Analysis and Improvements of Several (H)IBS/IBSC/PRBE Schemes

Jindan Zhang[1], Xu An Wang[2], Xiaoyuan Yang[2]
[1]Department of Electronic Information
Xianyang Vocational Technical College, 712000, P. R. China
[2]Key Laboratory of Information and Network Security
Engineering University of Chinese Armed Police Force, 710086, P. R. China
wangxahq@yahoo.com.cn

Abstract— **Constructing efficient (Hierarchical) identity based signature/signcryption ((H)IBS/IBSC) schemes in the standard model with full security remain as open problems for a long time. Ren et al. constructed efficient (H)IBS/IBSC schemes with full security without random oracle in ISDPE'07 and Chinacrypt'08, . They claimed their schemes can be proved to simultaneously achieve high efficiency, short public parameters and a tight reduction. But we shall show their schemes are not secure. Furthermore, we give improvements to these schemes which can resist the proposed attack. Proxy re-encryption is a primitive which allows the transformation from A's ciphertext to be B's ciphertext by using proxies, without the proxy knowing the corresponding plaintexts or secret keys of A or B. Proxy broadcast re-encryption aims at transforming ciphertext from one user to a group, which is a generalization of proxy re-encryption. Recently, Sun et al. proposed a CCA-secure unidirectional proxy broadcast re-encryption in the standard model, we also show their scheme has some flaws.**

Index Terms— **(Hierarchical) identity based signature/signcryption, Proxy broadcast re-encryption, Attack.**

I. Introduction

Background. Computer systems in modern time are always in large scale, decentralized, and heterogeneous environments. Inevitably they are often facing the diverse threats such as swarms , viruses, identity theft and session jack etc. Cryptography research seeks to help the computers and networks more safer, more dependable and less vulnerable to those attacks. In this paper, we will cryptanalyze three signature/signacryption schemes, which were proposed to ensure the authenticaiton/confidentiality of the messages flowed on internet. Therefore, our result can be seen as good examples to indicate that computational intelligence engineers should be more careful on using cryptographic schemes. Some of the cryptographic schemes maybe seem to be secure but actually are not secure at all. Our negative results could tell the engineers cryptographic schemes always can be broken in unexpected way, and thus they should be more cautious on which cryptographic scheme can be employed in practical applications.

IBE/IBS. In 1984, Shamir introduced the primitive of identity based encryption/signature (IBE/IBS), which aimed at avoiding the burden certificate management in traditional public key infrastructure [1]. In the IBE/IBS system, the public key of the users is their identity information (e.g., name). Hence the public key can be implicit authenticated by the owner of 'correct' private key. Thus the management of certificates can be avoided. Boneh and Franklin demonstrated the first practical IBE scheme in 2001 by relying on bilinear pairings [2]. In 2004, Boneh and Boyen gave the BB_1 IBE and BB_2 IBE, namely two new efficient selective identity secure IBE schemes in the standard model [3]. Later IBE scheme in the standard model with full security (the attacker can adaptively attack any identity) were proposed by Boneh and Boyen, and Waters also proposed new somehow tightly fully secure IBE scheme by using Waters' hash function [4], [5]. At Eurocrypt'06, based on a strong assumption, Gentry proposed a new efficient IBE scheme with tight security reduction without random oracle [6]. On the other hand, efficient identity based signature schemes have been successfully constructed much early before 2011. Some other constructions of IBS scheme can be found in [7]–[9].

IBSC. In 1997, Zheng et al. proposed the concept of Signcryption, which aimed at achieving both encryption and signature in one logic step, while avoiding separate encryption and signature [10]. The first ID-based signcryption (IBSC) scheme was proposed by Malone-Lee [11], by using the technique of Hess's signature [8] with modified BF IBE. Later an improved identity-based signcryption scheme was given by Chen and Malone-Lee [12]. Yuen et al. constructed a provably secure signcrypton scheme in the standard model in 2005 with its security being proved in a weaker " sample model" rather than "selective-ID" model.

HIBE/HIBS. In practice one always face organizations with hierarchical structures, thus we need the higher-level authority can delegate keys to its lower-level sub-authorities. In hierarchy identity based encryption/signature (HIBE/HIBS), the identities represents by

The second author is the corresponding author. This paper is an extended work of [29], [30] and supported by the National Natural Science Foundation of China under contract no. 61103230, 61103231, 61272492, 61202492, Natural Science Basic Research Plan in Shaanxi Province of China (program No. 2010JM8023 and 2011JM8012) and Natural Science Foundation of Engineering University of Chinese Armed Police Force.

doi:10.4304/jcp.8.9.2304-2312

vectors, while messages are encrypted/signed for these vectors. In Eurocrypt'02, the first HIBE/HIBS was proposed by Horwitz and Lynn [13], and the first fully functional HIBE/HIBS scheme was constructed by Gentry and Silverberg [14] in Asiacrypt'02. The first provable secure HIBS scheme in the random oracle was proposed by Chow et al. [15] in 2004. But their scheme has this feature: signature length grows linearly with the hierarchy depth. In 2005, Yuen et al. [16] gave a HIBS scheme with constant signature length in the selective-ID model in the standard model.

*PRE.*In 1998, Blaze et al. introduced the primitive of proxy re-encryption (PRE) [17]. Proxy re-encryption can transform the ciphertexts under Alice to be another ciphertext under Bob by using semi-trusted proxy. later Ateniese et al. proposed a few new PRE schemes and demonstrated its interesting applications [18]. Since then, PRE schemes received much attention by the researchers. In AisaCCS'09, Weng et al. [19] proposed the concept of conditional proxy re-encryption, which can enable the delegator fine-grained delegate his decryption right to the delegatee. Later, they revised the definition and security model for CPRE, and constructed a more efficient one [20]. Broadcast encryption is a primitive aimed at one-to-many secure message delivering and researchers often try to extending other cryptographic primitives to the broadcast scenario to solve the corresponding security problems in the broadcasting enviroment [27], [28]. In ACISP'09, Chu et al. then generalized the conditional proxy re-encryption to the broadcast setting, and constructed a provable secure conditional proxy broadcast re-encryption (CPBRE) scheme [21].

Our Contribution. In ISDPE'07, Ren et al. [23] proposed efficient fully secure IBS/IBSC schemes without random oracle. Later in Chinacrypt'08, they [24] also proposed a fully secure HIBS scheme in the standard model which can achieve short parameters, high efficiency and a tight reduction. However, in this paper we give attacks to show their schemes are not secure. Furthermore, we give improvements to these schemes. Recently Sun et al. proposed a CCA-secure unidirectional proxy broadcast re-encryption in the standard model. But we show their scheme also has some flaws.

Organization. We organize the paper as follows. In section 2, we first show Ren et al.'s IBS scheme is not secure, then give our improved scheme. In section 3, we first show Ren et al.'s IBSC scheme is not secure yet, then give our improved scheme. In section 4, we show Ren et al.'s HIBS scheme is not secure and then fix it. In section 5, we show Sun et al.'s scheme has some flaws. In the last section, we give our conclusion.

II. IBS Scheme

A. Definition and Security Model for IBS Scheme

An IBS scheme consists of the following algorithms:

1) Setup: When given a security parameter k, the PKG generates (msk, $param$). msk is the master secret key and $param$ is the corresponding public parameter.
2) KeyGen: When given an identity ID, the PKG generates the corresponding private key SK_{ID} and gives it to its owner in a secure way.
3) Sign: when given the private key SK_{ID} and a message M, it outputs a signature σ.
4) Verify: when given the signer's identity ID, a message M and signature σ, it outputs "Valid" if σ is a valid signature of M under ID, $param$. Otherwise, it outputs $\perp$.

Definition 1: The existential unforgeability against adaptive identity and adaptive chosen message attack for IBS (EU-ID-CMA) scheme is defined in the following game. We first define the following oracles:

- KGO(ID): The Key Generation Oracle with input ID and master secret key msk will output the secret key SK_{ID}.
- SO(ID, M): With input ID and message M, The signing Oracle outputs a valid signature σ.

We define the Game as follows:

1) (*Phase 1.*) Challenger $\mathcal{B}$ outputs system parameter $param$ and let it be publicly known.
2) (*Phase 2.*) $\mathcal{A}$ queries KGO(ID) and SO(ID, M) adaptively.
3) (*Phase 3.*) $\mathcal{A}$ delivers a signature σ^* for signer identity ID^* and message M^*. ID^* has never been input to a KGO and (ID^*, M^*) has never been input to a SO.

$\mathcal{A}$ wins if he completes the Game with $Valid = Verify(ID^*, M^*, \sigma^*)$. Its advantage is its probability of winning.

B. Review of Ren et al.'s IBS Scheme

Here we review Ren et al.'s IBS scheme in [23]:

1) Setup: G_1, G_2 are groups of order p where p is a large prime number. $e : G_1 \times G_1 \to G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^{\alpha}$, where $\alpha \in Z_p^*$ which is the private key of PKG. The PKG randomly chooses $h_i \in G_1$, $i = 0, 1, 2$. The public parameters are $param = (g, g_1, h_0, h_1, h_2)$ and the master secret key is $msk = \alpha$.
2) KeyGen: For a user U with identity $ID \in Z_p^*$, the PKG randomly chooses $r_{ID} \in Z_p^*$, and outputs
$$d_1 = (h_0 g^{-r_{ID}})^{\frac{1}{\alpha - ID}}, d_2 = r_{ID}$$
User U's private key is $d = (d_1, d_2)$. If $ID = \alpha$, aborts.
3) Sign: For signing a message $m \in Z_p^*$, U randomly chooses $s \in Z_p^*$ and computes
$$\sigma_1 = d_1 (h_2 h_1^{ID})^{sm}, \sigma_2 = (g_1 g^{-ID})^s$$
so the signature of m is $\sigma = (\sigma_1, \sigma_2, r_{ID})$.
4) Verify: The receiver verifies whether $e(g_1 g^{-ID}, \sigma_1) = e(g, h_0) e(g, g)^{-r_{ID}} e(\sigma_2, (h_2 h_1^{ID})^m)$ hold, If

this equation holds, it is a valid signature. Otherwise, it is an invalid signature.

Correctness:

$$\begin{aligned} & e(g_1g^{-ID}, \sigma_1) \\ = \; & e(g_1g^{-ID}, (h_0g^{-r_{ID}})^{\frac{1}{\alpha-ID}} \cdot (h_2h_1^{ID})^{sm}) \\ = \; & e(g, h_0)e(g, g)^{-r_{ID}}e((g_1g^{-ID})^s, (h_2h_1^{ID})^m) \\ = \; & e(g, h_0)e(g, g)^{-r_{ID}}e(\sigma_2, (h_2h_1^{ID})^m) \end{aligned}$$

C. Attack to IBS Scheme

Observe the verifying equation:

$$e(g_1g^{-ID}, \sigma_1) = e(g, h_0)e(g, g)^{-r_{ID}}e(\sigma_2, (h_2h_1^{ID})^m)$$

From this, we can modify the signature to be

$$\sigma_1' = \sigma_1, \ \sigma_2' = \sigma_2^{\frac{m}{m'}}$$

which is a valid signature for m'. Concretely, assume the target identity is ID^*, the attacker $\mathcal{A}$ attacks as follows:

1) $\mathcal{A}$ queries a signature on m of ID^* to the challenger $\mathcal{B}$, $\mathcal{B}$ will return $\sigma = (\sigma_1, \sigma_2, r_{ID})$.
2) Now $\mathcal{A}$ modifies the signature to be
$$\sigma_1' = \sigma_1, \ \sigma_2' = \sigma_2^{\frac{m}{m'}}, r_{ID}' = r_{ID}$$
$$\sigma' = (\sigma_1', \sigma_2', r_{ID}')$$
3) Then $\sigma' = (\sigma_1', \sigma_2', r_{ID})$ is a valid signature on m' because
$$\begin{aligned} & e(g_1g^{-ID}, \sigma_1') \\ = \; & e(g_1g^{-ID}, (h_0g^{-r_{ID}})^{\frac{1}{\alpha-ID}} \cdot (h_2h_1^{ID})^{sm}) \\ = \; & e(g, h_0)e(g, g)^{-r_{ID}}e((g_1g^{-ID})^s, (h_2h_1^{ID})^m) \\ = \; & e(g, h_0)e(g, g)^{-r_{ID}}e((g_1g^{-ID})^{s\cdot\frac{m}{m'}}, (h_2h_1^{ID})^{m'}) \\ = \; & e(g, h_0)e(g, g)^{-r_{ID}'}e(\sigma_2', (h_2h_1^{ID})^{m'}) \end{aligned}$$

D. Improved IBS Scheme

1) Setup: Let p be a large prime number, G_1, G_2 are groups of order p. $e : G_1 \times G_1 \to G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^\alpha$, where $\alpha \in Z_p^*$. α is the private key of PKG. Let $H : \{0, 1\}^* \times G_1 \to Z_p^*$ be a hash function. The PKG randomly chooses $h_i \in G_1$, $i = 0, 1, 2$. The public parameters and the master secret key are
$$param = (g, g_1, h_0, h_1, h_2, H), msk = \alpha$$
2) KeyGen: To a user U with identity $ID \in Z_p^*$, the PKG randomly chooses $r_{ID} \in Z_p^*$, and computes
$$d_1 = (h_0g^{-r_{ID}})^{\frac{1}{\alpha-ID}}, d_2 = r_{ID}$$
so the private key of user U is $d = (d_1, d_2)$. If $ID = \alpha$, aborts.
3) Sign: To sign a message $m \in Z_p^*$, U randomly chooses $s \in Z_p^*$ and computes
$$\sigma_2 = (g_1g^{-ID})^s, \sigma_1 = d_1(h_2h_1^{ID})^{sH(m,\sigma_2)},$$
so the signature of m is $\sigma = (\sigma_1, \sigma_2, r_{ID})$.
4) Verify: The receiver verifies whether
$$\begin{aligned} e(g_1g^{-ID}, \sigma_1) \;=\; & e(g, h_0)e(g, g)^{-r_{ID}} \\ & e(\sigma_2, (h_2h_1^{ID})^{H(m,\sigma_2)}) \end{aligned}$$
If σ can pass the above verification, it is a valid signature. Otherwise, it is an invalid signature.

Remark 1: In this scheme, $\sigma_1 = d_1(h_2h_1^{ID})^{sH(m,\sigma_2)}$ instead of $\sigma_1 = d_1(h_2h_1^{ID})^{sm}$. This little modification can resist the above attack, for this time $\sigma_1' = \sigma_1, \ \sigma_2' = \sigma_2^{\frac{m}{m'}}, r_{ID}' = r_{ID}$ can not pass the verified equation. Any modification on σ_2 will be detected.

III. IBSC Scheme

A. Definition and Security Model for IBSC Scheme

An IBSC scheme is defined by the following four algorithms:

1) Setup: When given a security parameter k, the PKG generates (msk,$param$) where msk is the randomly generated master secret key and $param$ is the corresponding public parameter.
2) Keygen: When given an identity ID, the PKG computes the corresponding private key SK_{ID} and gives it to its owner in a secure way.
3) Signcrypt: When given a message m for ID_b, ID_a computes Signcrypt(m, SK_{ID_a}, ID_b) to obtain the ciphertext σ.
4) Unsigncrypt: When ID_b receives σ, he computes Unsigncrypt(σ, SK_{ID_b}, ID_a) and obtains the plain text m or the symbol $\perp$ if σ was an invalid ciphertext between identities ID_a and ID_b.

Definition 2: We say that an IBSC scheme has the indistinguishability against adaptive chosen ciphertext attacks property (IND-IBSC-CCA) if no P.P.T adversary has a non-negligeable advantage in the following game.

1) The challenger runs the Setup algorithm and sends the system parameters to the adversary.
2) The adversary $\mathcal{A}$ performs a polynomially bounded number of requests:
   - Signcryption request: $\mathcal{A}$ queries two identities ID_i, ID_j and a plaintext M. The challenger computes $d_{ID_i} = Keygen(ID_i)$, $Signcrypt(m, d_{ID_i}, ID_j)$ and sends the result to $\mathcal{A}$.
   - Unsigncryption request: $\mathcal{A}$ queries two identities ID_i and ID_j, a ciphertext σ. The challenger computes the private key $d_{ID_j} = Keygen(ID_j)$ and sends the result of $Unsigncrypt(\sigma, d_{ID_j}, ID_i)$ to $\mathcal{A}$ (this result can be $\perp$ if σ is an invalid ciphertext).
   - Key extraction request: $\mathcal{A}$ produces an identity ID and receives the extracted private key $d_{ID} = Keygen(ID)$.

$\mathcal{A}$ can present its requests adaptively: every request may depend on the answer to the previous ones.

3) $\mathcal{A}$ chooses two plaintexts M_0, M_1 and two identities ID_A and ID_B on which he wishes to be challenged. He cannot ask the ID_A nor ID_B's private key in the first stage.
4) The challenger takes a bit $b \in \{0, 1\}$ and computes $C = Signcrypt(M_b, d_{ID_A}, ID_B)$ as the challenge ciphertext to $\mathcal{A}$.
5) $\mathcal{A}$ asks again a polynomially bounded number of requests as in the first stage. This time, he cannot make a key extraction request on ID_A nor ID_B and he cannot ask the Unsigncryption request which returns the plaintext corresponding to C.
6) Finally, $\mathcal{A}$ produces a bit b' and wins the game if $b' = b$.

The adversary's advantage is defined to be $\mid Adv(\mathcal{A}) := [2Pr[b' = b] - 1 \mid$

Definition 3: If no polynomially bounded adversary has a non-negligeable advantage in the following game, an IBSC scheme is said to be secure against an existential forgery for adaptive chosen messages attacks (EF-IBSC-CMA)

1) The challenger runs the $Setup$ algorithm with a security parameter k and gives the system parameters to the adversary.
2) The adversary $\mathcal{A}$ can query a polynomially bounded number of requests as in the previous definition.
3) Finally, $\mathcal{A}$ produces (σ^*, ID_A, ID_B) which was not produced by the signcryption oracle, where the private key of ID_A was not asked in the second stage and wins the game if the result of $Unsigncrypt(\sigma^*, d_{ID_A}, ID_B)$ is not the $\perp$ symbol.

The adversary's advantage is its probability of victory.

B. Review of Ren et al.'s IBSC Scheme

Here we review Ren et al.'s IBSC scheme in [23]:

1) Setup: G_1, G_2 are groups of order p where p be a large prime number. $e : G_1 \times G_1 \to G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^\alpha$, where $\alpha \in Z_p^*$. α is the private key of PKG. The PKG randomly chooses $h_i \in G_1$, $i = 0, l, 2$. $H : G_2 \to Z_p^*$ is a collision resistent hash function. The public parameters is $param = (g, g_1, h_0, h_1, h_2, H)$ and the master secret key is $msk = \alpha$.
2) KeyGen: For a user U with identity $ID \in Z_p^*$, the PKG randomly chooses $r_{ID} \in Z_p^*$, and outputs

$$d_1 = (h_0 g^{-r_{ID}})^{\frac{1}{\alpha - ID}}, d_2 = r_{ID}$$

so the private key of user U is $d = (d_1, d_2)$. If $ID = \alpha$, aborts.
3) Signcrypt: For a message $m \in G_2$, U randomly chooses $s \in Z_p^*$ and computes

$$c_1 = g_1^s g^{-sID}, c_2 = e(g, g)^s,$$

$$c_3 = m \cdot e(g, h_0)^{-s}, \sigma_1 = d_1 (h_2 h_1^{ID})^{sH(m)}$$

the signcryption of m is $c = (c_1, c_2, c_3, \sigma_1, H(m), r_{ID})$.
4) Unsigncrypt:
   a) The receiver verifies whether

$$\begin{aligned} & e(g_1 g^{-ID}, \sigma_1) \\ = \;& e(g, h_0) e(g, g)^{-r_{ID}} \cdot e(c_1, (h_2 h_1^{ID})^{H(m)}) \end{aligned}$$

   if it is true, c is a true signcryption.
   b) Decrypts $m = c_3 e(c_1, d_1) c_2^{r_{ID}}$

C. Attack to IBSC Scheme

The attacker $\mathcal{A}$ breaks the unforgeability as follows, assume the target identity is ID^*,

1) $\mathcal{A}$ queries a signcryption on m of ID^* to the challenger $\mathcal{B}$, $\mathcal{B}$ will return $c = (c_1, c_2, c_3, \sigma_1, H(m), r_{ID})$.
2) Now $\mathcal{A}$ modifies the signcryption ciphertext to be

$$c_1' = c_1^{\frac{H(m)}{H(m')}}, c_2' = c_2^{\frac{H(m)}{H(m')}}, c_3' = c_3^{\frac{H(m)}{H(m')}}, \sigma_1' = \sigma_1,$$

$$r_{ID}' = r_{ID}, c' = (c_1', c_2', c_3', \sigma_1', H(m'), r_{ID}')$$

where m' is a randomly message chosen from the plaintext place.
3) Then $c' = (c_1', c_2', c_3', \sigma_1', H(m'), r_{ID}')$ is a valid signcryption on m' because

$$\begin{aligned} & e(g_1 g^{-ID}, \sigma_1') \\ = \;& e(g_1 g^{-ID}, (h_0 g^{r_{ID}})^{\frac{1}{\alpha - ID}} \cdot (h_2 h_1^{ID})^{sH(m)}) \\ = \;& e(g, h_0) e(g, g)^{-r_{ID}} e((g_1 g^{-ID})^s, (h_2 h_1^{ID})^{H(m)}) \\ = \;& e(g, h_0) e(g, g)^{-r_{ID}} e((g_1 g^{-ID})^{s \cdot \frac{H(m)}{H(m')}}, (h_2 h_1^{ID})^{H(m')}) \\ = \;& e(g, h_0) e(g, g)^{-r_{ID}} e(c_1', (h_2 h_1^{ID})^{H(m')}) \end{aligned}$$

Clearly, this will break the unforgeability of the signcryption scheme.

Assume the target identity is ID^*, the attacker $\mathcal{A}$ breaks the confidentiality as follows

1) The challenger $\mathcal{B}$ give $\mathcal{A}$ the challenge ciphertext $c = (c_1^*, c_2^*, c_3^*, \sigma_1^*, H(m^*), r_{ID^*}^*)$.
2) Now $\mathcal{A}$ modifies the challenge ciphertext to be

$$c_1' = (c_1^*)^{\frac{H(m^*)}{H(m')}}, c_2' = (c_2^*)^{\frac{H(m^*)}{H(m')}}, c_3' = (c_3^*)^{\frac{H(m)}{H(m')}}, \sigma_1' = \sigma_1^*,$$

$$r_{ID^*}' = r_{ID^*}^*, c' = (c_1', c_2', c_3', \sigma_1', H(m'), r_{ID^*}')$$

where m' is a randomly chosen message.
3) $\mathcal{A}$ queries c' to the unsigncryption oracle of ID^*, and he will get $(m^*)^{\frac{H(m^*)}{H(m')}}$ which is enough to get m^* ($\mathcal{A}$ knows m', $H(m^*)$), for the following

$$\begin{aligned} & c_3' e(c_1', d_1) c_2'^{\prime r_{ID}'} \\ = \;& (c_3^* e(c_1^*, d_1) (c_2^*)^{r_{ID^*}})^{\frac{H(m^*)}{H(m')}} \\ = \;& (m^*)^{\frac{H(m^*)}{H(m')}} \end{aligned}$$

Clearly, this breaks the confidentiality of this signcryption scheme.

D. Improved IBSC Scheme

1) Setup: Let p be a large prime number, G_1, G_2 are groups of order p. $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^{\alpha}$, where $\alpha \in Z_p^*$. α is the private key of PKG. The PKG randomly chooses $h_i \in G_1$, $i = 0, l, 2$. $H : G_2 \rightarrow Z_p^*$ is a collision resistent hash function. Let $H' : \{0,1\}^* \times G_1 \rightarrow Z_p^*$ be a hash function. The public parameters and the master secret key are

$$param = (g, g_1, h_0, h_1, h_2, H, H'), msk = \alpha$$

2) KeyGen: To a user U with identity $ID \in Z_p^*$, the PKG randomly chooses $r_{ID} \in Z_p^*$, and computes

$$d_1 = (h_0 g^{-r_{ID}})^{\frac{1}{\alpha - ID}}, d_2 = r_{ID}$$

so the private key of user U is $d = (d_1, d_2)$. If $ID = \alpha$, aborts.

3) Signcrypt: To signcrypt a message $m \in G_2$, U randomly chooses $s \in Z_p^*$ and computes

$$c_1 = g_1^s g^{-sID}, c_2 = e(g,g)^s,$$

$$c_3 = m \cdot e(g, h_0)^{-s}, \sigma_1 = d_1 (h_2 h_1^{ID})^{sH'(H(m), c_1)}$$

so the signcryption of m is $c = (c_1, c_2, c_3, \sigma_1, H(m), r_{ID})$.

4) Unsigncrypt:

a) The receiver verifies whether

$$\begin{aligned} & e(g_1 g^{-ID}, \sigma_1) \\ = \; & e(g, h_0) e(g,g)^{-r_{ID}} \\ & \cdot e(c_1, (h_2 h_1^{ID})^{H'(H(m), c_1)}) \end{aligned}$$

if it is true, c is a true signcryption.

b) Decrypts $m = c_3 e(c_1, d_1) c_2^{r_{ID}}$

Remark 2: In this scheme, $\sigma_1 = d_1 (h_2 h_1^{ID})^{sH'(H(m), \sigma_2)}$ instead of $\sigma_1 = d_1 (h_2 h_1^{ID})^{sH(m)}$. This little modification can resist the above attack. Any modification on σ_2 will be detected.

IV. HIBS Scheme

A. Definition and Security Model for HIBS Scheme

The following four algorithms express a l-level HIBS scheme:

1) Setup: With a security parameter k, the PKG generates (msk, $param$) where msk is the randomly generated master secret key and $param$ is the corresponding public parameter.

2) KeyGen: With an identity vector ID (where $\mathsf{ID} \leq l$), the PKG outputs the corresponding private key SK_{ID} and gives it to its owner in a secure way. Note its prefix identities can also generate the private key SK_{ID}.

3) Sign: With the private key of the signer ID, SK_{ID} and a message M, it outputs a signature σ corresponding to $param$.

4) Verify: With the signer identity vector ID, a message M and signature σ, it outputs "Valid" if σ is a valid signature of M corresponding to ID, $param$. Otherwise, it outputs $\perp$.

Definition 4: We define the following oracles:

- KGO(ID): With input ID (where $\mathsf{ID} \leq l$), the Key Generation Oracle will output the secret key SK_{ID} corresponding to msk.
- SO(ID, M): With input signer ID (where $\mathsf{ID} \leq l$) and message M, the Signing Oracle will output a signature σ such that $Verify(\mathsf{ID}, M, \sigma) = Valid$.

We define the existential unforgeability against adaptive identity and adaptive chosen message attack for HIBS (EU-ID-CMA), as in the following game.

1) (*Phase 1.*) Challenger $\mathcal{B}$ generates system parameter $param$ and gives it to Adversary $\mathcal{A}$.

2) (*Phase 2.*) $\mathcal{A}$ queries KGO(ID) and SO(ID, M) adaptively.

3) (*Phase 3.*) $\mathcal{A}$ delivers a signature σ^* for signer identity ID^* (where $|ID^*| \leq l$) and message M^*. ID^* or its prefix have never been input to a KGO and (ID^*, M^*) has never been input to a SO.

$\mathcal{A}$ wins if he completes the Game with $Valid = Verify(\mathsf{ID}^*, M^*, \sigma^*)$. Its advantage is its probability of winning.

B. Review of Ren et al.'s HIBS Scheme

Here we review Ren et al.'s HIBS scheme in [24]:

1) Setup: Let G_1, G_2 are groups of order p which is a large prime number. $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^{\alpha}$, where $\alpha \in Z_p^*$. l is the maximum number of levels in the HIBS, The PKG randomly chooses $h_i \in G_1$, $i = 0, \cdots, l$, $H : G_1 \rightarrow Z_p^*$ is a hash function randomly chosen from hash family. The public parameter is $param = (g, g_1, h_0, \cdots, h_l, H)$ and the master secret key is $msk = \alpha$.

2) KeyGen: To a user U with identity $ID = (ID_1, \cdots, ID_i) \in (Z_p^*)^i$, the PKG randomly chooses $\widehat{r}, r_i \in Z_p^*$, where $H(g^{r_i}) \neq \alpha$, and computes

$$d_{0,i} = (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_i})}} (\prod_{k=1}^{i} h_k^{ID_k})^{r_i}, d_{-2,i} = \widehat{r}, d_{-1,i} = g_1^{r_i},$$

$$d_{1,i} = g^{r_i}, d_{i+1,i} = h_{i+1}^{r_i}, \cdots, d_{l,i} = h_l^{r_i}$$

and abandons r_i. The private key of U is $d = (d_{-2,i}, d_{-1,i}, d_{0,i}, d_{1,i}, d_{i+1,i}, \cdots, d_{l,i})$. The private key can also be generated by its parent $(ID_1, \cdots, ID_{i-1})$ having the secret key $(d_{-2,i-1}, d_{-1,i-1}, d_{0,i-1}, d_{1,i-1}, d_{i,i-1}, \cdots, d_{l,i-1})$. It computes

$$d_{0,i} = d_{0,i-1} \cdot d_{i,i-1}^{ID_i}, d_{-2,i} = d_{-2,i-1}, d_{-1,i} = d_{-1,i-1}$$

$$d_{1,i} = d_{1,i-1}, d_{k,i} = d_{k,i-1} (k = i+1, \cdots, l)$$

where $r_i = r_{i-1}$.

3) Sign: U randomly chooses $s \in Z_p^*$ to sign a message $m \in Z_p^*$ and computes

$$\sigma_1 = d_{0,i} \cdot d_{i+1,i}^m \cdot (h_{i+1}^m \cdot \prod_{k=1}^{i} h_k^{ID_k})^s, \sigma_2 = g_1^{r_i+s},$$

$$\sigma_3 = g^{r_i+s}, \sigma_4 = H(g^{r_i}), \sigma_5 = \widehat{r}$$

so the signature of m is $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$.

4) Verify: The receiver verifies whether

$$\begin{aligned} & e(\sigma_2, g) = e(\sigma_3, g_1) \\ & e(g_1 g^{-\sigma_4}, \sigma_1) \\ = \; & e(g, h_0) e(g, g)^{\sigma_5} e(\sigma_2 \sigma_3^{-\sigma_4}, h_{i+1}^m \cdot \prod_{k=1}^{i} h_k^{ID_k}) \end{aligned}$$

holds. It is a valid signature if σ can pass the above verification. Otherwise, it is an invalid signature.

C. Attack to HIBS Scheme

Assume the target identity is a first level identity $\mathsf{ID}^* = [ID_1^*, ID_2^*, \cdots, ID_i^*]$, the adversary $\mathcal{A}$ attacks as follows:

1) First he queries the Key Generation Oracle on a first level identity ID_1 where $ID_1 \neq ID_1^*$ to the challenger $\mathcal{B}$, and $\mathcal{B}$ will return

$$d_{0,1} = (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_1})}} (h_1^{ID_1})^{r_1}, d_{-2,1} = \widehat{r}, d_{-1,1} = g_1^{r_1},$$

$$d_{1,1} = g^{r_1}, d_{2,1} = h_2^{r_1}, \cdots, d_{l,1} = h_l^{r_1}$$

$\mathcal{A}$ can derive a "private key"

$$d_{0,i} = (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_1})}} (h_1^{ID_1})^{r_1} (\prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}})^{r_1}, d_{-2,i} = \widehat{r},$$

$$d_{-1,i} = g_1^{r_1}, d_{1,i} = g^{r_1}, d_{i+1,i} = h_{i+1}^{r_1}, \cdots, d_{l,i} = h_l^{r_1}$$

2) Then $\mathcal{A}$ "signs" message m by this "private key", he will get the signature

$$\sigma_1 = d_{0,i} \cdot d_{i+1,i}^m \cdot (h_{i+1}^m \cdot h_1^{ID_1} (\prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}}))^s,$$

$$\sigma_2 = g_1^{r_1+s}, \sigma_3 = g^{r_1+s}, \sigma_4 = H(g^{r_1}), \sigma_5 = \widehat{r}$$

3) $\mathcal{A}$ modifies this signature to be

$$\sigma_1' = \sigma_1, \sigma_2' = \sigma_2^{\frac{ID_1}{ID_1^*}} = g_1^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}},$$

$$\sigma_3' = \sigma_3^{\frac{ID_1}{ID_1^*}} = g^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}}, \sigma_4' = \sigma_4, \sigma_5' = \sigma_5$$

we will show this is a valid signature for $\mathsf{ID}^* = [ID_1^*, ID_2^*, \cdots, ID_i^*]$ on $(m \cdot \frac{ID_1^*}{ID_1})$.

4) The first equation for verifying will be satisfied. $e(\sigma_2', g) = e(\sigma_3', g_1)$. The second equation for verifying will also be satisfied.

$$\begin{aligned} & e(g_1 g^{-\sigma_4'}, \sigma_1') \\ = \; & e(g_1 g^{-H(g^{r_1})}, d_{0,i} \cdot d_{i+1,i}^m \cdot (h_{i+1}^m \cdot h_1^{ID_1} \\ & \cdot (\prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}}))^s) \\ = \; & e(g_1 g^{-H(g^{r_1})}, (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_1})}} (h_1^{ID_1})^{r_1} \\ & (\prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}})^{r_1} \cdot h_{i+1}^{r_1 m} \cdot (h_{i+1}^m \\ & \cdot h_1^{ID_1} (\prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}}))^s) \\ = \; & e(g_1 g^{-H(g^{r_1})}, (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_1})}}) \cdot e(g_1 g^{-H(g^{r_1})}, \\ & (h_1^{ID_1} \prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}})^{r_1+s}) \cdot e(g_1 g^{-H(g^{r_1})}, \\ & h_{i+1}^{(r_1+s)m}) \\ = \; & e(g, (h_0 g^{\widehat{r}})) \cdot e(g_1^{r_1+s} (g^{r_1+s})^{-H(g^{r_1})}, \\ & h_{i+1}^m h_1^{ID_1} \prod_{k=2}^{i} h_k^{ID_k^* \cdot \frac{ID_1}{ID_1^*}}) \\ = \; & e(g, (h_0 g^{\widehat{r}})) \cdot e(g_1^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}} (g^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}})^{-H(g^{r_1})}, \\ & (h_{i+1}^m h_1^{ID_1})^{\frac{ID_1^*}{ID_1}} \prod_{k=2}^{i} h_k^{ID_k^*}) \\ = \; & e(g, (h_0 g^{\widehat{r}})) \cdot e(g_1^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}} (g^{(r_1+s) \cdot \frac{ID_1}{ID_1^*}})^{-H(g^{r_1})}, \\ & (h_{i+1}^{m \cdot \frac{ID_1^*}{ID_1}} h_1^{ID_1^*} \prod_{k=2}^{i} h_k^{ID_k^*})) \\ = \; & e(g, h_0) e(g, g)^{\sigma_5'} \\ & e(\sigma_2' \sigma_3'^{-\sigma_4'}, h_{i+1}^{m \cdot \frac{ID_1^*}{ID_1}} \cdot \prod_{k=1}^{i} h_k^{ID_k^*}) \end{aligned}$$

D. Improved HIBS Scheme

1) Setup: Let p be a large prime number, G_1, G_2 are groups of order p. $e : G_1 \times G_1 \rightarrow G_2$ is a bilinear map, g is a generator of G_1, $g_1 = g^\alpha$, where $\alpha \in Z_p^*$. l is the maximum number of levels in the HIBS, The PKG randomly chooses $h_i \in G_1$, $i = 0, \cdots, l$, $H : G_1 \rightarrow Z_p^*$ is a Hash function randomly chosen from a family of universal one-way Hash functions. Let $H' : G_1 \times G_1 \times Z_p^* \times Z_p^* \rightarrow Z_p^*$ be a hash function. The public parameters and the master secret key are

$$param = (g, g_1, h_0, \cdots, h_l, H, H'), msk = \alpha$$

2) KeyGen: To a user U with identity $ID = (ID_1, \cdots, ID_i) \in (Z_p^*)^i$, the PKG randomly chooses $\widehat{r}, r_i \in Z_p^*$, where $H(g^{r_i}) \neq \alpha$, and

computes

$$d_{0,i} = (h_0 g^{\widehat{r}})^{\frac{1}{\alpha - H(g^{r_i})}} (\prod_{k=1}^{i} h_k^{ID_k})^{r_i}, d_{-2,i} = \widehat{r}, d_{-1,i} = g_1^{r_i},$$

$$d_{1,i} = g^{r_i}, d_{i+1,i} = h_{i+1}^{r_i}, \cdots, d_{l,i} = h_l^{r_i}$$

and abandons r_i. So the private key of U is $d = (d_{-2,i}, d_{-1,i}, d_{0,i}, d_{1,i}, d_{i+1,i}, \cdots, d_{l,i})$. The private key can also be generated by its parent $(ID_1, \cdots, ID_{i-1})$ having the secret key $(d_{-2,i-1}, d_{-1,i-1}, d_{0,i-1}, d_{1,i-1}, d_{i,i-1}, \cdots, d_{l,i-1})$. It computes

$$d_{0,i} = d_{0,i-1} \cdot d_{i,i-1}^{ID_i}, d_{-2,i} = d_{-2,i-1}, d_{-1,i} = d_{-1,i-1}$$

$$d_{1,i} = d_{1,i-1}, d_{k,i} = d_{k,i-1} (k = i+1, \cdots, l)$$

where $r_i = r_{i-1}$.

3) Sign: To sign a message $m \in Z_p^*$, U randomly chooses $s \in Z_p^*$ and computes

$$\sigma_1 = d_{0,i} \cdot d_{i+1,i}^{H'(m,\sigma_2,\sigma_3,\sigma_4,\sigma_5)} \cdot (h_{i+1}^{H'(m,\sigma_2,\sigma_3,\sigma_4,\sigma_5)} \cdot \prod_{k=1}^{i} h_k^{ID_k})^s,$$

$$\sigma_2 = g_1^{r_i+s}, \sigma_3 = g^{r_i+s}, \sigma_4 = H(g^{r_i}), \sigma_5 = \widehat{r}$$

so the signature of m is $\sigma = (\sigma_1, \sigma_2, \sigma_3, \sigma_4, \sigma_5)$.

4) Verify: The receiver verifies whether

$$\begin{aligned} & e(\sigma_2, g) = e(\sigma_3, g_1) e(g_1 g^{-\sigma_4}, \sigma_1) \\ = \; & e(g, h_0) e(g, g)^{\sigma_5} e(\sigma_2 \sigma_3^{-\sigma_4}, h_{i+1}^{H'(m,\sigma_2,\sigma_3,\sigma_4,\sigma_5)} \\ & \cdot \prod_{k=1}^{i} h_k^{ID_k}) \end{aligned}$$

If σ can pass the above verification, it is a valid signature. Otherwise, it is an invalid signature.

Remark 3: In this scheme, $\sigma_1 = d_{0,i} \cdot d_{i+1,i}^{H'(m,\sigma_2,\sigma_3,\sigma_4,\sigma_5)} \cdot (h_{i+1}^{H'(m,\sigma_2,\sigma_3,\sigma_4,\sigma_5)} \cdot \prod_{k=1}^{i} h_k^{ID_k})^s$ instead of $\sigma_1 = d_{0,i} \cdot d_{i+1,i}^m \cdot (h_{i+1}^m \cdot \prod_{k=1}^{i} h_k^{ID_k})^s$. This little modification can resist the above attack. Any modification on $\sigma_2, \sigma_3, \sigma_4, \sigma_5$ will be detected.

V. PRBE Scheme

A. Definition and Security Model for PBRE Scheme

Here we review the definition and security model of PBRE schemes. A PBRE scheme consists of the following algorithms:

1) Setup: When given a security parameter, outputs both the master public parameters $params$, and the master key mk which is kept private.
2) KeyGen: On input mk and an identity I as input, this algorithm generates a secret key sk_I associated with I.
3) Encrypt: This algorithm takes pk, a message M, and an identity I as input, and generates the second level ciphertext C_I, which can be re-encrypted by the proxy.
4) RKGen: On input master key mk, and receiver set S, the delegator's identity I, he non-interactively generates the re-encryption key $rk_{I \to S}$ and outputs it.
5) Reencrypt: On input a second level ciphertext C under identity I, and a re-encryption key $rk_{I \to S}$, outputs a first level re-encrypted ciphertext C_S.
6) Decrypt$_2$: On input a second level ciphertext C_I under identity I with secret key sk_I, if $I \in S$, decrypts the ciphertext C_I and outputs M, otherwise output $\perp$.
7) Decrypt$_2$: On input a first level ciphertext C_S under identity I with secret key sk_I, if $I \in S$, decrypts the re-encrypted ciphertext C_S and outputs M, otherwise output $\perp$.

Definition 5: The security of an unidirectional IBPBRE scheme is defined according to the following games:

1) Setup: The challenger runs the Setup algorithm and gives pk to the adversary $\mathcal{A}$.
2) Phase 1: Adversary $\mathcal{A}$ makes the following queries.
   a) Extract: $\mathcal{A}$ submits an identity I for a KeyGen query, the challenger gives $\mathcal{A}$ the secret key sk_I.
   b) RKExtract: $\mathcal{A}$ submits an pair (I, S), the challenger gives the re-encryption key $rk_{I \to S}$
   c) Reencrypt: $\mathcal{A}$ submits a ciphertext C_I, the challenger gives the adversary the re-encrypted ciphertext $C_s = Reencrypt(C_I, rk_{I \to S}$ where $rk_{I \to S} = RKGen(sk_I, S)$ and $sk_I = KeyGen(mk, I)$.
   d) Decrypt: $\mathcal{A}$ submits a ciphertext C_I encrypted for I, the challenger gives the corresponding plaintext $M = Decrypt(C_T, sk_I)$, where $sk_I = KeyGen(mk, I)$.
3) Challenge: $\mathcal{A}$ submits a challenge identity I^* and two equal length messages M_0, M_1 to $\mathcal{C}$. If the queries $Extract(I^*)$, $RKExtract(I^*, S)$ and $Extract(S)$ for any identity set S are never made, then $\mathcal{C}$ flips a random coin θ and passes the ciphertext $C^* = Encrypt(params, M_\theta, I)$ to $\mathcal{A}$.
4) Phase2: Phase 1 is repeated with the restriction that $\mathcal{A}$ cannot make the following queries:
   a) Extract(I^*), RKExtract(I^*, S) and Extract(S) for any identity set S;
   b) RKExtract(I^*, S) and Decrypt(C_S, S) for any identity set S and any ciphertext S_S;
5) Guess: $\mathcal{A}$ outputs its guess θ' of θ. The advantage of $\mathcal{A}$ in this game is defined as $Adv_A = |Pr[\theta' = \theta] - 1/2|$.

This is the security model for the second level ciphertext, and the security model for first level ciphertext is similar as the second level one except this time the adversary can query any re-encryption keys, and the first level ciphertext is provided for adversary $\mathcal{A}$ in the challenge phase. The more carefully security model can refer to [21].

B. Review of Sun et al's PBRE Scheme

1) $\mathsf{Setup}(1^k)$: Given a security parameter 1^k, algorithm firstly chooses bilinear groups (G, G_T) of prime order $p \geqq 2^k$, pick generators $g, h, u, u_1, u_2 \leftarrow G$ and set $W = e(g, g)$. Choose a collision-resistant hash function $H : G \times \{0,1\}^n \rightarrow Z_p^*$. A pseudo-random function family $F : G \times \{0,1\}^n \rightarrow Z_p^*$. A pseudo random function family $F : G_T \times G \rightarrow \{0,1\}^{n-n_0} \parallel \{0,1\}^{n_0}$, given a seed in G_T and an input in G, it outputs an n-bit pseudo random string. Here n and n_0 are security parameters. Then, the public parameters $params = (p, G, G_T, g, h, u, u_1, u_2, n, n_0, W, H, F)$.
2) $\mathsf{KeyGen(params)}$: User ID_i picks $r_i \leftarrow Z_p^*$, and sets his public key as $pk_i = g^{r_i}$ and private key as $sk_i = r_i$.
3) $\mathsf{ReKeyGen}(sk_i, S = \{1, \cdots, k\}, pk_j, j \in S)$: Input user ID_i's private key $sk_i = r_i$ and user j's public key $pk_j = g^{r_j}$, this algorithm generates the re-encryption key $rk_{i \rightarrow S} = (\prod_{j=1}^{k} pk_j)^{1/sk_i} = g^{\sum_{j=1}^{k} r_j / r_i}$.
4) $\mathsf{Enc}_2(pk_i, M)$: At the second level, to encrypt a message M under the public key pk_i, the sender first pick $s \leftarrow Z_p^*$, and set $C_1 = h^s, C_2 = pk_i^s$, and $C_3 = F(V, C_1)_{n-n_0} || (F(V, C_1)_{n_0} \bigoplus M)$ for $V = W^s$. Then he picks $t \leftarrow Z_p^*$, compute $v = H(C_1, C_3)$, and $C_4 = (u^v u_1^t u_2)^s$, finally outputs the second level ciphertext $C_i = (t, C_1, C_2, C_3, C_4)$.
5) $\mathsf{Enc}_1(pk_j, M)$: To encrypt a message M under the public key $pk_j, j \in S$ at the first level, the sender first picks $s \leftarrow Z_p^*$, and set $C_1 = h^s, C_2' = \prod_{j=1}^{k} e(pk_j, g)^s, V = W^s$, and $C_3 = [F(V, C_1)]_{n-n_0} \parallel (F(V, C_1)^{n_0} \bigoplus M)$ for $V = W^s$. Then he picks $t \leftarrow Z_p^*$, compute $v = H(C_1, C_3)$, and $C_4 = (u^v u_1^t u_2)^s$, finally outputs the second level ciphertext $C_i = (t, C_1, C_2', C_3, C_4)$.
6) $\mathsf{ReEnc}\ (rk, C)$: Input an second level ciphertext $C_i = (t, C_1, C_2, C_3, C_4)$ under public key pk_i, a re-encryption key $rk_{i \rightarrow S}$, and compute $h = H(C_1, C_3)$, then check the validity of C_i by testing whether the following equalities hold: $e(C_1, u^v u_1^t u_2) = e(C_4, h)$, $e(C_1, pk_i) = e(C_2, h)$. If not, output $\perp$. Otherwise, compute $C_2' = e(C_2, rk_{i \rightarrow S})$, and output the first level ciphertext under public key pk_j as $C_j = (t, C_1, C_2', C_3, C_4)$. Where the verification of equations can be alternately done by picking $s_1, s_2 \in Z_p^*$ and testing if $e(C_1, pk_i^{s_1} (u^v u_1^t u_2)^{s_2}) = e(C_2^{s_1} C_4^{s_2}, h)$.
7) $\mathsf{Dec}_2(sk_i, C_i)$: User ID_i with private key sk_i proceeds as follows to decrypt a second level ciphertext $C_i = (t, C_1, C_2, C_3, C_4)$: first check the validity of the ciphertext as in verification equations if the verification fails, output $\perp$; then compute $V = e(C_2, g)^{1/sk_i}$ and output $M = [F(V, C_1)]^{n_0} \bigoplus [C_3]^{n_0}$ if $[F(V, C_1)]^{n-n_0} = [C_3]_{n-n_0}$ holds; else output $\perp$.
8) $\mathsf{Dec}_1(sk_j, C_j)$: Input a first level ciphertext $C_j = (t, C_1, C_2', C_3, C_4)$ and a private key sk_j, user ID_j with private key sk_j proceeds as follows: first check the validity of the ciphertext as in verification equations if the verification fails, output $\perp$; then compute $V = (\frac{C_2'}{e(\prod_{j \neq i} pk_i, g)^s})^{1/sk_i}$ and output $M = [F(V, C_1)]^{n_0} \bigoplus [C_3]^{n_0}$ if $[F(V, C_1)]^{n-n_0} = [C_3]_{n-n_0}$ holds; else output $\perp$.

C. The Flaw in Sun et al's PBRE Scheme

In the $\mathsf{Dec}_1(sk_j, C_j)$ algorithm, user ID_j needs to compute $V = (\frac{C_2'}{e(\prod_{j \neq i} pk_i, g)^s})^{1/sk_i}$, but from $C_j = (t, C_1, C_2', C_3, C_4)$, he can not compute $e(\prod_{j \neq i} pk_i, g)^s$, because he can not know $\prod_{j \neq i} {pk_i}^s$ or g^s. Thus the user can not decrypt the first level ciphertext, which implies Sun et al's PBRE scheme can not be correct.

Furthermore, the re-encryption key $rk_{i \rightarrow S} = (\prod_{j=1}^{k} pk_j)^{1/sk_i} = g^{\sum_{j=1}^{k} r_j / r_i}$ maybe be not in a good form, for there maybe exist two sets S and S' such that $rk_{i \rightarrow S} = (\prod_{j=1}^{k} pk_j)^{1/sk_i} = g^{\sum_{j=1}^{k} r_j / r_i}$ equal to $rk_{i \rightarrow S'} = (\prod_{j=1}^{k} pk_j')^{1/sk_i} = g^{\sum_{j=1}^{k} r_j' / r_i}$ if and only if $\prod_{j=1}^{k} pk_j = \prod_{j=1}^{k} pk_j'$.

VI. Conclusion

In this paper, we cryptanalyze and improve three signature/signacryption schemes which proposed by Ren et al. at ISDPE'07 and Chinacrypt'08 [23], [24]. Probably the recent "dual system encryption" [25], [26] technique can help to construct new novel efficient (H)IBS/IBSC schemes in the standard model. Furthermore, we point out sun et al.'s PBRE scheme [22] can not be correct, and we need more considerations for constructing efficient PBRE schemes.

References

[1] A. Shamir, "Identity-based cryptosystems and signature Schemes," in: G.R. Blakley, D. Chaum (Eds.), *Advances in Cryptology – CRYPTO 1984*, Springer-Verlag, Berlin, **196**, 47–53 (1984).

[2] D. Boneh and M. Franklin, "Identity based encryption from the Weil pairing," in: J. Kilian (Eds.), *Advances in Cryptology – CRYPTO 2001*, Springer-Verlag, Berlin, **2139**, 213–229 (2001).

[3] D. Boneh and X. Boyen, "Efficient selective-id secure identity based encryption without random oracles," in: C. Cachin, J. Camenisch (Eds.), *Advances in Cryptology – EUROCRYPT 2004*, Springer-Verlag, Berlin, **3027**, 223–238 (2004).

[4] D. Boneh and X. Boyen, "Secure identity based encryption without random oracles," in: M. Franklin (Eds.), *Advances in Cryptology – CRYPTO 2004*, Springer-Verlag, Berlin, **3152**, 443–459 (2004).

[5] B. Waters, "Efficient identity-based encryption without random oracles," in R. Cramer (Eds.), *Advances in Cryptology – EUROCRYPT 2005*, Springer-Verlag, Berlin, **3494**, 114–127 (2005).

[6] C. Gentry, "Practical identity-based encryption without random oracles," in: S. Vaudenay (Eds.), *Advances in Cryptology – EUROCRYPT 2006*, Springer-Verlag, Berlin, **4004**, 445–464 (2004).

[7] K. Paterson, "Id-Based Signatures from Pairings on Elliptic Curves," *IEEE Communications Letters,* **38**(18): 1025-1026 (2002).

[8] F. Hess, "Efficient identity based signature schemes based on pairings,"in: K. Nyberg and H. Heys (Eds.), *Annual International Workshop on Selected Areas in Cryptography (SAC)*, Springer-Verlag, Berlin, **2595**, 310–324 (2002).

[9] K. Paterson and J. Schuldt, "Efficient identity-based signatures secure in the standard model," in: L. Batten and R. Safavi-Naini (Eds.), *Australasian Conference on Information Security and Privacy (ACISP)*, Springer-Verlag, Berlin, **4058**, 207–222 (2006).

[10] Y. Zheng, "Digital signcryption or how to achieve cost(signature and encryption) $\ll$ cost (signature) + cost (encryption)," in B. Kaliski (Eds.), *Advances in Cryptology – CRYPTO 1997*, Springer-Verlag, Berlin, **1294**, 165–179 (1997).

[11] J. Malone-Lee, "Identity based signcryption", http://eprint.iacr.org/2002/098 (2002).

[12] L. Chen and J. Malone-Lee, "Improved identity- based signcryption," in: S. Vaudenay (Eds.), *International Conference on Theory and Practice of Public Key Cryptography (PKC)*, Springer-Verlag, Berlin, **3386**, 362–379 (2005).

[13] J. Horwitz and B. Lynn, "Toward hierarchical identity-based encryption," in: L.R. Knudsen (Eds.), *Advances in Cryptology – EUROCRYPT 2002*, Springer-Verlag, Berlin, **2332**, 466–481 (2002).

[14] C. Gentry and A. Silverberg, "Hierarchical id-based cryptography," in: Y. Zheng (Eds.), *Advances in Cryptology – ASIACRYPT 2002*, Springer-Verlag, Berlin, **2501**, 548–566 (2002).

[15] S. Chow, L. Hui, S. Yiu and K. Chow, "Secure hierarchical identity based signature and its application," in: J. Lopez, S. Qing, and E. Okamoto (Eds.), *International Conference on Information and Communication Security (ICICS)*, Springer-Verlag, Berlin, **3269**, 480–494 (2004).

[16] T. Yuen and V. Wei, "Constant-size hierarchical identity based signature/signcryption without random oracles," http://eprint.iacr.org/2005/412 (2005).

[17] M. Blaze, G. Bleumer, M. Strauss. Divertible protocols and atomic proxy cryptography. In *EUROCRYPT 1998*, volume 1403 of *LNCS*, pages 127–144, 1998.

[18] G. Ateniese, K. Fu, M. Green and S. Hohenberger. Improved proxy re-encryption schemes with applications to secure distributed storage. In *ACM NDSS 2005*, pages 29–43, 2005.

[19] J. Weng, R. H. Deng, C. Chu, X. Ding, and J. Lai. Conditional proxy re-encryption secure against chosen-ciphertext attack. In *ACM ASIACCS 2009*, Pages 322–332, 2009.

[20] J. Weng, Y. Yang, Q. Tang, R. Deng, and F. Bao. Efficient conditional proxy re-encryption with chosen-ciphertext security. In *ISC 2009*, volume 5735 of *LNCS*, pages 151–166, 2009.

[21] C. Chu, J. Weng, S. Chow, et al. Conditional proxy broadcast re-encryption. In *ACISP 2009*, volume 5594 of *LNCS*, pages 327–334, 2009.

[22] J. Sun, Y. Hu. CCA-secure unidirectional proxy broadcast re-encryption in the standard model. In *Journal of Computational Information Systems* 8:14 (2012) 5909–5916.

[23] Y. Ren and D. Gu, "Efficient identity based signature/signcryption scheme in the standard model," *Fisrt International Symposium on Data, Privacy and E-commerce (ISDPE)*, 133–137 (2007).

[24] Y. Ren and D. Gu, "Efficient hierarchical identity based signature scheme in the standard model," *Wuhan University Journal of Natural Sciences, Also in ChinaCrypt'08,*, **13**(6): 665-669 (2008).

[25] B. Waters, "Dual system encryption: realizing fully secure ibe and hibe under simple assumptions," in S. Halevi (Eds.), *Advances in Cryptology – CRYPTO 2009*, Springer-Verlag, Berlin, **5667**, 619–636 (2009).

[26] A. Lewko and B. Waters, "New techniques for dual system encryption and fully secure hibe with short ciphertexts," in: D. Micciancio (Eds.), *Theory of Cryptography Conference (TCC)*, Springer-Verlag, Berlin, **5978**, 455–479 (2010).

[27] M. Luo, C. Zou, J. Xu. An efficient identity-based broadcast signcryption scheme. In *Journal of Software*, pages 366-373, Vol. 7, Num. 2, 2012.

[28] Q. Wu, W. Wang. New identity-based broadcast encryption with constant ciphertexts in the standard model. In *Journal of Software*, 1929-1936 Volume 6, Number 10, 2011.

[29] X. Wang, W. Zhong, H. Luo, "Cryptanalysis of identity based signature/signcryption schemes in the standard model," in: *2010 International symposium on Intelligence Information Processing and Trusted Computing (IPTC 2010)*, IEEE Press, 622-625. (2010).

[30] X. Wang, W. Zhong, H. Luo, "Cryptanalysis of an hierarchy identity based signature scheme in the standard model," in: *2010 International symposium on Intelligence Information Processing and Trusted Computing (IPTC 2010)*, IEEE Press, 619-621. (2010).