency with Zhang et al.'s scheme (the private keys and ciphertexts of O(1)size). And, the security proof of the improved scheme also is given in the selective-identity model.**

***Index Terms*—HIBE, selective-identity model, identity-based encryption, security analysis, constant size**

I. INTRODUCTION

In a basic ID-based encryption (IBE) proposed by Shamir[1], there is a single trusted server, private key generator (PKG), responsible for computing the private key of each user based on his public key. However, using a single PKG is not practical in large scale, so Gentry-Silverberg [2] extended ID-based encryption to hierarchical ID-based encryption (HIBE). The notion of the hierarchical ID-based encryption reduces the workload of the root PKG by delegating the private key generation task to lower level entities, i.e., PKGs who have already obtained their private keys. Due to the hierarchical property of HIBE, it is applied in many areas where there are hierarchical administrative issues, such as large companies or e-government systems. Recently HIBE also are applied to Health Record System [3] and cloud computing [4].

Manuscript received September 27, 2012; revised June, 2013; accepted July, 2013
Corresponding author: Xiaoming Hu (xmhu@sspu.cn).

From the first introduction of Gentry-Silverberg, many works have been done on HIBE and many HIBE schemes were proposed [5-17]. However, Zhang et al. [13] pointed out that all previous these schemes have the common drawback that the private key or the cipher text depends on the hierarchy or the maximum hierarchy. In order to overcome the drawback, Zhang et al. recently (2012) proposed a new HIBE scheme based on pairings [18] that ciphertext size as well as the private-key size is independent of the hierarchy depth, which is the first scheme whose private keys and ciphertexts achieve O(1)-size. However, in this paper, we will point out that Zhang et al.'s scheme exists a shortcoming which makes their scheme be insecure or non based on identity. In order to solve this problem, we propose an improved scheme which can overcome the drawback with almost the same efficiency that the private key and the ciphertext size are independent of the hierarchy depth.

II. PRELIMINARIES

A. Blinear Map

Let G and G_T be a cyclic addictive group and a cyclic multiplicative group with prime order p respectively, e be a mapping: $G \times G \to G_T$ which satisfies the following three propertied:

(1) Bilinear: for all u , $v \in G$ and a , $b \in Z_p$ $e(u^a, v^b) = e(u,v)^{ab}$.

(2) Non-degeneracy: $e(g,g) \neq 1$.

(3) Computability: for all u , $v \in G$, there exists an efficient algorithm to compute $e(u,v)$.

B. Hierarchical Identity-Based Encryption (HIBE)

In the subsection, we will show the concept of an HIBE scheme. Based on the [6], a l -level HIBE consists of four algorithms: Setup, Key generation, Encryption and Decryption, where l denotes the maximum level of an HIBE.

doi:10.4304/jsw.8.7.1613-1619

Setup: Input a security parameter k, and return the public system parameters *params* and the secrete master key *msk* which only is known by the private key generator (PKG).

Key generation: Input a k-level identity $ID=(v_1,\ldots, v_k)(1\le k\le l)$ and the private key of k-1 level identity, and return a private key d_{ID} of identity ID. If k=1, then the private key of ID is generated by PKG; If k>1, then the private key of ID is generated by PKG_{i-1} using the private key of k-1 lever identity.

Encryption: Input an i-lever identity ID and a message M, return a ciphertext C.

Decryption: Input a ciphertext C and the private key of i-lever identity ID, return the plaintext message M or bad.

III. Review and Security Analysis of Zhang et at.'s HIBE

A. ZHANG ET AT's Scheme

In the subsection, we will simply review Zhang et al. HIBE scheme [13]. Their scheme consists of four algorithms: Setup algorithm, Key Generation algorithm, Encryption algorithm and Decryption algorithm. Assume that g is random generator of group G with prime order p, and l is the maximum length of HIBE.

Setup algorithm: Chose randomly $g_2\in G$ and α, α_{i1}, α_{i2},…, α_{in}, β_{i1}, β_{i2},…, $\beta_{in}\in Z_p$ with $1\le i\le l$. Set $PK=\{g, g_1=g^{\alpha}, g_2\}$as the public key, g_2^{α} as the master key, and $Msk_i=\{\alpha_{i1},\alpha_{i2},\ldots,\alpha_{in},\beta_{i1},\beta_{i2},\ldots,\beta_{in}\}$as the shared master key for PKG_i at hierarchy depth i.

Key generation algorithm:

For the first level $ID=(v_1)$ where $v_1=(v_{11},\ldots, v_{1n}), v_{1j}\in\{0,1\}$. PKG picks randomly $r\in Z_p$ and sets $d_{ID}=(d_0=g_2^{\alpha}h_{1n}^{r}, d_1=g^{r})$as ID's private key, where

$$h_{1i}=(h_{1(i-1)})^{\alpha_{1j}^{v_{1j}}\beta_{1j}^{1-v_{1j}}} \text{ and } h_{10}=g \ (1\le i\le n).$$

For the k-th level $ID=(v_1,\ldots,v_k)$ $(k\le l)$where $v_i=(v_{i1},\ldots, v_{in}), v_{ij}\in\{0,1\}$. PKG_i can use the private key

$$d_{ID}'=(d_0'=g_2^{\alpha}(\prod_{i=1}^{k-1}h_{in})^{r}, d_1'=g^{r})$$

of k-1 th level ID to generate the k-th's private key

$$d_{ID}=(d_0, d_1)=(d_0'h_{kn}', d_1')=(g_2^{\alpha}(\prod_{i=1}^{k}h_{in})^{r}, g^{r}),$$

where

$$h_{k0}'=d_1', h_{kj}'=(h_{k(j-1)}')^{\alpha_{kj}^{v_{kj}}\beta_{kj}^{1-v_{kj}}},$$

$$h_{kn}=g^{\prod_{i=1}^{n}\alpha_{kii}^{v_{ki}}\beta_{ki}^{1-v_{ki}}}.$$

Encryption algorithm: The ciphertext on the message M is

$$C=(C_0,C_1,C_2)$$

$$=(M\cdot e(g_1,g_2)^{s}, g^{s}, (\prod_{i=1}^{k}h_{in})^{s}),$$

where $s\in_R Z_p$.

Decryption algorithm: The plaintext can be recovered with the private key $d_{ID}=(d_0,d_1)$ as follows:

$$M=C_0\frac{e(d_1,C_2)}{e(d_0,C_1)}.$$

B. Security Analysis

In the subsection, we will show the severe security weakness in the Zhang et al.'s scheme.

From Zhang et al.'s scheme described above, we can find that they didn't point out that where does the parameter h_{in} come from and who computes the $h_{in}(1\le i\le k)$? In order to describe conveniently, in this section, we denote h_{in} for all $h_{in}(1\le i\le k)$. By analyzing Zhang et al.'s scheme, it can be derived that h_{in} is generated by the encryption user or PKG_i. However, we will point out whoever generates h_{in} (the encryption user or PKG_i) will make their scheme loss of security or identity-based. Next is the details.

(1) Suppose the encryption user computes h_{in}. Then, the user will compute h_{in} for every encryption operation on chosen $ID_k=(v_1,v_2,\ldots, v_k)$. In this case, in order to compute

$$h_{in}=g^{\prod_{j=1}^{n}\alpha_{ij}^{v_{ij}}\beta_{ij}^{1-v_{ij}}} \ (1\le i\le k),$$

the encryption user must first obtain α_{ij} and β_{ij} $(1\le i\le k, 1\le j\le n)$. This is very dangerous, because from Zhang et al.'s scheme we know that α_{ij} and β_{ij} are the shared master keys that only are known by PKG_{i-1}, and are used to generate the private key for i^{th} $(1\le i\le l)$ level identity. Once α_{ij} and β_{ij} $(1\le i\le k, 1\le j\le n)$ are exposed to the encryption user, he/she can use the α_{ij} and β_{ij} $(1\le i\le k, 1\le j\le n)$ with a known private key $d_{ID}=(d_0=g_2^{\alpha}(\prod_{i=1}^{m}h_{in})^{r}, d_1=g^{r})$on the identity $ID_k=(v_1,v_2,\ldots, v_m)(1\le m\le k)$.in advance to compute the master key

$$g_2^{\alpha}\left(=d_0\Big/d_1^{\prod_{i=1}^{m}\prod_{j=1}^{n}a_{ij}^{v_{ij}}b_{ij}^{1-v_{ij}}}\right).$$

Note it is not hard for the encryption user to get a private key d_{ID}, because any one PKG_m $(1\le m\le k)$.can obtain the d_{ID} from the upper level PKG, then PKG_m can collude with the encryption user or the malicious user can directly request to PKG_m for a private key d_{ID}. With the master key g_2^{α}, the encryption user can decrypt any cipher text. So, h_{in} can't be computed by the encryption user otherwise Zhang et al.'s scheme is insecure.

(2) Suppose every level PKG$_i$ computes h_{in} on the $ID_k = (v_1, v_2, \ldots, v_k)$ respectively. In this case, all PKG$_i$ first use the shared master key α_{ij} and β_{ij} $(1 \le j \le n)$ to compute h_{in}, then publish it publicly. According to the encryption process of Zhang et al.'s scheme $C = (M \cdot e(g_1, g_2)^s, g^s, (\prod_{i=1}^{k} h_{in})^s)$, the encryption user can directly use only the published h_{in} and (g_1, g_2) to generate the cipher text $C = (M \cdot e(g_1, g_2)^s, g^s, (\prod_{i=1}^{k} h_{in})^s)$ on the message M without needing to know the identity $ID_k = (v_1, v_2, \ldots, v_k)$. In this case, the encryption user can't know which identity is the real identity ID_k used to generate the ciphertext $C = (C_0, C_1, C_2)$. Thus, a malicious enemy can issue that h_{in}'s corresponding identity is $ID_k' (\neq ID_k)$, the malicious enemy know the private key of ID_k') not ID_k because there is not any banding between h_{in} and ID_k. Thus, the encryption user generates a ciphertext $C = (C_0, C_1, C_2)$ on the uncorrect identity $ID_k' (\neq ID_k)$, but he/she believe that C is on ID_k.

Thus, the malicious enemy can decrypt the ciphertext C with the private key of ID_k'. This problem is very severe. The main reason is because that the encryption process of Zhang et al.' scheme don't need to use decryption user's identity ID_k, so the encryption user can't distinguish the real identity from the parameter h_{in}, unless every level PKG$_i$ makes a certification that binds the identity v_i and corresponding parameter h_{in} or the encryption user timely access every level PKG$_i$ to get the real h_{in} corresponding with v_i of an identity when he/she make a encryption operation. Both of them all are not a good idea. For one method, the encryption user must get the certification to verify h_{in}'s authenticity before the encryption user encrypt, which make the identity based scheme change into traditional certification based scheme which is in the contradict with the paper's goal. For two method, PKG$_i$ will compute h_{in} for every different identity v_i $(1 \le i \le k)$ of any height identity ID_k $(1 \le k \le l)$ and all level PKG must be online to support the h_{in} access operation from every encryption user, it will largely increase the cost and also is in the contradict with identity based scheme where encryption user can directly use the decryption user's identity information as the public key without accessing PKI or others. So, PKG$_i$ also can't compute h_{in} otherwise Zhang et al.'s scheme is non identity based.

IV. IMPROVED SCHEME

A. Our Scheme

In order to solve the previous problem, we propose an improved scheme of Zhanget al.'s HIBE [13]. It consists of four phases. In order to compare easily, we use the same symbols with Zhang et al's scheme.

Setup Phase: Let G be a group generated by g whose order is a prime p, and l be the max height of HIBE and n be the bit number of every identity. Pick randomly $\alpha \in Z_p$, $g_2 \in G$, and set $g_1 = g^{\alpha}$. Choose randomly λ_i, $\alpha_{i1}, \alpha_{i2}, \ldots \alpha_{in}$, $\beta_{i1}, \beta_{i2}, \ldots, \beta_{in} \in Z_p$, set $c_i = g^{\lambda_i}$, $a_{i1} = g^{\alpha_{i1}}$, $a_{i2} = g^{\alpha_{i2}}, \ldots, a_{in} = g^{\alpha_{in}}$, $b_{i1} = g^{b_{i1}}$, $b_{i2} = g^{b_{i2}}, \ldots, b_{in} = g^{b_{in}}$, where $1 \le i \le l$. Then g_2^{α} is the master key that only is known by the root PKG. At the $(i\text{-}1)^{\text{th}}$ level, PKG$_{i-1}$ is given the share master key $(\lambda_i, \alpha_{i1}, \ldots \alpha_{in}, \beta_{i1}, \ldots, \beta_{in})$ $(1 \le i \le l)$. The public parameters are $PK = \{g, g_1, g_2, c_1, \ldots, c_l, a_{i1}, \ldots, a_{in}, b_{i1}, \ldots, b_{in}\}$, where $1 \le i \le l$.

$$g, g_1, g_2, c_1, \ldots, c_l, a_{i1}, \ldots, a_{in}, b_{i1}, \ldots, b_{in}, 1 \le i \le l$$

Figure 1. public parameters

Key Generation Phase: Assumed that $ID_k = (v_1, v_2, \ldots, v_k)$ $(1 \le k \le l)$ with $v_i = (v_{i1}, v_{i2}, \ldots, v_{in})$ $(1 \le i \le k)$ is the identity for which the private key is required, where $v_{ij} \in \{0,1\}$ $(1 \le j \le n)$, then the private key for ID_k is generated by the following steps:
(1) Define a function

$$h_{in} = c_i \prod_{j=1}^{n} a_{ij}^{v_{ij}} b_{ij}^{1-v_{ij}}, \text{ where } 1 \le i \le l.$$

(2) Choose $r \in_R Z_p$, and computer the private key $d_{ID}^k = (d_0, d_1)$ for identity ID_k:

$$d_0 = g_2^{\alpha} (\prod_{i=1}^{k} c_i (\prod_{j=1}^{n} a_{ij}^{v_{ij}} b_{ij}^{1-v_{ij}}))^r = g_2^{\alpha} (\prod_{i=1}^{k} h_{in})^r,$$

$$d_1 = g^r.$$

The private key $d_{ID}^k = (d_0, d_1)$ of ID_k also can be computed by PKG$_{i-1}$ using the private key $d_{ID}^{k-1} = (d_0', d_1') = (g_2^{\alpha} (\prod_{i=1}^{k-1} h_{in})^r, g^r)$ and the shared master key $(\lambda_k, \alpha_{k1}, \ldots, \alpha_{kn}, \beta_{k1}, \ldots, \beta_{kn})$ of the parent $(k\text{-}1)$ level ID_{k-1}:
(1) Define a function

$$T(v_k) = \sum_{j=1}^{n} (\alpha_{kj} v_{kj} + \beta_{kj} (1 - v_{kj})), \text{ where } 1 \le i \le l.$$

(2) Compute $d_1 = d_1'$,

$$d_0 = d_0' \cdot (d_1')^{\lambda_k} \cdot (d_1')^{T(v_k)}$$

$$= g_2^{\alpha} (\prod_{i=1}^{k-1} h_{in})^r \cdot g^{r\lambda_k} \cdot g^{r \sum_{j=1}^{n} (\alpha_{kj} v_{kj} + b_{kj} (1 - v_{kj}))} =$$

$$g_2^{\alpha} (\prod_{i=1}^{k-1} h_{in})^r \cdot g^{r(\lambda_k + (\alpha_{k1} v_{k1} + \ldots + \alpha_{kn} v_{kn} + b_{k1} (1 - v_{k1}) + \ldots + b_{kn} (1 - v_{kn})))}$$

$$= g_2^{\alpha} (\prod_{i=1}^{k-1} h_{in})^r \cdot c_k^r \cdot \prod_{j=1}^{n} (a_{k1}^{v_{k1}} b_{k1}^{1-v_{k1}})^r$$

$= g_2^{\alpha}(\prod_{i=1}^{k-1} h_{in})^r \cdot (c_k \prod_{j=1}^{n}(a_{k1}^{v_{k1}} b_{k1}^{1-v_{k1}}))^r$

$= g_2^{\alpha}(\prod_{i=1}^{k-1} h_{in})^r \cdot (h_{kn})^r = g_2^{\alpha}(\prod_{i=1}^{k} h_{in})^r$

Then, $d_{ID_k} = (d_0, d_1)$ is the private key of ID_k.

$$d_0 = g_2^{\alpha}(\prod_{i=1}^{k} h_{in})^r$$
$$d_1 = g^r,\ 1 \le k \le l$$

Figure 2. Private Key of *k*th lever

Encryption Phase: Assumed that M is a message which the encryption is required and $ID_k = (v_1, v_2, \ldots, v_k)(1 \le k \le l)$ is the identity. The cipher text of M on the identity ID_k is generated as follow: pick randomly $s \in Z_p$, and compute

$C = (C_0, C_1, C_2)$

$= (M \cdot e(g_1, g_2)^s, g^s, (\prod_{i=1}^{k} c_i \cdot \prod_{j=1}^{n}(a_{ij}^{v_{ij}} b_{ij}^{1-v_{ij}}))^s)$

$= (M \cdot e(g_1, g_2)^s, g^s, (\prod_{i=1}^{k} h_{in})^s)$.

$$M \cdot e(g_1, g_2)^s, g^s, (\prod_{i=1}^{k} h_{in})^s, 1 \le k \le l$$

Figure 3. ciphertext for *k*th lever

Decryption Phase: After getting a cipher text $C = (C_0, C_1, C_2)$ for identity ID_k and message M, the plain text *M* can be recovered by using the private key $d_{ID}^k = (d_0, d_1)$ of ID_k:

$M = C_0 \cdot \frac{e(d_1, C_2)}{e(d_0, C_1)} = M \cdot e(g_1, g_2)^s \cdot \frac{e(g^r, (\prod_{i=1}^{k} h_{in})^s)}{e(g_2^{\alpha}(\prod_{i=1}^{k} h_{in})^r, g^s)}$

$= M \cdot e(g_1, g_2)^s \cdot \frac{e(g^r, (\prod_{i=1}^{k} h_{in})^s)}{e(g_2^{\alpha}, g^s) e((\prod_{i=1}^{k} h_{in})^r, g^s)}$

$= M \cdot e(g_1, g_2)^s \cdot \frac{1}{e(g_2, g^{\alpha s})} = M$.

$$C_0 \frac{e(d_1, C_2)}{e(d_0, C_1)}$$

Figure 4. Decryption Equation

So, (C_0, C_1, C_2) is a correctness cipher text.

B. Security Analysis

In this section, we show that the scheme constructed in the previous subsection is secure in the selective-identity model.

Theorem 1. Assuming that the (ε, t)–.decisional n+1-BDHE assumption holds in G, the proposed HIBE scheme is (ε, t', q_E)secure in the selective-identity model, where q_E is the number of the private key generation queries , $t' = t + O(\tau q_E)$, and τ is the time for an exponentiation in G.

Proof. We prove that if there exists a successful (ε, t', q_E) – adversary A against our scheme, then we can construct an algorithm B that solves the decisional n+1-BDHE problem in time at most t' with probability at least ε .This is contradicting with the (ε, t)– .decisional n+1-BDHE assumption.

The algorithm B is given a random instance $(g, y_0, y_1, \ldots, y_n, y_{n+2}, \ldots, y_{2n+2}, T)$ of the decisional n+1-BDHE problem, where $g \in G$ is a generator of group G and α, $c \in Z_p$, $y_i = g^{\alpha^i}$ $(1 \le i \le n$ or $n+2 \le j \le 2n+2)$ and $y_0 = g^{c}$. *B*'s goal is to output 1 if

$$T = e(g, g)^{g^{\alpha^{n+1}c}}$$

or output 0. To do so, B must be able to simulate a challenger for A, and such a simulation can be created by the following way.

(1) **Initiation Phase:** The adversary A outputs an identity $ID_k^* = (v_1^*, v_2^*, \ldots, v_k^*)$ $(1 \le k \le l)$with $v_i^* = (v_{i1}^*, v_{i2}^*, \ldots, v_{in}^*)$ and $v_{ij} \in \{0,1\}(1 \le i \le k$, $1 \le j \le n)$which he want to attack.

(2) **Setup Phase:** B chooses randomly $\mu \in Z_p$, and set $g_1 = g^{\alpha} = y_1$ and $g_2 = y_n g^{\mu} = g^{\alpha^n + \mu}$. Then, pick randomly $\lambda_i, \alpha_{i1}, \alpha_{i2}, \ldots \alpha_{in}, \beta_{i1}, \beta_{i2}, \ldots, \beta_{in} \in Z_p$, set $a_{i1} = g_1^{\alpha_{i1}} = y_1^{\alpha_{i1}}$, $a_{i2} = g_1^{\alpha_{i2}} = y_1^{\alpha_{i2}}, \ldots$, $a_{in} = g_1^{\alpha_{in}} = y_1^{\alpha_{in}}$, $b_{i1} = g_1^{\beta_{i1}} = y_1^{\beta_{i1}}$, $b_{i2} = g_1^{\beta_{i2}} = y_1^{\beta_{i2}}, \ldots$,

$b_{in} = g_1^{\beta_{in}} = y_1^{\beta_{in}}$,

$c_i = g^{\lambda_i} g_1^{\sum_{j=1}^{n}(-\alpha_{ij} v_{ij}^* - \beta_{ij}(1-v_{ij}^*))} = y_1^{\sum_{j=1}^{n}(-\alpha_{ij} v_{ij}^* - \beta_{ij}(1-v_{ij}^*))} g^{\lambda_i}$,

where$1 \le i \le l$. Note we can extend the *k* levels of ID_k^* to l levels if need. Then the master key is g_2^{α} that is not known to B, and the share master key in the i^{th} level is $(\lambda_i, \alpha_{i1}, \alpha_{i2}, \ldots \alpha_{in}, \beta_{i1}, \beta_{i2}, \ldots, \beta_{in})$ that is known to B. The public parameters $PK = \{g, g_1, g_2, c_1, \ldots, c_l, a_{i1}, \ldots, a_{in}, b_{i1}, \ldots, b_{in}\}$ is send to A, where $1 \le i \le l$.

(3) **Query Phase1:** In this phase, A is allowed to make q_E private key queries. Assuming that $ID_k = (v_1, v_2, \ldots, v_k)(1 \le k \le l)$ is the identity that A submits to ask for the private key, and the restriction is ID_k is not ID_k^* or a prefix of ID_k^*. Assuming that i is the smallest index that $v_i^* \ne v_i$ $(1 \le i \le k)$, namely $v_1 = v_1^*, \ldots, v_{i-1} = v_{i-1}^*$. B answers the query in the following way.

(1) Define two functions

$$F(v_m) = \sum_{j=1}^{n}(-\alpha_{mj} v_{mj}^* - \beta_{mj}(1 - v_{mj}^*)) +$$
$$\sum_{j=1}^{n}(\alpha_{mj} v_{mj} + \beta_{mj}(1 - v_{mj}))$$

$= \sum_{j=1}^{n}(\alpha_{mj}(v_{mj} - v_{mj}^*) + \beta_{mj}(1 - v_{mj} - (1 - v_{mj}^*)))$,

$h_{mn} = y_1^{F(v_m)} \cdot g^{\lambda_k}$, where $1 \le m \le k$.

It is obvious that for all $1 \le x \le i$-1, $F(v_x) = F(v_x^*) =$

$\sum_{j=1}^{n}(\alpha_{mj}(v_{mj}^* - v_{mj}^*) + \beta_{mj}(1 - v_{mj}^* - (1 - v_{mj}^*)))$ =0.

So, $h_{xn} = y_1^{F(v_x)} \cdot g^{\lambda_x} = g^{\lambda_x}$.

(2) B first generate the private key of $ID_i = (v_1, v_2, \ldots, v_i)$. Choose randomly $r' \in Z_p$, then the private key of ID_i is simulated as

$$d_{ID}^i = (d_0, d_1)(g_2^{\alpha}(\prod_{j=1}^{i} h_{jn})^r, g^r),$$

where $r = r' - \frac{\alpha^n}{F(v_i)}$. d_{ID}^i can be computed correctly because:

$$d_0 = g_2^{\alpha}(\prod_{j=1}^{i} h_{jn})^r$$
$$= y_{n+1} y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn}))^r \cdot h_{in}^r$$
$$= y_{n+1} y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn})^r \cdot (y_1^{F(v_i)} g^{\lambda_i})^r$$
$$= y_{n+1} y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_1^{F(v_i)} g^{\lambda_i})^{-\frac{\alpha^n}{F(v_i)}} =$$
$$y_{n+1} y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_1^{F(v_i)})^{-\frac{\alpha^n}{F(v_i)}} (g^{\lambda_i})^{-\frac{\alpha^n}{F(v_i)}}$$
$$= y_{n+1} y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} y_{n+1}^{-1} (g^{\lambda_i})^{-\frac{\alpha^n}{F(v_i)}}$$
$$= y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} (g^{\lambda_i})^{-\frac{\alpha^n}{F(v_i)}}$$
$$= y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn} y_1^{F(v_j)} g^{\lambda_j})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_n)^{-\frac{\lambda_i}{F(v_i)}}$$
$$= y_1^{\mu}(\prod_{j=1}^{i-1} h_{jn} g^{\lambda_j})^r (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_n)^{-\frac{\lambda_i}{F(v_i)}}$$
$$= y_1^{\mu}(\prod_{j=1}^{i-1} g^{\lambda_j})^{r' - \frac{\alpha^n}{F(v_i)}} (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_n)^{-\frac{\lambda_i}{F(v_i)}} =$$
$$y_1^{\mu}(\prod_{j=1}^{i-1} g^{\lambda_j})^{r'} (\prod_{j=1}^{i-1} y_n)^{-\frac{\lambda_j}{F(v_i)}} (y_1^{F(v_i)} g^{\lambda_i})^{r'} (y_n)^{-\frac{\lambda_j}{F(v_i)}}$$

(1)

$$d_1 = g^r = g^{r' - \frac{\alpha^n}{F(v_i)}} = g^{r'} g^{-\frac{\alpha^n}{F(v_i)}} = g^{r'} \cdot y_n^{-\frac{1}{F(v_i)}} \qquad (2)$$

It is obvious that all expression in (1) and (2) are known to B. So, B can compute the private key of ID_i. Using the private key d_{ID}^i of ID_i, B can generate the private key of ID_k. So, B can simulate perfectly.

(4) **Challenge Phase:** After the phase1, A outputs two equal length messages M_0 and $M_1 \in G$ which he want to challenge. B picks randomly $b \in \{0,1\}$ and generates the cipher text $C^* = (C_0^*, C_1^*, C_2^*)$ of M_b in the identity $ID_k^* = (v_1^*, v_2^*, \ldots, v_k^*)$ as follows:

$$C_0^* = M_b Te(y_1^{\mu}, y_0),\ C_1^* = y_0,\ C_2^* = y_0^{\sum_{j=1}^{k} \lambda_j}.$$

If $T = e(g,g)^{g^{\alpha^{n+1}}c}$, then C^* is a valid challenged cipher text of M_b with ID_k^*. Because $F(v_m^*) = 0$ $(1 \le m \le k)$ and

$$C_0^* = M_b Te(y_1^{\mu}, y_0) = M_b e(g,g)^{g^{\alpha^{n+1}}c} e(g^{\alpha\mu}, g^c)$$
$$= M_b e(y_n, y_1)^c e(g^{\mu}, y_1)^c = M_b e(y_n g^{\mu}, y_1)^c$$
$$= M_b e(g_2, g_1)^c$$

$$C_1^* = y_0 = g^c$$

$$C_2^* = y_0^{\sum_{j=1}^{k} \lambda_j} = (g^{\lambda_1 + \lambda_2 + \ldots + \lambda_k})^c = (g^{\lambda_1} g^{\lambda_2} \ldots g^{\lambda_k})^c$$
$$= (y_1^{F(v_1^*)} y_2^{F(v_2^*)} \ldots y_k^{F(v_k^*)} g^{\lambda_1} g^{\lambda_2} \ldots g^{\lambda_k})^c$$
$$= (y_1^{F(v_1^*)} g^{\lambda_1} y_2^{F(v_2^*)} g^{\lambda_2} \ldots y_k^{F(v_k^*)} g^{\lambda_k})^c$$
$$= (h_{1n} h_{2n} \ldots h_{kn})^c = (\prod_{j=1}^{k} h_{jn})^c$$

Else, if T is a random element of G, the cipher text will give no information about M_b to *A*.

(5) **Query Phase2:** A continues to issue queries as in Phase 1 and B responds as be before.

(6) **Guess:** Finally, A outputs a guess $b' \in \{0,1\}$. If $b = b'$, B outputs 1 as the solution to the decisional n+1-BDHE problem, namely $T = e(g,g)^{g^{\alpha^{n+1}}c}$, else *B* outputs 0, namely T is a random element of G.

Probability Analysis: When the input ($g, y_0, y_1, \ldots, y_n, y_{n+2}, \ldots, y_{2n+2}, T$) is sampled from decisional n+1-BDHE (where $T = e(g,g)^{g^{\alpha^{n+1}}c}$) then A's view is identical to its view in a real attack game and therefore A must have $|\Pr[b = b'] - 1/2| > \varepsilon$. On the other hand, when the input ($g, y_0, y_1, \ldots, y_n, y_{n+2}, \ldots, y_{2n+2}, T$) is sampled from decisional n+1-BDHE (where T is uniform in G) then $\Pr[b = b'] = 1/2$. Therefore, B can solve the decisional n+1-BDHE problem with probability ε.

Time Complexity: The time complexity of algorithm B is dominated by exponentiation in G in the private key generation queries. Each such query requires $O(l)$ exponentiations in G. Assuming that τ is the maximum time of an exponentiation in G. Since A makes at most q_E private key queries, so $t' = t + O(\tau l q_E)$. □

C. Efficiency Analysis

For simplicity, we use the same comparison items with [13], and only draw the comparison result with [13].

TABLE I.
COMPARISON OF EFFICIENCY

Scheme	Ciphertext size	pk size	PK size
[13]	$O(1)$	$O(1)$	$O(k)$
Our scheme	$O(1)$	$O(1)$	$O(nk)$

From table1 of this paper and table1 of Zhang et al.' scheme, we can see that the private key and the cipher text in our scheme achieve $O(1)$ size respectively, and have less computation complexity than other any one scheme[5-11],and have the same computation complexity with [13]. Thought our scheme needs a little more public parameters than [13], our scheme overcome the security drawback of Zhang et al's scheme.

D. Implementation

The HIBE of this paper is pairing-based scheme. So, we can use Pairing-Based Cryptography (PBC) Library [19] which provides routines such as elliptic curve generation, elliptic curve arithmetic and pairing computation to implement our HIBE scheme by choosing suitable Tate pairing. According to PBC, a fastest pairing only needs 11ms on a 1GHz Pentium III. In our scheme, it only needs a pairing operation on the stage of encryption. So, our scheme is very efficient.

V. CONCLUSION

In this paper, we review Zhang et al.'s HIBE scheme and point out a mistake of Zhang et al.'s HIBE. Then, we propose a new HIBE scheme which can overcome the drawback of Zhang et al.'s HIBE with the same efficiency. Finally, we analyze the security of the proposed scheme and the efficiency.

ACKNOWLEDGMENT

This work is supported by the National Natural Science Foundation of China (No.61103213) and the Innovation Program of Shanghai Municipal Education Commission (No.10YZ201) and the Shanghai Education Commission Foundation for Excellent Young High Education Teacher (No.RYQ309002)

REFERENCES

[1] A. Shamir, Identity-Based Cryptosystems and Signature Schemes, Proc. CRYPTO, LNCS, vol. 196, 1985, pp. 47-53.
[2] C. Gentry and A. Silverberg, Hierarchical ID-Based Cryptog raphy, Proc. ASIACRYPT, LNCS, vol. 2501, 2002, pp. 548 -566.
[3] J. Sun and Y. Fang, Cross-Domain Delegation for Sensitive Data Sharing in Distributed Electronic Health Record Systems, IEEE Trans. Parallel Distrib. Syst., vol. 21, no. 6, 2010, pp. 754-764.
[4] L. Yan, C.M. Rong, and G.S. Zhao, Strengthen Cloud Computing Security with Federal Identity Management Using Hierarchical Identity-Based Cryptography, CloudCom, LNCS, vol. 5931, 2009, pp. 167-177.
[5] D. Boneh and X. Boyen, Efficient Selective-ID Secure Identity-Based Encryption without Random Oracles, Proc. EUROCRYPT, LNCS, vol. 3027, 2004, pp. 223-238.
[6] D. Boneh, X. Boyen, and E. Goh, Hierarchical Identity Based Encryption with Constant Ciphertext, Proc. EUROCRYPT, LNCS, vol. 3494, 2005, pp. 440-456.
[7] B. Waters, Dual System Encryption: Realizing Fully Secure IBE and HIBE under Simple Assumptions, Proc. CRYPTO, LNCS, vol. 5677, 2009, pp. 619-636.
[8] Y. Lu and J. Li, Generic Construction of Forward-Secure Identity-Based Encryption, Journal of Computers, Vol.7, No.12, 2012, pp.3068-3074.
[9] Y.L. Ren and D.W. Gu. Efficient Hierarchical Identity Based Encryption Scheme in the Standard Model, Informatica, vol. 32, no. 2, 2008, pp. 207-211.
[10] D. Cash, D. Hofheinz, and E. Kiltz, How to Delegate a Lattice Basis, Cryptology ePrint Archive, Report 2009/351 (2009). http://eprint.iacr.org/
[11] S. Agrawal, D. Boneh, and X. Boyen, Efficient Lattice (H)IBE in the Standard Model, Proc. EUROCRYPT, LNCS, vol. 6110, 2010, pp. 553-572.
[12] W. Yuan, L. Hu, H. Li, et al., Analysis and Enhancement of Three Identity-based Signcryption Protocols, Journal of Computers, Vol.7, No.4, 2012, pp.1006-1013.
[13] L.y. Zhang, Q. Wu, and Y.P Hu. Hierarchical IdentityBased Encryption with Constant-Size Private Keys. ETRI Journal, Volume 34, Number 1, February 2012,pp. 142-145.
[14] Sai Krishna Parsha and Mohd Khaja Pasha. Enhancing Data Security in Cloud Computing using Hierarchical Identity Based Encryption. International Journal of Research and Reviews in Engineering Sciences, Vol 1, No 1, 2012, pp.1-3.
[15] W. Yuan, L. Hu, H. Li, et al., Cryptanalysis and Improvement of an ID-Based Threshold Signcryption Scheme, Journal of Computers, Vol.7, No.6, 2012, pp.1345-1352.
[16] A. Lewko and B. Waters, New Techniques for Dual System Encryption and Fully Secure HIBE with Short Ciphertexts, Proc. TCC, LNCS, vol. 5978, 2010, pp. 455-479.
[17] H.S. Jae and H.C. Jung. Fully Secure Anonymous Hierarchical Identity-Based Encryption with Constant Size Ciphertexts. http://eprint.iacr.org/2011/021.pdf, 2011.
[18] F.Li, W.Gao, Y.Wang, et al., An Efficient Certificateless Threshold Decryption Schemes Based On Pairings, Journal of Computers, Vol.7, No.12, 2012, pp.2987-2996.
[19] PBC Library: the pairing-based cryptography library. Available from: http://crypto.standard.edu/pbc.

Xiaoming Hu received the Ph.D. degree in Department of Computer Application Technology, Shanghai JiaoTong University in 2009. Now, she is working in School of Computer & Information, Shanghai Second Polytechnic University, Shanghai, China. Her current research interests include cryptography, information security and network security.

Huajie Xu received the Ph.D. degree in the School of Computer Science and Technology, Huazhong University of Science and Technology in 2008. He is working in the School of Computer and Electronic Information, Guangxi University, Nanning, China. His current research interests include network security and wireless sensor network.

Jian Wang received the Ph.D. degree in Department of Computer Application Technology, Shanghai Jiao Tong University in 2009. Now, he is working in School of Computer & Information, Shanghai Second Polytechnic University, Shanghai, China. His current research

interests include network security, P2P network and sensor network.

Yinchun Yang received the M.S. degree in College of Information Security, Shanghai Jiao Tong University in 2004. Now, she is working in School of Computer & Information, Shanghai Second Polytechnic University, Shanghai, China. Her current research interests include information security and wireless network security.

Xiaolin Xu received the M.S. degree in Shanghai Maritime University. Now, she is working in School of Computer & Information, Shanghai Second Polytechnic University, Shanghai, China. Her current research interests include network and network security.