ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.19, No.6, June 2008, pp.1555−1564
DOI: 10.3724/SP.J.1001.2008.01555

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

标准模型下基于身份的强密钥隔离签名*

翁 健[1,2+], 陈克非[1], 刘胜利[1] 李祥学[3]

[1](上海交通大学 计算机科学与工程系,上海 200240)

[2](暨南大学 计算机科学系,广东 广州 510632)

[3](上海交通大学 信息安全工程学院,上海 200240)

Identity-Based Strong Key-Insulated Signature Without Random Oracles

WENG Jian[1,2+], CHEN Ke-Fei[1], LIU Sheng-Li[1], LI Xiang-Xue[3]

[1](Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

[2](Department of Computer Science, Jinan University, Guangzhou 510632, China)

[3](School of Information Security Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

+ Corresponding author: E-mail: jianweng@sjtu.edu.cn

Weng J, Chen KF, Liu SL, Li XX. Identity-Based strong key-insulated signature without random oracles. *Journal of Software*, 2008,19(6):1555−1564. http://www.jos.org.cn/1000-9825/19/1555.htm

Abstract: It is a worthwhile challenge to deal with the key-exposure problem in identity-based signatures. To deal with this problem, this paper adopts Dodis, *et al*.'s key-insulation mechanism to identity-based signature scenarios, and proposes an identity-based key-insulated signature scheme. The proposed scheme enjoys two attractive features: (i) it is strong key-insulated; (ii) it is provably secure without random oracles.

Key words: key-insulated; identity-based signature; key-exposure; standard model

摘 要: 如何应对基于身份的签名系统中密钥泄漏的问题,是一项非常有意义的工作.为了处理这一问题,利用Dodis等人的密钥隔离机制,提出了一种基于身份的密钥隔离签名.所提出出的签名方案具有两个显著的特点:(i) 满足强密钥隔离安全性;(ii) 其安全性证明无须借助随机预言机模型.

关键词: 密钥隔离;基于身份的签名;密钥泄漏;标准模型

中图法分类号: TP309 **文献标识码**: A

1 Introduction

In 1984, Shamir[1] introduced an innovative concept called identity-based cryptography. In such a cryptosystem, user's public key is determined as his identity such as e-mail address, while the corresponding secret key is generated by a private key generator (PKG) according to this identity. Since the identity is a natural link to a user, there is no need to bind it by a digital certificate. Thus it can successfully eliminate the need for certificates as

* Supported by the National Natural Science Foundation of China under Grant Nos.60573030, 60673077 (国家自然科学基金)
Received 2006-10-09; Accepted 2006-12-28

used in traditional public key infrastructures. So far, a large number of identity-based signature (IBS) schemes have been proposed. Standard IBS schemes rely on the assumption that secret keys are kept perfectly secure. However, as more and more cryptographic primitives are applied to insecure environments (e.g. mobile devices), the problem of key-exposure seems inevitable. This problem is perhaps the most devastating attack on a cryptosystem, since it typically means that security is entirely lost.

To deal with the key-exposure problem, key-evolving protocols have been introduced. This mechanism includes forward security[2,3], intrusion-resilience[4] and key-insulation[5]. The latter was introduced by Dodis, *et al*.[5] in Eurocrypt'02. In this model, a physically-secure but computationally-limited device, named the base or helper, is involved. The full-fledged secret key is divided into two parts: a helper key and an initial temporary secret key. The former is stored in the helper, and the latter is kept by the user. The lifetime of the system is divided into discrete periods. The public key remains unchanged throughout the lifetime, while temporary secret keys are updated periodically: at the beginning of each period, the user obtains from the helper an update key for the current period; combining this update key with the temporary secret key for the previous period, he can derive the temporary secret key for the current period. A temporary secret key is used to sign a message during the corresponding period without further access to the physically secure device. Exposure of the temporary secret key at a given period will not enable an adversary to derive temporary secret keys for the remaining periods. Therefore, this mechanism can minimize the damage caused by key-exposure. More precisely, in a (*l*,*N*)-*key-insulated* scheme, the compromise of temporary secret keys for up to *l* periods does not expose temporary secret keys for any of the remaining *N*–*l* periods. Therefore, the public key needs not to be revoked unless up to *l* periods have been exposed. A scheme is called *perfectly key-insulated* if it is (*N*–1,*N*)-key-insulated. This is a desirable property for dealing with the key-exposure problem in ID-based cryptosystems. Additionally, *strong key-insulated* security guarantees that the helper (or an attacker compromising the helper key) is unable to derive the temporary secret key for any period. This is an extremely important property if the helper serves several different users or the helper is untrustworthy.

Following the pioneering work due to Dodis, *et al*.[5], several key-insulated encryption schemes including some ID-based key-insulated encryption ones have been proposed[6–11]. Following Dodis, *et al*.'s first key-insulated signature schemes[12], efforts have also been devoted to the key-insulated signatures, e.g. Ref.[13–16]. To minimize the damage caused by key-exposure in IBS scenarios, Zhou, *et al*.[17] applied the key-insulation mechanism to IBS and proposed the first ID-based key-insulated signature (IBKIS) scheme (ZCC scheme). However, the full-fledged secret key in ZCC scheme is just wholly stored in the helper. Consequently, it can not satisfy the strong key-insulated security. That is, if an adversary compromises a user's helper, he can derive all the temporary secret keys of this user. Moreover, ZCC scheme is provably secure in the random oracle model. As pointed out in Ref.[18], a proof in the random oracle model can only serve as a heuristic argument since it can not imply the security in the real world.

In this paper, we re-formalize the definition and security notions for IBKIS schemes, and then propose a new IBKIS scheme which is strongly key-insulated and provably secure without random oracles. The rest of this paper is organized as follows: Section 2 gives an introduction to bilinear pairings and the computational Diffie-Hellman (CDH) assumption. We re-formalize the definition and security notions for IBKIS schemes in Section 3. Our new IBKIS scheme is proposed in Section 4. In Section 5, we prove the security of our scheme in the standard model. Section 6 concludes this paper.

2 Preliminaries

2.1 Bilinear pairings

Let G_1 be a cyclic multiplicative group of prime order q, and G_2 be a cyclic multiplicative group of the same order q. A bilinear pairing is a map $\hat{e}: G_1 \times G_1 \to G_2$ with the following properties:

- Bilinearity: $\forall g_1, g_2 \in G_1, \forall a, b \in Z_q^*$, we have $\hat{e}(g_1^a, g_2^b) = \hat{e}(g_1, g_2)^{ab}$;
- Non-Degeneracy: There exist g_1, $g_2 \in G_1$ such that $\hat{e}(g_1, g_2) \neq 1$;
- Computability: There exists an efficient algorithm to compute $\hat{e}(g_1, g_2)$ for $\forall g_1$, $g_2 \in G_1$.

As shown in Ref.[19], such non-degenerate admissible maps over cyclic groups can be obtained from the Weil or Tate pairing over supersingular elliptic curves or Abelian varieties.

2.2 Computational Diffie-Hellman assumption

Definition 1. The CDH problem in group G_1 is, given $(g, g^a, g^b) \in G_1^3$ for some unknown $a, b \in_R Z_q^*$, to compute $g^{ab} \in G_1$. For a probabilistic polynomial-time (PPT) adversary A, we define his *advantage* against the CDH problem in group G_1 as $Adv_{A,G_1}^{CDH} = \Pr[g \in_R G_1, a, b \in Z_q^{\wedge} : A(g, g^a, g^b) = g^{ab}]$, where the probability is taken over the random coins consumed by A.

Definition 2. We say that the (t,ε)-CDH assumption holds in group G_1, if no t-time adversary A has advantage at least ε in solving the CDH problem in G_1.

3 Framework of ID-Based Key-Insulated Signature

3.1 ID-Based key-insulated signature

Definition 3. An IBKIS scheme is a tuple of six polynomial-time algorithms:

- *Setup*(k,N): The setup algorithm, taking as input a security parameter k and (possibly) a total number of periods N, returns a public parameter *para* and a master key *msk*. We write (*msk*,*para*)←*Setup*(k,N).
- *Extract*(*msk*,*para*,*ID*): The key extraction algorithm, taking as input the master key *msk*, the public parameter *para* and a user's identity *ID*, returns this user's initial temporary secret key $TSK_{ID,0}$ and helper key HK_{ID}. We write ($TSK_{ID,0}$,HK_{ID})←*Extract*(*msk*,*para*,*ID*).
- *UpdH*(t_1,t_2,*ID*,HK_{ID}): The key-update algorithm performed by the helper, taking as input two period indices t_1 and t_2, a user's identity *ID* and a helper key HK_{ID}, returns an update key UK_{ID,t_1,t_2}. We write $UK_{ID,t_1,t_2} \leftarrow UpdH(t_1, t_2, ID, HK_{ID})$.
- $UpdS(t_1, ID, UK_{ID,t_1,t_2}, TSK_{ID,t_2})$: The key-update algorithm performed by the user, taking as input a period index t_1, a signer's identity *ID*, a temporary secret key TSK_{ID,t_2} and an update key UK_{ID,t_1,t_2}, returns the temporary secret key TSK_{ID,t_1}. We write $TSK_{ID,t_1} \leftarrow UpdS(t_1, ID, UK_{ID,t_1}, TSK_{ID,t_1,t_2})$.
- *Sign*(t,m,$TSK_{ID,t}$): The signing algorithm, taking as input a period index t, a message m and the temporary secret key $TSK_{ID,t}$, returns a pair (t,σ) composed of the period index t and a signature σ. We write (t,σ)=*Sign*(t,m,$TSK_{ID,t}$).
- *Verify*((t,σ),m,*ID*)←*Sign*(t,m,$TSK_{ID,t}$): The verification algorithm taking as input a message m, a candidate signature (t,σ) on m and the signer's identity *ID*, returns 1 if (t,σ) is a valid signature, and 0 otherwise.

Consistency requires that $\forall t \in \{1,\ldots,N\}$, $\forall m \in M$, $\forall ID \in \{0,1\}^*$, *Verify*((t,σ),m,*ID*)=1, where (t,σ)=*Sign*(t,m, $TSK_{ID,t}$) and M denotes the message space.

Note that there exist only five algorithms in Zhou, *et al.*'s definition[17] for IBKIS. In fact, their definition does

not include the key-update algorithm of the signer, and the full-fledged secret key simply acts as the helper key and is wholly stored in the helper. Obviously, schemes satisfying their definition can not achieve strong key-insulated security.

3.2 Security notions for IBKIS

Dodis, *et al*.[12] formalized the security notions of *key-insulation*, *strong key-insulation* and *secure key-updates* for key-insulated signatures. In this section, we also formalize these security notions for IBKIS schemes. Note that Zhou, *et al*.[17] did not consider notions of strong key-insulation and secure key-updates.

Before giving these security notions for IBKIS schemes, we consider the following oracles which together model the abilities of an adversary:

- Key-Extraction oracle $KEO(\cdot)$: On input a user's identity ID, it returns this user's initial temporary secret key $TSK_{ID,0}$ and his helper key HK_{ID};
- Helper key oracle $HKO(\cdot)$: On inputting a user's identity ID, it returns his helper key HK_{ID};
- Temporary secret key oracle $TKO(\cdot,\cdot)$: Upon receiving a tuple $\langle ID,t\rangle$ consisting of a user's identity ID and a period index t, it returns the user's temporary secret key $TSK_{ID,t}$;
- Signing oracle $SO(\cdot,\cdot,\cdot)$: upon receiving a tuple $\langle ID,t,m\rangle$ consisting of a signer's identity ID, a period index t, and a message m, it returns a signature $Sign(t,m,TSK_{ID,t})$.

Definition 4. Let $\Pi=(Setup,Extract,UpdH,UpdS,Sign,Verify)$ be an IBKIS scheme. We define the *advantage* of an adversary A as

$$Adv^{KI}_{A,\Pi}(k)=[(msk,para)\leftarrow Setup(k,N);((t^*,\sigma^*),m^*,ID^*)\leftarrow A^{KEO(\cdot),TKO(\cdot,\cdot),SO(\cdot,\cdot,\cdot)}(para):Verify((t^*,\sigma^*),m^*,ID^*)=1],$$

where it is mandated that: (1) ID^* was not submitted to oracle $KEO(\cdot)$; (2) $\langle ID^*,t^*\rangle$ was not submitted to oracle $TKO(\cdot,\cdot)$; (3) $\langle ID^*,t^*,m^*\rangle$ was not submitted to oracle $SO(\cdot,\cdot,\cdot)$. We say that Π is *perfectly key-insulated* if for any PPT adversary A, $Adv^{KI}_{A,\Pi}(k)$ is negligible.

Remark 1. For those non-challenged identities, oracle $TKO(\cdot,\cdot)$ is of no help for adversary A, since he can derive any temporary secret key for these identities by querying oracle $KEO(\cdot)$. Therefore, without loss of generality, we require that adversary A only query oracle $TKO(\cdot,\cdot)$ on the challenged identity.

It is possible for an adversary to compromise the physically-secure helpers (this includes the attacks by the helpers themselves, in case they are untrustworthy). Zhou, *et al*.[17] did not address this kind of attack. Here, we model this attack by allowing the adversary to query oracle $HKO(\cdot)$ on any identity (even including the challenged identity). However, as in Ref.[12], the adversary is prohibited to query oracle $TKO(\cdot,\cdot)$ on the challenged identity for any period. Moreover, since oracle $TKO(\cdot,\cdot)$ is of no help for those non-challenged identities, we do not provide it for adversary A in the following definition.

Definition 5. Let $\Pi=(Setup,Extract,UpdH,UpdS,Sign,Verify)$ be an IBKIS scheme. We define the *advantage* of an adversary A as

$$Adv^{SKI}_{A,\Pi}(k)=[(msk,para)\leftarrow Setup(k,N);((t^*,\sigma^*),m^*,ID^*)\leftarrow A^{KEO(\cdot),HKO(\cdot),SO(\cdot,\cdot,\cdot)}(para):Verify((t^*,\sigma^*),m^*,ID^*)=1],$$

where it is mandated that: (1) ID^* was not submitted to oracle $KEO(\cdot)$; (2) $\langle ID^*,t^*,m^*\rangle$ was not submitted to oracle $SO(\cdot,\cdot,\cdot)$. We say that Π is *strong key-insulated* if for any PPT adversary A, $Adv^{KI}_{A,\Pi}(k)$ is negligible.

Finally, as in Ref.[12], we address an adversary who compromises the user's storage while a key is being updated from TSK_{ID,t_2} to TSK_{ID,t_1}, and we call it a key-update exposure at (ID,t_2,t_1). When this occurs, the adversary gets $TSK_{ID,t_2},UK_{ID,t_1,t_2}$ and TSK_{ID,t_1} (actually, the latter can be computed from the formers).

Definition 6. An IBKIS scheme *has secure key-updates* if the view of any adversary A making a key-update exposure at (ID,t_2,t_1) can be perfectly simulated by an adversary A' issuing oracle $TKO(\cdot,\cdot)$ queries on $\langle ID,t_2\rangle$ and

$\langle ID,t_1\rangle$.

4 Our Proposed Scheme

Based on Paterson-Schuldt's IBS scheme[20], which is based on Water's ID-based encryption scheme[21], we propose a new IBKIS scheme in this section.

4.1 Construction

Let G_1 and G_2 be two groups with prime order q of size k, g be a random generator of G_1, and $\hat{e}$ be a bilinear map such that $\hat{e}: G_1 \times G_1 \to G_2$. Let $H_1: \{0,1\}^* \to \{0,1\}^{n_u}$ and $H_1: \{0,1\}^* \to \{0,1\}^{n_m}$ be two collision-resistant hash functions for some n_u, $n_m \in Z$. Let F be a pseudo random function (PRF)[22] such that given a k-bit seed s and a k-bit argument x, it outputs a k-bit string $F_s(x)$. The proposed IBKIS scheme consists of the following six algorithms:

Setup: Given a security parameter k, PKG first picks $\alpha \in_R Z_q^*, g_2 \in_R G_1$ and defines $g_1=g^{\alpha}$. It then chooses u', $m' \in_R G_1$ and two vectors such that

$$\vec{U}=(\hat{u}_i) \text{ with } \hat{u}_i \in_R G_1 \text{ for } i=1,...,n_u;\ \vec{M}=(\hat{m}_j) \text{ with } \hat{m}_j \in_R G_1 \text{ for } j=1,...,n_m.$$

For easy explanation, we define two functions L_1 and L_2 such that

$$L_1(S)=u'\prod_{i\in S}\hat{u}_i \text{ for any set } S \subseteq \{1,...,n_u\};\ L_2(S')=m'\prod_{j\in S'}\hat{m}_j \text{ for any set } S' \subseteq \{1,...,n_m\}.$$

Then the master key is $msk=g_2^{\alpha}$ and the public parameters is

$$para=(G_1,G_2,\hat{e},q,g,g_1,g_2,u',m',\vec{U},\vec{M},H_1,H_2,L_1,L_2).$$

To make the notation easy to follow, hereafter, we use $U_{ID,t}, U'_{ID}$ and M_m to denote the following sets for a given identity ID, a period index t and a message m as follows.

$$\begin{cases} U_{ID,t}=\{i \mid S_1[i]=1,\ S_1=H_1(ID,t)\} \subseteq \{1,...,n_u\}, \\ U'_{ID}=\{j \mid S_2[j]=1,\ S_2=H_1(ID)\} \subseteq \{1,...,n_u\}, \\ M_m=\{k \mid S_3[k]=1,\ S_3=H_2(m)\} \subseteq \{1,...,n_m\}. \end{cases}$$

Extract: Given an identity ID, the PKG first randomly chooses a helper key $HK_{ID} \in_R \{0,1\}^k$ and Computes $k_{ID,0}=F_{HK_{ID}}(0 \| ID)$. Note that if the length of the input for F is less than k, we can add some "0"s as the prefix to meet the length requirement. Then PKG chooses $r \in_R Z_q^*$ and defines the initial temporary secret key as

$$TSK_{ID,0}=(g_2^{\alpha}L_1(U'_{ID})^r L_1(U_{ID,0})^{k_{ID,0}}, g^{k_{ID,0}}, g^r) \quad (1)$$

UpdH: Given an identity ID and two period indices t_1, t_2, the helper for user ID first computes $k_{ID,t_1}=F_{HK_{ID}}(t_1 \| ID)$ and $k_{ID,t_2}=F_{HK_{ID}}(t_2 \| ID)$, then defines and returns the update key as

$$UK_{ID,t_1,t_2}=\left(\frac{L_1(U_{ID,t_1})^{k_{ID,t_1}}}{L_1(U_{ID,t_2})^{k_{ID,t_2}}}, g^{k_{ID,t_1}}\right).$$

UpdS: Given a period index t_1, an update key $UK_{ID,t_1,t_2}=(\hat{U}_{ID,t_1,t_2}, \hat{R}_{ID,t_1})$ and a temporary secret key $TSK_{ID,t_2}=(U_{ID,t_2}, R_{ID,t_2}, R)$, the temporary secret key for user ID in period t_1 can be computed as

$$TSK_{ID,t_1}=(U_{ID,t_2}\cdot\hat{U}_{ID,t_1,t_2}, \hat{R}_{ID,t_1}, R).$$

Note that for a given identity ID and a given period index t, the corresponding temporary secret key is always set to

$$TSK_{ID,t}=(g_2^{\alpha}L_1(U'_{ID})^r L_1(U_{ID,t})^{k_{ID,t}}, g^{k_{ID,t}}, g^r) \quad (2)$$

where $k_{ID,t}=F_{HK_{ID}}(t \| ID)$.

Sign: in period t, the signer ID with temporary secret key $TSK_{ID,t}=(U_{ID,t},R_{ID,t},R)$ can produce the signature on

message m as follows: Choose $r'_t, r_m \in_R Z_q^*$, and then compute the signature as

$$\sigma = (t, U_{ID,t} \cdot L_1(U_{ID,t})^{r'_t} \cdot L_2(M_m)^{r_m}, R_{ID,t} \cdot g^{r'_t}, R, g^{r_m}) \quad (3)$$

Note that let $r_t = k_{ID,t} + r'_t$ with $k_{ID,t} = F_{HK_{ID}}(t \| ID)$, the signature is always set to be

$$\sigma = (t, g_2^{\alpha} \cdot L_1(U'_{ID})^{r} \cdot L_1(U_{ID,t})^{r_t} \cdot L_2(M_m)^{r_m}, g^{r_t}, g^{r}, g^{r_m}) \quad (4)$$

Verify: Given a purported signature $\sigma=(t,V,R_t,R,R_m)$ on an identity ID and a message m, a verifier accepts σ iff. the following equality holds

$$\hat{e}(g,V) = \hat{e}(g_1,g_2)\hat{e}(R, L_1(U'_{ID}))\hat{e}(R_t, L_1(U_{ID,t}))\hat{e}(R_m, L_2(M_m)) \quad (5)$$

4.2 Correctness

The consistency of this scheme can be explained as follows:

$$\begin{aligned}\hat{e}(g,V) &= \hat{e}(g, g_2^{\alpha} L_1(U'_{ID})^{r} \cdot L_1(U_{ID,t})^{r_t} \cdot L_2(M_m)^{r_m}) \\ &= \hat{e}(g, g_2^{\alpha}) \cdot \hat{e}(g, L_1(U'_{ID})^{r})\hat{e} \cdot (g, L_1(U_{ID,t})^{r_t}) \cdot \hat{e}(g, L_2(M_m)^{r_m}) \\ &= \hat{e}(g^{\alpha}, g_2) \cdot \hat{e}(g^{r}, L_1(U'_{ID}))\hat{e} \cdot (g^{r_t}, L_1(U_{ID,t})) \cdot \hat{e}(g^{r_m}, L_2(M_m)) \\ &= \hat{e}(g_1,g_2)\hat{e}(R, L_1(U'_{ID}))\hat{e}(R_t, L_1(U_{ID,t}))\hat{e}(R_m, L_2(M_m)).\end{aligned}$$

4.3 Desirable properties

Our scheme supports unbounded number of time periods[6], i.e., the the total number of periods, say N, is not involved in algorithm Setup. Algorithms *UpdH* and *UpdS* further show that our scheme supports random-access key-updates[6], since one can update TSK_{ID,t_2} to TSK_{ID,t_1} in one "step" for any time period indices t_1, t_2. In Section 5, we will prove that our scheme is perfectly key-insulated, strong key-insulated and has secure key-updates.

5 Security Analysis

In this section, to support our scheme, we will give its provable security in the standard model.

Theorem 1. The proposed scheme is perfectly key-insulated in the standard model, assuming that (1) the CDH assumption holds in G_1; (2) the hash function H is collision-resistant; (3) the function F is a pseudo random function.

Proof: Without loss of generality, we assume that the hash function H is collision-resistant and the function F is a pseudo random function, then given an adversary A that has advantage ε against the perfectly key-insulated security of our proposed scheme by running in time T, asking at most q_k, q_t and q_s queries to oracles $KEO(\cdot)$, $TKO(\cdot,\cdot)$ and $SO(\cdot,\cdot,\cdot)$ respectively, there exists a (t',ε') adversary B against the CDH assumption in G_1 with

$$\begin{cases} T' \le T + O((q_k + q_t + q_s)t_e + (n_u(q_k + q_t) + (n_u + n_m)q_s)t_m) \\ \varepsilon' \ge \dfrac{\varepsilon}{27(n_u+1)^2(n_m+1)(q_k + q_t + 2q_s)^2 q_s} \end{cases},$$

where t_e and t_m denote the running time of an exponentiation and a multiplication in G_1 respectively.

We will show how to construct a (t',ε')-adversary B against the CDH assumption in group G_1. On inputting $(g, g^a, g^b) \in G_1^3$ for some unknown $a,b \in_R Z_q^*$, B's goal is to compute g^{ab}. B plays the role of A's challenger and works by interacting with A in a game defined as follows:

Setup: B first sets $l_u = \dfrac{3(q_k + q_t + 2q_s)}{2}$, $l_m = 2q_s$. Here we assume $l_u(n_u+1)<q$ and $l_m(n_m+1)<q$. Next it randomly chooses two integers k_u and k_m with $0 \le k_u \le n_u$ and $0 \le k_m \le n_m$. Besides, the following integers are chosen:

$$x' \in_R Z_{l_u},\ z' \in_R Z_{l_m},\ y', w' \in_R Z_q,\ \{\hat{x}_i \in_R Z_{l_u}\}_{i=1,\ldots,n_u},\ \{\hat{z}_j \in_R Z_{l_m}\}_{j=1,\ldots,n_m},\ \{\hat{y}_i \in_R Z_q\}_{i=1,\ldots,n_u},\ \{\hat{w}_i \in_R Z_q\}_{i=1,\ldots,n_m}.$$

Then a set of public parameters defined below are passed to A

$$\begin{cases} g_1 = g^a,\ g_2 = g^b,\ u' = g_2^{x'-l_u k_u} g^{y'},\ m' = g_2^{z'-l_m k_m} g^{w'}. \\ \vec{U} = (\hat{u}_i) \text{ with } \hat{u}_i = g_2^{\hat{x}_i} g^{\hat{y}_i} \text{ for } i = 1,...,n_u. \\ \vec{M} = (\hat{m}_j) \text{ with } \hat{m}_j = g_2^{\hat{z}_j} g^{\hat{w}_j} \text{ for } j = 1,...,n_m. \end{cases}$$

Observe that from the perspective of the adversary, the distribution of these public parameters are identical to the real construction. Note that the master key is implicitly set to be $g_2^{\alpha} = g_2^{a} = g^{ab}$.

To make the notation easy to follow, we also define four functions J_1, J_2, K_1 and K_2 such that for any set $S\subseteq\{1,\ldots,n_u\}$ and $S'\subseteq\{1,\ldots,n_m\}$,

$$K_1(S) = x' - l_u k_u + \sum_{i\in S}\hat{x}_i,\ J_1(S) = y' + \sum_{i\in S}\hat{y}_i,\ K_2(S') = z' - l_m k_m + \sum_{j\in S'}\hat{z}_j,\ J_2(S') = w' + \sum_{j\in S'}\hat{w}_j .$$

Note that for any set $S\subseteq\{1,\ldots,n_u\}$, $S'\subseteq\{1,\ldots,n_m\}$ the following equalities always hold

$$g_2^{K_1(S)J_1(S)} = L_1(S),\ g_2^{K_2(S')} g^{J_2(S')} = L_2(S').$$

Before describing the simulation, we point out that some implicit relations exit in our scheme: according to Eq.(2), all the temporary secret keys of a given user share the same exponent r; and according to Eq.(4), all the signatures generated by a given signer also share the same exponent r. To embody these relations in the simulation, B forms an initially empty list named R^{list} as explained below. For easy explanation, an algorithm named $RQuery(ID)$ is also defined such that for an input ID, if there exists a tuple $(ID,\hat{r})$ in R^{list} then $\hat{r}$ is returned, otherwise, it chooses $\hat{r}\in_R Z_q^*$, and adds $(ID,\hat{r})$ into R^{list} and returns $\hat{r}$.

Oracles Simulation: B answers a series of oracle queries for A in the following way:

Oracle KEO(·) *simulation*: B maintains a list HK^{list} which is initially empty. Upon receiving an extract query on identity ID, B outputs "*failure*" and aborts if $K_1(U'_{ID}) \equiv 0 \bmod q$ (denote this event by **E1**). Otherwise, B first searches HK^{list} for tuple (ID,HK_{ID}) (if HK^{list} does not contain this tuple, it chooses $HK_{ID}\in_R Z_q^*$ and adds (ID,HK_{ID}) into HK^{list}), then it computes $\hat{r} = RQuery(ID)$, $k_{ID,0} = F_{HK_{ID}}(0\,\|\,ID)$ and defines $TSK_{ID,0}$ as

$$TSK_{ID,0} = \left(g_1^{-\frac{J_1(U'_{ID})}{K_1(U'_{ID})}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,0})^{k_{ID,0}}, g^{k_{ID,0}}, g_1^{\frac{-1}{K_1(U'_{ID})}} g^{\hat{r}} \right) \tag{6}$$

At last, B responds with $TSK_{ID,0}$ and HK_{ID} to A. Observe that if let $r = \hat{r} - \frac{a}{K_1(U'_{ID})}$, then it can be verified that $TSK_{ID,0}$ has the correct form as Eq.(1).

Oracle TKO(·) *simulation*: As argued in Remark 2, we require that A only queries oracle $TKO(\cdot)$ on the challenged identity. Upon receiving a temporary secret key query $\langle ID,t\rangle$, B outputs "*failure*" and aborts if $L_1(U'_{ID}) \equiv L_1(U_{ID,t}) \equiv 0 \bmod q$ holds (denote this event by **E2**). Otherwise, due to the fact that $k_{ID,t}$ is the output of a PRF and A does not know HK_{ID}, B can freely define $k_{ID,t}$ himself. B constructs $TSK_{ID,t}$ for A as follows: It first chooses $\hat{k}_{ID,t}\in_R Z_q^*$, computes $\hat{r} = RQuery(ID)$, and then defines $TSK_{ID,t}$ according to two cases

$$TSK_{ID,t} = \begin{cases} \left(g_1^{-\frac{J_1(U'_{ID})}{K_1(U'_{ID})}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,t})^{\hat{k}_{ID,t}}, g^{\hat{k}_{ID,t}}, g_1^{\frac{-1}{K_1(U'_{ID})}} g^{\hat{r}} \right), & \text{if } L_1(U'_{ID}) \neq 0 \bmod q \\ \left(g_1^{-\frac{J_1(U_{ID,t})}{K_1(U_{ID,t})}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,t})^{\hat{k}_{ID,t}}, g_1^{\frac{-1}{K_1(U_{ID,t})}} g^{\hat{k}_{ID,t}}, g^{\hat{r}} \right), & \text{else if } L_1(U_{ID,t}) \neq 0 \bmod q \end{cases}.$$

Note that in both cases, it can be verified that $TSK_{ID,t}$ has the correct form as Eq.(2).

Oracle SO(·,·,·) *simulation*: Upon receiving a signing query $\langle ID,t,m\rangle$, B outputs "*failure*" and aborts if $K_1(U'_{ID}) \equiv K_1(U_{ID,t}) \equiv K_2(M_m) \equiv 0 \bmod q$ holds (denote this event by **E3**). Otherwise, B first computes

$\hat{r} = RQuery(ID)$, chooses $r_t, r_m \in_R Z_q^*$, and then constructs the signature σ for A according to three cases

$$\sigma = \begin{cases} \left(t, g_1^{-\frac{J_1(U'_{ID})}{K_1(U'_{ID})}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,t})^{r_t} L_2(M_m)^{r_m}, g^{r_t}, g_1^{\frac{-1}{K_1(U'_{ID})}} g^{\hat{r}}, g^{r_m}\right), & \text{if } K_1(U'_{ID}) \neq 0 \bmod q \\ \left(t, g_1^{-\frac{J_1(U_{ID,t})}{K_1(U_{ID,t})}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,t})^{r_t} L_2(M_m)^{r_m}, g_1^{\frac{-1}{K_1(U_{ID,t})}} g^{r_t}, g^{\hat{r}}, g^{r_m}\right), & \text{else if } K_1(U_{ID,t}) \neq 0 \bmod q \\ \left(t, g_1^{-\frac{J_2(M_m)}{K_2(M_m)}} L_1(U'_{ID})^{\hat{r}} L_1(U_{ID,t})^{r_t} L_2(M_m)^{r_m}, g^{r_t}, g^{\hat{r}}, g_1^{\frac{-1}{K_2(M_m)}} g^{r_m}\right), & \text{else if } K_2(M_m) \neq 0 \bmod q \end{cases}.$$

Observe that σ is indeed a valid signature in all cases.

Forge: Eventually, A returns a forged signature $\sigma^* = (t^*, V^*, R_{t^*}, R^*, R_{m^*})$ on message m^* and identity ID^* with the constraint described in Definition 4. B outputs "*failure*" and aborts if $K_1(U'_{ID^*}) \equiv K_1(U_{ID^*,t^*}) \equiv K_2(M_{m^*}) \equiv 0 \bmod q$ does not hold (denote this event by **E4**). Otherwise, B can successfully derive g^{ab} as

$$g^{ab} = \frac{V^*}{R^{*J_1(U'_{ID^*})} R_{t^*}^{J_1(U_{ID^*,t^*})} R_{m^*}^{J_2(M_{m^*})}}.$$

This completes the simulation. From the description of B, we know that the time complexity of B is dominated by the exponentiations and the multiplications in the oracle simulations. Since there are $O(1)$ exponentiations in each oracle simulation, and $O(n_u)$, $O(n_u)$ and $O(n_u+n_m)$ multiplications in the simulation of oracles $KEO(\cdot)$, $TKO(\cdot,\cdot)$ and $SO(\cdot,\cdot,\cdot)$ respectively, we known that the time complexity of B is bounded by

$$T' \leq T + O((q_k+q_t+q_s)t_e + (n_u(q_k+q_t) + (n_u+n_m)q_s)t_m).$$

Let $\Pr[\neg abort]$ denote the probability of B's not aborting. Similarly to the analysis in Ref.[20], we can have

$$\Pr[\neg abort] \geq \frac{1}{27(n_u+1)^2(n_m+1)(q_k+q_t+2q_s)^2 q_s}.$$

It can be seen that in the above simulation, all the temporary secret keys of a given user share the same exponent r, and all the signatures generated for a given user also share this same exponent r. From the description of the simulation, we know that if B does not abort, the responses for A's oracle queries are identical to the real environment, and A can successfully return a valid forged signature with advantage ε. Therefore, B can solve the CDH problem instance with advantage

$$\varepsilon' \geq \frac{\varepsilon}{27(n_u+1)^2(n_m+1)(q_k+q_t+2q_s)^2 q_s}.$$

This concludes the proof of the theorem.

Theorem 2. The proposed scheme is strongly key-insulated in the standard model, assuming that (1) the CDH assumption holds in G_1; (2) the hash function H is collision-resistant; (3) the function F is a pseudo random function.

Proof: Without loss of generality, we assume that the hash function H is collision-resistant and the function F is a pseudo random function, then given an adversary A that has advantage ε against the strong key-insulated security of our proposed scheme by running within time T, asking at most q_k, q_h and q_s queries to oracles $KEO(\cdot)$, $HKO(\cdot,\cdot)$ and $SO(\cdot,\cdot,\cdot)$ respectively, there exists a (T', ε') adversary B against the CDH assumption in G_1 with

$$\begin{cases} T' \leq T + O((q_k+q_s)t_e + (n_u q_k + (n_u+n_m)q_s)t_m) \\ \varepsilon' \geq \dfrac{\varepsilon}{27(n_u+1)^2(n_m+1)(q_k+2q_s)^2 q_s} \end{cases},$$

where t_e and t_m have the same meaning as Theorem 1.

On inputting $(g, g^a, g^b) \in G_1^3$ for some unknown $a,b \in_R Z_q^*$, B interacts with A as follows:

Setup: The same as Theorem 1 except that l_u is set to be $l_u = \frac{3(q_k + 2q_s)}{2}$.

Oracles Simulation: B provides the simulation of oracles $KEO(\cdot)$ and $SO(\cdot,\cdot,\cdot)$ for A in the same way as in Theorem 1. Besides, B provides the simulation of oracle $HKO(\cdot)$ for A in the following way:

Oracle $HKO(\cdot)$ *simulation*: B maintains a list HK^{list} which is initially empty. Upon receiving a helper key query on ID. B first checks whether HK^{list} contains a tuple (ID,HK_{ID}). If it does, HK_{ID} is returned to A. Otherwise, B chooses $HK_{ID} \in \{0,1\}^k$, adds (ID,HK_{ID}) into HK^{list} and returns HK_{ID} to A.

Forge: Eventually, A returns a forged signature σ^* with the constraint described in Definition 5. B can derive g^{ab} in the same way as Theorem 1.

Similarly to Theorem 1, we can bound the complexity of B by

$$T' \le T + O((q_k+q_s)t_e + (n_u q_k + (n_u+n_m)q_s)t_m),$$

and the advantage of B by

$$\varepsilon' \ge \frac{\varepsilon}{27(n_u+1)^2(n_m+1)(q_k+2q_s)^2 q_s}.$$

Theorem 3. The proposed scheme has secure key-updates.

This theorem follows from the fact that for any period indices t_1, t_2 and any identity ID, the update key UK_{ID,t_1,t_2} can be derived from TSK_{ID,t_1} and TSK_{ID,t_2}.

6 Conclusions

With more and more cryptographic primitives applied to insecure environments such as mobile devices, key-exposure seems inevitable. This problem is perhaps the most dangerous attack on a cryptosystem since it typically means that security is entirely lost. To minimize the damage caused by key-exposure in ID-based signature scenarios, Zhou, *et al*.[17] adopted the key-insulation method and proposed an IBKIS scheme. However, their scheme is not strong key-insulated and their probably security is based on the random oracle model. In this paper, we re-formalize the definition and security notions for IBKIS schemes, and then propose a new IBKIS scheme with strong key-insulated security. Moreover, our scheme is provably secure in the standard model without resorting to the random oracle methodology. This is an attractive property since a proof in the random oracle model can only serve as a heuristic argument and can not imply the security in the implementation.

References:

[1] Shamir A. Identity-Based cryptosystems and signature schemes. In: Blakley GR, Chaum D, eds. Proc. of the Crypto'84. LNCS 196, Berlin: Springer-Verlag, 1984. 47−53.

[2] Anderson R. Two remarks on public-key cryptology. Invited lecture. In: Proc. of the CCCS'97. 1997. http://www.cl.cam.ac.uk/users/rja14/

[3] Bellare M, Miner S. A forward-secure digital signature scheme. In: Wiener M, ed. Proc. of the CRYPTO'99. LNCS 1666, Berlin: Springer-Verlag, 1999. 431−448.

[4] Itkis G, Reyzin L. SiBIR: Signer-base intrusion-resilient signatures. In: Yung M, ed. Proc. of the Crypto 2002. LNCS 2442, Berlin: Springer-Verlag, 2002. 499−514.

[5] Dodis Y, Katz J, Xu S, Yung M. Key-Insulated public-key cryptosystems. In: Knudsen LR, ed. Proc. of the Eurocrypt 2002. LNCS 2332, Berlin: Springer-Verlag, 2002. 65−82.

[6] Bellare M, Palacio A. Protecting against key exposure: Strongly key-insulated encryption with optimal threshold. 2002. http://eprint.iacr.org/2002/064

[7] Hanaoka Y, Hanaoka G, Shikata J, Imai H. Unconditionally secure key insulated cryptosystems: Models, bounds and constructions. In: Deng R, Qing S, Bao F, Zhou J, eds. Proc. of the ICICS 2002. LNCS 2513, Berlin: Springer-Verlag, 2002. 85−96.

[8] Dodis Y, Yung M. Exposure-Resilience for free: The hierarchical ID-based encryption case. In: Proc. of the IEEE SISW 2002. 2002. 45−52.

[9] Cheon JH, Hopper N, Kim Y, Osipkov I. Authenticated key-insulated public key encryption and timed release cryptography. 2004. http://eprint.iacr.org/2004/231

[10] Hanaoka Y, Hanaoka G, Shikata J, Imai H. Identity-Based hierarchical strongly key-insulated encryption and its application. In: Roy B, ed. Proc. of the ASIACRYPT 2005. LNCS 3788, Berlin: Springer-Verlag, 2005. 495−514.

[11] Hanaoka G, Hanaoka Y, Imai H. Parallel key-insulated public key encryption. In: Yung M, Dodis Y, Kiayias A, *et al*., eds. Proc. of the PKC 2006. LNCS 3958, Berlin: Springer-Verlag, 2006. 105−122.

[12] Dodis Y, Katz J, Xu S, Yung M. Strong key-insulated signature schemes. In: Desmedt YG, ed. Proc. of the PKC 2003. LNCS 2567, Berlin: Springer-Verlag, 2003. 130−144.

[13] Le Z, Ouyang Y, Ford J, Makedon F. A hierarchical key-insulated signature scheme in the CA trust model. In: Zhang K, Zheng Y, eds. Proc. of the ISC 2004. LNCS 3225, Berlin: Springer-Verlag, 2004. 280−291.

[14] González-Deleito N, Markowitch O, Dall'Olio E. A new key-insulated signature scheme. In: López J, Qing S, Okamoto E, eds. Proc. of the ICICS 2004. LNCS 3269, Berlin: Springer-Verlag, 2004. 465−479.

[15] Cao Z. Universal forgeability of Wang-Wu-Wang key-insulated signature scheme. 2004. http://eprint.iacr.org/2004/307.pdf

[16] Yum DH, Lee PJ. Efficient key updating signature schemes based on IBS. In: Paterson KG, ed. Proc. of the Cryptography and Coding 2003. LNCS 2898, Berlin: Springer-Verlag, 2003. 16−18.

[17] Zhou Y, Cao Z, Chai Z. Identity based key insulated signature. In: Chen K, Deng R, Lai X, *et al*., eds. Proc. of the ISPEC 2006. LNCS 3903, Berlin: Springer-Verlag, 2006. 226−234.

[18] Canetti R, Goldreich O, Halevi S. The random oracle methodology, revisited. Journal of the ACM, 2004,51(4):557−594.

[19] Boneh D, Franklin M. Identity based encryption from the Weil pairing. In: Kilian J, ed. Proc. of the Crypto 2001. LNCS 2139, Berlin: Springer-Verlag, 2001. 213−229.

[20] Paterson K, Schuldt J. Efficient identity-based signatures secure in the standard model. In: Batten L, Safavi-Naini R, eds. Proc. of the ACISP 2006. LNCS 4058, Berlin: Springer-Verlag, 2006. 207−222.

[21] Waters B. Efficient identity-based encryption without random oracles. In: Cramer R, ed. Proc. of the Eurocrypt 2005. LNCS 3494, Berlin: Springer-Verlag, 2005. 114−127.

[22] Goldreich O, Goldwasser S, Micali S. How to construct random functions. Journal of the ACM, 1984,33(4):792−807.

WENG Jian was born in 1976. He is a Ph.D. candidate at Shanghai Jiaotong University. His current research areas are cryptography and information security.

CHEN Ke-Fei was born in 1959. He is a professor and doctoral supervisor at the Shanghai Jiaotong University. His research areas are cryptography and information security.

LIU Sheng-Li was born in 1975. She is an associate professor and master supervisor at the Shanghai Jiaotong University. Her current research areas are cryptography and information security.

LI Xiang-Xue was born in 1974. He is a doctor at the Shanghai Jiaotong University. His current research areas are provable security and pairing-based cryptography.