SECURITY AND COMMUNICATION NETWORKS
Security Comm. Networks 2010; **00**:1–11
DOI: 10.1002/sec

RESEARCH ARTICLE

An Escrow-Free Online/Offline HIBS Scheme for Privacy Protection of People-Centric Sensing

Peixin Chen[1], Jinshu Su[2,1], Baokang Zhao[1], Xiaofeng Wang[1] and Ilsun You[3]*

[1] College of Computer, National University of Defense Technology, Changsha 410073, China
[2] National Key Laboratory for Parallel and Distributed Processing, National University of Defense Technology, Changsha 410073, China
[3] Department of Information Security Engineering, Soonchunhyang University, Asan-si, Republic of Korea

ABSTRACT

People-Centric Sensing (PCS), which collects information closely related to human activity and interactions in societies, is stepping into a flourishing time. Along with its great benefits, PCS poses new security challenges such as data integrity, participant privacy. Hierarchical Identity-Based Signature (HIBS) scheme can efficiently provide high integrity messaging, secure communication and privacy protection to PCS. However, key escrow problem and low computation efficiency primarily hinder the adoption of HIBS scheme. In this paper, we propose an escrow-free online/offline HIBS (EF-HIBOOS) scheme for securing PCS. By utilizing user-selected-secret signing algorithm and splitting the signing phase into online and offline procedures, our scheme solves the key escrow problem and achieve high scheme performance. Copyright © 2010 John Wiley & Sons, Ltd.

KEYWORDS

Internet of Things; People-Centric Sensing; Hierarchical Identity-Based Signature; Online/Offline Signature; Key Escrow Problem

***Correspondence**

Department of Information Security Engineering, Soonchunhyang University, Asan-si, Republic of Korea.

Received . . .

1. INTRODUCTION

With enormous improvements of sensing technology and embed computation, more and more ubiquitous devices are utilized to build a new kind of mobile sensing system, which is referred to as People-Centric Sensing (PCS) system [1]. However, the PCS is facing a more serious privacy problem than the traditional wireless sensor network. Works on protecting the privacy of PCS has been proposed [2–5]. Most of the work assume that participant of PCS application are supported from a public key infrastructure (PKI). Nevertheless, building and operating a PKI are quite burden jobs, which significantly reduce the practicability of the PKI-based scheme. The Identity-Based Signature (IBS) scheme can be efficiently utilized to build a lightweight PKI that is appropriate for the PCS application.

IBS is a public key signature scheme which allows a receiver to verify message using the signer's identity as the public key [6]. Choon et al. propose the first practical IBS scheme that utilizes a Private Key Generator (PKG) to generate private key for users [7]. On this basis, Gentry et al. present the first Hierarchical IBS (HIBS) scheme, which greatly reduces the workload on PKG and resolves the problem of single-point failure in IBS scheme [8]. Improving the scheme security and efficiency, several IBS and hierarchical IBS (HIBS) schemes [9–14] have been presented.

Deployment of an HIBS scheme has to take two problems into consideration: key escrow problem and scheme performance problem. Since an HIBS scheme uses PKGs to generate private key for users, it inevitably leads to the *key escrow problem*. That is, the PKG knows the private keys and thus can unscrupulously sign messages intended for the users [15]. Besides the key escrow problem, the low computation efficiency of HIBS scheme is another concern while deploying the identity-based signature scheme. Most IBS schemes involve computations including pairings over points on

† Please ensure that you use the most up to date class file, available from the SEC Home Page at
http://www3.interscience.wiley.com/journal/114299116/home

elliptic cure and point multiplications in groups, which might be too costly to be applied in lightweight devices. Online/offline (OO) signature mechanism that divides the process of message signing into offline phase and online phase is an effective method to reduce the computational cost of signature generation [16]. The offline phase is performed prior to the knowledge of the message to be signed, and the online phase is performed after knowing the message. In an OO mechanism, most of the computation is implemented in offline signing and online signing is typically very fast while generating a signature. Imposing the OO mechanism, numerous identity-based online/offline signature (IBOOS) scheme have been proposed [17–22]. Yang et al. present a comprehensive IBOOS scheme, which solves the key escrow problem utilizing a threshold model and achieves high performance thanks to the OO signing algorithm [22]. However, there are few work on hierarchical IBOOS (HIBOOS) scheme and its key escrow problem.

In this paper, we propose an escrow-free online/offline hierarchical identity-based signature (EF-HIBOOS) scheme, which only executes two group element exponentiation operations in online procedure. We use a user-selected secret besides the signing key to sign messages so that any bogus signatures can be recognized and forgery behaviors can be detected and blamed. We prove that our scheme is selective-identity and adaptive chosen-message secure for existential-forgery attack (EF-sID-CMA) secure and can efficiently solve the key escrow problem.

The rest of this paper is organized as follows. We overview our escrow-free HIBOOS model in Section 2. The construction and security proof of the HIBOOS scheme are presented in Section 3. We discuss the performance and advantages of our HIBOOS scheme in Section 4. Related works are reviewed in Section 5 and our work is concluded in Section 6.

2. OVERVIEW OF OUR ESCROW FREE HIBOOS MODEL

In this section, we firstly review some background knowledge, including the bilinear pairing and the complexity assumption used in our proof. Then, we introduce the intuition of our solution to the key escrow problem of HIBOOS, and briefly describe the construction of our scheme. Finally, we present the security definition by illuminating the *EF-sID-CMA*, *EKA-ID-CMA* and *EUS-sID-CMA* attack games for our escrow-free approach.

2.1. Preliminaries

2.1.1. Bilinear Pairing

Let G_1 and G_2 be two cyclic multiplicative groups of the same order p. A map $e : G_1 \times G_1 \to G_2$ is referred to as a bilinear paring if it has the following properties:

1. Bilinear: $\forall u, v \in G_1, a, b \in \mathbb{Z}_N$, there is $e(u^a, v^b) = e(u, v)^{ab}$;
2. Non-degenerate: $\exists g \in G_1$, s.t. $e(g, g) \neq 1$, where 1 denotes the identity in G_2;
3. Computable: $\forall u, v \in G_1$, there is an efficient algorithm to compute $e(u, v)$.

2.1.2. Complexity Assumption

We prove the security of our scheme based on the CDH assumption. The assumption is defined as follows:

Definition 2.1 (CDH Assumption)**.** Let G be a cyclic multiplicative group generated by g and $a, b \in \mathbb{Z}_p$. Given g, g^a, g^b, there is no probabilistic polynomial time algorithm $\mathcal{A}$ has a non-negligible advantage to compute the value g^{ab}.

2.2. Intuition of Escrow-free

In the HIBOOS scheme, a user private key is generated by a domain PKG. Therefore, either the domain PKG or the user can sign a message to obtain a valid signature. Since the signature verifier cannot determine the actual signer, two problems should be addressed in those primitive HIBS scheme:

- *Key abusing problem.* A domain PKG is able to sign messages with the user keys generated by it without being detected;
- *User slandering problem.* The dishonest user is able to sign a message and slander that the PKG abuses its private key. That is, the undeniable property is missing in the primitive HIBS scheme.

For the key abusing problem, an intuitive solution is to limit the signing ability of the PKG by a well designed signing algorithm. Therefore, we use a user-selected secret apart from the private key to generate the signature for a message. We also compute a user public parameter with the secret as input. The user sends the parameter along with the message and signature to the receivers. Signature verifying needs to take user parameter and signature as input. Since the PKG cannot obtain the user secret, it cannot generate a valid signature with respect to the user parameter. However, the PKG can generate a well-formed signature with a fake user parameter. Receiver will thereby accept the PKG generated signature-parameter pair. To solve such problem, we introduce an Arbitral Party (AP) to keep the users' public parameters. User publishes its user parameter, and attaches these same parameter to each signature. A receiver is not constrained to compare the user parameter attaching in the signature with the one publishing in the AP. Nevertheless, the PKG will be detected and blamed once it abuses a user private key to sign messages. Note that, an AP does not keep any confidential contents.

Since PKG is able to generate well-formed signatures with distinct user parameters, a user can slander the PKG by signing messages with randomly picked secret and sending the receiver a corresponding fake user parameter along with the signature. To solve this problem, we modify the signing algorithm so that the user can only generate well-formed signatures with regard to the parameter it published. After publishing the public parameter, user also needs to ask for a PKG signing factor from the root PKG. The root PKG computes the factor with the user parameter as well as the master secret as input and returns the factor to the user. User is desired to sign messages with the PKG signing factor.

Combining these two technologies, we can present a full secure escrow-free HIBS model. Generally speaking, it can be applied to any secure HIBS schemes by modifying the signing and verifying algorithms. In this paper, we instantiate an escrow-free HIBOOS scheme with the above intuition. The construction and security proof are described in Section 3.

2.3. Generic Construction

A primitive hierarchical identity-based online/offline signature scheme generally consists of five algorithms: Setup, KeyGen, Offline Signing, Offline Signing and Verification. To achieve the escrow-free property, we add Publish algorithm and Blame algorithm to our HIBOOS scheme. The generic construction of our HIBOOS scheme is shown as in Figure 1. The scheme consists of the following 7 algorithms:

Setup. The setup algorithm takes a security parameter as input and outputs the HIBS public parameters.

KeyGen. The key generation algorithm takes a secret key and an identity ID as input and outputs the user private key. More specifically, the root PKG takes the master secret as input and can generate private keys for any user. Each domain PKG takes its private key as input and generates private keys for its descendant user.

Publish. The algorithm takes the user secret as input and outputs the user parameter. User uploads the parameter to the AP and get PKG signing factor from the root PKG.

Offline Signing. The offline signing algorithm is performed prior to obtaining the message. It takes a private key as well as the public parameter as input, and outputs the offline signature.

Online Signing. The online signing algorithm takes a message, a private key and the offline signature as input, and outputs a final signature.

Verification. The verification algorithm takes an identity ID, a message m and a signature as input. If σ is valid, outputs 1. Otherwise, outputs 0.

Blame.. The algorithm takes a message-signature pair $\{m, \sigma\}$ and a user parameter as input. If σ is generated by an honest signer, outputs 0. Otherwise, outputs 1.

2.4. Security Model

We claim that an secure escrow-free hierarchical identity-based online/offline signature scheme should be uncrackable against three attack games: *EF-sID-CMA*, *EKA-ID-CMA* and *EUS-sID-CMA*.

Definition 2.2 (*EF-sID-CMA* Security)**.** We say that a hierarchical identity-based online/offline signature scheme is secure if no probabilistic polynomial time (*PPT*) adversary $\mathcal{A}$ has a non-negligible advantage against the challenger $\mathcal{C}$ in the above *EF-sID-CMA* game. As shorthand, we say that the HIBS scheme is *EF-sID-CMA* secure.

Definition 2.3 (*EKA-ID-CMA* Security)**.** We say that a hierarchical identity-based online/offline signature scheme is secure if no *PPT* adversary $\mathcal{A}$ has a non-negligible advantage against the challenger $\mathcal{C}$ in the above *EKA-ID-CMA* game. As shorthand, we say that the HIBS scheme is *EKA-ID-CMA* secure.

Definition 2.4 (*EUS-sID-CMA* Security)**.** We say that a hierarchical identity-based online/offline signature scheme is secure if no *PPT* adversary $\mathcal{A}$ has a non-negligible advantage against the challenger $\mathcal{C}$ in the above *EUS-ID-CMA* game. As shorthand, we say that the HIBS scheme is *EUS-sID-CMA* secure.

Details of the security model can be found in our previous work [33].

3. ESCROW-FREE ONLINE/OFFLINE HIBS SCHEME

In this section, we present the construction of our escrow-free HIBOOS scheme, and prove the security of our scheme via there attack games.

3.1. Construction

Let K be the security parameter given to the setup algorithm, and let $\mathcal{G}$ be a BDH parameter generator.
Setup. Given a security parameter, the PKG works as follows:

1. runs $\mathcal{G}$ on input K to generate multiplicative groups G_1, G_2 of same prime order, and a bilinear pairing $\hat{e} : G_1 \times G_1 \rightarrow G_2$;
2. chooses random $\alpha \in \mathbb{Z}_p^*$ and two generators $g, g_2 \in G_1$, computes $g_1 = g^\alpha$;
3. randomly picks $h_1, \ldots, h_\ell \in G_1$;
4. chooses two cryptographic hash functions $H_1 : \{0,1\}^* \times G_1 \rightarrow \mathbb{Z}_p^*$ and $H_2 : G_1 \times \{0,1\}^* \rightarrow G_1$;
5. publishes $Param = \{\hat{e}, g, g_1, g_2, h_1, \ldots, h_\ell, H_1, H_2\}$ as public parameters and keeps $\mathrm{MSK} = g_2^\alpha$ as master secret.

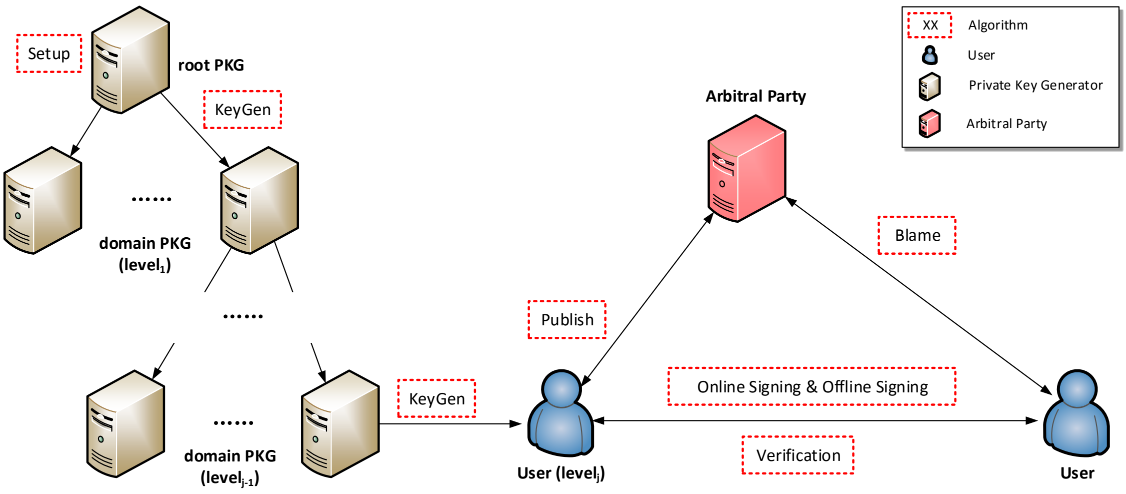

Figure 1. Generic construction of the HIBOOS scheme.

KeyGen. For an input $\mathrm{ID}=\{I_1,\ldots,I_k\}$, the $level_{k-1}$ domain PKG with private key $d_{\mathrm{ID}|_{k-1}}=\{d'_0,\ldots,d'_{k-1}\}$ generates the key d_{ID} as follows:

1. picks random $r_k\in\mathbb{Z}_p^*$;
2. set $d_{\mathrm{ID}}=\{d'_0F_k(I_k)^{r_k},d'_1,\ldots,d'_{k-1},g^{r_k}\}$, where $F_k(x)=g_1^x h_k$.

We refer to g_2^α as $d_{\mathrm{ID}|_0}$, and the user private key can be presented as $d_{\mathrm{ID}}=\{d_0,d_1,\ldots,d_k\}=\{g_2^\alpha\prod_{j=1}^k F_j(I_j)^{r_j},g^{r_1},\ldots,g^{r_k}\}$.

Publish. In this phase, user publishes a public parameter and gets PKG signing factor from the root PKG. It does the work as follows:

1. randomly picks $s_{\mathrm{ID}}\in\mathbb{Z}_p^*$ as user secret;
2. computes $g_{\mathrm{ID}}=g^{s_{\mathrm{ID}}}$;
3. publishes g_{ID} by submitting it to the AP;
4. sends g_{ID} to the root PKG, the root PKG computes and returns $f^\alpha=H_2(g_{\mathrm{ID}},\mathrm{ID})^\alpha$.

Although the root PKG has to computes the signing factor for all users, it brings in little overhead because of the following reasons:

- it has not to authenticate the users before computing the signing factors for them;
- it has not to maintain a secure channel to transmit the signing factor;
- it only needs to do an exponentiation computation for each user.

Offline Signing. During this phase, the signer computes the followings: To sign a message $m\in\{0,1\}^*$ with respect to identity $\mathrm{ID}=\{I_1,\ldots,I_k\}$, user takes the private key $d_{\mathrm{ID}}=\{d_0,d_1,\ldots,d_k\}$, secret s_{ID} and PKG signing factor f^α as input, running the signing algorithm as follows:

1. picks a random $s\in\mathbb{Z}_p^*$ and computes $x=g_2^s$;
2. for $j=1,\ldots,k$, computes $y_j=d_j^s$;
3. computes $f=H_2(g_{\mathrm{ID}},\mathrm{ID})$;
4. computes $E_{off}=\hat{e}(g_1,\prod_{j=1}^k d_j^{I_j})\prod_{j=1}^k\hat{e}(d_j,h_j)$;
5. computes $z_{off}=d_0^s f^{s_{\mathrm{ID}}}f^\alpha=d_0^s f^{s_{\mathrm{ID}}+\alpha}$;
6. sets the offline signature as $\sigma_{off}=\{x,y_1,\ldots,y_k,z_{off},E_{off},g_{\mathrm{ID}}\}$.

Online Signing. During this phase, the signer computes the followings:

1. computes $h=H_1(m,x)$;
2. computes $z=d_0^h z_{off}=d_0^{s+h}f^{(s_{\mathrm{ID}}+\alpha)}$;
3. computes $E=E_{off}^h$;
4. sets signature as $\sigma_{on}=\{z,E\}$.

The signature is $\sigma=\{x,y_1,\ldots,y_k,z,E,g_{\mathrm{ID}}\}$.

Verification. To verify $\sigma=\{x,y_1,\ldots,y_k,z,E,g_{\mathrm{ID}}\}$ on message m with respect to identity $\mathrm{ID}=\{I_1,\ldots,I_k\}$, the verification algorithm works as follows:

1. computes $h=H_1(m,x)$ and $f=H_2(g_{\mathrm{ID}},\mathrm{ID})$;
2. checks whether the equation holds: $\hat{e}(g,z)=E\cdot\hat{e}(g_1,g_2^h x\prod_{j=1}^k y_j^{I_j})\hat{e}(f,g_{\mathrm{ID}}g_1)\prod_{j=1}^k\hat{e}(y_j,h_j)$ If so, outputs 1. Otherwise, outputs 0.

Actually, if the signature is valid, there is

$$\begin{aligned}
&E \cdot \hat{e}(g_1, g_2^h x \prod_{j=1}^{k} y_j^{I_j})\hat{e}(f, g_{\mathrm{ID}} g_1)\prod_{j=1}^{k}\hat{e}(y_j, h_j)\\
=&\hat{e}(g_1, \prod_{j=1}^{k} d_j^{I_j})^h \left(\prod_{j=1}^{k}\hat{e}(d_j, h_j)^h\right)\hat{e}(g_1, g_2^{(s+h)}\prod_{j=1}^{k} y_j^{I_j})\cdot\\
&\hat{e}(f, g_{\mathrm{ID}} g_1)\prod_{j=1}^{k}\hat{e}(y_j, h_j)\\
=&\hat{e}(g_1, g_2^{(s+h)}\prod_{j=1}^{k} d_j^{(s+h)I_j})\hat{e}(f, g^{s_{\mathrm{ID}}} g^{\alpha})\prod_{j=1}^{k}\hat{e}(d_j^{(s+h)}, h_j)\\
=&\left(\hat{e}(g^{\alpha}, g_2 g^{\sum_{j=1}^{k} r_j I_j})\hat{e}(\prod_{j=1}^{k} g, h_j^{r_j})\right)^{(s+h)}\hat{e}(g, f^{s_{\mathrm{ID}}} f^{\alpha})\\
=&\hat{e}(g, g_2^{\alpha}\prod_{j=1}^{k} g_1^{r_j I_j} h_j^{r_j})^{(s+h)}\hat{e}(g, f^{(s_{\mathrm{ID}}+\alpha)})\\
=&\hat{e}(g, d_0^{(s+h)} f^{(s_{\mathrm{ID}}+\alpha)})\\
=&\hat{e}(g, z).
\end{aligned}$$

Note that, although the user public parameter is carried as a part of the signature, the receiver is not constrained to verify whether the signature is generated by the user by checking the parameter. That is, the receiver implicitly believes a message is signed by the sender if the signature is correctly verified. The sender himself has to run the *Blame* procedure if he argue that the signature with respect to his identity is bogus.

Blame. Given $\{\mathrm{ID}, m, \sigma = \{x, y_1, \ldots, y_k, z, g_{\mathrm{ID}}\}\}$, the algorithm requires the user parameter g'_{ID} with respect to the identity ID from the AP. It outputs 1 if and only if $g_{\mathrm{ID}} \neq g'_{\mathrm{ID}}$ and 0 otherwise.

3.2. Security Proof

The security of our escrow-free HIBOOS scheme is proved according to the following Lemmas.

Lemma 3.1
If there exists an EF-sID-CMA algorithm $\mathcal{A}$ that has non-negligible advantage against our HIBOOS scheme, then there exists an algorithm $\mathcal{B}$ that breaks the CDH assumption with non-negligible advantage.

Proof
We show how to construct algorithm $\mathcal{B}$ to win the *EF-sID-CMA* game against the CDH assumption. Given g, g^a, g^b, $\mathcal{B}$ interacts with $\mathcal{A}$ as follows:

Init. $\mathcal{A}$ selects an identity $\mathrm{ID}^* = \{I_1^*, \ldots, I_k^*\}(k \leqslant \ell)$ which will be used in the challenge phase.

Setup. $\mathcal{B}$ generates a bilinear map $\hat{e}: G_1 \times G_1 \to G_2$ with g as a generator of G_1. It randomly picks $\alpha_1, \ldots, \alpha_\ell \in \mathbb{Z}_p^*$, and sets $g_1 = g^a$, $g_2 = g^b$, $h_j = g^{-I_j^* \alpha_j}$ for $j = 1, \ldots, \ell$. Function $F_j : \mathbb{Z}_p \to G_1$ is defined as $F_j(x) = g_1^x h_j = g_1^{x - I_j^*} g^{\alpha_j}$. $\mathcal{B}$ also maintains hash oracle $H_1 : \{0,1\}^* \times G_1 \to \mathbb{Z}_p^*$. Parameters $\{\hat{e}, g, g_1, g_2, h_1, \ldots, h_\ell, H_1\}$ are sent to $\mathcal{A}$.

Query. $\mathcal{B}$ answers queries made by $\mathcal{A}$, where $\mathcal{A}$ is allowed to make up to q_1 hash queries, q_2 key queries and q_3 signing queries.

- *Hash queries.* $\mathcal{B}$ maintains lists L_1 and L_2 to store the answers of the H_1 and H_2 oracle, respectively.
  - $\mathcal{A}$ submits the H_1 hash query with input (m, x), $\mathcal{B}$ checks the list L. If an entry is found, the same answer is returned to $\mathcal{A}$; otherwise, $\mathcal{B}$ randomly picks $h \in \mathbb{Z}_p^*$ and returns it to $\mathcal{A}$. $\mathcal{B}$ stores (m, x, h) to the list L_1.
  - $\mathcal{A}$ submits the H_2 hash query with input $(g_{\mathrm{ID}}, \mathrm{ID})$, $\mathcal{B}$ checks the list L_2. If an entry for the query is found, the same answer will be returned to $\mathcal{A}$. Otherwise, if $\mathrm{ID} \neq \mathrm{ID}^*$, $\mathcal{B}$ randomly picks $\beta_i \in \mathbb{Z}_p^*$ and sets $f = H_2(g_{\mathrm{ID}}, \mathrm{ID}) = g^{\beta_i}$; if $\mathrm{ID} = \mathrm{ID}^*$, sets $f = g^b$. $\mathcal{B}$ stores $(g_{\mathrm{ID}}, \mathrm{ID}, \beta_i)$ to the list L_2.
- *Key queries.* $\mathcal{A}$ submits a private key query with input $\mathrm{ID} = \{I_1, \ldots, I_u\}(u \leqslant \ell)$, where $\mathrm{ID} \neq \mathrm{ID}^*$. Let j be the smallest index such that $I_j \neq I_j^*(1 \leqslant j \leqslant u)$, $\mathcal{B}$ randomly picks $r_1, \ldots, r_j \in \mathbb{Z}_p^*$ and sets

  $$d_0 = g_2^{\frac{-\alpha_j}{I_j - I_j^*}}\prod_{v=1}^{j} F_v(I_v)^{r_v},\ d_1 = g^{r_1},\ \ldots,$$

  $$d_{j-1} = g^{r_{j-1}},\ d_j = g_2^{\frac{-1}{I_j - I_j^*}} g^{r_j}.$$

  Note that, there is

  $$\begin{aligned}
  d_j = g_2^{\frac{-1}{I_j - I_j^*}} g^{r_j} =& g^{\frac{-b}{I_j - I_j^*}} g^{r_j} = g^{r_j - \frac{b}{I_j - I_j^*}},\\
  g_2^{\frac{-\alpha_j}{I_j - I_j^*}} F_j(I_j)^{r_j} =& g_2^{\frac{-\alpha_j}{I_j - I_j^*}}(g_1^{I_j - I_j^*} g^{\alpha_j})^{r_j}\\
  =& g_2^{\alpha}(g_1^{I_j - I_j^*} g^{\alpha_j})^{r_k - \frac{b}{I_j - I^* j}}\\
  =& g_2^{\alpha} F_j(I_j)^{r_j - \frac{b}{I_j - I_j^*}}.
  \end{aligned}$$

  Thus we can get the private key of $\{I_1, \ldots, I_j\}$:

  $$d_0 = g_2^{\alpha}\left(\prod_{v=1}^{j-1} F_v(I_v)^{r_v}\right) F_j(I_j)^{r_j - \frac{b}{I_j - I_j^*}},$$

  $$d_1 = g^{r_1}, \ldots, d_{j-1} = g^{r_{j-1}},\ d_j = g^{r_j - \frac{b}{I_j - I_j^*}},$$

  Therefore, the private key returned to $\mathcal{A}$ is well-formed.
  For each identity whose prefix equals $\{I_1, \ldots, I_j\}$, $\mathcal{B}$ can generate a private key with the key $d_{\mathrm{ID}|j}$.
- *Signing queries.* $\mathcal{A}$ submits a signing query with $\mathrm{ID} = \{I_1, \ldots, I_k\}$, g_{ID} and m as input, where $\mathrm{ID} \neq \mathrm{ID}^*$. If $\mathcal{B}$ does not have a private key d_{ID}

with respect to ID, it generates the private key by implementing the *KeyGen* algorithm. $\mathcal{B}$ signs the message m according to the *Offline* and *Online* algorithm, and returns $\mathcal{A}$ the signature.

Challenge. $\mathcal{A}$ finally outputs $(\mathrm{ID}^*, m^*, \sigma)$, where $\sigma = \{x, y_1, \ldots, y_k, z, E, g^*_{\mathrm{ID}}\}$. Since $\mathcal{A}$ can break our HIBOOS scheme against the *EF-sID-CMA* game, there is $\hat{e}(g, z) = E \cdot \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j^*}) \hat{e}(f, g^*_{\mathrm{ID}} g_1) \cdot \prod_{j=1}^k \hat{e}(y_j, h_j)$.
According to the principle of forking lemma [32], $\mathcal{B}$ can replay $\mathcal{A}$ with the same random tape but different choices of H_1. It then obtains two valid signatures $(x, y_1, \ldots, y_k, z, E, g^*_{\mathrm{ID}})$ and $(x, y_1, \ldots, y_k, \bar{z}, E, g^*_{\mathrm{ID}})$ on message m^* with respect to hash functions H_1 and $\bar{H}_1$ having different values $h \neq \bar{h}$ on (m^*, x), respectively.

Thus, there is

$$\begin{aligned} \frac{\hat{e}(g, z)}{\hat{e}(g, \bar{z})} &= \frac{\hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j})}{\hat{e}(g_1, g_2^{\bar{h}} x \prod_{j=1}^k y_j^{I_j})} \\ \Rightarrow \quad & \hat{e}(g, z\bar{z}^{-1}) = \hat{e}(g_1, g_2^{h-\bar{h}}) = \hat{e}(g^a, g^{b(h-\bar{h})}) \\ \Rightarrow \quad & z\bar{z}^{-1} = g^{ab(h-\bar{h})} \\ \Rightarrow \quad & g^{ab} = (z^* \bar{z}^{-1})^{1/(h-\bar{h})}. \end{aligned}$$

Therefore, $\mathcal{B}$ can breaks the CDH assumption. However, Joux and Nguyen have pointed out that the CDH problem in cyclic group is hard [31]. That is, our HIBOOS scheme is ES-sID-CMA secure. □

Lemma 3.2
If there exists an EKA-ID-CMA algorithm $\mathcal{A}$ that has non-negligible advantage against our HIBOOS scheme, then there is an EKA-ID-CMA algorithm $\mathcal{B}$ that breaks the CWS-HIBS scheme with the same advantage.

Proof
Supposing *EKA-sID-CMA* algorithm $\mathcal{A}$ can break our HIBOOS scheme, we show how to construct a PPT algorithm $\mathcal{B}$ to win the EF-sID-CMA game against the CWS-HIBS scheme. $\mathcal{B}$ is given parameters of CWS-HIBS scheme $parm = \{g, g_1, g_2, h_1, \ldots, h_\ell, H_1, H_2\}$, where $\hat{e} : G_1 \times G_1 \rightarrow G_2$ is a bilinear map, g, g_2 are generators of cyclic multiplicative group G_1, and H_1 is a hash function. As a simulator, $\mathcal{B}$ provides an *EF-sID-CMA* game to $\mathcal{A}$ and uses the final challenge information to break the CWS-HIBS scheme.
$\mathcal{A}$ play the game as below:

Setup. $\mathcal{B}$ obtains the public parameters $param = \{g, g_1, g_2, h_1, \ldots, h_\ell, H_1, H_2\}$ of the CWS-HIBS scheme, Parameters $param$ are sent to $\mathcal{A}$ as the parameters of our HIBOOS scheme.

Query. $\mathcal{B}$ answers queries made by $\mathcal{A}$.

- Key queries. $\mathcal{A}$ submits an identity ID. $\mathcal{B}$ maintains an ID-key list L. If the queried key d_{ID} is in the list, $\mathcal{B}$ returns d_{ID} to $\mathcal{A}$. Otherwise, $\mathcal{B}$ takes ID as the input and queries d_{ID} from the CWS-HIBS scheme, and returns d_{ID} to $\mathcal{A}$.
- Signing queries. $\mathcal{A}$ submits $\{\mathrm{ID}, m\}$. If $\mathcal{B}$ does not have a private key d_{ID} with respect to ID, it queries the private key from the CWS-HIBS scheme and picks a random $s_{\mathrm{ID}} \in \mathbb{Z}_p^*$. $\mathcal{B}$ inputs d_{ID} and s_{ID} to sign message m using the signing algorithm provided by our HIBOOS scheme, and gets signature $\sigma = \{x, y_1, \ldots, y_k, z, E\}$. Signature σ are sent to $\mathcal{A}$.

Challenge. $\mathcal{A}$ finally outputs $(\mathrm{ID}^*, m^*, \sigma)$, where $\sigma = \{x, y_1, \ldots, y_k, z, E, g^*_{\mathrm{ID}}\}$. According to the attack game, there exists a signature query of which the return values is $(\mathrm{ID}^*, m, \sigma)$ with same user public parameter, i.e. $g^*_{\mathrm{ID}} = g_{\mathrm{ID}}$. $\mathcal{B}$ retrieves the private key $d_{\mathrm{ID}^*} = \{d_0, \ldots, d_k\}$ of ID^* from the list L. Since $\mathcal{A}$ can break our HIBOOS scheme against the *EKA-sID-CMA* game, there is $\hat{e}(g, z) = E \cdot \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j^*}) \hat{e}(f, g^*_{\mathrm{ID}} g_1) \prod_{j=1}^k \hat{e}(y_j, h_j)$.

$\mathcal{B}$ computes $h = H_1(m, x)$, and let $y'_j = y_j d_j^h$, for $j = 1, \ldots, k$, $x' = x$, $z' = z$, there is

$$\begin{aligned} &\hat{e}(g, z') = \hat{e}(g, z) \\ &= E \cdot \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j^*}) \hat{e}(f, g^*_{\mathrm{ID}} g_1) \prod_{j=1}^k \hat{e}(y_j, h_j) \\ &= \hat{e}(g_1, \prod_{j=1}^k d_j^{hI_j^*}) \prod_{j=1}^k \hat{e}(d_j^h, h_j) \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j^*}) \cdot \\ &\quad \hat{e}(f, g^*_{\mathrm{ID}} g_1) \prod_{j=1}^k \hat{e}(y_j, h_j) \\ &= \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j^*} d_j^{hI_j^*}) \hat{e}(f, g^*_{\mathrm{ID}} g_1) \prod_{j=1}^k \hat{e}(y_j d_j^h, h_j) \\ &= \hat{e}(g_1, g_2^h x' \prod_{j=1}^k {y'}_j^{I_j^*}) \hat{e}(f, g^*_{\mathrm{ID}} g_1) \prod_{j=1}^k \hat{e}(y'_j, h_j). \end{aligned}$$

Thus, $\sigma' = \{x', y'_1, \ldots, y'_k, z', g^*_{\mathrm{ID}}\}$ is a valid challenge signature. That is, PPT algorithm $\mathcal{B}$ can break the CWS-HIBS scheme. □

Lemma 3.3
If there exists an EKA-ID-CMA algorithm $\mathcal{A}$ that has non-negligible advantage against our HIBS scheme, then there is an algorithm $\mathcal{B}$ that breaks the CDH assumption.

Proof
The lemma is proved in [33] (proof of *Lemma 3*). □

Lemma 3.4
If there exists an EUS-sID-CMA algorithm $\mathcal{A}$ that has non-negligible advantage against our HIBOOS scheme, then there is an algorithm $\mathcal{B}$ that breaks the CDH assumption.

Proof
Supposing *EUS-sID-CMA* algorithm $\mathcal{A}$ can break our

HIBS scheme, we show how to construct a PPT algorithm $\mathcal{B}$ to violate the CDH assumption. Given g, g^a, g^b, $\mathcal{B}$ interacts with $\mathcal{A}$ in a selective identity game as follows:

Init. $\mathcal{A}$ selects an identity $\mathrm{ID}^* = \{I_1^*, \ldots, I_k^*\} (k \leqslant \ell)$ which will be used in the challenge phase.

Setup. $\mathcal{B}$ generates a bilinear map $\hat{e} : G_1 \times G_1 \to G_2$ with g as a generator of G_1. It randomly picks $\alpha, \alpha_1, \ldots, \alpha_\ell \in \mathbb{Z}_p^*$, and sets $g_1 = g^a$, $g_2 = g^b$, $h_j = g^{\alpha_j}$ for $j = 1, \ldots, \ell$. Function $F_j : \mathbb{Z}_p \to G_1$ is defined as $F_j(x) = g_1^x h_j = g_1^x g^{\alpha_j}$. $\mathcal{B}$ also maintains hash oracles $H_1 : \{0,1\}^* \times G_1 \to \mathbb{Z}_p^*$ and $H_2 : G_1 \times \{0,1\}^* \to G_1$. Parameters $Param = \{\hat{e}, g, g_1, g_2, h_1, \ldots, h_\ell, H_1, H_2\}$ are sent to $\mathcal{A}$.

Query. $\mathcal{B}$ answers queries made by $\mathcal{A}$.

- *Hash queries.* $\mathcal{B}$ maintains lists L_1 and L_2 to store the answer of the H_1 oracle and H_2 oracle respectively.
  - $\mathcal{A}$ submits the i^{th} H_1 hash query with input $\{m, x\}$, $\mathcal{B}$ checks the list L_1. If an entry for the query is found, the same answer will be returned to $\mathcal{A}$; otherwise, $\mathcal{B}$ randomly picks $h \in \mathbb{Z}_p^*$ and returns to $\mathcal{A}$. $\mathcal{B}$ stores $\{m, x, h\}$ to the list L_1;
  - $\mathcal{A}$ submits the i^{th} H_2 hash query with input $\{g_{\mathrm{ID}}, \mathrm{ID}\}$, $\mathcal{B}$ checks the list L_2. If an entry for the query is found, the same answer will be returned to $\mathcal{A}$; otherwise, $\mathcal{B}$ randomly picks $\beta_i \in \mathbb{Z}_p^*$ and sets $f = H_2(g_{\mathrm{ID}}, \mathrm{ID}) = g^{\beta_i}$. $\mathcal{B}$ stores $\{g_{\mathrm{ID}}, \mathrm{ID}, \beta_i\}$ to the list L_2.
- *KeyGen queries.* $\mathcal{A}$ submits a private key query with input $\mathrm{ID} = \{I_1, \ldots, i_k\}$, $\mathcal{B}$ randomly picks $r_1, \ldots, r_k \in \mathbb{Z}_p^*$ and sets

  $$d_0 = g_2^{\frac{-\alpha_k}{I_k}} \prod_{j=1}^{k} F_j(I_j)^{r_j},\ d_1 = g^{r_1},\ \ldots,$$

  $$d_{k-1} = g^{r_{k-1}},\ d_k = g_2^{\frac{-1}{I_k}} g^{r_k}$$

  Note that, there is

  $$\begin{aligned} d_k &= g_2^{\frac{-1}{I_k}} g^{r_k} = g^{\frac{-b}{I_k}} g^{r_k} = g^{r_k - \frac{b}{I_k}}, \\ g_2^{\frac{-\alpha_k}{I_k}} F_k(I_k)^{r_k} &= g_2^{\frac{-\alpha_k}{I_k}} (g_1^{I_k} g^{\alpha_k})^{r_k} \\ &= g_2^{\alpha} (g_1^{I_k} g^{\alpha_k})^{r_k - \frac{b}{I_k}} \\ &= g_2^{\alpha} F_k(I_k)^{r_k - \frac{b}{I_k}}. \end{aligned}$$

  Thus we can get

  $$d_0 = g_2^{\alpha} \left(\prod_{j=1}^{k-1} F_j(I_j)^{r_j} \right) F_k(I_k)^{r_k - \frac{b}{I_k}},$$

  $$d_1 = g^{r_1},\ \ldots,\ d_{k-1} = g^{r_{k-1}},\ d_k = g^{r_k - \frac{b}{I_k}}.$$

  Therefore, the private key returned to $\mathcal{A}$ is well-formed.

  For each identity ID, $\mathcal{B}$ maintains a list L_3 to store the user key information. It randomly picks $s_{\mathrm{ID}} \in \mathbb{Z}_p^*$, and stores $(\mathrm{ID}, d_{\mathrm{ID}}, r_1, \ldots, r_k, s_{\mathrm{ID}})$ into the list L_3. Both the private key d_{ID} and user secret s_{ID} are returned to $\mathcal{A}$.
- *Signing queries.* $\mathcal{A}$ submits a signing query with input $\mathrm{ID} = \{I_1, \ldots, I_k\}$ and m. If $\mathcal{B}$ does not have a private key d_{ID} with respect to ID, it generates the private key by implementing the algorithm in *KeyGen queries*. With key d_{ID} as well as s_{ID}, $\mathcal{B}$ replies the signing query as below:
  - computes $g_{\mathrm{ID}} = g^{s_{\mathrm{ID}}}$;
  - queries the hash oracle H_2 to obtain β_i and sets $f = g^{\beta_i}$;
  - randomly picks $s \in \mathbb{Z}_p^*$, and sets $x = g_2^s$;
  - queries the hash oracle H_1 to obtain h;
  - sets $y_j = d_j^s$ for $j = 1, \ldots, k$;
  - sets $z = d_0^s f^{s_{\mathrm{ID}}+\alpha} = d_0^s g^{\beta_i(s_{\mathrm{ID}}+\alpha)}$
    $= d_0^s g_{\mathrm{ID}}^{\beta_i} g_1^{\beta_i}$;
  - sets $E = \hat{e}(g_1, \prod_{j=1}^k d_j^{hI_j}) \prod_{j=1}^k \hat{e}(d_j, h_j)^h$.

  Signature $\{x, y_1, \ldots, y_k, z, E, g_{\mathrm{ID}}\}$ is returned to $\mathcal{A}$.

Challenge. $\mathcal{A}$ finally outputs $(\mathrm{ID}^*, m^*, \sigma)$, where $\sigma = \{x, y_1, \ldots, y_k, z, E, g_{\mathrm{ID}}^*\}$ and the private key of ID^* has been queried during the *Query Phase*. $\mathcal{B}$ extracts the private key entry $(\mathrm{ID}^*, d_{\mathrm{ID}^*}, r_1, \ldots, r_k, s_{\mathrm{ID}'})$ from list L_3. Note that, since $\mathcal{A}$ can break our HIBS scheme against the *EUS-ID-CMA* game, there is $g'_{\mathrm{ID}} \neq g^*_{\mathrm{ID}}$, where $g'_{\mathrm{ID}} = g^{s_{\mathrm{ID}'}}$ is the extracted public parameter of ID^*.

Following the principle of forking lemma, $\mathcal{B}$ can replay $\mathcal{A}$ with the same random tape but different choices of H_1. It then obtains two valid signatures $\{x, y_1, \ldots, y_k, z, E, g_{\mathrm{ID}}^*\}$ and $\{x, y_1, \ldots, y_k, \bar{z}, \bar{E}, g'_{\mathrm{ID}}\}$ on message m^*.

Since $E = \hat{e}(g_1, \prod_{j=1}^k d_j^{hI_j}) \prod_{j=1}^k \hat{e}(d_j, h_j)^h$ and $h_j = g^{\alpha_j}$, we can calculate

$$E/\bar{E} = \hat{e}(g_1, \prod_{j=1}^{k} d_j^{(h-\bar{h})I_j}) \prod_{j=1}^{k} \hat{e}(d_j^{h-\bar{h}}, g^{\alpha_j}).$$

Thus, there is

$$\begin{aligned} &\hat{e}(g, z)/\hat{e}(g, \bar{z}) \\ =&\frac{E \cdot \hat{e}(g_1, g_2^h x \prod_{j=1}^k y_j^{I_j}) \hat{e}(f, g_{\mathrm{ID}}^* g_1) \prod_{j=1}^k \hat{e}(y_j, h_j)}{\bar{E} \cdot \hat{e}(g_1, g_2^{\bar{h}} x \prod_{j=1}^k {y_j}^{I_j}) \hat{e}(f', g_{\mathrm{ID}}^{\bar{*}} g_1) \prod_{j=1}^k \hat{e}(y_j, h_j)} \\ =&E/\bar{E} \cdot \hat{e}(g_1, g_2^{h/\bar{h}}) \cdot \frac{\hat{e}(g^{\beta_u}, g_{\mathrm{ID}}^* g_1)}{\hat{e}(g^{\beta_v}, g_{\mathrm{ID}}^{\bar{*}} g_1)} \\ =&\hat{e}(g_1, g_2^{h-\bar{h}} \prod_{j=1}^{k} d_j^{(h-\bar{h})I_j}) \hat{e}(g, (g_{\mathrm{ID}}^* g_1)^{\beta_u} (g_{\mathrm{ID}}^{\bar{*}} g_1)^{-\beta_v}) \cdot \\ &\prod_{j=1}^{k} \hat{e}(d_j^{h-\bar{h}}, g^{\alpha_j}) \end{aligned}$$

$$=\hat{e}(g, g^{ab(h-\bar{h})} \prod_{j=1}^{k} (g^{(h-\bar{h})r_j I_j})^a) \hat{e}(g, g_{\mathrm{ID}}^{*\beta_u} {g_{\mathrm{ID}}^{\bar{*}}}^{-\beta_v} g_1^{\beta_u-\beta_v}) \cdot$$

$$\hat{e}(\prod_{j=1}^{k} g^{(h-\bar{h})\alpha_j r_j}, g)$$

$$=\hat{e}(g, g^{ab(h-\bar{h})} g_{\mathrm{ID}}^{*\beta_u} {g_{\mathrm{ID}}^{\bar{*}}}^{-\beta_v} g^{a(\beta_u-\beta_v)} \prod_{j=1}^{k} (g^{aI_j} g^{\alpha_j})^{(h-\bar{h})r_j})$$

$$=\hat{e}(g, z^* \bar{z^*}^{-1}).$$

So $\mathcal{B}$ gets $z^* \bar{z^*}^{-1} = g^{ab(h-\bar{h})} g_{\mathrm{ID}}^{*\beta_u} {g_{\mathrm{ID}}^{\bar{*}}}^{-\beta_v} g^{a(\beta_u-\beta_v)} \cdot \prod_{j=1}^{k} (g^{aI_j} g^{\alpha_j})^{(h-\bar{h})r_j}$, and computes

$$g^{ab} = \left(\frac{z^* (g_{\mathrm{ID}}^{\bar{*}} g^a)^{\beta_v}}{\bar{z^*} \left(\prod_{j=1}^{k} (g^{aI_j} g^{\alpha_j})^{(h-\bar{h})r_j} \right) (g_{\mathrm{ID}}^{*} g^a)^{\beta_u}} \right)^{\frac{1}{h-\bar{h}}}.$$

Therefore, PPT algorithm $\mathcal{B}$ can break the CDH assumption. □

Theorem 3.5
Our construction of HIBOOS scheme possesses EF-sID-CMA and escrow-free under the CDH assumption in the random oracle model.

Proof
According to the *Lemma* 3.1 to 3.4, if CDH assumption holds, our HIBS scheme is secure against *EF-sID-CMA*, *EKA-ID-CMA* and *EUS-ID-CMA* attacks . Therefore, our escrow-free HIBOOS scheme is full secure. □

4. PERFORMANCE ANALYSIS

In this section, we analysis the computational costs of algorithms in our HIBOOS scheme, and make a theoretical comparison of computational cost between the scheme and some other schemes.

4.1. Computational Costs

Varying the identity lengths, Figure 2 plots a detailed view of the computational costs in terms of computation times. These timings were obtained through implementations of our HIBOOS scheme based on the PBC library [34]. Our implementation uses the curves and parameters defined in Section 2.1.1 and is written in C and compiled with GCC 4.8.4 in Ubuntu 14.04. We run the program at a Desktop, which equips Intel(R) Core Duo CPU i5-3550 at 3.30 GHz and 8G memory.

The results show that the computational costs of *Offline Signing* and *Verification* algorithms scale linearly as the identity lengths increase, while the costs of *KeyGen*, *Publish* and *Online Signing* algorithms are basically constants. Apparently, *Online Signing* is mush faster than *Offline Signing* algorithm. Let k be the length of an identity, evaluation shows that *Online Signing* is above 15 times faster than *Offline Signing* when $k = 2$, and rises to above 62 times when $k = 19$. Because a user performs offline signing beforehand and only executes the online signing when messages are ready, our online/offline model can effectively improve the signing efficiency.

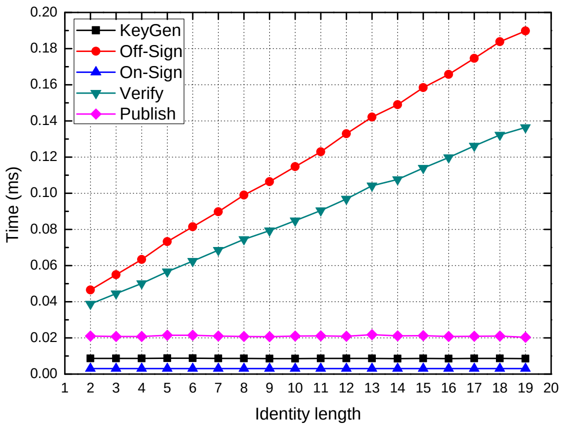

Figure 2. Algorithm computaional costs.

4.2. Scheme Comparison

We theoretically compare the performance of our HIBOOS scheme with other schemes in Table I. Online/offline signing algorithm computation, verifying algorithm computation are compared between the the schemes. The q denotes the order of group, k denotes the length of identity, E represents a group exponentiation computation and P represents a pairing computation. Since the overhead of hash computation and group point multiplicative computation are negligible comparing with the E and P computation, we don not take them into consideration in our scheme comparison.

Although introducing heavy computations, especially the three pairing computations in the offline singing phase, our HIBOOS scheme can achieve high online efficiency that the computation overhead is constant, which is also better than the K-IBOOS scheme. Our HIBOOS scheme extends the CWS-HIBOOS scheme [35]. The comparing results show that our escrow-free scheme introduces low extra overhead to the primitive HIBOOS scheme.

5. RELATED WORK

Even et al. first introduce the notion of online/offline signatures to reduce the computational cost of signature generation [16]. An OO signature scheme divides the process of message signing into offline phase and online phase. The offline phase with most of the heavy computations is performed prior to obtaining the message to be signed. And the online phase performing

Table I. Comparison of different schemes. H denotes hierarchical, OO denotes online/offline model, EF denotes escrow-free model. (* The computation overhead depends on the value a random $\beta \in \mathbb{Z}_q^*$, which up to $|q|-1$)

Scheme	H	OO	EF	Offline Sign Comp.	Online Sign Comp.	Verify Comp.
SHER-IBS [9]	√	×	×	–	$(k+2)E$	$(k+1)E+(k+2)P$
K-IBOOS [20]	×	√	×	$(\|q\|-1)P$	*	$1E+3P$
CWS-HIBOOS [35]	√	√	×	$(2k+2)E+2P$	$2E$	$(k+1)E+(k+2)P$
CWS-EF-HIBS [33]	√	×	√	–	$(k+3)E$	$(k+1)E+(k+3)P$
Our scheme	√	√	√	$(2k+3)E+(k+1)P$	$2E$	$(k+1)E+(k+2)P$

the light computations is executed when messages are ready. According to the OO signature model, several identity-based online/offline signature scheme have been proposed [17–20]. Liu et al. propose an online/offline IBS scheme (LJY-IBOOS) for securing wireless sensor network [19]. Yang et al. illuminate that the LJY-IBOOS scheme is not secure and propose an identity-based online/offline threshold signature scheme, which achieves higher security [22]. Both of their schemes are not appropriate for hierarchical IBS. Since each cryptosystem needs to be proved security utilizing formal mathematical methodology, it cannot directly extend a proved-secure IBS scheme to a secure HIBS scheme. Moreover, a secure HIBS scheme has to solve the domain-PKG collusion problem in addition.

Works on key escrow problem can be categorized into two types: IBC-applicable solutions [15, 23–26] and IBS-applicable solutions [27–29]. **IBC-applicable solution**. This type of solutions can be apply to solve the key escrow problems in both the (H)IBE and (H)IBS schemes. Most of the solutions impose extra organizations to distribute the ability of PKG so that a single PKG is not able to generate and obtain user private keys. Boneh et al. first apply the threshold method to suggest an multi-PKG mechanism [15]. Kate and Goldberg improve their model and apply the model to three well-known IBE schemes [23]. Besides, some researchers introduce Key Privacy Authorities (KPAs) to restrict the power of PKG [24–26]. However, either the multi-PKG or PKG-KPAs models brings significant overhead because of the extra identity authentication and the more complicated key generating algorithms. **IBS-applicable solution**. This type of solutions relies on the fact that the signer can attach extra information to the signature so that the message receivers can utilize the information along with the public key of the signer to verify the signature. Yuen et al. propose an escrow-free IBS model that each signer uses a public key and a secret key to sign messages [28]. Two signatures are generated and verified for one message. Besides, a judge and a Trusted Third Party are required in their model. Zhang et al. propose an escrow-free IBS scheme that unnecessarily depends on any judges [27]. The essence of their scheme is that a user-selected secret is added while generating the private key so that the PKG cannot obtain a complete key. However, either Yuen's model or Zhang's scheme is only for IBS schemes.

6. CONCLUSION

In this paper we propose a provably-secure escrow-free hierarchical Identity-based Online/Offline Signature scheme. The main idea of the escrow-free model lay on 1) imposing user-selected secret while signing message so as to restrict the PKGs' ability of generating an identical signature as user; and 2) introducing PKG signing factor to the signature so that the user could not generate well-formed signatures with different user parameter. According to the idea, we presented an escrow-free HIBOOS scheme based on the CWS-HIBOOS scheme. Based on the CDH assumption, we proved the security of our escrow-free HIBOOS scheme. Our scheme only introduced acceptable overhead to the CWS-HIBS scheme and our escrow-free model was flexible and compatible to the primitive schemes.

ACKNOWLEDGEMENT

This work was partially supported by National Science Foundation of China(NSFC) under grant no. 61303264; National Science Foundation of China(NSFC) under grant no. 61202488, and also in part supported by Soonchunhyang University Research Fund; the Basic Science Research Program through the National Research Foundation of Korea (NRF) funded by the Ministry of Science, ICT and Future Planning (2014R1A1A1005915).

REFERENCES

1. A. T. Campbell, S. B. Eisenman, N. D. Lane, E. Miluzzo, R. Peterson, H. Lu, X. Zheng, M. Musolesi, K. Fodor, G.-S. Ahn *et al.*, The rise of people-centric sensing, *Internet Computing, IEEE*, vol. 12, no. 4, pp. 12–21, 2008.
2. C. Cornelius, A. Kapadia, D. Kotz, D. Peebles, M. Shin, and N. Triandopoulos, Anonysense: privacy-aware people-centric sensing, *Proceedings of the 6th international conference on Mobile systems, applications, and services.* ACM, 2008, pp. 211–224.

DOI: 10.1002/sec
Prepared using secauth.cls

3. J. Shi, R. Zhang, Y. Liu, and Y. Zhang, Prisense: privacy-preserving data aggregation in people-centric urban sensing systems, *INFOCOM, 2010 Proceedings IEEE*. IEEE, 2010, pp. 1–9.
4. K. P. Puttaswamy and B. Y. Zhao, Preserving privacy in location-based mobile social applications, *Proceedings of the Eleventh Workshop on Mobile Computing Systems & Applications*. ACM, 2010, pp. 1–6.
5. P. Johnson, A. Kapadia, D. Kotz, N. Triandopoulos, and N. Hanover, People-centric urban sensing: Security challenges for the new paradigm, *Dept. of Computer Science, Dartmouth College. URL http://www. cs. dartmouth. edu/˜ dfk/papers/johnson-metrosec-challenges-tr. pdf.–Zugriffsdatum*, vol. 26, 2007.
6. A. Shamir, Identity-based cryptosystems and signature schemes, in *Advances in cryptology*. Springer, 1985, pp. 47–53.
7. J. C. Choon and J. H. Cheon, An identity-based signature from gap diffie-hellman groups, *Public key cryptographyPKC 2003*. Springer, 2002, pp. 18–30.
8. C. Gentry and A. Silverberg. Hierarchical id-based cryptography. In *Advances in cryptologyASIACRYPT 2002*, pages 548–566. Springer, 2002.
9. S. S. Chow, L. C. Hui, S. M. Yiu, and K. Chow, Secure hierarchical identity based signature and its application, *Information and Communications Security*. Springer, 2004, pp. 480–494.
10. D. Boneh, B. Lynn, and H. Shacham. Short signatures from the weil pairing. In *Advances in CryptologyASIACRYPT 2001*, pages 514–532. Springer, 2001.
11. J. Camenisch and A. Lysyanskaya. Signature schemes and anonymous credentials from bilinear maps. In *Advances in Cryptology–CRYPTO 2004*, pages 56–72. Springer, 2004.
12. D. Boneh and X. Boyen. Short signatures without random oracles. In *Advances in Cryptology-EUROCRYPT 2004*, pages 56–73. Springer, 2004.
13. Y. Yao, and Z. Li, A novel fuzzy identity based signature scheme based on the short integer solution problem, *Computers & Electrical Engineering*, vol. 40, no. 6, pp. 1930–1939, Elsevier 2014.
14. M. Gerbush, A. Lewko, A. ONeill, and B. Waters, Dual form signatures: An approach for proving security from static assumptions, *Advances in Cryptology–ASIACRYPT 2012*. Springer, 2012, pp. 25–42.
15. D. Boneh and M. Franklin, Identity-based encryption from the weil pairing, in *Advances in Cryptology-CRYPTO 2001*. Springer, 2001, pp. 213–229.
16. S. Even, O. Goldreich, and S. Micali, On-line/off-line digital signatures, in *Advances in CryptologyCRYPTO89 Proceedings*. Springer, 1990, pp. 263–275.
17. J. K. Liu, J. Baek, and J. Zhou, Online/offline identity-based signcryption revisited, *Information Security and Cryptology*. Springer, 2011, pp. 36–51.
18. R. Yasmin, E. Ritter, and G. Wang, An authentication framework for wireless sensor networks using identity-based signatures, *Computer and Information Technology (CIT), 2010 IEEE 10th International Conference on*. IEEE, 2010, pp. 882–889.
19. J. K. Liu, J. Baek, J. Zhou, Y. Yang, and J. W. Wong, Efficient online/offline identity-based signature for wireless sensor network, *International Journal of Information Security*, vol. 9, no. 4, pp. 287–296, 2010.
20. J. Kar, Provably secure online/off-line identity-based signature scheme for wireless sensor network.” *IJ Network Security*, vol. 16, no. 1, pp. 29–39, 2014.
21. J. Lai, Y. Mu, F. Guo, and W. Susilo. Improved Identity-Based Online/Offline Encryption, *Information Security and Privacy, Springer International Publishing*, pp. 160–173, 2015.
22. X. Yang, C. Li, T. Xu, and C. Wang, Id-based on-line/off-line threshold signature scheme without bilinear pairing, *Journal on Communications*, vol. 8, 2013.
23. A. Kate and I. Goldberg, Distributed private-key generators for identity-based cryptography, *Security and Cryptography for Networks*. Springer, 2010, pp. 436–453.
24. D. Cao, X.-F. Wang, F. Wang, Q.-L. Hu, and J.-S. Su, Sa-ibe: A secure and accountable identity-based encryption scheme, *Dianzi Yu Xinxi Xuebao(Journal of Electronics and Information Technology)*, vol. 33, no. 12, pp. 2922–2928, 2011.
25. B. Lee, C. Boyd, E. Dawson, K. Kim, J. Yang, and S. Yoo. Secure key issuing in id-based cryptography. In *Proceedings of the second workshop on Australasian information security, Data Mining and Web Intelligence, and Software Internationalisation-Volume 32*, pages 69–74. Australian Computer Society, Inc., 2004.
26. X. Wang, P. Chen, H. Zhou, and J. Su. T-hibe: A trustworthy and secure hierarchical identity-based encryption system [in press]. *Chinese Journal of Electronics*, page 0, 2015.
27. Y. Zhang, J. K. Liu, X. Huang, M. H. Au, and W. Susilo, Efficient escrow-free identity-based signature, *Provable Security*. Springer, 2012, pp. 161–174.
28. T. H. Yuen, W. Susilo, and Y. Mu, How to construct identity-based signatures without the key escrow problem, *International Journal of Information Security*, vol. 9, no. 4, pp. 297–311, 2010.
29. M. R. OgielaU. Ogiela, Linguistic protocols for secure information management and sharing, *Computers & Mathematics with Applications*, vol. 63,

no. 2, pp. 564–572, 2012. Elsevier, 2012.

30. P. Chen, X. Wang, S. Hao, and J. Su, An escrow-free hierarchical identity-based signature scheme from composite order bilinear groups [in press], *The Fifth International Workshop on Cloud, Wireless and e-Commerce Security (CWECS 2015)*. IEEE, 2015.
31. A. Joux and K. Nguyen, Separating decision diffie–hellman from computational diffie–hellman in cryptographic groups, *Journal of cryptology*, vol. 16, no. 4, pp. 239–247, 2003.
32. D. Pointcheval and J. Stern, Security proofs for signature schemes, in *Advances in CryptologyEUROCRYPT96*. Springer, 1996, pp. 387–398.
33. P. Chen, X. Wang, and J. Su, An Escrow-Free Hierarchical Identity-Based Signature Model for Cloud Storage, *Algorithms and Architectures for Parallel Processing*, Springer International Publishing, pp. 633–647, 2015. Springer, 2015.
34. B. Lynn, The Pairing-Based Cryptography Library. Available at https://crypto.stanford.edu/pbc/
35. P. Chen, X. Wang, and J. Su, An Online/Offline HIBS Scheme for Privacy Protection of People-Centric Sensing [in press], *2015 International Conference on Industrial Informatics –Computing Technology, Intelligent Technology, Industrial Information Integration (ICIICII 2015)*. IEEE, 2015.