Cryptanalysis of Two Strongly Unforgeable Identity-based Signatures in the Standard Model

Wenjie Yang[1,2], Min-Rong Chen[2], and Guo-Qiang Zeng[3]
(Corresponding author: Min-Rong Chen)
College of Cyber Security, College of Information Science and Technology, Jinan University[1]
School of Computer, South China Normal University[2]
Guangzhou 510631, China
(Email: mrongchen@126.com)
National-Local Joint Engineering Laboratory of Digitalize Electrical Design Technology, Wenzhou University[3]

(Received July 14, 2017; revised and accepted Oct. 22, 2017)

Abstract

The strong unforgeability of digital signature means that no attacker can forge a valid signature on a message, even given some previous signatures on the message, which has been widely accepted as a common security requirement. Recently, Tsai *et al.* and Hung *et al.* presented an efficient identity-based signature scheme and a revocable identity-based signature scheme, respectively. Meanwhile, they all claimed that their scheme is strongly unforgeable against chosen message attacks. In this paper, we point out that the two schemes cannot meet the requirements of strong unforgeability by giving some concrete attacks and briefly analyze the reasons why the provably-secure schemes are insecure following their security model.

Keywords: Cryptanalysis; Identity-Based Signature; Strong Unforgeability

1 Introduction

To simplify the complicated certificate management in traditional public key systems, Shamir [11] introduced the concept of identity-based public key cryptography (IB-PKC). In IB-PKC settings, an entity's public key is some unique public information such as ID card, email address, while the corresponding private key are directly derived by the private key generator (PKG) from these public identity information. Moreover, the author addressed the first identity-based signature (IBS) scheme. Since then, a few classic IBS schemes [3,5] were presented in random oracles following Shamir's idea.

As shown in [2], however, a security proof in random oracles can only serve as a heuristic argument and does not necessarily imply the security in the real implementation. It arises interest to construct a IBS scheme provably secure without random oracles. Until 2006, the first practical IBS scheme provably secure in the standard model was presented in [9]. Unfortunately, it does not cover the strong unforgeability [1] which is needed in a variety of applications. Afterwards, lots of improved IBS schemes were proposed to meet the requirements of strong unfrogeability in the standard model [7,8,10].

Recently, Tsai *et al.* [12] analyzed the existing strongly unforgeable IBS schemes without using random oracles and proposed an efficient and practical IBS scheme with short signature that is secure without random oracles. In the same year, Hung *et al.* [4] introduced a revocable identity-based signature (RIBS) scheme in the standard model. They all claimed that their (R)IBS scheme is strongly unforgeable against chosen message attacks.

In this paper, we first illustrate that Tsai *et al.*'s IBS scheme cannot meet the requirements of strong unforgeability by giving some concrete attacks. Then, we demonstrate that an attacker can easily discover the difference between simulated signatures and real signatures by interacting with the challenger. Finally, we show that Hung *et al.*'s RIBS scheme is actually based on Tsai *et al.*'s IBS scheme and can give some similar cryptanalysis according to the same ideas.

The rest of this paper is organized as follows. Section 2 gives some preliminaries. In Section 3, we review Tsai *et al.*'s IBS scheme and cryptanalyze its security. In Section 4, we look back Hung *et al.*'s RIBS scheme and briefly do some cryptanalysis on it. Finally, the conclusion is given in Section 5.

2 Preliminaries

2.1 Bilinear Groups and Complexity Assumption

Bilinear groups: Let $\mathbb{G}_1$ and $\mathbb{G}_2$ be two multiplicative cyclic groups of same prime order p and g be a generator of $\mathbb{G}_1$. The bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ has the following properties [6,13]:

- Bilinearity: $\forall g, h \in \mathbb{G}_1$ and $\forall a, b \in \mathbb{Z}_p^*$, we have $\hat{e}(g^a, h^b) = \hat{e}(g, h)^{ab}$;
- Non-degenracy: $\hat{e}(g, g) \neq 1_{\mathbb{G}_2}$ for $1_{\mathbb{G}_2}$ denotes the identity element of $\mathbb{G}_2$;
- Computability: There exists an efficient algorithm to compute $\hat{e}(g, h)$.

Computational Diffie-Hellman (CDH) Assumption: Let $(\mathbb{G}_1, \mathbb{G}_2, p, \hat{e})$ be a description of the bilinear group of prime order p. g is a generator of subgroup $\mathbb{G}_1$. The CDH assumption is that if the challenge tuple $D = ((\mathbb{G}_1, \mathbb{G}_2, p, \hat{e}), g, g^a, g^b)$ is given, no probabilistic polynomial time (PPT) algorithm $\mathcal{A}$ can output $g^{ab} \in \mathbb{G}_1$ with more than a negligible advantage. The advantage of $\mathcal{A}$ is defined as $\mathbf{Adv}_{\mathcal{A}}^{CDH}(\lambda) = Pr[\mathcal{A}(D) = g^{ab}]$ where the probability is taken over random choices of $a, b \in \mathbb{Z}_p$.

2.2 Collision Resistant Hash (CRH) Assumption

Let $H_k : \{0,1\}^* \rightarrow \{0,1\}^n$ be a collision-resistant hash family of functions, where n is a fixed length and k is an index. We say that the $(\epsilon, t)-$CRH assumption holds if no polynomial time adversary $\mathcal{A}$ running in time at most t can break the collision resistance of H_k with probability ϵ. Here, the successful probability of the adversary $\mathcal{A}$ is presented as $Pr[\mathcal{A}(k) = (m_0, m_1) : m_0 \neq m_1, H_k(m_0) = H_k(m_1)]$, where the probability is over the random choice consumed by the adversary $\mathcal{A}$.

2.3 Frameworks and Security Notions

Identity-Based Signature (IBS) Scheme consists of four polynomial-time algorithms as follows:

Setup. Given a security parameter κ, this algorithm produces a master secret key msk and the corresponding public parameters $params$. Then $params$ are published and msk is kept by itself.

Extract. Given a user's identity ID, the public parameters $params$ and the master secret key msk, this algorithm computes a private key D_{ID} for ID, which is transmitted to the user ID through a secure channel.

Sign. Given a private key D_{ID} of a user ID and a message m, this algorithm running by the user ID generates and outputs a signature σ of ID on m.

Verify. Given a user's identity ID, a message m and a signature σ, a verifier checks the validity of σ. More precisely, the algorithm outputs 1 if accepted, or 0 if rejected.

Strong Unforgeability for Identity-Based Signature Here, we denote by $\mathcal{O}_E$ an oracle simulating the algorithm *Extract*, and by $\mathcal{O}_S$ an oracle simulating the algorithm *Sign*. Strong unforgeability under an adaptive chosen-message attack is defined using the following game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:

Setup: $\mathcal{C}$ picks a security parameter κ and runs the algorithm *Setup*. It keeps the master secret key msk to itself and gives $\mathcal{A}$ the resulting parameters $params$.

Extract queries: $\mathcal{A}$ adaptively asks for the private key of any identity ID_i. To each extraction query of ID_i, $\mathcal{C}$ responds by running $\mathcal{O}_E$ to generate the private key D_{ID_i} of ID_i and forwarding D_{ID_i} to $\mathcal{A}$.

Signing queries: $\mathcal{A}$ adaptively asks for the signature of any identity ID_i on any message m_i. To each signing query of ID_i on M_i, $\mathcal{C}$ responds by running $\mathcal{O}_S$ to generate a signature σ, and sending σ to $\mathcal{A}$.

Forgery: $\mathcal{A}$ outputs (ID^*, m^*, σ^*) and wins if the followings hold:

1) σ^* is a valid signature of ID^* on m^*;
2) ID^* is not queried during extract queries;
3) (ID^*, m^*, σ^*) is not queried during the sign queries.

We define $\mathbf{Adv}_{\mathcal{A}}$ to be the probability that $\mathcal{A}$ wins the above game, taken over all coin toss of $\mathcal{C}$ and $\mathcal{A}$. In this paper, $\mathcal{A}$ is said to $(t, q_e, q_s, \varepsilon)$-strongly break an identity-based signature (IBS) scheme if $\mathcal{A}$ runs in time at most t, makes at most q_e *extract* queries, at most q_s *signing* queries, and $\mathbf{Adv}_{\mathcal{A}}$ is at least ε. An IBS scheme is $(t, q_e, q_s, \varepsilon)$-strongly existential unforgeable under an adaptive chosen message attack if no adversary $(t, q_e, q_s, \varepsilon)$-strongly breaks it.

Revocable Identity-Based Signature (RIBS) Scheme consists of five polynomial-time algorithms as follows.

Setup. Given a security parameter κ and the total number z of all periods, this algorithm outputs a master secret key msk and the corresponding public parameters $params$. Then $params$ are published and msk is kept by the PKG.

Initial key extract. Given an identity ID, the public parameters $params$ and the master secret key msk, this algorithm outputs the initial key D_{ID} which is transmitted to the user ID through a secure channel.

Time key update. Given an identity ID, a time period t, the public parameters $params$ and the master secret key msk, this algorithm outputs the time key T_{ID} which is transmitted to the user through a public channel. The user can combine the initial key D_{ID} and the time key T_{ID} to obtain the full private key $S_{ID,t}$.

Sign. Given an identity ID, the corresponding private key $S_{ID,t}$, a time period t and a message m, this algorithm outputs a signature σ of ID on m in t.

Verify. Given an identity ID, a message m and a signature σ, a verifier checks the validity of σ in the period t. More precisely, the algorithm outputs 1 if accepted, or 0 if rejected.

Strong Unforgeability for Revocable Identity-Based Signature Here, we denote by $\mathcal{O}_E$ an oracle simulating the algorithm *Initial key extract*, by $\mathcal{O}_T$ an oracle simulating the algorithm *Time key update*, and by $\mathcal{O}_S$ an oracle simulating the algorithm *Sign*. Strong unforgeability under an adaptive chosen-message attack is defined using the following game between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$:

Setup: $\mathcal{C}$ picks a security parameter κ and runs the algorithm *Setup*. It keeps the master secret key msk to itself and gives $\mathcal{A}$ the resulting parameters $params$.

Extract queries: $\mathcal{A}$ adaptively asks for the initial key of any identity ID. To each extraction query of ID, $\mathcal{C}$ responds by running $\mathcal{O}_E$ to generate the initial key D_{ID} and forwarding D_{ID} to $\mathcal{A}$.

Update queries: $\mathcal{A}$ adaptively asks for the time key of any identity ID in period t. To each update query of ID, $\mathcal{C}$ responds by running $\mathcal{O}_T$ to generate the time key T_{ID} and forwarding T_{ID} to $\mathcal{A}$.

Signing queries: $\mathcal{A}$ adaptively asks for the signature of any identity ID on any message m in period t. To each signing query, $\mathcal{C}$ responds by running $\mathcal{O}_S$ to generate a signature σ and sending σ to $\mathcal{A}$.

Forgery: $\mathcal{A}$ outputs $(ID^*, m^*, t^*, \sigma^*)$ and wins if the following holds:

1) σ^* is a valid signature of ID^* on m^* in t^*;
2) σ^* has not been outputted in the signing queries on (ID^*, m^*, t^*);
3) Either ID^* or (ID^*, t^*) has not appeared in the extract queries or the update queries, respectively.

The adversary $\mathcal{A}$'s advantage is defined as the probability that $\mathcal{A}$ wins the above game. In addition, to simplify the security analysis, we consider two types of adversaries, namely, outside adversary and inside adversary (or revoked user). Note that if the adversary is an outsider, it is allowed to issue all queries in the above game except for the initial key extract query on the target identity ID^*. If the adversary is an insider, it is allowed to issue all queries in the above game except for the time key update query on (ID^*, t^*).

3 Tsai *et al.*'s IBS Scheme

3.1 Review of Tsai *et al.*'s Scheme

The strongly unforgeable identity-based signature scheme [12] is specified by the following four algorithms.

Setup. Given a security parameter κ, the PKG chooses two groups $\mathbb{G}_1$, $\mathbb{G}_2$ of sufficiently large prime order $p > 2^\kappa$, a generator g of $\mathbb{G}_1$ and an admissible bilinear map $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \to \mathbb{G}_2$. The PKG sets three collision resistant hash functions, namely, $H_1 : \{0,1\}^* \to \{0,1\}^m$, H_2 and $H_3 : \{0,1\}^* \to \{0,1\}^n$, where m and n are fixed lengths. We assume $p > 2^m$ and $p > 2^n$ so that the hash outputs can be viewed as the elements of $\mathbb{Z}_p$. The PKG chooses two random values $u', w' \in \mathbb{G}_1$ as well as two vectors $\vec{u} = (u_i)$ of length m and $\vec{w} = (w_j)$ of length n, where u_i, $w_j \in \mathbb{G}_1$ for $i = 1, 2, ..., m$ and $j = 1, 2, ..., n$. The PKG then chooses a secret random value $\alpha \in \mathbb{Z}_p^*$ and sets $g_1 = g^\alpha \in \mathbb{G}_1$. Finally, the PKG randomly chooses $g_2 \in \mathbb{G}_2$ and sets the master secret key $msk = g_2^\alpha$ and the public parameters $params = (\mathbb{G}_1, \mathbb{G}_2, \hat{e}, g, g_1, g_2, H_1, H_2, H_3, u', \vec{u}, w', \vec{w})$.

Extract. Given a user's identity $ID \in \{0,1\}^*$, the PKG sets $v = H_1(ID)$. Here, $v = (v_1, v_2, ..., v_m)$ is a bit string of length m. Let $U \subset \{1, 2, ..., m\}$ be the set of index i such that $v_i = 1$, for $i = 1, 2, ..., m$. The PKG chooses a random value $r_v \in \mathbb{Z}_p^*$ and computes the user's private key $D_{ID} = (D_1, D_2) = (g_2^\alpha (u' \prod_{i \in U} u_i)^{r_v}, g^{r_v})$. The PKG transmits D_{ID} to the user via a secure channel.

Sign. Given a message $m \in \{0,1\}^*$, let $vm = H_2(m)$ be a bit string of length n and let vm_j denote the jth bit of vm. Let $W \subset \{1, 2, ..., n\}$ be the set of index j such that $vm_j = 1$, for $j = 1, 2, ..., n$. The signer with identity ID chooses a random number $r_m \in \mathbb{Z}_p^*$ and then computes $h = H_3(m \| g^{r_m})$. The signer uses her/his private key $D_{ID} = (D_1, D_2)$ to create a signature on the message m by $\sigma = (D_1^h (w' \prod_{j \in W} w_j)^{r_m}, D_2^h, g^{r_m})$.

Verify. Given a signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$ of a signer ID on a message m, a verifier can compute $h = H_3(m \| \sigma_3)$ to validate the signature tuple by the equation

$$\hat{e}(\sigma_1, g) \overset{?}{=} \hat{e}(g_1, g_2)^h \hat{e}(u' \prod_{i \in U} u_i, \sigma_2) \hat{e}(w' \prod_{i \in W} w_i, \sigma_3).$$

The algorithm outputs "accept" if the above equation holds, and "reject" otherwise.

3.2 A Concrete Attack

Now, we shall in detail show how an attacker $\mathcal{A}$ forges a new signature σ' for a previously signed message m by interacting with the challenger $\mathcal{C}$ according to the security model [12].

1) $\mathcal{C}$ takes a security parameter κ and runs the *Setup* algorithm to produce a master secret key $msk = g_2^\alpha$ and public parameters $params = (\mathbb{G}_1, \mathbb{G}_2, \hat{e}, g, g_1, g_2, H_1, H_2, H_3, u', \vec{u}, w', \vec{w})$. $\mathcal{C}$ then gives $params$ to $\mathcal{A}$ and keeps msk by itself.

2) Given any user's identity ID and any message m, $\mathcal{C}$ runs the *Sign* algorithms in Tsai *et al.*'s scheme and produces the corresponding signature σ for m under ID. The signature's concrete forms are as follows, where $h = H_3(m\|g^{r_m})$.

$$\begin{aligned}\sigma &= (\sigma_1, \sigma_2, \sigma_3)\\ &= (D_1^h(w'\prod_{j\in W} w_j)^{r_m}, D_2^h, g^{r_m})\\ &= ((g_2^\alpha(u'\prod_{i\in U} u_i)^{r_v})^h(w'\prod_{j\in W} w_j)^{r_m}, g^{r_v h}, g^{r_m})\end{aligned}$$

3) $\mathcal{A}$ picks $r'_v \in_R \mathbb{Z}_p^*$ and computes a new signature σ' for m under ID with $\sigma'_1 = \sigma_1((u'\prod_{i\in U} u_i)^{r'_v})^h = (g_2^\alpha(u'\prod_{i\in U} u_i)^{r_v+r'_v})^h(w'\prod_{j\in W} w_j)^{r_m}$, $\sigma'_2 = \sigma_2 g^{r'_v h} = g^{(r_v+r'_v)h}$ and $\sigma'_3 = \sigma_3$, where $h = H_3(m\|g^{r_m})$.

It is clear that σ' is a new valid signature for m on ID since

$$\begin{aligned}&\hat{e}(\sigma'_1, g)\\ =\ &\hat{e}((g_2^\alpha(u'\prod_{i\in U} u_i)^{r_v+r'_v})^h(w'\prod_{j\in W} w_j)^{r_m}, g)\\ =\ &\hat{e}((g_2^\alpha)^h, g)\hat{e}(((u'\prod_{i\in U} u_i)^{r_v+r'_v})^h, g)\hat{e}((w'\prod_{j\in W} w_j)^{r_m}, g)\\ =\ &\hat{e}(g_1, g_2)^h\hat{e}(u'\prod_{i\in U} u_i, \sigma'_2)\hat{e}(w'\prod_{i\in W} w_i, \sigma'_3).\end{aligned}$$

Thus, the scheme fails to satisfy the requirement of strong unforgeability.

3.3 Flaws in the Security Proof

It is well known that a provably-secure cryptographic scheme should resist all attacks under the appropriate adversarial model. Then, why is Tsai *et al.*'s IBS scheme which is strictly proven under their security model not secure? In fact, there exist some fatal flaws in Tsai *et al.*'s security proof as follows.

- *Signing query*(ID, m). Upon receiving the query along with (ID, m), the challenger $\mathcal{C}$ sets $v = H_1(ID)$ and $vm = H_2(m)$.

Case 1. If $F(v) \neq 0 \bmod l_v$, the challenger $\mathcal{C}$ can construct the private key for $v = H_1(ID)$ as in the extract query, and then use the Signing algorithm to respond a signature σ on m.

Case 2. If $F(v) = 0 \bmod l_v$ and $K(vm) = 0 \bmod l_m$, the challenger $\mathcal{C}$ reports failure and terminates. Otherwise, if $F(v) = 0 \bmod l_v$ and $K(vm) \neq 0 \bmod l_m$, the challenger $\mathcal{C}$ chooses two random values $r_v, r_m \in \mathbb{Z}_p^*$ and then computes $h = H_3(m\|g^{r_m})$ to generate the simulated signature $\sigma = (\sigma_1, \sigma_2, \sigma_3)$, where $\sigma_3 = g_1^{\frac{-h}{K(vm)}} \cdot g^{r_m}$.

From the above descriptions, we notice that if $F(v) = 0 \bmod l_v$ (where $v = H_1(ID)$), $\mathcal{C}$ cannot respond a valid signature σ on message m under the identity ID since $H_3(m\|g^{r_m}) \neq H_3(m\|\sigma_3)$. That is to say, an adversary can easily distinguish the distribution of simulated signatures from that of real signatures by making some signing queries under the target identity ID^* and message m^*. Therefore, the security argument of Tsai *et al.*'s IBS scheme did not work out exactly as their simulated game definition.

4 Hung *et al.*'s RIBS Scheme

4.1 Review of Hung et al.'s Scheme

Here, we review the strongly secure revocable identity-based signature scheme [4] by the five algorithms below.

Setup. Given a security parameter κ and the total number z of all periods, the PKG chooses two cyclic groups $\mathbb{G}_1$ and $\mathbb{G}_2$ of sufficiently large prime order $p > 2^\kappa$. Let g be a generator of $\mathbb{G}_1$ and $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ be an admissible map. The PKG sets the master secret key and the public parameters by running the following tasks.

1) Pick two secret values $\alpha, \beta \in Z_p^*$ at random and compute $g_1 = g^{\alpha+\beta} \in \mathbb{G}_1$. Select a random $g_2 \in \mathbb{G}_1$ and compute g_2^α and g_2^β.

2) Set four collision resistant hash functions $H_1 : \{0,1\}^* \rightarrow \{0,1\}^{n_u}$, $H_2 : \{0,1\}^* \rightarrow \{0,1\}^{n_t}$, $H_3, H_4 : \{0,1\}^* \rightarrow \{0,1\}^{n_m}$, where n_u, n_t and n_m are fixed lengths.

3) Choose three random values $u', t', w' \in \mathbb{G}_1$ and three vectors $\vec{U} = (u_i)$, $\vec{T} = (t_j)$, $\vec{W} = (w_k)$, where $u_i, t_j, w_k \in \mathbb{G}_1$ for $i = 1, 2, ..., n_u$, $j = 1, 2, ..., n_t$ and $k = 1, 2, ..., n_m$.

4) Finally, the PKG sets the master secret key $msk = (g_2^\alpha, g_2^\beta)$ and the public parameters $params$= $(\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, g, g_1, g_2, H_1, H_2, H_3, H_4, u', $\vec{U}$, t', $\vec{T}$, w', $\vec{W})$.

Initial key extract. Given a user's identity $ID \in \{0,1\}^*$, the PKG sets $vu = H_1(ID)$. Here, $vu = (vu_1, vu_2, ..., vu_{n_u})$ is a bit string of length n_u. Let $U \subset \{1, 2, ..., n_u\}$ be the set of indices i such that $vu_i = 1$, for $i = 1, 2, ..., n_u$. The PKG chooses a random value $r_u \in \mathbb{Z}_p^*$ and computes the user's private key $D_{ID} = (D_1, D_2) = (g_2^\alpha(u'\prod_{i\in U} u_i)^{r_u}, g^{r_u})$. The PKG transmits D_{ID} to the user via a secure channel.

Time key update. Given a user's identity $ID \in \{0,1\}^*$ and a period t, the PKG sets $vt = H_2(ID, t)$. Here,

$vt = (vt_1, vt_2, ..., vt_{n_t})$ is a bit string of length n_t. Let $T \subset \{1, 2, ..., n_t\}$ be the set of indices j such that $vt_j = 1$, for $j = 1, 2, ..., n_t$. The PKG chooses a random value $r_t \in \mathbb{Z}_p^*$ and computes the user's private key $T_{ID} = (T_1, T_2) = (g_2^{\beta}(t' \prod_{j \in T} t_j)^{r_t}, g^{r_t})$. The PKG sends T_{ID} to the user via a public channel. Upon receiving T_{ID}, the user combines it with his/her initial secret key $D_{ID} = (D_1, D_2)$ to obtain the signing key $S_{ID,t}$ =(S_1, S_2, S_3)=(D_1T_1, D_2, T_2)= $(g^{\alpha+\beta}(u' \prod_{i \in U} u_i)^{r_u}(t' \prod_{j \in T} t_j)^{r_t}, g^{r_u}, g^{r_t})$.

Sign. For a period t, given a non-revoked user's identity $ID \in \{0, 1\}^*$, a message $m \in \{0, 1\}^*$, the user first computes a string $vm = H_3(m)$ of length n_m. Let vm_k denote the k-th bit of the string vm and let $W \subset \{1, 2, ..., n_m\}$ be the set of indices k such that $vm_k = 1$ for $k = 1, 2, ..., n_m$. Then the user chooses a random number $r_m \in Z_p^*$ and computes g^{r_m} and $h = H_4(m \| g^{r_m})$. Finally, the user generates a signature σ on the message m as follows:

$$\begin{aligned}
\sigma &= (\sigma_1, \sigma_2, \sigma_3, \sigma_4) \\
&= (S_1^h(w' \prod_{k \in W} w_k)^{r_m}, S_2^h, S_3^h, g^{r_m}) \\
&= (g_2^{(\alpha+\beta)h}(u' \prod_{i \in U} u_i)^{r_u h}(t' \prod_{j \in T} t_j)^{r_t h}(w' \prod_{k \in W} w_k)^{r_m}, \\
&\quad g^{r_u h}, g^{r_t h}, g^{r_m}),
\end{aligned}$$

where (S_1, S_2, S_3) is the signing key $S_{ID,t}$ obtained above.

Verify. Given a signature $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4)$ of a signer ID on a message m in a period t, a verifier can compute $h = H_4(m \| \sigma_4)$ to validate the signature tuple by the following equation

$$\begin{aligned}
\hat{e}(\sigma_1, g) &\overset{?}{=} \hat{e}(g_1, g_2)^h \hat{e}(u' \prod_{i \in U} u_i, \sigma_2) \cdot \\
&\quad \hat{e}(t' \prod_{j \in T} t_j, \sigma_3) \hat{e}(w' \prod_{k \in W} w_k, \sigma_4).
\end{aligned}$$

The algorithm outputs "accept" if the above equation holds, and "reject" otherwise.

4.2 Some Concrete Attacks

In fact, Hung *et al.*'s RIBS scheme is based on Tsai *et al.*'s IBS scheme. Thus, we can easily give a similar cryptanalysis according to the same ideas in Subsection 3.2. Here, we shall show how an attacker $\mathcal{A}$ forges a new signature σ' for a previously signed message m by interacting with the challenger $\mathcal{C}$ under the security model defined in [4].

1) $\mathcal{C}$ takes a security parameter κ and runs the *Setup* algorithm to produce the master secret key $msk = (g_2^{\alpha}, g_2^{\beta})$ and the public parameters $params$= ($\mathbb{G}_1$, $\mathbb{G}_2$, $\hat{e}$, g, g_1, g_2, H_1, H_2, H_3, H_4, u', $\vec{U}$, t', $\vec{T}$, w', $\vec{W}$). $\mathcal{C}$ then gives $params$ to $\mathcal{A}$ and keeps msk by itself.

2) Given any user's identity ID, a period t and any message m, $\mathcal{C}$ runs the *Sign* algorithms in Hung *et al.*'s scheme and outputs the corresponding signature σ for m under ID in t. The signature's concrete forms are as follows, where $h = H_3(m \| g^{r_m})$.

$$\begin{aligned}
\sigma &= (\sigma_1, \sigma_2, \sigma_3, \sigma_4) \\
&= (S_1^h(w' \prod_{k \in W} w_k)^{r_m}, S_2^h, S_3^h, g^{r_m}) \\
&= (g_2^{(\alpha+\beta)h}(u' \prod_{i \in U} u_i)^{r_u h}(t' \prod_{j \in T} t_j)^{r_t h}(w' \prod_{k \in W} w_k)^{r_m}, \\
&\quad g^{r_u h}, g^{r_t h}, g^{r_m})
\end{aligned}$$

3) $\mathcal{A}$ picks two random values r'_u and r'_t from $\mathbb{Z}_p^*$ and forges a new signature σ' on m under ID in t as follows, where $h = H_3(m \| g^{r_m})$.

$$\begin{aligned}
\sigma'_1 &= \sigma_1((u' \prod_{i \in U} u_i)^{r'_u})^h((t' \prod_{j \in T} t_j)^{r'_t})^h \\
&= (g_2^{\alpha+\beta}(u' \prod_{i \in U} u_i)^{r_u+r'_u}(t' \prod_{j \in T} t_j)^{r_t+r'_t})^h(w' \prod_{j \in W} w_j)^{r_m}
\end{aligned}$$

σ'_2=$\sigma_2 g^{r'_u h}$=$g^{(r_u+r'_u)h}$, σ'_3=$\sigma_3 g^{r'_t h}$ = $g^{(r_t+r'_t)h}$ and $\sigma'_4 = \sigma_4$.

It is clear that σ' is a new valid signature for m on ID in t since

$$\begin{aligned}
&\hat{e}(\sigma'_1, g) \\
= \ &\hat{e}(g_2^{(\alpha+\beta)h}(u' \prod_{i \in U} u_i)^{(r_u+r'_u)h}(t' \prod_{j \in T} t_j)^{(r_t+r'_t)h} \\
&\quad (w' \prod_{k \in W} w_k)^{r_m}, g) \\
= \ &\hat{e}(g_2^{(\alpha+\beta)h}, g)\hat{e}((u' \prod_{i \in U} u_i)^{(r_u+r'_u)h}, g) \cdot \\
&\qquad \hat{e}((t' \prod_{j \in T} t_j)^{(r_t+r'_t)h}, g)\hat{e}((w' \prod_{k \in W} w_k)^{r_m}, g) \\
= \ &\hat{e}(g_1, g_2)^h \hat{e}(u' \prod_{i \in U} u_i, \sigma'_2)\hat{e}(t' \prod_{j \in T} t_j, \sigma'_3)\hat{e}(w' \prod_{k \in W} w_k, \sigma'_4).
\end{aligned}$$

Therefore, the RIBS scheme fails to meet the requirement of strong unforgeability under their security model. For more details about cause analysis, the interested readers are referred to Section 3.3.

5 Conclusion

Strong unforgeability has been widely accepted as a common security requirement for signature schemes. In this paper, we first reviewed two so-called strongly unforgeable identity-based signatures presented by Tsai *et al.* and Hung *et al.*, respectively. Then, we demonstrated that both of them are not strongly unforgeable by giving some concrete attacks. Finally, we illustrated that there exist some serious errors in their proving process.

Acknowledgements

The authors would like to thank Prof. Jian Weng for his helpful comments. This work was supported by National Science Foundation of China (Grant Nos. 61472165, 61373158 and 61732021), Guangdong Provincial Engineering Technology Research Center on Network Security Detection and Defence (Grant No. 2014B090904067), Guangdong Provincial Special Funds for Applied Technology Research and development and Transformation of Important Scientific and Technological Achieve (Grant No. 2016B010124009), the Zhuhai Top Discipline–Information Security, Guangzhou Key Laboratory of Data Security and Privacy Preserving, Guangdong Key Laboratory of Data Security and Privacy Preserving, Zhejiang Provincial Natural Science Foundation of China (Grant Nos. LY16F030011 and LZ16E050002).

References

[1] D. Boneh, E. Shen, and B. Waters, "Strongly unforgeable signatures based on computational diffiehellman," in *Proceedings of The nineth international Conference on Theory and Practice in Public Key Cryptography (PKC2006)*, pp. 229–240, June 2006.

[2] R. Canetti, O. Goldreich, and S. Halevi, "The random oracle methodology, revisited," *Journal of the ACM*, vol. 51, no. 35, 2004.

[3] L. Deng, H. Huang, and Y. Qu, "Identity based proxy signature from RSA without pairings," *International Journal of Network Security*, vol. 19, no. 2, 2017.

[4] Y. H. Hung, T. T. Tsai, Y. M. Tseng, and S. S. Huang, "Strongly secure revocable id-based signature without random oracles," *Information Technology and Control*, vol. 43, no. 3, 2014.

[5] M. S. Hwang, J. W. Lo, S. C. Lin, "An efficient user identification scheme based on ID-based cryptosystem", *Computer Standards & Interfaces*, vol. 26, no. 6, pp. 565–569, 2004.

[6] M. S. Hwang, S. F. Tzeng, C. S. Tsai, "Generalization of Proxy Signature Based on Elliptic Curves", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 73–84, 2004.

[7] S. Kwon, "An identity-based strongly unforgeable signature without random oracles from bilinear pairings," *Information Sciences*, vol. 276, no. 32, pp. 1–9, 2014.

[8] Y. Ming and Y. M. Wang, "Cryptanalysis of an identity based signcryption scheme in the standard model," *International Journal of Network Security*, vol. 18, no. 1, pp. 165–171, 2016.

[9] K. G. Paterson and J. C. Schuldt, "Efficient identity-based signatures secure in the standard model," in *Proceedings of The Eleventh Australasian Conference on Information Security and Privacy (ACISP'06)*, pp. 207–222, June 2006.

[10] C. Sato, T. Okamoto, and E. Okamoto, "Strongly unforgeable id-based signatures without random oracles," in *Proceedings of The Fifth International Conference on Information Security Practice and Experience (ISPEC'09)*, pp. 35–45, June 2009.

[11] A. Shamir, "Identity-based cryptosystems and signature schemes," in *Proceedings of The Fourth Annual International Cryptology Conference*, pp. 47–53, June 1984.

[12] T. T. Tsai, Y. M. Tseng, and S. S. Huang, "Efficient strongly unforgeable id-based signature without random oracles," *Informatica*, vol. 25, no. 3, pp. 505–521, 2014.

[13] S. F. Tzeng, M. S. Hwang, "Digital signature with message recovery and its variants based on elliptic curve discrete logarithm problem", *Computer Standards & Interfaces*, vol. 26, no. 2, pp. 61–71, Mar. 2004.

Biography

Wenjie Yang is working toward his PhD degree in College of Cyber Security/College of Information Science and Technology, Jinan University at Guangzhou, P.R. China. His research interests focus on Cryptography and Information Security.

Min-Rong Chen received the B.S. degree and the M.S degree from South China University of Technology, Guangzhou, China, in 2000 and 2004, respectively, and the PhD degree from Shanghai Jiaotong University, Shanghai, China, in 2008. From 2008 to 2015, she was an Associate Professor with the College of Information Engineering, Shenzhen University, Shenzhen, China. She is currently an Associate Professor with the School of Computer, South China Normal University, Guangzhou, China. She has headed two national projects. She has authored or co-authored the book Extremal optimization: Fundamentals, algorithms, and applications (CRC press, 2016), over 30 journal and conference papers. She holds one patent. Her research interests include computational intelligence and information security.

Guo-Qiang Zeng received the B.S. degree in automation from China Jiliang University, Hangzhou, China, in 2006, and the PhD degree in control science and engineering from Zhejiang University, China, in 2011. From 2011 to 2014, he was a Lecturer with the Department of Electrical and Electronic Engineering, Wenzhou University, Wenzhou, China. Since 2015, he has been an Associate Professor with the National-Local Joint Engineering Laboratory of Digitalize Electrical Design Technology, Wenzhou University. He has headed three national and provincial projects. He has authored or co-authored the book Extremal optimization: Fundamentals, algorithms, and applications (CRC press, 2016), over 30 journal and conference papers, and over ten inventions. He holds seven patents. His research interests include computational intelligence and information security.