Fast and Proven Secure Blind Identity-Based Signcryption from Pairings

Tsz Hon Yuen and Victor K. Wei

Department of Information Engineering
The Chinese University of Hong Kong
Shatin, Hong Kong
`thyuen1,kwwei@ie.cuhk.edu.hk`

Abstract. We introduce the first blind identity-based signcryption (BIBSC). The Warden and the blind signcryption oracle conduct a 3-move interactive protocol, in which the oracle commits and then blindly sign any challenge, and Warden outputs a signcryption untraceable from the 3-move conversation. We introduce security models to define the security notions of blindness and parallel one-more unforgeability (`p1m-uf`) by active attackers. We present an efficient construction from pairings, then prove its security in the random oracle model. The `p1m-uf` is reduced to Schnorr's ROS Problem or the co-CDH Problem. In the process, we also introduce a new security model for (non-blind) identity-based signcryption (IBSC) which is a strenthening of Boyen's. We construct protocols proven secure in that model which has no more complexity (counting the number of pairings and exponentiations) and cost no more bandwidth than any secure IBSC in the literature. We also show several existing IBSC schemes failing our new security model.

1 Introduction

Identity based cryptography is a kind of asymmetric key cryptography using recipient's identity as the public key. In 1984, Shamir [19] firstly proposed the idea of identity based cryptography. Since then, there are many suggestions for the implementation of identity based encryption ([12], [21], [16], [10]). However they are not fully satisfactory. In 2001, Boneh and Franklin [4] proposed the first practical identity based encryption scheme using pairings on elliptic curves.

The basic idea of identity based cryptography is to use the recipient's identity as the public key. The identity can be name, email address or combining any other strings that can help to identify a person uniquely. Usually a trusted authority (TA) is needed to generate private keys according to the public keys. The advantage of identity based cryptography over traditional public key cryptography is that distribution of public key in advance is not needed.

Since the first practical identity based encryption scheme was proposed in 2001 [4], there are many new development in identity based cryptography, like identity based signatures [6], authenticated key agreement [20], [8]. Identity-based encryptions prior to that result either requires high complexity to compute the key pair (e.g. RSA-based) or is insecure against colluders who can jointly extract one more secret key (e.g. DL-based).

Privacy and authenticity are the basic aims of public-key cryptography. We have encryption and signature to achieve these aims. There are many researches for encryption or signature separately. Yet, there are some applications that requires the use of both encryption and signature, like signing an e-mail and then encrypted before sent. Zheng [25] proposed that encryption and signature can be combined as ”signcryption” which can be more efficient in computation than running encryption and signature separately. The security of signcryption is discussed by An et al. [1]

1.1 Contributions

We introduce the literature's first blind identity-based signcryption (BIBSC). Upon request from Warden, a blind signcryption oracle makes a commitment, then blindly signs and computes the randomness term in the encryption for Warden. Warden deblinds and proceeds to produce a signcryption untraceable from the conversation with the oracle.

We formulate the first BIBSC security models to define blindness and to define parallel one-more unforgeability by active adversary (p1m-UF).

We present the first BIBSC from pairings, and prove its security. The blindness of our BIBSC from pairings is statistical ZK, and the p1m-UF is reduced to Schnorr's ROS Problem or the co-CDH Problem, in the random oracle model.

We also introduce a strengthening of Boyen's security model for (non-blind) identify-based signcryption (IBSC) to support authenticated encryption. We give an efficient and secure constructing satisfying the strengthened model. It has no more complexity (in terms of pairings and exponentiations) and costs no more bandwidth than any secure IBSC in the literature. The shortcomings of several existing IBSC in the strengthened model are shown.

1.2 Organization

In Section 2, we will define preliminaries. In Section 3, we will define the IBSC and BIBSC security model. In Section 4, we will introduce our schemes. In Section 5, we will compare our scheme with existing schemes. In Section 6, we will introduce the additional functionalities of our scheme.

2 Preliminaries

2.1 Related Results

Shamir [19] suggested an identity based signature scheme. Boneh and Franklin [4] proposed an identity based encryption scheme. There are some papers [15], [5], [13], [11], [9], [14] concerning the combination of signature and encryption to form a new IBSC scheme. The advantage of IBSC is that it involves less computation and usually has a shorter ciphertext than using encryption and signature scheme separately.

Let us consider the efficiency and proven security of known IBSC schemes from pairings. The most expensive single operation is the pairing computations. The scheme of [15], [5] and [14] use 5 pairings, while [13] and [9] use 6, [11] uses 4. The scheme of [5] is proven secure in a stronger model than [15] and [13]. The scheme of [11] has no security proof. The detailed comparison of our scheme and other schemes will be discussed in Section 5.

The concept of blind signatures was introduced by Chaum [7], which provides anonymity of users in applications such as e-cash. It allows users to get a signature of a message in a way that the signer learns neither the message nor the resulting signature. Some ID-based blind signature schemes is developed recently [22], [23], [24].

2.2 Pairings

Some candidate hard problems from pairings that will be used later.

Definition 1. *(co-BDH problem)The co-Bilinear Diffie-Hellman problem is, given $P, P^{\alpha}, P^{\beta} \in G_1$, $Q \in G_2$, for unknown $\alpha, \beta \in Z_q$, to compute $e(P,Q)^{\alpha\beta}$.*

Definition 2. *(co-CDH problem) The co-Computational Diffie-Hellman problem is, given $P, P^{\alpha} \in G_1$, $Q \in G_2$ for unknown $\alpha \in Z_q$, to compute Q^{α}.*

2.3 Blind signatures and Schnorr's ROS Problem

Schnorr [18] reduced the parallel one-more unforgeability of the blind Schnorr signature (resp. blind Okamoto-Schnorr signature) to the ROS Problem in the random oracle (RO) plus generic group model.

Definition 3. *(ROS problem) Find an overdetermined, solvable system of linear equations modulo q with random inhomogeneities. Specifically, given an oracle random function $F : Z_q^l \leftarrow Z_q$, find coefficients $a_{k,i} \in Z_q$ and a solvable system of $l + 1$ distinct equations of Eq. (1) in the unknowns $c_1, \ldots, c_l$ over Z_q:*

$$a_{k,1}c_1 + \ldots + a_{k,l}c_l = F(a_{k,1}, \ldots, a_{k,l}) \text{ for } k = 1, \ldots, t. \quad (1)$$

2.4 Extended generic group model for pairings

We will need to extend the generic group model (GMM) to include pairing-related operations. Briefly, we add the paring $\hat{e}(\cdot,\cdot)$, the homomorphic mapping $\phi : G_2 \rightarrow G_1$. There are three kinds of group elements, in G_1, G_2, or G_T. Non-group date are integers. Each computation step is a computation of the form

$$(g_{1,1}, \cdots, g_{1,n_1}, g_{2,1}, \cdots, g_{2,n_2}, g_{3,1}, \cdots, g_{3,n_3}, a_1, \cdots, a_{n_1}, b_1, \cdots, b_{n_2}, c_1, \cdots, c_{n_3}) \\ | \rightarrow (\prod_i g_{1,i}^{a_i}, \prod_j g_{2,j}^{b_j}, \prod_k g_{3,k}^{c_k})$$

or a pairing $\hat{e}$, or a mapping ϕ.

Full details will be provided later.

3 Introducing BIBSC Security Model and Enhancing IBSC Security Model

We define the first security models for BIBSC (Blind Identity-Based SignCryption). We also define an enhancement of Boyen's security model for IBSC (Identity-Based SignCryption). For logistics, we present the latter first.

3.1 Enhanced IBSC Security Model

3.1.1 Primitives An IBSC scheme consists of four algorithms: (Setup, Extract, Signcrypt, Unsigncrypt). The algorithms are specified as follows:
Setup: On input a security parameter k, the TA generates $\langle\zeta,\pi\rangle$ where ζ is the randomly generated master key, and π is the corresponding public parameter.
Extract: On input ID, the TA computes its corresponding private key S_{ID} (corresponding to $\langle\zeta,\pi\rangle$) and sends back to its owner in a secure channel.
Signcrypt: On input the private key of sender A, S_A, recipient identity ID_B and a message m, outputs a ciphertext σ corresponding to π.
Unsigncrypt: On input private key of recipient B, S_B, and ciphertext σ, decrypt to get sender identity ID_A, message m and signature s corresponding to π. Verify s and verify if encryptor = signer. Output $\top$ for "true" or $\perp$ for "false".

We make the consistency constraint that if $\sigma \leftarrow \mathit{Signcrypt}(S_A, ID_B, m)$, then $m \leftarrow \mathit{Unsigncrypt}(S_B, \sigma)$.

3.1.2 Indistinguishability Indistinguishability for IBSC against adaptive chosen ciphertext attack (IND-IBSC-CCA2) is defined as in the following game. It is similar to the IND-CCA2 for traditional public key encryption scheme.

In this game, the Adversary is allowed to query the random oracles, key extraction oracle, signcryption oracle and unsigncryption oracle adaptively. The game is defined as follows:

1. Simulator selects the public parameter and sends to the Adversary.
2. Adversary performs polynomial number of oracle queries adaptively.
3. Adversary generates m_1, ID_{A1}, ID_{B1}, and sends to Simulator. Adversary knows S_{A1}. Simulator generates m_0, ID_{A0}, ID_{B0}, randomly chooses $b \in_R \{0,1\}$. Simulator delivers $\sigma \leftarrow \mathit{Signcrypt}(S_{Ab}, ID_{Bb}, m_b)$ to Adversary.
4. Adversary performs polynomial number of oracle queries adaptively.
5. Adversary tries to compute b, in the following three sub-games
   (a) Simulator ensures $B0 = B1$, $m_0 = m_1$, Adversary computes b.
   (b) Simulator ensures $A0 = A1$, $m_0 = m_1$, Adversary computes b.
   (c) Simulator ensures $A0 = A1$, $B0 = B1$, Adversary computes b.

The Adversary wins the game if he can guess b correctly.

The oracles are defined as follows:
Key extraction oracle $\mathcal{KEO}$: Upon input an identity, the key extraction oracle outputs the private key corresponding to this identity.

Signcryption oracle $\mathcal{SO}$: Upon input m, ID_A, ID_B, produce valid signcryption σ for the triple of input.
Unsigncryption oracle $\mathcal{UO}$: Upon input ciphertext σ and receiver ID, the unsigncryption oracle outputs the decryption result, verification outcome of signature and verification outcome of encryptor=signer.

Oracle query to $\mathcal{KEO}$ to extract private key of ID_{B0}, ID_{B1} is not allowed. Oracle query to $\mathcal{SO}$ for m_1, ID_{A1}, ID_{B1} is not allowed. Oracle query to $\mathcal{UO}$ for the challenge ciphertext from Simulator is not allowed.

The *advantage* of the adversary is the probability, over half, that he can compute b accurately.

Definition 4. *(Indistinguishability) The IBSC is* IND-IBSC-CCA2 *secure if no PPT adversary has non-negligible advantage in any of the three sub-games above.*

In this game, the Adversary is allowed to know the private key of sender S_A of the challenge ciphertext. This gives us a strong *insider-security* for indistinguishability in [1]. Notice that the original definition for indistinguishability for IBSC in Malone-Lee's [15] paper is similar to the IND sub-game-c (IND-C) here. We combine the security model of "anonymity" for IBSC in Boyen's [5] paper as the IND sub-game-a (IND-A) and IND sub-game-b (IND-B). This new IND definition provides a more comprehensive view of indistinguishability.

3.1.3 Existential unforgeability

Existential unforgeability against adaptive chosen message attack for identity based signcryption (EU-IBSC-CMA) is defined as in the following game. It is similar to the EU-CMA for traditional signature scheme.

In this game, the Adversary is allowed to query the random oracles, $\mathcal{KEO}$, $\mathcal{SO}$ and $\mathcal{UO}$ adaptively. The definition for oracles are same as above section. The game is defined as follows:

1. Simulator selects the public parameter and sends to the Adversary.
2. Adversary performs polynomially number of oracle queries adaptively.
3. Adversary delivers valid (σ, ID_B) where σ is not produced by any signcryption oracle query, and Adversary never extracted the secret key of ID_A.

The Adversary wins the game if he can produce a valid (σ, ID_B) that can be decrypted, under the private key of ID_B, to a message m, sender identity ID_A and a signature s. It is required that s pass the verification test for ID_A, and σ passes the verification that tests if encryptor = signer.

Oracle query to $\mathcal{KEO}$ to extract private key of ID_A is not allowed. The Adversary's answer (σ, ID_B) should not be computed by the $\mathcal{SO}$ before.

Definition 5. *(Existential Unforgeability) A signcryption is secure against* EU-IBSC-CMA *if no PPT adversary has non-negligible in successful completion of the game above.*

The Adversary is allowed to ask private key of ID_B, the recipient identity in the Adversary's answer. This gives us a strong *insider-security* for existential unforgeability in [1]. It is stronger than Boyen's [5] existential unforgeability in the sense that our model provides non-repudiation for the ciphertext while Boyen's provides non-repudiation for the decrypted signature only.

3.2 Introducing BIBSC security model

We will propose a blind version of IBSC and then define the security against parallel one-more forgery for ciphertext attack for blind IBSC (BIBSC). It means that any adversary cannot produce $L+1$ ciphertext from L valid ciphertext.

3.2.1 Primitives A BIBSC is a five-tuple (Setup, Extract, BlindSigncrypt, Warden, Unsigncrypt) where Setup, Extract and Unsigncrypt primitives are identical as primitives in IBSC. (BlindSigncrypt, Warden) is a 3-move interactive protocol. Input to BlindSigncrypt is sender's identity ID_A and private key S_A, and recipient's identity ID_B. Input to Warden is ID_A, ID_B and a message m. The 3-move interactive protocol is as follows:

1. BlindSigncrypt sends a commit X to Warden.
2. Warden challenges BlindSigncrypt with h.
3. BlindSigncrypt sends back the response W and V to Warden.

Finally Warden outputs a ciphertext σ.

3.2.2 Blindness We give a formal definition of the blindness of BIBSC scheme. Adversary makes q_B query to blind signcryption oracle $\mathcal{BSO}$, q_H query to random oracles, q_S query to $\mathcal{SO}$, and q_U query to $\mathcal{UO}$. Let the Adversary keeps the transcript $\mathcal{T}$of the interaction between BlindSigncrypt and Warden. Then given a valid ciphertext $\sigma = (X, Y, Z)$, we say that BIBSC is blind if:

$$Prob\{\sigma \; by \; Warden\} = Prob\{\sigma \; by \; Warden | \mathcal{T}\}$$

3.2.3 Parallel One-more Unforgeability Parallel one-more unforgeability for BIBSC (p1m-UF) is defined as in the following game. It is similar to the one-more forgery for traditional blind signature scheme [2], [3], [24].

The game is defined as follows:

1. Sender identity ID_A is given to Adversary.
2. Adversary makes a total of q_B queries to blind signcryption oracles $\mathcal{BSO}_{ID_k}$, $1 \leq k \leq K$, and q_H (resp. q_S) queries to random (resp. Signcryption) Oracle.
3. Adversary delivers $q_B + 1$ tuples (ID_i, m_i, σ_i) to Simulator, $1 \leq i \leq q_B + 1$.

The Adversary wins the game if he can produce $q_B + 1$ valid tuples (ID_i, m_i, σ_i) that can decrypts, under the private key of ID_i, to message m_i and sender identity ID_A.

The $\mathcal{UO}$and $\mathcal{KEO}$are same as the one in IBSC. It is required that the private key of ID_A is never extracted by $\mathcal{KEO}$. We have the new interactive $\mathcal{BSO}$: $\mathcal{BSO}_{ID_A}$: Upon input ID_B, it returns a number X. Then inputs a number h. It produces an output (W, V) based on sender ID_A, recipient ID_B, X and h.

Definition 6. *(Parallel One-more Unforgeability) The* advantage *of the adversary is the probability that he can produce* $q_B + 1$ *distinct pairs of* (ID_{Bi}, σ_i) *to win the above game. The BIBSC is* p1m-UF *secure if no PPT adversary has non-negligible advantage in this game.*

4 Efficient and Secure BIBSC (resp. IBSC) Schemes from Pairings

We present our constructions of efficient and secure BIBSC and IBSC schemes. For logistics of presentation, we present the IBSC first.

4.1 A new efficient and secure IBSC scheme

This IBSC scheme follows the primitives in Section 2. Let G_1, G_2 be two (multiplicative) cyclic groups of prime order p. The bilinear mapping is given as $e : G_1 \times G_2 \rightarrow G_T$. Then for all $P \in G_1$, $Q \in G_2$ and for all $a, b \in Z$ we have $e(P^a, Q^b) = e(P, Q)^{ab} = e(P^b, Q^a)$. Now we define our scheme as follows.

Setup: The setup of the TA is similar to the setup in [4]. On inputting a security parameter $n \in N$, the BDH parameter generator $G[1^n]$ will generates G_1, G_2, G_T, p and e. The TA chooses a generator $P \in G_1^*$ and pick a random $s \in Z_p^*$ as master key. Then the TA sets $P_{TA} = P^s$. After that the TA chooses cryptographic hash functions $H_0 : \{0,1\}^* \rightarrow G_2^*, H_1 : \{0,1\}^* \times G_2^* \rightarrow F_p^*, H_2 : G_T \rightarrow \{0,1\}^*, H_3 : G_T \times \{0,1\}^* \rightarrow \{0,1\}^*$. The system parameters are $\langle p, G_1, G_2, G_T, e, P, P_{TA}, H_0, H_1, H_2, H_3 \rangle$.
Extract: Given a user with identity string $ID \in \{0,1\}^*$. His public key is $Q_{ID} = H_0(ID) \in G_2^*$. His private key S_{ID} is calculated by the TA where $S_{ID} = (Q_{ID})^s$.

Signcrypt: Suppose Alice wants to signcrypt a message m to Bob. Alice firstly signs the message and then encrypts it and sends to Bob.

- Sign: Assume Alice's identity is ID_A. The public key and private key of Alice are Q_A and S_A respectively. Alice chooses a random $r \in F_p^*$ and computes:

$$\begin{aligned} X &= P^r \\ h &= H_1(m, X) \oplus ID_B \\ W &= {S_A}^h {Q_A}^r \end{aligned}$$

  Alice forwards the parameters $\langle X, W, m, r \rangle$ for using in Encrypt.
- Encrypt: Assume Bob's identity is ID_B. Alice computes:

$$\begin{aligned} Q_B &= H_0(ID_B) \\ V &= e({P_{TA}}^r, Q_B) \\ Y &= H_3(V, ID_A) \oplus W \\ Z &= H_2(V) \oplus \langle ID_A, m \rangle \end{aligned}$$

Alice outputs ciphertext $\sigma = \langle X, Y, Z \rangle$ after encryption and sends to Bob.

Unsigncrypt: Bob receives the ciphertext and decrypts it. After that Bob verifies if the signature is indeed come from Alice.

– Decrypt: Assume the private key of Bob is S_B from Extract. Let $\sigma = \langle X, Y, Z \rangle$ be the ciphertext received. Bob decrypts by computing:

$$\begin{aligned} V' &= e(X, S_B) \\ \langle ID_A, m \rangle &= H_2(V') \oplus Z \end{aligned}$$

Output $\langle ID_A, m \rangle$ together with $\langle X, Y, V' \rangle$ to Verify.

– Verify: Alice verifies the signature by computing $W' = H_3(V', ID_A) \oplus Y$. Accept the message if:

$$e(P, W') = e(X{P_{TA}}^h, Q_A) \quad \text{where } h = H_1(m, X) \oplus ID_B$$

Output $\top$ if the above verification is true, or output $\perp$ if false.

In Section 3.1, the Unsigncrypt requires decryption of the ciphertext, verification of the signature, and verification for checking encryptor = signer. The first two parts are done in the previous steps. The last one is implicitly done in Decrypt and Verify as both of them use the same X in σ to decrypt and verify.

Finally, we show the consistency constraint is satisfied in Decrypt and Verify. In Decrypt, V can be recovered as:

$$e(X, S_B) = e(P^r, {Q_B}^s) = e(P^{rs}, Q_B) = e({P_{TA}}^r, Q_B)$$

In Verify, if the signature is valid, both sides should be equivalent because:

$$e(P, W) = e(P, {S_A}^h {Q_A}^r) = e(P, {Q_A}^{(sh+r)}) = e(P^{(r+sh)}, Q_A) = e(X{P_{TA}}^h, Q_A)$$

We find that our IBSC scheme satisfy security models for indistinguishability and existential unforgeability. The security analysis results are given as follows:

Theorem 1. *Our IBSC scheme is* IND-IBSC-CCA2 *secure provided the co-BDH Problem is hard in the random oracle model.*

Theorem 2. *Our IBSC scheme is* EU-IBSC-CMA *secure provided the co-CDH Problem is hard, in the random oracle model.*

The security proof for the above theorems will be given in Appendix A.

Ciphertext unlinkability (CU) and authenticated encryption (AE) One of the main difference between our scheme in previous section and Boyen's scheme [5] is that our scheme has linkability (AE) while Boyen's scheme has unlinkability (CU). As unlinkability may also be important in some applications, we provide the CU version of our scheme.

The only change to our scheme is to change h in Sign into $h = H_1(m, X)$. All other steps remains the same. Therefore this unlinkable version is as efficient as the original AE version.

Notice that by changing to CU, unforgeability for ciphertext reduces to unforgeability for signature only, as in [5]. Other security levels remains the same as AE version.

4.2 The first BIBSC scheme

In this BIBSC, the Setup, Extract and Unsigncrypt are same as Section 4.1. Now, we describe the interactive protocol for BlindSigncrypt and Warden in the following table:

BlindSigncrypt		Warden
randomly choose r		
send $X = P^r$	$\longrightarrow$	
		randomly choose α, β
		computes $\hat{X} = X^\alpha P^\beta$, $\hat{h} = H(m, \hat{X})$
	$\longleftarrow$	send $h = \alpha^{-1}\hat{h}$
send $W = {S_A}^h {Q_A}^r$, $V = e({P_{TA}}^r, Q_B)$	$\longrightarrow$	
		computes $\hat{W} = W^\alpha {Q_A}^\beta$
		computes $\hat{V} = V^\alpha e({P_{TA}}^\beta, Q_B)$
		computes $\hat{Y} = H_3(\hat{V}, ID_A) \oplus \hat{W}$
		computes $\hat{Z} = H_2(\hat{V}) \oplus \langle ID_A, m\rangle$
		outputs $\sigma = \langle \hat{X}, \hat{Y}, \hat{Z}\rangle$

Consistency is verified as:

$$\begin{aligned} e(P, \hat{W}) &= e(P, W^\alpha {Q_A}^\beta) & \text{and } \hat{V} &= V^\alpha e({P_{TA}}^\beta, Q_B) \\ &= e(P, Q_A)^{s\hat{h}+\alpha r+\beta} & &= e(P^{s(r\alpha+\beta)}, Q_B) \\ &= e({P_{TA}}^{\hat{h}} X^\alpha P^\beta, Q_A) & &= e(X^\alpha P^\beta, S_B) \\ &= e(\hat{X} {P_{TA}}^{\hat{h}}, Q_A) & &= e(\hat{X}, S_B) \end{aligned}$$

For the BIBSC, we have the following security analysis:

Theorem 3. *Our BIBSC scheme is blind even if the transcript of* BlindSigncrypt *is given.*

Theorem 4. *Our BIBSC scheme is* p1m-UF-BIBSC *secure provided Schnorr's ROS Problem is hard and the co-CDH Problem is hard, in the random oracle model plus the extended generic group model for pairings.*

The security proof for the above theorems will be given in Appendix A.

5 Comparing Performance

In this Section, we will compare our IBSC scheme with existing schemes from Malone-Lee(M) [15], Libert and Quisquater scheme 1(LQ1) [13] and 2(LQ2)[14], Nalla and Reddy(NR) [11], Boyen(B) [5], and Chow et al.(CYSC) [9]. We also include the Sign-then-Encrypt(StE) and Encrypt-then-Sign(EtS) using ID-based encryption from [4] and ID-based signature from [6]. We will compare in terms of security, size of ciphertext and computation time.

For security analysis, we divide into the followings: IND-A implies anonymity of sender. IND-B implies anonymity of recipient. IND-C implies message confidentiality. EU implies ciphertext non-repudiation. The computation time of IBSC scheme includes the number of pairings and exponential computation as they are the most expensive in IBSC scheme. The comparisons are summarized in the following table.

Scheme	Security				Ciphertext Size	Signcrypt Time		Unsigncrypt Time	
	IND			EU					
	A	B	C			#pair	#exp	#pair	#exp
EtS	×	√	√	√	$(2k+1)G_1+2\Vert m\Vert(+ID)$	1	4 (1)	3	1 (1)
StE	√	√	√	×	$(2k+1)G_1+2\Vert m\Vert+ID$	1	4 (1)	3	1 (1)
M [15]	×	√	×	√	$(k+1)G_1+\Vert m\Vert(+ID)$	1	3 (1)	4	1 (1)
LQ1 [13]	×	×	*	√	$k(G_1+F_p)+\Vert m\Vert(+ID)$	2	2 (1)	4	1 (1)
NR [11]	×	×	*	×	$(k+1)G_1+\Vert m\Vert(+ID)$	1	3 (2)	3	1 (1)
B [5]	√	√	√	*	$(k+1)G_1+\Vert m\Vert+ID$	1	4 (3)	4	2 (2)
CYSC [9]	×	√	√	√	$k(G_1+F_p)+\Vert m\Vert(+ID)$	2	2 (1)	4	1 (1)
LQ2 [14]	√	√	√	*	$(k+1)G_1+\Vert m+\delta\Vert+ID$	1	4 (3)	4	1 (1)
This scheme	√	√	√	√	$(k+1)G_1+\Vert m\Vert+ID$	1	4 (1)	3	1 (1)

5.1 Security

The security analysis follows our definition of security models in Section 2: IND-A, IND-B, IND-C, EU.

- IND-A: The schemes of M, LQ1, NR and CYSC are not IND-A secure. It is because the unsigncryption of ciphertext requires the knowledge of sender's identity in advance.
- IND-B: The schemes of LQ1 and NR are not IND-B secure. Any adversary which knows the sender's identity, private key and the message signcrypted can distinguish the identity of the recipient.
- IND-C: The scheme of M is not IND-CCA2 secure shown in [13]. Schemes of LQ1 and NR are IND secure according to security model in LQ1, but not secure in Boyen's and our security models, where private key of sender is known to Adversary.
- EU: NR's scheme is not EU-CMA secure. Any adversary can forge a signcryption from any sender to recipient ID_B, where private key of ID_B is known to adversary. Boyen's scheme has unforgeability for the signature only. It does

not satisfy the unforgeability for ciphertext as required in our security model and also the security model of standard signcryption in [1]. It is related to the property of "unlinkability" in Boyen's scheme. LQ2 scheme is similar to Boyen's in this aspect. Our IBSC scheme avoids this controversial property of unlinkability and achieves unforgeability for ciphertext.

Some comments based on the above definitions are given in Appendix B, C and D.

5.2 Computation Time

The computation of pairings is the most expensive computation in IBSC scheme. From the above table, we can see that our scheme is the fastest among existing schemes, with similar running time as NR [11], EtS and StE.

If we look further to the number of exponential computation involved, our scheme is in the middle place in exponential calculation. However, there are some components in our scheme that can be pre-computed before knowing the recipient identity and message. For any random number r, X, ${Q_A}^r$ and ${P_{TA}}^r$ can be pre-computed. Therefore the actual number of exponential in our scheme which cannot be pre-computed is two, which is shown in bracket in the table. We can see that our scheme is again the fastest in terms of exponential computation.

5.3 Ciphertext Size

For fair comparison on ciphertext size, we assume that a message m of length $||m||$ have to cut into k pieces for signcryption. Also, sender's identity must be known in advance to unsigncryption for the schemes which do not pass IND-A test. Therefore sender's identity is also included in those schemes. Parameters for signcryption of same m is reused whenever possible.

In LQ2 [14], δ is 160 bits for ciphertext unlinkability, and is 0 bit for ciphertext linkability. As shown in the table, we can see that our scheme has the shortest ciphertext size.

6 Important Functionality of Our Scheme

From our new efficient IBSC scheme, we can achieve further functionalities which are useful in reality. They are the TA compatibility and forward secrecy.

6.1 TA Compatibility

In the reality, it is quite often that sender and recipient use different TAs. If this situation happens, our scheme can still be used without major changes.

Assume all TAs use same pairing e, hash functions and $P \in G_1$. Now let Alice uses $TA1$ with master key s_1. Hence $P_{TA1} = P_{s1}$ and $S_A = {Q_A}^{s_1}$. Similarly Bob uses $TA2$ with master key s_2. Hence $P_{TA2} = P_{s2}$ and $S_B = {Q_A}^{s_2}$.

In our scheme, Sign remains unchanged. In Encrypt, $V = e({Q_B}^r, P_{TA2})$ and others remain unchanged. Decrypt remains unchanged. In Verify, $e(P, Y) = e({P_{TA1}}^h X, Q_A)$ and others remain unchanged. Consistency is verified as:

$$\begin{aligned} V &= e(P_{TA2}, {Q_B}^r) \text{ and } e(P, W) &&= e(P, {S_A}^h {Q_A}^r) \\ &= e(P^{s2}, {Q_B}^r) &&= e(P, {Q_A}^{(r+hs1)}) \\ &= e(X, S_B) &&= e({P_{TA1}}^h X, Q_A) \end{aligned}$$

The security and efficiency of our scheme remains unaffected. Therefore, our scheme can have the TA compatibility function.

6.2 Forward secrecy

Our scheme can achieve forward secrecy. It means that even if the private key of the sender is compromised in the future, the past communications will not be compromised. It can be achieved as in our scheme:

$$V = e(P_{TA}, {Q_B}^r)$$

where r cannot be known even if sender private key is compromised in the future. Therefore Adversary cannot compute V and hence cannot recover m from Z.

If sender and recipient use different TAs as in Section 6.1, then our scheme can even achieve partial TA forward secrecy. If master key of $TA1$ is compromised, then the past communications with users using different TAs will not be compromised, since the computation of V requires the knowledge of r or s_2:

$$V = e(P_{TA2}, {Q_B}^r) = e(P^{s_2}, {Q_B}^r)$$

Therefore even $s1$ is compromised in the future, the adversary still cannot compute V and hence cannot recover m from Z.

7 Conclusion

In this paper, we have proposed a new BIBSC scheme. It is secure against parallel one-more forgery attack.

For the IBSC scheme, our scheme is the fastest, have maximum security and have a short ciphertext when comparing with existing scheme. It is proven secure in a stronger security model than the models in existing schemes. We provide the flexibility for choosing linkability of ciphertext or not.

Moreover, our scheme provides practical features of TA compatibility and forward secrecy.

References

1. J.H. An, Y. Dodis, and T. Rabin. On the security of joint signature and encryption. In *Proc. CRYPTO 2002*, pages 83–107. Springer-Verlag, 2002. Lecture Notes in Computer Science No. 2332.
2. M. Bellare, C. Namprempre, D. Pointcheval, and M. Semanko. The one-more-RSA-inversion problem and the security of Chaum's blind signature scheme. *J. of Cryptology*, pages 185–215, 2003.
3. A. Boldyreva. Efficient threshold signature, multisignature, and blind signature schemes based on the Gap-Diffie-Hellman-group signature scheme. In *PKC'03*, pages 31–46. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 567.
4. D. Boneh and M. Franklin. Identity-based encryption from the weil paring. In *Proc. CRYPTO 2001*, pages 213–229. Springer-Verlag, 2001. Lecture Notes in Computer Science No. 2139.
5. X. Boyen. Multipurpose identity-based signcryption: A swiss army knife for identity-based cryptography. In *Proc. CRYPTO 2003*, pages 382–398. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 2729.
6. J.C. Cha and J.H. Cheon. An identity-based signature from gap diffie-hellman groups. In *Practice and Theory in Public Key Cryptography – PKC'2003*, pages 18–30. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 2567.
7. D. Chaum. Blind signatures for untraceable payments. In *Proc. CRYPTO 82*, pages 199–203. NY, 1983. Plenum.
8. L. Chen and C. Kudla. Identity based authenticated key agreement from pairings. Cryptology ePrint Archive, Report 2002/184, 2002. http://eprint.iacr.org/.
9. S. Chow, S.M. Yiu, L. Hui, and K.P. Chow. Efficient forward and provably secure ID-based signcryption scheme with public verifiability and public ciphertext authenticity. In *ICISC 2003*, pages 352–369. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 2971.
10. C. Cocks. Non-interactive public-key cryptography. In *Cryptography and Coding*, pages 360–363. Springer-Verlag, 2001. Lecture Notes in Computer Science No. 2260.
11. K.C. Reddy D. Nalla. Signcryption scheme for identity-based cryptosystems. Cryptology ePrint Archive, Report 2003/066, 2003. http://eprint.iacr.org/.
12. Y. Desmedt and J. Quisquater. Public-key systems based on the difficulty of tampering. In *Proc. CRYPTO 86*, pages 111–117. Springer-Verlag, 1986. Lecture Notes in Computer Science No. 263.
13. B. Libert and J.-J. Quisquater. New identity based signcryption schemes from pairings. IEEE Information Theory Workshop, Paris (France), 2003.
14. B. Libert and J.-J. Quisquater. The exact security of an identity based signature and its applications. Cryptology ePrint Archive, Report 2004/102, 2004. http://eprint.iacr.org/.
15. J. Malone-Lee. Identity-based signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. http://eprint.iacr.org/.
16. U. Maurer and Y. Yacobi. Non-interactive public-key cryptography. In *Proc. CRYPTO 91*, pages 498–507. Springer-Verlag, 1991. Lecture Notes in Computer Science No. 547.
17. C. P. Schnorr. Practical security in public-key cryptography. In *Proc. ICISC*. Springer, 2001. Lecture Notes in Computer Science.
18. C. P. Schnorr. Security of blind discrete log signatures against interactive attacks. In *Proc. ICISC*, pages 1–12. Springer-Verlag, 2001. Lecture Notes in Computer Science No. 2229.

19. A. Shamir. Identity-based cryptosystems and signature schemes. In *Proc. CRYPTO 84*, pages 47–53. Springer-Verlag, 1984. Lecture Notes in Computer Science No. 196.
20. N.P. Smart. An identity based authenticated key agreement protocol based on the weil pairing. Electronic Letters 38, pp.630-632, 2002.
21. S. Tsuji and T. Itoh. An ID-based cryptosystem based on the discrete logarithm problem. *IEEE Journal on Selected Areas in Communication*, 7(4):467–473, 1989.
22. F. Zhang and K. Kim. ID-Based blind signature and ring signature from pairings. In *Proc. ASIACRYPT 2002*, pages 533–547. Springer-Verlag, 2002. Lecture Notes in Computer Science No. 2501.
23. F. Zhang and K. Kim. Efficient ID-based blind signature and proxy signature from bilinear pairings. In *Proc. ACISP'03*, pages 312–323. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 2727.
24. F. Zhang, R. Safavi-Naini, and W. Susilo. Efficient verifiably encrypted signature and partially blind signature from bilinear pairings. In *Proc. INDOCRYPT03*, pages 191–204. Springer-Verlag, 2003. Lecture Notes in Computer Science No. 2904.
25. Y. Zheng. Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost (encryption). In *Proc. CRYPTO 97*, pages 165–179. Springer-Verlag, 1997. Lecture Notes in Computer Science No. 1294.

A Proof Sketch of Security

A.1 Proof of Theorem 1

Setting up: Dealer D gives $(P, P^{\alpha}, P^{\beta}, Q)$ to Simulator S and wants S to compute $e(P,Q)^{\alpha\beta}$. S sends the system parameter to F with $P_{TA} = P^{\beta}$ as in Setup. S picks a random number η_Q from $\{1, 2, ..., \mu_0\}$, where μ_0 is the number of query to H_0.

Simulating Oracles: As regards queries to the random oracles:

- Query on H_0 for identity ID is handled as follows:
  - The η_Q-th distinct query to H_0 is back patched to the value Q. The corresponding identity is denoted as ID_Q. Adds the entry $\langle ID_Q, Q\rangle$ to tape L_0, and returns the public key Q.
  - Otherwise, picks a random $\lambda \in F_p^*$, adds the entry $\langle ID, \lambda\rangle$ to the tape L_0, and return the public key $Q_{ID} = P^{\lambda}$.
- Queries on H_1, H_2 and H_3 are handled by producing a random element from the codomain, and adding both query and answer to tape L_1, L_2 and L_3.

As regards to oracle queries for:

- $\mathcal{KEO}$: For input identity ID_A.
  - If $ID_A = ID_Q$, then D terminates its interaction with F, having failed to guess the targeted recipient among those in L_0.
  - Otherwise, S retrieves $\langle ID_A, \lambda_A\rangle$ from L_0 and returns $S_A = (P^{\beta})^{\lambda_A}$.
- $\mathcal{SO}$: For input message m, sender ID_A, and recipient ID_B.

  - If $ID_A = ID_Q$, then S randomly chooses $r, h \in F_p^*$, and lets $X = P^r(P^\beta)^{-h}$, $W = (Q)^r$. Then, S adds the tuple $\langle m, X, h \oplus ID_B \rangle$ to L_1 to force the random oracle $H_1(m, X) = h \oplus ID_B$. Finally, S uses $\langle X, W, m, r, ID_B \rangle$ to run Signcrypt to produce the desired ciphertext σ.
  - Otherwise, S retrieves $\langle ID_A, \lambda_A \rangle$ from L_0 and computes $S_A = (P^\beta)^{\lambda_A}$. Then S will run Signcrypt using S_A and get ciphertext σ.
- $\mathcal{UO}$: For input recipient ID_B and ciphertext $\sigma = \langle X, Y, Z \rangle$.
  - If $ID_B = ID_Q$, then S searches all combinations $\langle ID_A, m, X, W \rangle$ such that $\langle m, X, h_1 \rangle \in L_1$, $\langle V, h_2 \rangle \in L_2$, $\langle V, ID_A, h_3 \rangle \in L_3$, for some h_1, h_2, h_3, V, under the constraints that $h_3 \oplus Y = W$, $h_2 \oplus Z = \langle ID_A, m \rangle$ and Verify$[ID_A, m, X, W, ID_B] = \top$. Pick a $\langle ID_A, m \rangle$ in one of the combinations above to return as answer. If no such tuple is found, the oracle signals that the ciphertext is invalid.
  - Otherwise, S retrieves $\langle ID_B, \lambda_B \rangle$ from L_0 and computes $S_B = (P^\beta)^{\lambda_B}$. Then S will run Unsigncrypt using S_B to get $\langle ID_A, m \rangle$ or $\perp$.

Witness Extraction: As in the IND-IBSC-CCA2 game, at some point F chooses plaintext m_1, sender ID_{A1}, and recipient ID_{B1} on which he wishes to be challenged. S responds with challenge ciphertext $\langle X, Y, Z \rangle$, where:

$$X = P^\alpha$$

Y and Z are random strings of appropriate size. All further queries by F are processed adaptively as in the oracles above.

Finally, F returns its final guess. S ignores the answer from F, randomly picks an entry $\langle V, h_2 \rangle$ in L_2, and returns V as the solution to the co-BDH problem.

If the recipient identity $ID_{A1} = ID_Q$ selected by S, to recognize the challenge ciphertext $\langle X, Y, Z \rangle$ with $X = P^\alpha$ is incorrect, F needs to query random oracle $H_2(V)$ with

$$V = e(X, S_Q) = e(P^\alpha, Q^\beta) = e(P, Q)^{\alpha\beta}$$

It will leave an entry $\langle V, h_2 \rangle$ on L_2, from which B can then extract $V = e(P, Q)^{\alpha\beta}$. □

A.2 Proof Sketch of Theorem 2

Setting up: Dealer D gives (P, P^β, Q) to Simulator S and wants S to compute Q^β. Others are same as in the proof of Theorem 1.

Oracle Simulation: The signcryption oracle, the unsigncryption oracle, and the key extraction oracle are simulated in the same way as in the proof of Theorem 1.

Witness Extraction: Assume $\mathcal{F}$ is a PPT forger. Rewind $\mathcal{F}$ to the random oracle query whose output appears in the verification in unsigncryption. Then we obtain $W = S_A^h Q_A^r$ and $W' = S_A^{h'} Q_A^r$ in respective forks. Combining, we can compute the co-CDH Problem $S_a = (W'/W)^{(h'-h)^{-1}}$. □

A.3 Proof Sketch of Theorem 3

To prove the blindness of BIBSC, we show that given a valid ciphertext $\langle \hat{X}, \hat{Y}, \hat{Z} \rangle$ and any transcript of blind signcryption (X, h, W, V), there always exist a unique pair of blinding factors $\alpha, \beta \in Z_q^*$. Since the blinding factors are randomly chosen, the blindness of BIBSC is achieved.

Given a valid ciphertext $\langle \hat{X}, \hat{Y}, \hat{Z} \rangle$, then there exist a unique $(\hat{X}, \hat{W}, \hat{V}, m)$ for this ciphertext. Then for any transcript of blind signcryption (X, h, W, V), the following equations must hold for $\alpha, \beta \in Z_q^*$:

$$\begin{aligned}
\hat{X} &= X^{\alpha} P^{\beta} \\
h &= \alpha^{-1} H_1(m, \hat{X}) \\
\hat{W} &= W^{\alpha} {Q_A}^{\beta} \\
\hat{V} &= V^{\alpha} e({P_{TA}}^{\beta}, Q_B)
\end{aligned}$$

From the second equation, we see that there exist a blinding factor $\alpha = H_1(m, \hat{X})/h$. For this α, there exist a blinding factor β from the first equation and $\beta = log_P(\hat{X} X^{-\alpha})$. Therefore we have to show that these blinding factors α, β satisfy the last two equations.

Notice that there exists a S_B which is the private key for Q_B. Then:

$$\begin{aligned}
\hat{V} &= e(\hat{X}, S_B) \\
&= e(X^{\alpha} P^{\beta}, S_B) \\
&= e(X, S_B)^{\alpha} e(P^{\beta}, S_B) \\
&= V^{\alpha} e({P_{TA}}^{\beta}, Q_B)
\end{aligned}$$

Furthermore, $\langle \hat{X}, \hat{W}, m \rangle$ is a valid signature. Therefore we have:

$$\begin{aligned}
e(P, \hat{W}) &= e(\hat{X}, Q_A) e(P_{TA}, Q_A)^{H_1(m, \hat{X})} \\
&= e(X^{\alpha} P^{\beta}, Q_A) e(P_{TA}, Q_A)^{\alpha h} \\
&= e(X {P_{TA}}^{h}, Q_A)^{\alpha} e(P^{\beta}, Q_A) \\
&= e(P, W)^{\alpha} e(P, {Q_A}^{\beta}) \\
&= e(P, W^{\alpha} {Q_A}^{\beta})
\end{aligned}$$

Hence, given a valid ciphertext $\langle \hat{X}, \hat{Y}, \hat{Z} \rangle$ and any any transcript of blind signcryption (X, h, W, V), there always exists a unique pair of blinding factors $\alpha, \beta \in Z_q^*$. Therefore, $Prob\{\sigma \ by \ Warden\} = Prob\{\sigma \ by \ Warden | \mathcal{T}\}$. The blindness of BIBSC is proved. □

A.4 Proof Sketch of Theorem 4

That solving either hard problems imply forgery is easy. We proceed to prove the other direction.

The proof is in random oracle model (ROM) plus extended generic group model for pairings. The latter is a modification of GGM to include pairing-related operations.

Now we mimick Schnorr's proof [17]. **Generic Adversary: Algorithm GA1:** A generic adversary can be formulated as follows:

1. Input: ID_a, ID_b.
2. Obtain q_B commitments X_i, $1 \leq i \leq q_B$, from BlindSigncrypt Oracle.
3. Compute. Makes q_H queries to the random oracle H_1.
4. Send challenges h_i, $1 \leq i \leq q_B$, to BlindSigncrypt Oracle.
5. Receive responses (W_i, V_i), $1 \leq i \leq q_B$. Outputs message-signcryptions $(\hat{m}_j, (\hat{X}_j, \hat{Y}_j, \hat{Z}_j))$, $1 \leq j \leq q_B + 1$.

All conversations can be interleaved arbitrarily. For simplicity, we fixe ID_a and ID_b.

The queries $H_1(\hat{m}_j, \hat{X}_j) = \hat{h}_j$ must all have been made. Assume $\hat{X}_j$ is computed at Step σ_j of the algorithm. Bu the generic group model for pairings, we have

$$\hat{X}_j = \phi(Q_a)^{a_{\sigma_j,-2}} P^{a_{\sigma_j,-1}} P_{TA}^{\sigma_j,0} \prod_{i=1}^{q_B} X_i^{a_{\sigma_j,i}}$$

Simulating the generic adversary: Algorithm $GA2$:

1. Input: ID_a, ID_b.
2. Obtain commitments in the form of a q_B-row-vector $\mathbf{X}$ from BlindSigncrypt Oracle.
3. Randomly generate $q_B \times q_H$ matrix $\mathbf{D}$. Compute q_H-vector $\mathbf{X}' = \mathbf{XD}$. Randomly generate q_H-vector $\mathbf{m}'$. Compute q_H-vector $\mathbf{h}' = H_1(\mathbf{X}', \mathbf{m}')$, i.e. $h'_j = H_1(X'_j, m'_j)$, $1 \leq j \leq q_H$.
4. Simulate Steps (3-5) of the generic adversary. Except to backpatch all its H_1 oracle query outputs to $\mathbf{h}'$.
5. Outputs the tuples $(\hat{m}_j, (\hat{X}_j, \hat{Y}_j, \hat{Z}_j))$, $1 \leq j \leq q_B + 1$.

Let J'=$\{\ell_1,\cdots,\ell_{q_B+1}\}$ where for each member of J' we have $H_1(\hat{X}_j, \hat{m}_j) = h'_{\ell_j}$ which implies $\hat{X}_j = X'_{\ell_j}$ and $\hat{m}_j = m'_{\ell_j}$. Let $\mathbf{E} = \mathbf{D}_{|J'}$ denote the restriction (cropping) of $\mathbf{D}$ to columns whose column index is in J'. Note $\mathbf{E}$ is a $q_B \times (q_B+1)$ matrix. For each $j \in J'$, let

$$\begin{aligned} \Delta_{h,j} &= (\sum_i h_i E_{i,j}) - \hat{h}_j \\ \Delta_{X,j} &= (\prod_i X_i^{E_{i,j}})/\hat{X}_j \\ \Delta_{W,j} &= (\prod_i W_i^{E_{i,j}})/\hat{W}_j \end{aligned}$$

where r_i and r'_j are such that $X_i = g^{r_i}$ and $X'_j = g^{r'_j}$. Signature verifications ensure

$$e(P, W_i) = e(X_i P_{TA}^{h_i}, Q_a), 1 \leq i \leq q \text{ (BlindSigncrypt Oracle side);}$$

$$e(P, \hat{W}_j) = e(\hat{X}_j P_{TA}^{\hat{h}_j}, Q_a), 1 \leq j \leq q_B + 1 \text{ (}GA1\text{ side) .}$$

Combining, we obtain

$$e(P, \Delta_{W,j}) = e(\Delta_{X,j} P_{TA}^{\Delta_{h,j}}, Q_a) = e(P, Q_a^{s\Delta_{h,j}})$$

for each $j \in J'$. Therefore $Q_a^{s\Delta_{h,j}} = \Delta_{W,j}$.

Similar to Schnorr's proof [17], each entry on both sides of the equation are of the form dictated by the generic group model for pairings. They can be computed by the Adversary but not the Simulator, so rewinding does not help Simulator extract the witness. But in the generic group model, it can be deducted that $\Delta_{h,j} = 0$ and $\Delta_{W,j} = 1$, for all $j \in J'$; for otherwise Adversary can compute $Q_a^s = \Delta_{W,j}^{1/\Delta_{h,j}}$, which is a co-CDH Problem.

In summary, there are two cases. Case 1, $\Delta_{W,j} = 1$ and $\Delta_{h,j} = 0$ for all $j \in J'$. Then $GA1$ is reduced to solving the ROS Problem. Case 2, the opposite, then $GA1$ is reduced to solving a co-CDH Problem.

B Comments on various IBSC's w.r.t. our security model

B.1 Comment for IND-B

In the following, please refer to the original paper for original scheme and the definition of the symbols used. In the IND sub-game (b), the Adversary chooses message m, sender ID_A and recipient ID_{B1}. The Adversary knows the private key of ID_A. Simulator chooses a recipient ID_{B0}, and randomly picks $b \in \{0, 1\}$. Simulator signcrypts the message m from sender ID_A to recipient ID_{Bb} and returns the ciphertext to the Adversary. The Adversary has to guess b.

Libert and Quisquater's scheme 1 [13] The Adversary has the ciphertext $\langle c, r, S\rangle$ and d_A, the private key of ID_A. The Adversary computes:

$$\begin{aligned} k_2 &= H_2(e(S, Q_{B1}) e(d_A, Q_{B1})^r) \\ m' &= D_{k_2}(c) \end{aligned}$$

The Adversary outputs $b = 1$ if $m' = m$. Otherwise, the Adversary outputs $b = 0$. Then the Adversary wins the IND game with probability 1.

Nalla and Reddy's scheme [11] The Adversary has the ciphertext $\langle R, S, C\rangle$ and S_A, the private key of ID_A. The Adversary computes:

$$\begin{aligned} R' &= (R || H_1(e(Q_{B1}, S_A)) || m) \\ k_A &= H''(e(Q_{B1}, R)^{H'(R')}) \\ C' &= k_A \oplus m \end{aligned}$$

The Adversary outputs $b = 1$ if $C' = C$. Otherwise, the Adversary outputs $b = 0$. Then the Adversary wins the IND game with probability 1.

B.2 Comment for IND-C

In the IND sub-game (c), the Adversary chooses message m_1, sender ID_A and recipient ID_B. The Adversary knows the private key of ID_A. Simulator chooses a message m_0, and randomly picks $b \in \{0, 1\}$. Simulator signcrypts the message m_b from sender ID_A to recipient ID_B and returns the ciphertext to the Adversary. The Adversary has to guess b.

Nalla and Reddy's scheme [11] The Adversary has the ciphertext $\langle R, S, C\rangle$ and S_A, the private key of ID_A. The Adversary computes:

$$\begin{aligned} R' &= (R||H_1(e(Q_B, S_A))||m_1) \\ k_A &= H''(e(Q_B, R)^{H'(R')}) \\ C' &= k_A \oplus m_1 \end{aligned}$$

The Adversary outputs $b = 1$ if $C' = C$. Otherwise, the Adversary outputs $b = 0$. Then the Adversary wins the IND game with probability 1.

B.3 Comment for EU

In the EU game, the Adversary chooses message m, sender ID_A and recipient ID_B. The Adversary knows the private key of ID_B. The Adversary returns a ciphertext σ and recipient identity ID_B to the Simulator.

Nalla and Reddy's scheme [11] The Adversary has S_B, the private key of ID_B. The Adversary randomly chooses $a \in R$ and computes:

$$\begin{aligned} R &= {S_B}^a \\ R' &= (R||H_1(e(S_B, Q_A))||m) \\ S &= {Q_B}^{aH'(R')} \\ k_A &= H''(e(Q_B, S_B)^{aH'(R')}) \\ C &= k_A \oplus m \end{aligned}$$

The Adversary outputs the ciphertext $\sigma = \langle R, S, C\rangle$, sender identity ID_A and recipient identity ID_B to the Simulator.

The Simulator decrypts by computing:

$$\begin{aligned} k_B &= H''(e(S, S_B)) \\ m &= k_B \oplus C \end{aligned}$$

The decryption succeeds. Then in verification, the Simulator computes $R' = (R||H_1(e(S_B, Q_A))||m)$ and checks if:

$$e(S_B, S) = e(Q_B, R)^{H'(R')}$$

By the above construction, the ciphertext must pass the verification. Then the Adversary wins the EU game with probability 1.