Identity Based Online/Offline Encryption and Signcryption Schemes Revisited

S. Sharmila Deva Selvi, S. Sree Vivek, and C. Pandu Rangan*

Theoretical Computer Science Laboratory,
Department of Computer Science and Engineering,
Indian Institute of Technology Madras,
Chennai, India
{sharmila,svivek}@cse.iitm.ac.in, prangan@cse.iitm.ac.in.

Abstract. Consider the situation where a low power device with limited computational power has to perform cryptographic operation in order to do secure communication to the base station where the computational power is not limited. The most obvious way is to split each and every cryptographic operations into resource consuming, heavy operations and the fast light weight operations. This concept can be efficiently implemented through online/offline cryptography. In this paper, we show the security weakness of an identity based online offline encryption scheme proposed in ACNS 09 by Liu et al. [9]. The scheme in [9] is the first identity based online offline encryption scheme in the random oracle model, in which the message and recipient are not known during the offline phase. We have shown that this scheme is not CCA secure. We have also proposed a new identity based online offline encryption scheme in which the message and receiver are not known during the offline phase and is efficient than the scheme in [9].

Online/Offline signcryption is a cryptographic primitive where the signcryption process is divided into two phases - online and offline phase. To the best of our knowledge there exists three online/offline signcryption schemes in the literature: we propose various attacks on two of the existing schemes. Then, we present an efficient and provably secure identity based online/offline signcryption scheme. We formally prove the security of the new scheme in the random oracle model.

Keywords: Identity Based Cryptography, Encryption, Signcryption, Confidentiality, Unforgeability, Online/Offline, Cryptanalysis, Random Oracle Model.

1 Introduction

Separating the process of signing or encrypting into two phases namely, online phase and offline phase is the concept of "Online/Offline" cryptography. This notion was first introduced in the context of digital signatures by Even, Goldreich and Micali [5]. Their construction is inefficient as it increases the size of

* Currently Head, Indian Statistical Institute, Chennai, India.

M. Joye et al. (Eds.): InfoSecHiComNet 2011, LNCS 7011, pp. 111–127, 2011.

each signature by a quadratic factor. Shamir and Tauman [13] proposed an improved version which makes use of a new paradigm called "hash-sign-switch" to design more efficient online/offline signature schemes. During the offline phase, heavy computations like exponentiation and bilinear pairing are done and in the online phase, only light weight integer operations (multiplication and addition) and hashing are performed to make the computations faster. In an online/offline signature scheme the message is not known in the offline phase and in an online/offline encryption scheme both the message and receiver are not known in the offline phase. Thus, online/offline schemes find use in low power devices such as PDA's, sensor networks, hand held devices including mobile phones and smart-cards.

Adi Shamir introduced the concept of identity based cryptography and proposed the first identity based signature scheme. The idea of identity based cryptography is to enable an user to use any arbitrary string that uniquely identifies him as his public key. Identity based cryptography serves as an efficient alternative to Public Key Infrastructure (PKI) based systems. Most of the identity based encryption (IBE) schemes use the costly bilinear pairing operation and the concept of online/offline computation is an important area of research with respect to IBE. The first identity based online/offline encryption scheme was proposed by Guo et al.[7]. It should be noted that, the major difference between online/offline signature and encryption schemes is that, the message and the receiver are not known during the offline phase of encryption schemes. This makes it subtle and interesting to explore for new directions in constructing efficient and elegant online/offline encryption schemes. Few motivating examples for online/offline encryption schemes can be found in [7] and [9].

Related Works

Online/Offline Encryption: Guo et al. [7] have shown natural extension of the IBE of Boneh and Boyen [2] and Gentry [6]. They have also given constructions which efficiently divide the IBE schemes in [2] and [6]. All the schemes are in the standard model. In 2009, Joseph. K. Liu et al. [9] have proposed an identity based online/offline encryption scheme. It was proved to be chosen ciphertext (CCA) secure in the random oracle model and was claimed to be much efficient that the scheme in [7] (obviously true due to random oracle assumption). Recently, Chow et al. in [3] proposed a CPA secure identity based online/offline encryption scheme and have given a KEM (Key Encapsulation Mechanism) based CCA construction. Although they are giving a generic transformation from identity based online/offline KEM (IBOOKEM) to CCA secure identity based online/offline encryption, there is no concrete IBOOKEM scheme discussed in the paper. Hence, we do not compare our results with the results reported in [3].

Online/Offline Signcryption: Confidentiality and authenticity are two fundamental properties offered by public key cryptography which are achieved through encryption schemes and digital signatures respectively. In scenarios where both these properties are needed, a Sign-then-Encrypt approach was used earlier. In 1997, Zheng [17] introduced the concept of signcryption where both these properties are achieved in a single logical step, but in a more efficient way. The notion

of online/offline signcryption was first discussed in An et al. [1]. In their paper, they did not give any concrete method, but they have given general security proof notions for signcryption schemes. Zhang et al. [16] extended the work of An et al. [1] and provided a concrete scheme making use of short signatures. However, Zhang's scheme [16] is PKI based scheme and the focus of our paper is on identity based signcryption schemes. Sun et al. [14] were the first to propose an identity based online/offline signcryption scheme. In their paper, they formally defined the identity based online/offline signcryption and its security model. They also proposed a new scheme where the offline computations can be done before the message is available and the online computations are done after the message is received. After this, Sun et al. proposed another generic scheme in [15].

Our Contribution: In this paper, we show that the scheme in [9] is not CCA secure, i.e. an adversary can distinguish the challenge ciphertext by accessing the decryption oracle. Although the authors of [9] (footnote 4) claim that a bug in [9] was identified and presented in the conference, we are unable to trace any record of its presence. In view of this we present the details of the attack here explicitly. We provide a fix for the bug in the scheme and also propose a new efficient construction for identity based online/offline encryption. We prove the new scheme in the random oracle model.

Moreover, to the best of our knowledge there are three online/offline signcryption schemes in the literature: two schemes by sun et al. [15], [14] and one scheme by Liu et al. [8]. In this paper, we point out some weaknesses in the generic scheme by Sun et al. [15] and forgeability attack on the specific scheme by Sun et al [14]. Then, we present a new online/offline identity based signcryption scheme. In our scheme the online phase includes only modular addition operations and an XOR operation. The striking feature of our scheme is that the sender does not require the knowledge of receiver identity as well as the message in the offline phase. The security of the scheme is proved under random oracle model.

2 Preliminaries

2.1 Bilinear Pairing

Let $\mathbb{G}_1$ be an additive cyclic group generated by P, with prime order q, and $\mathbb{G}_2$ be a multiplicative cyclic group of the same order q. Let $\hat{e}$ be a bilinear pairing $\hat{e} : \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$.

2.2 Computational Assumptions

In this section, we recall the computational assumptions related to bilinear maps[4] that are relevant to the security of our scheme.

***Modified BDHI for k values (k-mBDHIP)*:** k-mBDHIP is the bilinear variant of the k-CAA problem. Given $(P, aP, (x_1+a)^{-1}P, ..., (x_k+a)^{-1}P) \in \mathbb{G}_1^{k+2}$

for unknown $a \in {Z_q}^*$ and known $x_1, ..., x_k \in {Z_q}^*$, the k-mBDHIP problem is to compute $\hat{e}(P,P)^{(a+x^*)^{-1}}$ for some $x^* \notin \{x_1, ..., x_k\}$.

The advantage of any probabilistic polynomial time algorithm $\mathcal{A}$ in solving the k-mBDHIP problem in $\mathbb{G}_1$ is defined as

$$\begin{aligned}\mathrm{Adv}_{\mathcal{A}}{}^{k-mBDHIP} &= \Pr[\mathcal{A}(P, aP, (x_1+a)^{-1}P, ..., (x_k+a)^{-1}P, x_1, ..., x^k) \\ &= \hat{e}(P,P)^{(a+x^*)^{-1}} | a, x^* \in_R \mathrm{Z}_q{}^*,\ x^* \notin \{x_1, ..., x_k\}].\end{aligned}$$

We say that the k-mBDHIP problem is (t, ϵ) hard if for any t time probabilistic algorithm $\mathcal{A}$, the advantage $\mathrm{Adv}_{\mathcal{A}}{}^{k-mBDHIP} < \epsilon$.

The q-Computation Diffie-Hellman Inverse problem (q-CDHIP): Given an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$, all with prime order p and $(q+1)$ tuples $(G, sG, s^2G, \ldots, s^qG)$, computing $(1/s)P$ is the q-Computation Diffie-Hellman Inverse problem.

The q-Bilinear Diffie-Hellman Inversion problem (q-BDHIP): Given an additive group $\mathbb{G}_1$ and a multiplicative group $\mathbb{G}_2$, all with prime order p and $(q+1)$ tuples $(G, sG, s^2G, \ldots, s^qG)$, computing $\hat{e}(G,G)^{1/s} \in \mathbb{G}_2$ is the q-Bilinear Diffie-Hellman Inversion problem.

2.3 Identity Based Online/Offline Encryption Schemes(IBOOE)

An identity based online/offline encryption scheme consists of the following algorithms.

Setup(1^κ)**:** Given a security parameter κ, the Private Key Generator(PKG) generates a master private key msk and public parameters $Params$. $Params$ is made public while msk is kept secret by the PKG.

Extract(ID)**:** Given an identity ID, the PKG executes this algorithm to generate the private key D_{ID} corresponding to ID and transmits D_{ID} to the user with identity ID via. secure channel.

Off-Encrypt $(Params)$**:** To generate the offline share of the encryption, this algorithm is executed without the knowledge of message to be encrypted and the receiver of the encryption. The offline ciphertext is represented as ϕ.

On-Encrypt (m, ID_A, ϕ)**:** For encrypting a message m to user with identity ID_A, any sender can run this algorithm to generate the encryption σ of message m. This algorithm uses a new offline ciphertext ϕ and generates the full encryption σ.

Decrypt(σ, ID_A, D_A)**:** For decryption of σ, the receiver ID_A uses his private key D_A and run this algorithm to get back the message m.

Definition 1. *An ID-Based online/offline encryption scheme is said to be indistinguishable against adaptive chosen ciphertext attacks (IND-IBOOE-CCA2) if no polynomially bounded adversary has a non-negligible advantage in the following game.*

***Setup*:** The challenger $\mathcal{C}$ runs the *Setup* algorithm with a security parameter κ and obtains public parameters $Params$ and the master private key msk. $\mathcal{C}$ sends $Params$ to the adversary $\mathcal{A}$ and keeps msk secret.

***Phase I*:** The adversary $\mathcal{A}$ performs a polynomially bounded number of queries. These queries may be adaptive, i.e. current query may depend on the answers to the previous queries.

- ***Key extraction queries(Oracle*** $\mathcal{O}_{Extract}(ID)$***):*** $\mathcal{A}$ produces an identity ID and receives the private key D_{ID}.
- ***Decryption queries(Oracle*** $\mathcal{O}_{Decrypt}(\sigma, ID_A)$***):*** $\mathcal{A}$ produces the receiver identity $ID_{\mathbb{A}}$ and the ciphertext σ. $\mathcal{C}$ generates the private key D_A and sends the result of $Decrypt(\sigma, ID_A, D_A)$ to $\mathcal{A}$. This result will be "$Invalid''$ if σ is not a valid ciphertext or the message m if σ is a valid encryption of message m to ID_A.

***Challenge*:** $\mathcal{A}$ chooses two plaintexts, m_0 and m_1 and the receiver identity $ID_{\mathbb{R}}$, on which $\mathcal{A}$ wishes to be challenged. $\mathcal{A}$ should not have queried for the private key corresponding to $ID_{\mathbb{R}}$ in Phase I. $\mathcal{C}$ chooses randomly a bit $b \in \{0, 1\}$, computes $\sigma = Encrypt(m_b, ID_{\mathbb{R}})$ and sends it to $\mathcal{A}$.

***Phase II*:** $\mathcal{A}$ is now allowed to get training as in $Phase-I$. During this interaction, $\mathcal{A}$ is not allowed to extract the private key corresponding to $ID_{\mathbb{R}}$. Also, $\mathcal{A}$ cannot query the decryption oracle with $\sigma, ID_{\mathbb{R}}$ as input, i.e. $\mathcal{O}_{Decrypt}(\sigma, ID_{\mathbb{R}})$.

***Guess*:** Finally, $\mathcal{A}$ produces a bit $b^{'}$ and wins the game if $b^{'} = b$.

$\mathcal{A}$'s advantage is defined as $Adv(\mathcal{A})=2\left|Pr\left[b^{'} = b\right] - 1\right|$, where $Pr\left[b^{'} = b\right]$ denotes the probability that $b^{'} = b$.

2.4 Identity Based Online/Offline Signcryption

Identity based online/offline signcryption scheme consists of the following algorithms.

***Setup*(κ):** Given a security parameter κ, the Private Key Generator (PKG) generates the systems public parameters *params* and the corresponding master private key *msk* that is kept secret by PKG.

***Key Extract*(ID_i):** Given a user identity ID_i by user U_i, the PKG computes the corresponding private key D_i and sends D_i to U_i via. a secure channel.

***OffSigncrypt*($ID_{\mathbb{S}}$, $D_{\mathbb{S}}$):** Given the sender identity $ID_{\mathbb{S}}$ and the private key $D_{\mathbb{S}}$ of $ID_{\mathbb{S}}$, this algorithm outputs an offline signcryption σ'. This is executed by the sender with identity $ID_{\mathbb{S}}$.

***OnSigncrypt*(m, $ID_{\mathbb{S}}$, $ID_{\mathbb{R}}$, σ'):** This algorithm takes as input a message $m \in \mathcal{M}$, the sender identity $ID_{\mathbb{S}}$, the receiver identity $\mathrm{ID}_{\mathbb{R}}$ and the offline signcryption σ' by $ID_{\mathbb{S}}$ as input and outputs the signcryption σ. This algorithm is executed by the sender with identity $ID_{\mathbb{S}}$.

***Unsigncrypt*(σ, $ID_{\mathbb{S}}$, $ID_{\mathbb{R}}$, $D_{\mathbb{R}}$):** This algorithm takes as input the signcryption σ, sender's identity $\mathrm{ID}_{\mathbb{S}}$, the receiver identity $ID_{\mathbb{R}}$ and the receiver's private key

$D_{\mathbb{R}}$ as input and produces the plaintext m, if σ is a valid signcryption of m from the sender $\mathrm{ID}_{\mathbb{S}}$ to $ID_{\mathbb{R}}$ or "Invalid" otherwise.

Definition 2. *(Confidentiality) An identity based online/offline signcryption (IBOOSC) is indistinguishable against adaptive chosen ciphertext attacks (IND-IBOOSC-CCA2) if there exists no polynomially bounded adversary having non-negligible advantage in the following game:*

***Setup Phase*:** The challenger $\mathcal{C}$ runs the **Setup** algorithm with the security parameter κ as input and sends the system parameters **params** to the adversary $\mathcal{A}$ and keeps the master private key **msk** secret.

***Phase-I*:** $\mathcal{A}$ performs polynomially bounded number of queries to the oracles provided to $\mathcal{A}$ by $\mathcal{C}$. The description of the queries in the first phase are listed below:

- ***Key Extract query*:** $\mathcal{A}$ produces an identity ID_i and receives the private key D_i corresponding to ID_i.
- ***Signcryption query*:** $\mathcal{A}$ produces a message m, the sender identity $ID_{\mathbb{S}}$, and the receiver identity $ID_{\mathbb{R}}$ to the challenger $\mathcal{C}$. $\mathcal{C}$ computes $ID_{\mathbb{S}}$'s private key $D_{\mathbb{S}}$ and runs the algorithm **OffSigncrypt**$(ID_{\mathbb{S}}, D_{\mathbb{S}})$ to obtain an offline signcryption σ'. Finally $\mathcal{C}$ returns $\sigma =$ **OnSigncrypt**$(m, ID_{\mathbb{R}}, \sigma')$ to $\mathcal{A}$.
- ***Unsigncryption query*:** $\mathcal{A}$ produces the signcryption σ, the sender identity $ID_{\mathbb{S}}$, and the receiver identity $ID_{\mathbb{R}}$ to $\mathcal{C}$. $\mathcal{C}$ generates the private key $D_{\mathbb{R}}$ by querying the **Key Extraction oracle**. $\mathcal{C}$ unsigncrypts σ using $D_{\mathbb{R}}$ and returns m if σ is a valid signcryption from $ID_{\mathbb{S}}$ to $ID_{\mathbb{R}}$, else outputs "Invalid".

$\mathcal{A}$ can present its queries adaptively, i.e. every request may depend on the response to the previous queries.

***Challenge*:** $\mathcal{A}$ chooses two plaintexts $\{m_0, m_1\} \in \mathcal{M}$ of equal length and ID_A and ID_B as the sender and receiver identities on which $\mathcal{A}$ wishes to be challenged. The restriction is that $\mathcal{A}$ should not have queried the private key corresponding to ID_B in Phase-I. $\mathcal{C}$ now chooses a bit $\bar{\delta} \in_R \{0, 1\}$ and computes the challenge signcryption σ^* of $\mathrm{m}_{\bar{\delta}}$ and sends σ^* to $\mathcal{A}$.

***Phase-II*:** $\mathcal{A}$ performs polynomially bounded number of requests just like the Phase-I, with the restrictions that $\mathcal{A}$ cannot make **Key Extraction query** on ID_B and should not query for unsigncryption query on C^*.

***Guess*:** Finally, $\mathcal{A}$ produces a bit $\bar{\delta}'$ and wins the game if $\bar{\delta}' = \bar{\delta}$. The success probability is defined by:

$$\mathrm{Succ}_{\mathcal{A}}{}^{IND-IBOOSC-CCA2}(\kappa) = \tfrac{1}{2} + \epsilon$$

Here, ϵ is called the advantage for the adversary in the above game.

Definition 3. *(Unforgeability) An identity based online/offline signcryption scheme (IBOOSC) is said to be existentially unforgeable against adaptive chosen messages attacks (EUF-IBOOSC-CMA) if no polynomially bounded adversary has a non-negligible advantage in the following game:*

***Setup Phase*:** The challenger runs the **Setup** algorithm with a security parameter κ and gives the system parameters **params** to the adversary $\mathcal{A}$ and keeps **msk** secret.

***Training Phase*:** $\mathcal{A}$ performs polynomially bounded number of queries as described in Phase-I of **Definition 2**. The queries may be adaptive, i.e. the current query may depend on the previous query responses.

***Existential Forgery*:** Finally, $\mathcal{A}$ produces a new triple $(ID_{\mathbb{A}}, ID_{\mathbb{B}}, C^*)$ (i.e. a triple that was not produced by the signcryption oracle), where the private key of $ID_{\mathbb{A}}$ was not queried in the **training phase**. $\mathcal{A}$ wins the game if the result of the unsigncryption of $(ID_{\mathbb{A}}, ID_{\mathbb{B}}, C^*)$ is $\neq$ "Invalid", in other words C* is a valid signcrypt of some message $m \in \mathcal{M}$.

3 Review and Attack of IBOOE in [9]

In this section we review the identity based online/offline encryption scheme proposed in [9].

3.1 Review of of Liu et al.'s Scheme (L-IBOOE) [9]

Let $\mathbb{G}$ and $\mathbb{G}_T$ be groups of prime order q , and let $\hat{e}$:$\mathbb{G} \times \mathbb{G}_T \rightarrow \mathbb{G}_T$ be the bilinear pairing. We use a multiplicative notation for the operation in $\mathbb{G}$ and $\mathbb{G}_T$.

Setup: The PKG selects a generator $P \in \mathbb{G}$ and randomly chooses s, $w \in \mathbb{Z}_q^*$. It sets $P_{pub} = sP$, $P'_{pub} = s^2P$ and $W = (w+s)^{-1}P$. Define $\mathcal{M}$ to be the message space. Let $n_M = |\mathcal{M}|$. Let H_1:$\{0,1\}^* \rightarrow \mathbb{Z}_q^*$ H_2:$\{0,1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_q^*$ and H_3:$\{0,1\}^* \rightarrow \{0,1\}^{n_M}$ be two cryptographic hash functions. The public parameters $Params$ and master private key msk are given by,

$$Params = \langle \mathbb{G}, \mathbb{G}_T, q, P_{pub}, P'_{pub}, W, w, \mathcal{M}, H_1, H_2, H_3 \rangle \; msk = s.$$

Extract(ID):
- $q_{ID} = H_1(ID)$
- $D_{ID} = \dfrac{1}{q_{ID} + s} P.$

Off-Encrypt($Params$):
- $u, x, \alpha, \beta, \gamma, \delta \in_R \mathbb{Z}_q^*$
- $U = W - uP$
- $R = \hat{e}(wP + P_{pub}, P)^x$
- $T_0 = x(w \quad \alpha P \quad + \quad (w+ \gamma)P_{pub} + P'_{pub})$
- $T_1 = xw\beta P.$
- $T_2 = x\delta P_{pub}.$
- Output the offline ciphertext $\phi = \langle u, x, \alpha, \beta, \gamma, \delta, U, R, T_0, T_1, T_2 \rangle.$

On-Encrypt(m, ID_A, ϕ):
- $t_1 = \beta^{-1}(H_1(ID_A) - \alpha) \bmod q$
- $t_2 = \delta^{-1}(H_1(ID_A) - \gamma) \bmod q$
- $t = H_2(m, R)x + u \bmod q$
- $c = H_3(R) \oplus m$
- Output the ciphertext $\sigma = \langle U, T_0, T_1, T_2, t, t_1, t_2, c \rangle$

Decrypt(σ,ID_A, D_A):
- $R = \hat{e}(T_0 + t_1 T_1 + t_2 T_2, D_A)$
- $m = c \oplus H_3(R)$
- and if $R^{H_2(m,R)} \stackrel{?}{=} \hat{e}(tP + U, wP + P_{pub})\, \hat{e}(P,P)^{-1}$, output m else output $\perp$

3.2 Attack on Confidentiality

[1]During the confidentiality game, after the completion of Phase-1 of training, the adversary $\mathcal{A}$ picks two messages, (m_0, m_1) of equal length and an identity $ID_{\mathbb{R}}(D_{\mathbb{R}}$ is not known to $\mathcal{A})$, and submits them to $\mathcal{C}$. $\mathcal{C}$ chooses a bit $b \in_R \{0,1\}$, generates the challenge ciphertext $\sigma^* = \langle U,\ T_0, T_1,\ T_2,\ t_1^{'}, t_2^{'}, t,\ c\rangle$ of message m_b and gives σ^* to $\mathcal{A}$. Now, we show that $\mathcal{A}$ can cook up another valid ciphertext $\delta = (U^*,\ T_0^*,\ T_1^*,\ T_2^*,\ t_1^*,\ t_2^*,\ t^*,\ c^*)$ as given below:

- Chooses $r^*,\ t_1^*,\ t_2^* \in_R \mathbb{Z}_q^*$.
- Computes $U^* = U - r^*P = W - (u + r^*)$P.
- Chooses $T_1^*,\ T_2^* \in_R \mathbb{G}$.
- Computes $T_0^* = T_0 - (t_1^* T_1^* + t_2^* T_2^*) + (t_1 T_1 + t_2 T_2) = x(w + s)(q_A + s)P-(t_1^* T_1^* + t_2^* T_2^*)$ (since $T_0 + t_1 T_1 + t_2 T_2 = x(w + s)(q_A + s)$P).
- Computes $t^* = t + r^* \bmod q$
- Sets $c^* = c$
- Now, $\mathcal{A}$ queries the decrypt oracle with δ as input during $Phase - 2$ of training. Here, the relations between σ^* and δ are $R = R^* = \hat{e}(P,P)^{(w+s)x}$ and $c = c^*$. Hence, the decryption of δ will give the message $m_b = c \oplus H_3(R) = c^* \oplus H_3(R^*)$. So, $\mathcal{A}$ can obtain m_b by constructing δ from σ^* and querying the decrypt oracle with δ as input (which is allowed in the security model of [9], i.e. δ is totally different from the challenge ciphertext). The only restriction for $\mathcal{A}$ during Phase - 2 is that $\mathcal{A}$ should not query the decryption of the challenge ciphertext σ^* and the extract of $ID_{\mathbb{R}}$. Also, it should be noted that the check $R^{*H_2(m_b, R^*)} \stackrel{?}{=} \hat{e}(t^*P + U^*,\ wP + P_{pub})\ \hat{e}(P,P)^{-1}$ should hold.

Proof of Correctness: The equality of R and R^* can be shown by,

$$\begin{aligned}
R^* &= \hat{e}(T_0^* + t_1^* T_1^* + t_2^* T_2^*, D_R) \\
&= \hat{e}(x(w+s)(q_R+s)P - (t_1^* T_1^* + t_2^* T_2^*) + t_1^* T_1^* + t_2^* T_2^*, D_R) \\
&= \hat{e}(x(w+s)(q_R+s)P, D_R) \\
&= \hat{e}(x(w+s)(q_R+s)P, \frac{1}{q_R+s}P) \\
&= \hat{e}(x(w+s)P, P) = \hat{e}((w+s)P, xP) = \hat{e}(wP + P_{pub}, P)^x = R
\end{aligned}$$

Also, the derived ciphertext δ will pass the verification test, which can be shown as,

$$\begin{aligned}
\hat{e}(t^*P + U^*, wP + P_{pub})\hat{e}(P,P)^{-1} &= \hat{e}((t + r^*)P + U - r^*P, wP + P_{pub})\hat{e}(P,P)^{-1} \\
&= \hat{e}((xH_2(m_b, R^*) + u + r^*)P, wP + P_{pub}) \\
&\qquad \hat{e}(W - (u + r^*)P, wP + P_{pub})\hat{e}(P,P)^{-1} \\
&= \hat{e}(xH_2(m_b, R)P + W, wP + P_{pub})\hat{e}(P,P)^{-1} \text{ (Since } R^* = R) \\
&= \hat{e}(xH_2(m_b, R)P, wP + P_{pub})\hat{e}(W, wP + P_{pub})\hat{e}(P,P)^{-1} \\
&= \hat{e}(xH_2(m_b, R)P, wP + P_{pub})\hat{e}(P,P)\hat{e}(P,P)^{-1} \\
&= \hat{e}(wP + P_{pub}, P)^{xH_2(m_b,R)} = R^{H_2(m_b,R)} = R^{*H_2(m_b,R^*)}
\end{aligned}$$

[1] Although the authors of [9] have claimed that an attack was discussed in a private communication, to the best of our knowledge, it is not recorded anywhere. The attack is subtle and non-trivial. We report the same here.

3.3 A Possible Fix for the Weakness in [9]

The security weakness of [9] shown in section 3.2 can be fixed by providing the modifications to the $On - Encrypt$ algorithm and the definition of the hash function H_2 allowing all other algorithms unaltered. The improved On-Encrypt protocol can be given by,

On-Encrypt(m, ID_A, ϕ)

- $t_1 = \beta^{-1}(H_1(ID_A)\text{-}\ \alpha) \bmod q$
- $t_2 = \beta^{-1}(H_1(ID_A)\text{-}\ \gamma) \bmod q$
- $t = H_2(m, U, R, T_0, T_1, T_2, t_1, t_2)x + u \bmod q$
- $c = H_3(R) \oplus m$
- Output the ciphertext $\sigma = \langle U, T_0, T_1, T_2, t, t_1, t_2, c\rangle$

The hash function H_2 is redefined as $H_2 : \{0,1\}^* \times \mathbb{G}_T \times \mathbb{G}^3 \times \mathbb{Z}_q^* \times \mathbb{Z}_q^* \rightarrow \mathbb{Z}_q^*$

4 The New IBOOE

In this section we provide a new identity based online/offline encryption scheme (New-IBOOE), which is more efficient than the fixed version of [9].

4.1 The Scheme

Let $\mathbb{G}$ be a cyclic additive group and $\mathbb{G}_T$ be a cyclic multiplicative group. Both the groups have prime order, q and let $\hat{e}$:$\mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ be the bilinear pairing. The algorithms in the scheme are described below:

Setup: The PKG selects a generator $P \in_R \mathbb{G}$ and randomly chooses $s \in \mathbb{Z}_q^*$. It computes $P_{pub} = sP$ and $\alpha = \hat{e}(P, P)$. Let $\mathcal{M}$ denotes the message space and $n_M = \mid \mathcal{M} \mid$. Let $H_1 : \{0,1\}^* \rightarrow \mathbb{Z}_q^*$, $H_2 : \{0,1\}^* \times \mathbb{G}_T \times \mathbb{G}^4 \rightarrow \mathbb{Z}_q^*$ and $H_3 : \{0,1\}^* \rightarrow \{0,1\}^{n_M}$ be three cryptographic hash functions. The public parameters $Params$ and master private key msk are given as:

$$Params = \langle \mathbb{G}, \mathbb{G}_T, q, P_{pub}, \alpha, \mathcal{M}, H_1, H_2, H_3\rangle \text{ and } msk = s.$$

Extract(ID_A)**:**
- $q_A = H_1(ID_A)$
- $D_A = \dfrac{1}{q_A + s} P.$

Off-Encrypt$(Params)$**:**
- $u, x, a, \hat{b} \in_R \mathbb{Z}_q^*$
- $U = uP$
- $R = \alpha^x$
- $\beta = H_3(R)$
- $T_1 = a^{-1}xP$
- $T_2 = x(\hat{b} + s)P.$
- Outputs the offline ciphertext $\phi = \langle u, x, a, \hat{b}, U, R, T_1, T_2, \beta\rangle.$

On-Encrypt(m, ID_A, ϕ)**:**
- $t_1 = a(q_A - \hat{b}) \, mod \, q$
- $t_2 = H_2(m, R, U, T_1, T_2, t_1)x + u \, mod \, q$
- $c = \beta \oplus m$
- Outputs the ciphertext $\sigma = \langle U, T_1, T_2, t_1, t_2, c\rangle.$

Decrypt(σ, ID_A, D_A)**:**
- $R = \hat{e}(T_2 + t_1 T_1, D_A)$
- $m = c \oplus H_3(R)$
- $h = H_2(m, R, U, T_1, T_2, t_1)$
- If $R^h \stackrel{?}{=} \hat{e}(t_2 P - U, P)$, output m else output $\perp$

It should be noted that the offline encryption process is carried out before knowing the message m as well as the receiver identity ID_A. These are the attracting features of our scheme. The correctness of the verification of the equation $R^h \stackrel{?}{=} \hat{e}(t_2P - U, P)$ done during the decryption process is given below:

$$\begin{aligned}\text{LHS} = R^h &= \hat{e}(T_2 + t_1T_1, D_A)^h \\ &= \hat{e}(x(\hat{b}+s)P + a(q_A - \hat{b})a^{-1}xP, \tfrac{1}{q_A+s}P)^h \\ &= \hat{e}(x\hat{b}P + xsP + q_AxP - \hat{b}xP, \tfrac{1}{q_A+s}P)^h \\ &= \hat{e}(x(s+q_A)P, \tfrac{1}{q_A+s}P)^h \\ &= \hat{e}(xP, P)^h\end{aligned}$$

$$\text{RHS} = \hat{e}(t_2P - U, P) = \hat{e}((hx+u)P - U, P) = \hat{e}(hxP + uP - U, P) = \hat{e}(xP, P)^h$$

Since LHS=RHS, the verification of a well formed ciphertext holds.

Theorem 1. *If there exists an adversary $\mathcal{A}$ that breaks the IND-IBOOE-CCA2 security of the New-IBOOE scheme then, there exists an algorithm $\mathcal{C}$ to solve the k-modified Bilinear Diffie Hellman Inversion Problem (k-mBDHIP).*

Please refer the proof of this theorem in the full version of the paper [11].

5 Review and Attack of IBOOSC Schemes

In this section, we recall the identity based online/offline schemes by Sun et al. presented in [14] and [15]. We demonstrate attacks on both these schemes in this section.

5.1 Scheme by Sun et al.[14]

Review of the Scheme: The scheme consists of five algorithms - **Setup**, **Extract**, **OffSigncrypt**, **Onsigncrypt** and **UnSigncrypt**. A secure symmetric key encryption scheme $(\mathcal{E}, \mathcal{D})$ is employed in this scheme where $\mathcal{E}$ and $\mathcal{D}$ are the secure symmetric encryption and decryption algorithms respectively.

Setup: Given security parameters κ, n and $\mathbb{G}_1$,$\mathbb{G}_2$ of order q and generator P of $\mathbb{G}_1$, PKG picks a random $s \in \mathbb{Z}_q^*$, ands sets $P_{pub} = sP$. Choose cryptographic hash functions H_0: $\{0,1\}^* \rightarrow \mathbb{G}_1$, H_1: $\{0,1\}^* \times \mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{Z}_q^*$, H_2: $Z_q^* \rightarrow \{0,1\}^n$, H_3: $\mathbb{G}_2 \rightarrow \mathbb{Z}_q^* \times \mathbb{Z}_q^*$. The system parameters are $\langle P, P_{pub}, H_0, H_1, H_2, H_3\rangle$. The master secret key is s.

Key Extract: Given an identity ID_i, the algorithm computes the public key as $Q_i = H_0(ID_i)$ and the corresponding private key as $D_i = sH_0(ID_i)$. The private key is returned to the user via a secure channel.

OffSigncrypt: To send a message m to user $U_{\mathbb{R}}$ with identity $ID_{\mathbb{R}}$, the sender $U_{\mathbb{S}}$ with identity $ID_{\mathbb{S}}$ follows the steps below.

1. Computes $Q_{\mathbb{R}} = H_0(ID_{\mathbb{R}})$.
2. Picks random x, y $\in \mathbb{Z}_q^*$, and sets $k = H_3(e(P_{pub}, Q_{\mathbb{R}})^x)$.

3. Splits k into k_1, k_2 such that $k_1 \in \mathbb{Z}_q^*$ and $k_2 \in \mathbb{Z}_q^*$, then stores them for future use.
4. Using the private key $D_{\mathbb{S}}$, $U_{\mathbb{S}}$ outputs the offline signcryption (S, U), where $S = D_{\mathbb{S}} - xP_{Pub}$, $U = (y - k_1)P$; also stores x, y for future use.

OnSigncrypt: Given a message $m \in \mathbb{Z}_q^*$, and an off-line signcryption (S, U), this algorithm sets $k_3 = H_2(k_2)$ first. The message encryption is done with k_3 and a symmetric-key encryption algorithm $\mathcal{E}$ such as AES. The ciphertext is $c = \mathcal{E}_{k_3}(m)$. Computes $r = H_1(c, S, U)$ and on-line signcryption $\sigma = rx + y$; returns signcryption (c, S, U, σ).

UnSigncrypt: Given a signcryption (c, S, U, σ), the receiver with identity $ID_{\mathbb{R}}$ does the following:

1. Computes $T = e(-S, Q_{\mathbb{R}})e(Q_{\mathbb{S}}, D_{\mathbb{R}})$.
2. Sets $k = H_3(T)$,then splits k into k_1, k_2.
3. Sets $k_3 = H_2(k_2)$ and decrypts the message $\mathcal{D}_{k_3}(c) = m$. m is valid if $e(\sigma P_{pub} + rS, P) \stackrel{?}{=} e(U + k_1 P + rQ_{ID_A}, P_{pub})$ holds, where $r = H_1(c, S, U)$.

Existential Forgeability of the Scheme: This scheme is not secure against existential forgery. A forger $\mathcal{F}$ can forge a signcryption for an identity whose private key is not queried. This can be done as follows:

- $\mathcal{F}$ sets an identity ID_A as the target identity for which the forged signcryption is to be generated.
- During unforgeability game, a forger is allowed to extract the private key of receiver (used for generating the forgery) according to the model given by Sun et al [14]
- During the Training phase, $\mathcal{F}$ asks for the signcryption of a message m from ID_A to an arbitrary receiver ID_B. Let the response be (c, S, U, σ). On receiving this, $\mathcal{F}$ computes the following
  - Gets the private key of ID_B using a Key_Extract query on ID_B.
  - Computes $T = \hat{e}(-S, Q_B)\hat{e}(Q_A, D_B)$
  - Sets $k = H_3(T)$ and divides k into two parts: k_1 and k_2.
- $\mathcal{F}$ can now modify the above ciphertext (c, S, U, σ) so that it becomes a valid signcryption on some message m' from ID_A to an arbitrary ID_C. For achieving this $\mathcal{F}$ computes following:
  - $T' = \hat{e}(-S, Q_C)\hat{e}(Q_A, D_C)$
  - $k' = H_3(T')$ and it is divided into two parts: k_1' and k_2'
  - $\Delta k = k_1' - k_1$ and $\sigma' = rx + y + \Delta k$
  - Outputs the new signcryption (c, S, U, σ')

This will pass through the verification because

$$\begin{aligned}\text{LHS} &= \hat{e}(\sigma' P_{pub} + rS, P)\\ &= \hat{e}((rx + y + \Delta k)P_{pub} + r(D_A - xP_{pub}), P)\\ &= \hat{e}((y + \Delta k)P_{pub} + rsQ_A, P)\end{aligned}$$

$= \hat{e}((y + k_1' - k_1)P + rQ_A, sP)$
$= \hat{e}((y - k_1)P + k_1'P + rQ_A, P_{pub})$
$= \hat{e}(U + k_1'P + rQ_A, P_{pub})$
$=$ RHS

5.2 Generic Scheme by Sun et al. [15]

Review of the Scheme: We review the generic online/offline signcryption scheme by Sun et al. [15] in this section.

Systems Parameter Generation: Let t be a prime power, and $E(\mathbb{F}_t)$ an elliptic curve over finite field $\mathbb{F}_t$. Let $\#E(\mathbb{F}_t)$ be the number of points of $\#E(\mathbb{F}_t)$, and P be a point of $E(\mathbb{F}_t)$ with prime order q where $q|\#E(\mathbb{F}_t)$. $\mathbb{G}_1$ is the subgroup generated by P. $\mathbb{G}_2$ is a finite group of order q. Choose cryptographic hash function $H_1 : \mathbb{G}_2 \rightarrow \{0,1\}^n$. Let $(\mathcal{L}, \mathcal{H})$ be the chameleon hash family, which will be sent to the designated user on request, based on the discrete logarithm assumption and $(\mathcal{G}, \mathcal{S}, \mathcal{V})$ be any identity-based signature scheme. The system parameters are $SP = (\#E(\mathbb{F}_t), t, q, P, \mathbb{G}_1, \mathbb{G}_2, (\mathcal{G}, \mathcal{S}, \mathcal{V}), H_1)$.

Key Extract: Given an identity ID, run the key extract algorithm of the original identity-based signature scheme to obtain the private/public key pair (D_{ID}, Q_{ID}). On input 1^k, the sender runs the key generation algorithm of the trapdoor hash family $(\mathcal{L}, \mathcal{H})$ to obtain the hash/trapdoor key pair $(Y = xP, x)$.

Assume user $U_{\mathbb{S}}$ with identity $ID_{\mathbb{S}}$ sends m to user $U_{\mathbb{R}}$ with identity $ID_{\mathbb{R}}$. $U_{\mathbb{S}}$ obtains private key and hash/trapdoor key $\{D_{\mathbb{S}}, Y, x\}$. $U_{\mathbb{R}}$ obtains private key $D_{\mathbb{R}}$. $\{Q_{\mathbb{S}}, Q_{\mathbb{R}}\}$ are public to both of them.

OffSigncrypt: Offline signcryption is done as follows:

- Choose at random $(m, r) \in_R \mathcal{M} \times \mathcal{R}$, where $\mathcal{M}$ is a message space and $\mathcal{R}$ is a finite space, and compute the chameleon hash value $h = H_Y(m', r') = m'P + r'Y$.
- Run the signing algorithm $\mathcal{S}$ with the signing key $D_{\mathbb{S}}$ to sign the hash value h. Let the output be $\sigma = S_{D_{\mathbb{S}}}(h||H_Y)$, where H_Y is the description of the chameleon hash.
- Choose at random $y \in_R Z_q^*$ and compute $X = yP$ then compute $\omega = e(yP_{pub}, Q_{\mathbb{R}})$. Finally set $y' = H_1(\omega)$.
- Store the pair (m', r') and y' for future use.

OnSigncrypt: Online signcryption is done as follows:

- For a given message m, retrieve from the memory x^{-1} and the pair (m, r).
- Compute $r = x^{-1}(m' - m) + r' \bmod q$.
- The message encryption is done with y' and a symmetric-key encryption algorithm such as AES. The ciphertext is $c = Enc_{y'}(\sigma||ID_{\mathbb{S}}||m||r||H_Y)$.
- Final ciphertext is (c, X).

Unsigncrypt: Given ciphertext (c, X), unsigncryption is done as follows:

- Compute $\omega = e(X, d_{ID_B})$ and $y' = H_1(\omega)$.
- Decrypt c as $\sigma||ID_{\mathbb{S}}||m||r||H_Y = Dec_{y'}(c)$.
- Compute $h = H_Y(m, r) = mP + rY$.
- Verify that σ is indeed a signature of the value $h||H_Y$ with respect to the verification key $Q_{\mathbb{S}}$.

Attack on the Scheme: In the scheme proposed by Sun et al. [15], there is no binding between the encryption and the signature. Therefore, a signcryption on a message m from ID_A to ID_B can be changed to a valid signcryption on the same message m from ID_A to ID_C. This can be done as follows:

- Get the signcryption of message m from the sender ID_A to receiver ID_B and decrypt it using the secret key D_B of ID_B to get $\sigma||ID_A||m||r||H_Y$.
- Choose $\eta \in_R Z_q^*$ and compute $\omega^* = \hat{e}(P_{pub}, Q_C)^\eta$ and set $X^* = \eta P$ and $y^* = H_1(\omega)$.
- Compute $c^* = Enc_{y^*}(\sigma||ID_A||m||r||H_Y)$
- Output the signcryption as (c^*, X^*)

Note that Q_C is the public key of the user with identity ID_C whose private key is not known. The new signcryption (c^*, X^*) is a valid signcryption from ID_A to ID_C.

6 The New IBOOSC

In this section, we present a provably secure identity based online/offline signcryption scheme. It should be noted that the scheme presented in this section is more efficient than the naive combination of online/offline identity based signature and online/offline identity based encryption because, we have considered the case where the receiver identity is not known during the offline phase and more over it is explicit that just combining a signature scheme and an encryption scheme is not signcryption but signcryption should involve cheap computation than the naive combination. The size of the ciphertext and the computations done during the unsigncryption process are bulky than normal signcryption but we consider only the computation complexity of the online signcryption algorithm where we have only modular addition, multiplication and bit-wise exclusive-OR operations. This is considered as the highlight of any online/offline primitive. The IBOOSC scheme consists of the following algorithms:

Setup(1^κ)**:** Given the security parameter 1^κ as input, PKG chooses two groups $\mathbb{G}_1$, $\mathbb{G}_2$ of prime order q, a bilinear map $\hat{e}$: $\mathbb{G}_1 \times \mathbb{G}_1 \rightarrow \mathbb{G}_2$ and a generator $P \in_R \mathbb{G}_1$. The PKG chooses $s \in_R \mathbb{Z}_q^*$ and sets master secret key $msk = s$ and also sets master public key $P_{pub} = sP$. PKG then computes $\alpha = \hat{e}(P, P)$ and defines five cryptographic hash functions:

- H_1: $\{0,1\}^* \rightarrow \mathbb{G}_1$.
- H_2: $\mathbb{G}_1 \times \mathbb{G}_1 \times \mathbb{G}_1 \times \{0,1\}^{n_1} \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.
- H_3: $\{0,1\}^{n_1} \times \{0,1\}^* \times \mathbb{G}_1 \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.
- H_4: $\mathbb{G}_2 \rightarrow \{0,1\}^{n_1+n_m}$. where n_m is the message size n_1 is the number of random bits concatenated to message.
- H_5: $\{0,1\}^{n_m} \times \mathbb{G}_2 \times \{0,1\}^{n_1} \times \mathbb{Z}_q^* \times \mathbb{Z}_q^* \times \{0,1\}^{n_1+n_m} \times \{0,1\}^* \times \{0,1\}^* \rightarrow \mathbb{Z}_q^*$.

The public parameters *Params* of the system are set to be *Params* = $\langle \mathbb{G}_1, \mathbb{G}_2, \hat{e}, P, R, P_{pub}, H_1, H_2, H_3, H_4, H_5, \alpha \rangle$.

Key Extract(ID_i)**:** On input of identity ID_i of user U_i, the private key D_i is computed as $D_i = (\frac{1}{q_i+s})P)$, where $q_i = H_1(ID_i)$. D_i is given to user by PKG via. secure channel.

Off-Signcrypt$(ID_{\mathbb{S}},\ D_{\mathbb{S}})$**:** This algorithm is run by the sender $U_{\mathbb{S}}$ with identity $ID_{\mathbb{S}}$ for sending any message to any receiver. Note that the sender carries out these computations without the knowledge message and receiver information.

1. Selects $\delta \in_R \{0,1\}^{n_1}$ and $b, x, y, z, r \in_R \mathbb{Z}_q^*$.
2. Computes $U_1 = \alpha^r \in \mathbb{G}_2$, $U_2 = yP \in \mathbb{G}_1$ and $U_3 = zP \in \mathbb{G}_1$.
3. Computes $V = (r + h_2)D_{\mathbb{S}} \in \mathbb{G}_1$, where $h_2 = H_2(U_1, U_2, U_3, \delta, ID_S)$.
4. Computes $a = H_3(\delta, V, ID_{\mathbb{S}})$.
5. Computes $C_1 = a^{-1}xP$, $C_2 = x(b + s)P$.
6. Sets $k = H_4(\omega = \alpha^x)$.

Outputs the offline signcryption $\sigma' = \langle C_1, C_2, V, U_1, U_2\rangle$, while $\sigma_{secret} = \langle k, \omega, a, b, y, z\rangle$ are kept as secret for future use in online phase and they are not made public. Note here that the output of the $Off - Signcrypt$ algorithm can be used only once to generate an online signcryption.

Remark: It should be noted that above offline signcryption σ' does not require the knowledge of the message or the receiver.

On-Signcrypt$(m,\ ID_{\mathbb{S}},\ ID_{\mathbb{R}},\ \sigma',\ \sigma_{secret})$: This algorithm is run by the sender, once the message $m \in \mathcal{M}$ and the receiver identity $\mathrm{ID}_{\mathbb{R}}$ are available and makes use of the offline signature $\sigma' = \langle C_1, C_2, V, U_1, U_2\rangle$, along with the stored values $\sigma_{secret} = \langle k, \omega, a, b, y, z\rangle$.

1. Compute $C_3 = a(q_{\mathbb{R}} - b) \bmod q$.
2. Compute $C_4 = (m\|\delta) \oplus k$.
3. Compute $v = yh + z \bmod q$ where $h = H_5(m, \omega, \delta, h_2, C_3, C_4, ID_{\mathbb{S}}, ID_{\mathbb{R}})$.
4. Outputs the signcryption $\sigma = \langle \{C_i\}_{i=1\,to\,4}, U_1, U_2, U_3, V, v\rangle$.

Remark: Here, the $On - Signcrypt$ phase includes only one hash computation.

Unsigncrypt$(\sigma,\ ID_{\mathbb{S}},\ ID_{\mathbb{R}},\ D_{\mathbb{R}})$: When the receiver $U_{\mathbb{R}}$ with identity $ID_{\mathbb{R}}$ is provided with the signcryption $\langle \sigma, U_{\mathbb{S}}, U_{\mathbb{R}}\rangle$ uses the following steps to unsigncrypt the signcryption $\sigma = \langle \{C_i\}_{i=1\,to\,4}, U_1, U_2, U_3, V, v\rangle$ from $ID_{\mathbb{R}}$:

1. Computes $\omega' = \hat{e}(C_3C_1 + C_2, D_{\mathbb{R}})$ and $k' = H_3(\omega')$.
2. $(m'\|\delta') = C_4 \oplus k'$.
3. Computes $h_2' = H_2(U_1, U_2, U_3, \delta', ID_S)$ and $h' = H_5(m', \omega', \delta', h_2', C_3, C_4, ID_{\mathbb{S}}, ID_{\mathbb{R}})$.
4. Verify $h'U_2 + U_3 \stackrel{?}{=} vP$, $\hat{e}(P, C_1)^{H_3(\delta', V, ID_{\mathbb{S}})} \stackrel{?}{=} \omega'$ and $\hat{e}(V, (q_{\mathbb{S}} + s)P)\alpha^{-h_2'} \stackrel{?}{=} U_1$
5. If all the checks in the above step holds, then output the message m', else output "$Invalid$".

Correctness: We show the correctness of the unsigncryption algorithm here.

$$\begin{aligned}\omega' = \hat{e}(C_3C_1 + C_2, D_{\mathbb{R}}) &= \hat{e}((q_{\mathbb{R}} - b)xp + x(b + s)P, \tfrac{1}{q_{\mathbb{R}}+s}P)\\ &= \hat{e}((q_{\mathbb{R}} + s)xP, \tfrac{1}{q_{\mathbb{R}}+s}P)\\ &= \hat{e}(xP, P) = \hat{e}(P, P)^x = \alpha^x = \omega\end{aligned}$$

The correctness of the verification tests $U_2h' + U_3 \stackrel{?}{=} vP$, $\hat{e}(P, C_1)^{H_3(\delta', V, ID_{\mathbb{S}})} \stackrel{?}{=} \omega'$ and $\hat{e}(V, (q_{\mathbb{S}} + s)P)\alpha^{-h'} \stackrel{?}{=} U_1$ is shown below:

Correctness of $U_2h' + U_3 \stackrel{?}{=} vP$

$h'U_3 + U_1 = h'(yP) + rP = (h'y + r)P = vP$

Correctness of $\hat{e}(P, C_1)^{H_3(\delta', V, ID_{\mathbb{S}})} \stackrel{?}{=} \omega'$

$\hat{e}(P, C_1)^{H_3(\delta', V, ID_{\mathbb{S}})} = \hat{e}(P, a^{-1}xP)^a = \hat{e}(P, P)^x = \omega' = \omega$

Correctness of $\hat{e}(V, (q_{\mathbb{S}} + s)P)\alpha^{-h'} \stackrel{?}{=} U_1$

$$\begin{aligned}\hat{e}(V, (q_{\mathbb{S}} + s)P)\alpha^{-h'_2} &= \hat{e}((r + h_2)D_{\mathbb{S}}, (q_{\mathbb{S}} + s)P)\hat{e}(P, P)^{-h'_2} \\ &= \hat{e}((r + h_2)\frac{1}{q_{\mathbb{S}} + sP}, (q_{\mathbb{S}} + s)P)\hat{e}(P, P)^{-h'_2} \\ &= \hat{e}(P, P)^{r+h_2}\hat{e}(P, P)^{-h'_2} = \hat{e}(P, P)^r = U_1\end{aligned}$$

7 Security Analysis of Our IBOOSC

In the new identity based online/offline signcryption scheme proposed above, we are not directly signing the message, instead two randomness are signed which are acting as the public keys for signing the message using a one-time schnorr signature[10].

Theorem 2. *If there exists an attacker $\mathcal{A}$ that can break the IND-IBOOSC-CCA2 security (confidentiality) of IBOOSC, then there exists an algorithm $\mathcal{C}$ that is capable of solving the q-SDHIP.*

Please refer the proof of this theorem in the full version of the paper [12].

Theorem 3. *If there exists an attacker $\mathcal{A}$ who can break the EUF-IBOOSC-CMA security of IBOOSC, then there exists an algorithm $\mathcal{C}$ that is capable of solving the q-CDHIP.*

Please refer the proof of this theorem in the full version of the paper [12].

Conclusion

Identity based encryption schemes wherein the encryption is carried out in two phases namely, offline and online phase according to the complexity of the operations performed is known to be identity based online/offline encryption scheme. The subtle issue in designing an identity based online/offline encryption scheme is to split the operations into heavy weight (for offline phase) and light weight (for online phase) without knowing the message and receiver. [9] gives a solution for this problem in the random oracle model. In this paper, we have pointed out that the scheme in [9] is not CCA secure. We have proposed a possible fix for the same and have also given a more efficient identity based online/offline encryption scheme. We have formally proved the security of the new scheme in the random oracle model. The complexity figure of our scheme is given below:

Table 1. Comparison of Complexity

Scheme	Encrypt					Decrypt				
	Offline			Online						
	BP	SPM	EXP	M	Ex	BP	SPM	EXP	M	Ex
Improved L-IBOOE (Sec. 3.3)	1	7	1	3	1	3	4	1	-	1
New-IBOOE	-	4	1	2	1	2	2	1	-	1

SPM - Scalar Point Multiplication, BP - Bilinear Pairing, EXP - Exponentiation in $\mathbb{G}_T$, M - Modular Computation in $\mathbb{Z}_q^*$, Ex - Exclusive OR

We have also showed security weaknesses in two existing identity based online/offline signcryption schemes[14,15]. Also, we proposed a provably secure identity based online/offline signcryption scheme which does not require the knowledge of the message and receiver. We proved the security of our scheme in the random oracle model. Since two existing identity based online/offline signcryption schemes are showed to be flawed in one way or the other we compare our scheme only with [8]. The IBOOSC scheme presented in this paper has efficiency gain in the online signcryption phase and unsigncryption with one less modular arithmetic and one less hashing, and has one less pairing during respectively when compared with [8]

Table 2. Comparison of Complexity

Scheme	Signcrypt						Unsigncrypt				
	Offline			Online							
	BP	SPM	EXP	M	Ex	HF	BP	SPM	EXP	M	Ex
[8]	-	6	1	3	1	3	3	4	-	-	1
IBOOSC	-	6	2	2	1	2	2	5	-	-	1

SPM - Scalar Point Multiplication, BP - Bilinear Pairing, EXP - Exponentiation in $\mathbb{G}_T$, M - Modular Computation in $\mathbb{Z}_q^*$, Ex - Exclusive OR, HF - Hash Computation

References

1. An, J.H., Dodis, Y., Rabin, T.: On the Security of Joint Signature and Encryption. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 83–107. Springer, Heidelberg (2002)
2. Boneh, D., Boyen, X.: Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In: Cachin, C., Camenisch, J.L. (eds.) EUROCRYPT 2004. LNCS, vol. 3027, pp. 223–238. Springer, Heidelberg (2004)
3. Chow, S.S.M., Liu, J.K., Zhou, J.: Identity-based online/offline key encapsulation and encryption. Cryptology ePrint Archive, Report 2010/194 (2010)
4. Dutta, R., Barua, R., Sarkar, P.: Pairing-based cryptographic protocols: A survey. In: Cryptology ePrint Archive, Report 2004/064 (2004)

5. Even, S., Goldreich, O., Micali, S.: On-line/off-line digital signatures. Journal of Cryptology 9(1) (1996)
6. Gentry, C.: Practical Identity-Based Encryption Without Random Oracles. In: Vaudenay, S. (ed.) EUROCRYPT 2006. LNCS, vol. 4004, pp. 445–464. Springer, Heidelberg (2006)
7. Guo, F., Mu, Y., Chen, Z.: Identity-Based Online/Offline Encryption. In: Tsudik, G. (ed.) FC 2008. LNCS, vol. 5143, pp. 247–261. Springer, Heidelberg (2008)
8. Liu, J.K., Baek, J., Zhou, J.: Online/Offline identity-based signcryption revisited. In: Lai, X., Yung, M., Lin, D. (eds.) Inscrypt 2010. LNCS, vol. 6584, pp. 36–51. Springer, Heidelberg (2011), http://eprint.iacr.org/
9. Liu, J.K., Zhou, J.: An Efficient Identity-Based Online/Offline Encryption Scheme. In: Abdalla, M., Pointcheval, D., Fouque, P.-A., Vergnaud, D. (eds.) ACNS 2009. LNCS, vol. 5536, pp. 156–167. Springer, Heidelberg (2009)
10. Schnorr, C.-P.: Efficient signature generation by smart cards. J. Cryptology 4(3) (1991)
11. Sharmila Deva Selvi, S., Sree Vivek, S., Pandu Rangan, C.: Identity based online/offline encryption scheme. Cryptology ePrint Archive, Report 2010/178 (2010)
12. Sharmila Deva Selvi, S., Sree Vivek,S., Pandu Rangan, C.: Identity based online/offline signcryption scheme. Cryptology ePrint Archive, Report 2010/376 (2010)
13. Shamir, A., Tauman, Y.: Improved Online/Offline Signature Schemes. In: Kilian, J. (ed.) CRYPTO 2001. LNCS, vol. 2139, pp. 355–367. Springer, Heidelberg (2001)
14. Sun, D., Huang, X., Mu, Y., Susilo, W.: Identity-based on-line/off-line signcryption. In: Cao, J., Li, M., Wu, M.-Y., Chen, J. (eds.) NPC 2008. LNCS, vol. 5245, pp. 34–41. Springer, Heidelberg (2008)
15. Sun, D., Mu, Y., Susilo, W.: A generic construction of identity-based online/offline signcryption. In: ISPA, pp. 707–712. IEEE, Los Alamitos (2008)
16. Zhang, F., Mu, Y., Susilo, W.: Reducing security overhead for mobile networks. In: AINA 2005: Proceedings of the 19th International Conference on Advanced Information Networking and Applications, pp. 398–403. IEEE Computer Society, Los Alamitos (2005)
17. Zheng, Y.: Digital Signcryption or How to Achieve Cost (Signature & Encryption) << Cost(Signature) + Cost(Encryption). In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 165–179. Springer, Heidelberg (1997)