Online/Offline Identity-Based Signcryption Re-visited

Joseph K. Liu, Joonsang Baek, and Jianying Zhou

Cryptography and Security Department
Institute for Infocomm Research (I^2R), Singapore
{ksliu, jsbaek, jyzhou}@i2r.a-star.edu.sg

Abstract. In this paper, we re-define a cryptographic notion called *Online/Offline Identity-Based Signcryption*. It is an "online/offline" version of identity-based signcryption, where most of the computations are carried out offline and the online part does not require any heavy computations such as pairings or multiplications on elliptic curve. It is particularly suitable for power-constrained devices such as smart cards. We give a concrete implementation of online/offline identity-based signcryption. The construction is very efficient and flexible. Unlike all previous schemes in the literature, our scheme does *not* require the knowledge of receiver's information (either public key or identity) in the offline stage. The receiver's identity and the message to be signcrypted are only needed in the online stage. This feature provides great flexibility to our scheme and makes it practical to use in real-world applications. We prove that the proposed scheme meets strong security requirements in the random oracle model, assuming the Strong Diffie-Hellman (SDH) and Bilinear Diffie-Hellman Inversion (BDHI) are computationally hard.

1 Introduction

1.1 Motivation

Providing efficient mechanisms for authentication and confidentiality is probably the most important requirement in electronic transactions, especially in mobile devices or smart cards. Since attackers can easily access the physical layer and launch some potential attacks in such devices, inclusion of cryptographic protection as a countermeasure should be very effective. However, due to the power-constrained nature of these devices, only light operations are allowed to be implemented. For this reason, efficiency becomes the main concern in the design of cryptographic algorithm for such environment.

IDENTITY-BASED CRYPTOGRAPHY. Identity (ID)-based Cryptography, introduced by Shamir [16], eliminates the need for checking the validity of certificates in traditional public key infrastructure (PKI). In an ID-based cryptography, public key of each user is easily computable from an arbitrary string corresponding to this user's identity (e.g. an email address, a telephone number, and etc.). Using its master key, the private key generator (PKG) then computes a private key for each user. This property avoids the requirement of using digital certificates (which contain Certificate Authority (CA)'s signature on each user's public key) and associates implicitly a public key (i.e. user identity) to each user within the system. One only needs to know the recipient's identity in order to send an encrypted message to him. It avoids the complicated and costly certificate (chain) verification for the authentication purpose. In the case of signature schemes, verification takes only the identity together with the message and a signature pair as input and executes the algorithm directly. In contrast, the traditional PKI needs an additional certification verification process, which is in fact equivalent to the computation of *two* signature verifications.

We argue that ID-based cryptography is particularly suitable for smart cards. The most important reason is that it eliminates the costly certificate verification process and the storage of the lengthy certificate. In addition, when there is a new card issued, other terminals or payment gateways do not need to have its certificate verified in order to communicate in a secure and authenticated way. This can greatly reduce communication overhead and computation cost.

Signcryption. Signcryption, whose concept was introduced by Zheng [21], is a cryptographic primitive aiming to provide unforgeability and confidentiality simultaneously as typical signature-then-encryption technique does but with less computational complexity and lower communication cost. Due to the efficiency one can obtain, signcryption is suitable for many applications which require secure and authenticated message delivery using resource-constrained devices.

The idea of ID-based signcryption was first proposed by Malone-Lee [15]. It was further improved in [7, 13, 2, 8] for efficiency and security.

Online/Offline Signature. Online/Offline Signature was first introduced by Even, Goldreich and Micali [9]. The main idea is to perform signature generation in two phases. The first phase is performed offline (before the message to be signed is given) and the second phase is performed online (after the message to be signed is given). Online/offline signature schemes are useful, since in many applications the signer has a very limited response time once the message is presented, but he can carry out costly computations between consecutive signing requests. We note that smart card applications may take full advantages of online/offline signature schemes: The offline phase is implemented during the card manufacturing process, while the online phase uses the stored result of the offline phase to sign actual messages. The online phase is typically very fast, and hence can be executed efficiently even on a weak processor.

Online/Offline Signcryption. The notion of online/offline signcryption was first introduced by An, Dodis and Rabin [1]. As in the case of online/offline signature schemes, online/offline signcryption schemes should satisfy a basic property that online computation should be performed very efficiently. All expensive operations such as exponentiation or pairing computation should be conducted offline in the first phase of the scheme. Similar to online/offline signature, it is also reasonable to assume that the offline operations are independent of the particular message to be signed and encrypted, since the message only becomes available at a later stage. The second phase is performed online, once the message is presented.

An, Dodis and Rabin [1] did not give any concrete construction of online/offline signcryption, but focused mainly on establishing formal security model for signcryption and analysis of some generic constructions. The first concrete online/offline signcryption scheme was given by Zhang, Mu and Susilo [20], and it requires an additional symmetric key encryption scheme to achieve confidentiality. Another scheme can be found in [19]. However, its practicality is dubious since the scheme requires every user to execute a key exchange protocol with the remaining users in the system. Moreover, both of them are in the PKI (non ID-based) setting. The first ID-based online/offline signcryption scheme was given by Sun, Mu and Susilo [17] in a semi-generic setting, from any ID-based signature scheme.

1.2 Limitation of existing schemes

All of the schemes mentioned above have a restriction which renders them impractical in many situations: They require the receiver's public key / identity to be known in the offline phase, which can result in serious performance degradation. Smartcard is one of the examples. Suppose there are some sensitive data stored in a smartcard, which has only very limited computation power. In order to send the sensitive data to a recipient in a secure and authenticated way, it should be encrypted using the recipient's public key and signed with the card owner's private key. To ensure timely and efficient delivery, it would be much better if part of the signcryption process could be done *prior* to know the data to be encrypted *and* the recipient's public key. Wireless sensor network (WSN) or mobile devices can be another example. Similar to smartcard, wireless sensors or mobile devices such as PDA or smart phone have only limited resources. It may take very long time, or even impossible to execute heavy computations on those tiny devices. Yet the data they process may be sensitive which is necessary to be encrypted and authenticated before sending off to the terminal stations. By using online/offline signcryption, the offline part (containing all heavy computation) can be done by a third party at the setup or manufacturing stage or when external power is connected. However, it is obvious that the data to be processed and the receiver's information is unknown at this stage.

In the above examples, the previous online/offline signcryption schemes (such as [1, 20]) cannot be used, since they require the receiver's public key in the offline stage. This maybe one of the very important reasons that previous online/offline signcryption schemes are not practical to be used in daily life applications.

1.3 Our Contributions

In this paper, we make the following contributions.

1. We re-formulate the notion of *online/offline signcryption in the ID-based setting*. We argue that it would be the best solution to provide authentication and confidentiality to smart cards or mobile devices for the following reasons. First, it combines the separate process of *sign* and *encrypt* into one "*signcryption*". Second, it even splits the signcryption process into *online* and *offline* stages, so that all the heavy computations can be performed in the offline stage, leaving only light operations such as hashing or integer multiplication to be done on tiny devices when the signcrypted message is known. Third, it is in the identity-based setting which gets rid of the costly process of certificate verification.
2. We present a concrete online/offline ID-based signcryption scheme, which does not require any heavy computation (such as pairing or elliptic curve multiplication) in the online stage. The security is proven using two assumptions, namely the Strong Diffie-Hellman (SDH) and Bilinear Diffie-Hellman Inversion (BDHI) assumptions in the random oracle model.
3. More importantly, unlike all other previous schemes, our proposed scheme does *not* require the receiver's information (in our case, the identity) in the offline stage. The receiver's identity, together with the message to be signcrypted, are needed only in the online stage. This feature greatly increases the practicality of online/offline signcryption scheme. Our scheme is the first in the literature to allow this kind of flexibility.
4. When compared to the combination of online/offline ID-based encryption and online/offline ID-based signature, although the combination may achieve the same features as our scheme, efficiency is far more behind. Our scheme is about 30%−50% more efficient than any combination of the state of the art online/offline ID-based encryption and signature schemes.

1.4 Organization

The rest of the paper are organized as follows. We review some definitions in Section 2. It is followed by our proposed scheme in Section 3. We analyze the performance of our scheme in Section 4. Our paper is concluded in Section 5.

2 Definitions

2.1 Pairings

We briefly review the bilinear pairing. Let $\mathbb{G}$ and $\mathbb{G}_T$ be two multiplicative cyclic groups of prime order q. Let g be a generator of $\mathbb{G}$, and e be a bilinear map such that $e : \mathbb{G} \times \mathbb{G} \to \mathbb{G}_T$ with the following properties:

1. *Bilinearity*: For all $u, v \in \mathbb{G}$, and $a, b \in \mathbb{Z}$, $e(u^a, v^b) = e(u, v)^{ab}$.
2. *Non-degeneracy*: $e(g, g) \neq 1$.
3. *Computability*: It is efficient to compute $e(u, v)$ for all $u, v \in \mathbb{G}$.

2.2 Intractability Assumption

Definition 1 (ℓ-Strong Diffie-Hellman Assumption (ℓ-SDH)). *[4] The ℓ-Strong Diffie-Hellman (ℓ-SDH) problem in $\mathbb{G}$ is defined as follow: On input a $(\ell+1)$-tuple $(g,\ g^{\alpha}, g^{\alpha^2}, \cdots, g^{\alpha^\ell}) \in \mathbb{G}^{\ell+1}$, output a pair $(g^{\frac{1}{\alpha+c}}, c)$ where $c \in \mathbb{Z}_q^*$. We say that the (t, ϵ, ℓ)-SDH assumption holds in $\mathbb{G}$ if no t-time algorithm has advantage at least ϵ in solving the ℓ-SDH problem in $\mathbb{G}$.*

Definition 2 (ℓ-Bilinear Diffie-Hellman Inversion Assumption (ℓ-BDHI)). *[3] The ℓ-Diffie-Hellman (ℓ-BDHI) problem in $\mathbb{G}$ is defined as follow: On input a $(\ell{+}1)$-tuple $(g, g^{\alpha}, g^{\alpha^2}, \cdots, g^{\alpha^{\ell}}) \in \mathbb{G}^{\ell+1}$, output $e(g,g)^{\frac{1}{\alpha}} \in \mathbb{G}_T$. We say that the (t, ϵ, ℓ)-BDHI assumption holds in $\mathbb{G}$ if no t-time algorithm has advantage at least ϵ in solving the ℓ-BDHI problem in $\mathbb{G}$.*

2.3 Definition of Signcryption

An ID-based online/offline signcryption scheme consists of the following six probabilistic polynomial time (PPT) algorithms:

- $(param, msk) \leftarrow \mathsf{Setup}(1^k)$ takes a security parameter $k \in \mathbb{N}$ and generates $param$ the global public parameters and msk the master secret key of the PKG.
- $D_{ID} \leftarrow \mathsf{Extract}(1^k, param, msk, ID)$ takes a security parameter k, a global parameters $param$, a master secret key msk and an identity ID to generate a secret key D_{ID} corresponding to this identity.
- $\bar{\phi} \leftarrow \mathsf{Offline\text{-}Signcrypt}(1^k, param, D_{ID_s})$ takes a security parameter k, a global parameters $param$, a secret key of the sender D_{ID_s}, to generate an offline ciphertext $\bar{\phi}$.
- $\phi \leftarrow \mathsf{Online\text{-}Signcrypt}(1^k, param, m, \bar{\phi}, ID_r)$ takes a security parameter k, a global parameters $param$, a message m, an identity of the receiver ID_r where $ID_s \neq ID_r$, an offline ciphertext $\bar{\phi}$ to generate a ciphertext ϕ.
- $(m, \sigma, ID_s)/ \perp \leftarrow \mathsf{De\text{-}Signcrypt}(1^k, param, \phi, D_{ID_r})$ takes a security parameter k, a global parameters $param$, a ciphertext ϕ, a secret key of the receiver D_{ID_r} to generate a message m, a signature σ and an identity ID_s, or $\perp$ which indicates the failure of de-signcryption.
- $\mathsf{valid}/ \perp \leftarrow \mathsf{Verify}(1^k, param, m, \sigma, ID_s)$ takes a security parameter k, a global parameters $param$, a message m, a signature σ, an identity ID_s to output valid of $\perp$ for an invalid signature.

For simplicity, we omit the notation of 1^k and $param$ from the input arguments of the above algorithms in the rest of this paper. For correctness, if

$$\begin{aligned} \bar{\phi} &\leftarrow \mathsf{Offline\text{-}Signcrypt}(D_{ID_s}) \\ \phi &\leftarrow \mathsf{Online\text{-}Signcrypt}(m, \bar{\phi}, ID_r) \\ (\tilde{m}, \tilde{\sigma}, \tilde{ID}_s) &\leftarrow \mathsf{De\text{-}Signcrypt}(\phi, D_{ID_r}) \end{aligned}$$

we require that

$$\begin{aligned} \tilde{m} &= m \\ \tilde{ID}_s &= ID_s \\ \mathsf{valid} &\leftarrow \mathsf{Verify}(\tilde{m}, \tilde{\sigma}, \tilde{ID}_s) \end{aligned}$$

Note that our definition differs from the one in [17] in the way where the offline signcrypt stage does *not* require the receiver's identity as input. In our definition, the receiver's identity is only required in the online signcrypt stage.

2.4 Security of Signcryption

Definition 3 (Confidentiality). *An ID-based online/offline signcryption scheme is semantically secure against chosen ciphertext insider attack (SC-IND-CCA) if no PPT adversary has a non-negligible advantage in the following game:*

1. *The challenger runs Setup and gives the resulting param to adversary $\mathcal{A}$. It keeps msk secret.*
2. *In the first stage, $\mathcal{A}$ makes a number of queries to the following oracles which are simulated by the challenger:*
   (a) ***Extraction oracle:** $\mathcal{A}$ submits an identity ID to the extraction oracle for the result of $\mathsf{Extract}(msk, ID)$.*

*(b) **Signcryption oracle**: $\mathcal{A}$ submits a sender identity ID_s, a receiver identity ID_r and a message m to the signcryption oracle for the result of $\textsf{Online-Signcrypt}(m, \textsf{Offline-Signcrypt}(D_{ID_s}), ID_r)$.*

*(c) **De-signcryption oracle**: $\mathcal{A}$ submits a ciphertext ϕ and a receiver identity ID_r to the oracle for the result of $\textsf{De-Signcrypt}(\phi, D_{ID_r})$. The result is made of a message, a signature and the sender's identity if the de-signcryption is successful and the signature is valid under the recovered sender's identity. Otherwise, a symbol $\perp$ is returned for rejection.*

These queries can be asked adaptively. That is, each query may depend on the answers of previous ones.

3. *$\mathcal{A}$ produces two messages m_0, m_1, two identities ID_s^*, ID_r^* and a valid secret key $D_{ID_s^*}$ corresponding to ID_s^*. The challenger chooses a random bit $b \in \{0, 1\}$ and computes a signcryption ciphertext $\phi^* = \textsf{Online-Signcrypt}(m_b, \textsf{Offline-Signcrypt}(D_{ID_s^*}), ID_r^*)$. ϕ^* is sent to $\mathcal{A}$.*
4. *$\mathcal{A}$ makes a number of new queries as in the first stage with the restriction that it cannot query the de-signcryption oracle with (ϕ^*, ID_r^*) and the extraction oracle with ID_r^*.*
5. *At the end of the game, $\mathcal{A}$ outputs a bit b' and wins if $b' = b$.*

$\mathcal{A}$'s advantage is defined as $\textbf{\textit{Adv}}^{IND-CCA}(\mathcal{A}) = |\Pr[b' = b] - \frac{1}{2}|$.

Definition 4 (Unforgeability). *A signcryption scheme is existentially unforgeable against chosen-message insider attack (SC-EUF-CMA) if no PPT adversary has a non-negligible advantage in the following game:*

1. *The challenger runs $\textsf{Setup}$ and gives the resulting param to adversary $\mathcal{A}$. It keeps msk secret.*
2. *$\mathcal{A}$ makes a number of queries to the following oracles which are simulated by the challenger:*
   *(a) **Extraction oracle**: $\mathcal{A}$ submits an identity ID to the extraction oracle for the result of $\textsf{Extract}(msk, ID)$.*
   *(b) **Signcryption oracle**: $\mathcal{A}$ submits a sender identity ID_s, a receiver identity ID_r and a message m to the signcryption oracle for the result of $\textsf{Online-Signcrypt}(m, \textsf{Offline-Signcrypt}(D_{ID_s}), ID_r)$.*
   *(c) **De-signcryption oracle**: $\mathcal{A}$ submits a ciphertext ϕ and a receiver identity ID_r to the oracle for the result of $\textsf{De-Signcrypt}(\phi, D_{ID_r})$. The result is made of a message, a signature and the sender's identity if the de-signcryption is successful and the signature is valid under the recovered sender's identity. Otherwise, a symbol $\perp$ is returned for rejection.*

   These queries can be asked adaptively. That is, each query may depend on the answers of previous ones.
3. *$\mathcal{A}$ produces a signcryption ciphertext ϕ^* and an identity ID_r^*. $\mathcal{A}$ wins if*
   (a) $\textsf{De-Signcrypt}(\phi^*, D_{ID_r^*})$ returns a tuple (m^*, σ^*, ID_s^*) such that $\textsf{valid} \leftarrow \textsf{Verify}(m^*, \sigma^*, ID_s^*)$;
   (b) No output of the signcryption oracle decrypts to (m^*, σ^*, ID_s^*); and
   (c) No extraction query was made on ID_s^*.

$\mathcal{A}$'s advantage is defined as $\textbf{\textit{Adv}}^{EUF-CMA}(\mathcal{A}) = \Pr[\mathcal{A}\ wins\ ]$

Definition 5 (Ciphertext (sender) Anonymity). *A signcryption scheme is ciphertext (sender) anonymous against chosen-ciphertext insider attack (SC-ANON-CCA) if no PPT adversary has a non-negligible advantage in the following game:*

1. *The challenger runs $\textsf{Setup}$ and gives the resulting param to adversary $\mathcal{A}$. It keeps msk secret.*
2. *In the first stage, $\mathcal{A}$ adaptively makes a number of queries to the extraction oracle, signcryption oracle and de-signcryption oracle as in the confidentiality game. At the end of this stage, $\mathcal{A}$ outputs a message m, two sender identities $\{ID_{s,0}, ID_{s,1}\}$, two corresponding secret key $\{D_{ID_{s,0}}, D_{ID_{s,1}}\}$ and a receiver identity $\{ID_r\}$. $\mathcal{A}$ must not have made an extraction query on $\{ID_r\}$.*
3. *The challenger flips a coin $b \in \{0, 1\}$ and computes a challenge ciphertext $\phi^* = \textsf{Online-Signcrypt}$ $(m, \textsf{Offline-Signcrypt}(D_{ID_{s,b}}), ID_r)$ and sends ϕ^* to $\mathcal{A}$.*
4. *$\mathcal{A}$ adaptively makes a number of new queries as above with restriction as in the first stage and it is not allowed to ask for the de-signcryption query of ϕ^*.*
5. *At the end of the game, $\mathcal{A}$ outputs a bit b'' and wins the game if $b = b'$.*

$\mathcal{A}$'s advantage is defined as $\textbf{\textit{Adv}}^{ANON-CCA}(\mathcal{A}) = |\Pr[b = b'] - \frac{1}{2}|$.

3 The Proposed Online/Offline ID-Based Signcryption Scheme

3.1 Construction

Let $\mathbb{G}$ and $\mathbb{G}_T$ be groups of prime-order q, and let $e : \mathbb{G} \times \mathbb{G} \rightarrow \mathbb{G}_T$ 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 $g \in \mathbb{G}$ and randomly chooses $s \in_R \mathbb{Z}_q^*$. It sets $g_1 = g^s$ and $g_2 = g_1^s$. Define $\mathcal{M}$ to be the message space. Let $n_M = |\mathcal{M}|$. Also let n_d be the length of an identity, $H_1 : \{0,1\}^{n_d} \rightarrow \mathbb{Z}_q^*$, $H_2 : \{0,1\}^* \times \mathbb{G}_T \rightarrow \mathbb{Z}_q^*$ and $H_3 : \{0,1\}^* \rightarrow \{0,1\}^{n_M+n_d}$ be some cryptographic hash functions. The public parameters *param* and master secret key *msk* are given by

$$param = (\mathbb{G}, \mathbb{G}_T, q, g, g_1, g_2, \mathcal{M}, H_1, H_2, H_3) \qquad msk = s$$

Extract: To generate a secret key for a user with identity $ID \in \{0,1\}^{n_d}$, the PKG computes:

$$d_{ID} \leftarrow g^{\frac{1}{H_1(ID)+s}}$$

The user also computes and stores $\overline{G_{ID}} = e\big(g^{H_1(ID)} g_1, g\big)$ for future use.

Offline-Signcrypt: The user with identity $ID_s \in \{0,1\}^{n_d}$, with secret key d_{ID_s}, at the offline stage first randomly generates $u, x, \alpha, \beta, \gamma, \delta \in_R \mathbb{Z}_q^*$ and computes:

$$U \leftarrow d_{ID_s} g^{-u} \qquad R \leftarrow (\overline{G_{ID_s}})^x$$

$$T_0 \leftarrow \Big(g^{\alpha H_1(ID_s)} g_1^{H_1(ID_s)+\gamma} g_2\Big)^x$$

$$T_1 \leftarrow g^{x\beta^{-1}H_1(ID_s)} \qquad T_2 \leftarrow g_1^{x\delta^{-1}}$$

Outputs the offline ciphertext $\bar{\phi} = (U, R, x, u, T_0, T_1, T_2, \alpha, \beta, \gamma, \delta)$.

Online-Signcrypt: At the online stage, to encrypt a message $m \in \mathcal{M}$ to a user with identity $ID_r \in \{0,1\}^{n_d}$ computes:

$$t_1' \leftarrow \beta\Big(H_1(ID_r) - \alpha\Big) \bmod q \qquad t_2' \leftarrow \delta\Big(H_1(ID_r) - \gamma\Big) \bmod q$$

$$t \leftarrow h_2 x + u \bmod q \qquad c \leftarrow h_3 \oplus (m||ID_s)$$

where $h_2 = H_2(m, ID_s, R, T_0, T_1, T_2, t_1', t_2', U)$ and $h_3 = H_3(R, T_1, T_2, U)$. Outputs the ciphertext $\phi = (U, t, c, T_0, T_1, T_2, t_1', t_2')$.

De-Signcrypt: To de-signcrypt ϕ using secret key D_{ID_r}, computes

$$R \leftarrow e(T_0 T_1^{t_1'} T_2^{t_2'}, d_{ID_r}) \qquad (m||ID_s) \leftarrow c \oplus H_3(R, T_1, T_2, U)$$

and outputs (m, σ, ID_s) where $\sigma = \{R, t, U, T_0, T_1, T_2, t_1', t_2'\}$.

Verify: Computes $h_2 = H_2(m, ID_s, R, T_0, T_1, T_2, t_1', t_2', U)$ and checks whether

$$R^{h_2} \stackrel{?}{=} e\big(g^t U, g^{H_1(ID_s)} g_1\big) \cdot e(g,g)^{-1} \tag{1}$$

Outputs valid if it is equal. Otherwise outputs $\perp$.

We note that the term $e(g,g)^{-1}$ can be pre-computed or published as part of the public parameter by the PKG. Thus the number of pairing required in the whole de-signcryption process is just 2, while there is no pairing required in either offline signcrypt or online signcrypt stage.

3.2 Security Analysis

Theorem 1 (Confidentiality). *If there is a SC-IND-CCA adversary $\mathcal{A}$ of the proposed scheme in Section 3 that succeeds with probability ϵ, then there is a simulator $\mathcal{B}$ running in polynomial time that solves the $(\ell+1)$-BDHI problem with probability at least*

$$\epsilon \cdot \frac{1}{q_1}\Big(1 - q_s\frac{q_s+q_2}{q}\Big)\Big(1-\frac{q_d}{q}\Big)$$

where q_1, q_2, q_3, q_s, q_d are the number of queries allowed to the random oracle H_1, H_2, H_3, signcryption oracle and de-signcryption oracle respectively and we assume $q_1 = \ell$.

Proof. Setup: Suppose $\mathcal{B}$ is given a random instance of the $(\ell+1)$-BDHI problem $(g, g^{\alpha}, g^{\alpha^2}, \ldots, g^{\alpha^{\ell}}, g^{\alpha^{\ell+1}})$, $\mathcal{B}$ runs $\mathcal{A}$ as a subroutine to output $e(g,g)^{\frac{1}{\alpha}}$. $\mathcal{B}$ sets up a simulated environment for $\mathcal{A}$ as follow.

$\mathcal{B}$ first randomly selects $\pi \in_R \{1,\ldots,q_1\}$, $I_\pi \in_R \mathbb{Z}_q^*$ and $w_1, \ldots, w_{\pi-1}, w_{\pi+1}, \ldots, w_\ell \in_R \mathbb{Z}_q^*$. For $i \in \{1,\ldots,\ell\} \setminus \{\pi\}$, it computes $I_i = I_\pi - w_i$. Construct a polynomial with degree $\ell - 1$ as

$$f(z) = \prod_{i=1, i\neq\pi}^{\ell} (z + w_i)$$

to obtain $c_0, \ldots, c_{\ell-1} \in \mathbb{Z}_q^*$ such that $f(z) = \sum_{i=0}^{\ell-1} c_i z^i$. Then it sets generator $\hat{g} = g^{\sum_{i=0}^{\ell-1} c_i \alpha^i} = g^{f(\alpha)}$.

For $i \in \{1,\ldots,\ell\} \setminus \{\pi\}$, $\mathcal{B}$ expands $f_i(z) = \frac{f(z)}{(z+w_i)} = \sum_{j=0}^{\ell-2} d_{i,j} z^j$ to obtain $d_{i,1}, \ldots, d_{i,\ell-2} \in \mathbb{Z}_q^*$ and sets

$$\tilde{H}_i = g^{\sum_{j=0}^{\ell-2} d_{i,j}\alpha^j} = g^{f_i(\alpha)} = g^{\frac{f(\alpha)}{\alpha+w_i}} = \hat{g}^{\frac{1}{\alpha+w_i}}$$

It computes the public key g_1 and g_2 as

$$g_1 = \hat{g}^{-\alpha}\hat{g}^{-I_\pi} = \hat{g}^{-\alpha-I_\pi} \qquad g_2 = \hat{g}^{\alpha^2}\hat{g}^{2I_\pi\alpha}\hat{g}^{I_\pi^2} = \hat{g}^{(\alpha+I_\pi)^2}$$

where $\hat{g}^{\alpha} = g^{\sum_{i=0}^{\ell-1} c_i\alpha^{i+1}}$ and $\hat{g}^{\alpha^2} = g^{\sum_{i=0}^{\ell-1} c_i\alpha^{i+2}}$ so that its unknown private key is implicitly set to $x = -\alpha - I_\pi \in \mathbb{Z}_q^*$. For all $i \in \{1,\ldots,\ell\} \setminus \{\pi\}$, we have $(I_i, -\tilde{H}_i) = (I_i, \hat{g}^{\frac{1}{I_i+x}})$.

Oracle Simulation: $\mathcal{B}$ first initializes a counter ν to 1 and starts $\mathcal{A}$. Throughout the game, we assume that H_1-queries are distinct, that the target identity ID_r^* is submitted to H_1 at some point.

1. *Random Oracle:* For H_1-queries (we denote ID_ν the input of the ν^{th} one of such queries), $\mathcal{B}$ answers I_ν and increments ν.
   For H_2-queries on input $(m, ID_s, R, T_0, T_1, T_2, t_1', t_2', U)$ and H_3-queries on input (R, T_1, T_2, U), $\mathcal{B}$ returns the defined value if it exists and a randomly chosen $h_2 \in_R \mathbb{Z}_q^*$ for H_2 and $h_3 \in_R \{0,1\}^{n_M+n_d}$ for H_3 respectively, otherwise. $\mathcal{B}$ stores the information $\{h_2, (m, ID_s, R, T_0, T_1, T_2, t_1', t_2', U)\}$ in L_2 and $\{h_3, (R, T_1, T_2, U)\}$ in L_3.
2. *Extraction Oracle:* On input ID_ν, if $\nu = \pi$, $\mathcal{B}$ aborts. Otherwise, it knows that $H_1(ID_\nu) = I_\nu$ and returns $-\tilde{H}_\nu = \hat{g}^{1/(I_\nu+x)}$.
3. *Signcryption Oracle:* On input a plaintext m and identities $(ID_s, ID_r) = (ID_\mu, ID_\nu)$ for $\mu, \nu \in \{1,\ldots,q_1\}$, we observe that if $\mu \neq \pi$, $\mathcal{B}$ knows the sender's private key $d_{ID_\mu} = -\tilde{H}_\mu$ and can answer the query according to the specification of the algorithm. We thus assume $\mu = \pi$ and hence $\nu \neq \pi$. Also observe that $\mathcal{B}$ knows the receiver's private key $d_{ID_\nu} = -\tilde{H}_\nu$. The remaining task is to find a triple $(U, t, T_0, T_1, T_2, t_1', t_2', h)$ such that

$$e(T, d_{ID_\nu})^h = e(\hat{g}^t U, g_{ID_\pi}) \cdot e(\hat{g},\hat{g})^{-1}$$

   where $T = T_0 T_1^{t_1'} T_2^{t_2'}$, $g_{ID_\pi} = \hat{g}^{I_\pi} g_1$. To do so, $\mathcal{B}$ randomly generates $t, t', h, t_1', t_2', \tilde{t}_1, \tilde{t}_2 \in_R \mathbb{Z}_q^*$, computes

$$U = {d_{ID_\nu}}^{t'+t_1'\tilde{t}_1+t_2'\tilde{t}_2}\hat{g}^{-t} \qquad R = e(T, d_{ID_\nu})$$

$$T_0 = {g_{ID_\pi}}^{\frac{t'}{h}} {g_{ID_\nu}}^{-\frac{1}{h}} \qquad T_1 = {g_{ID_\pi}}^{\frac{\tilde{t}_1}{h}} \qquad T_2 = {g_{ID_\pi}}^{\frac{\tilde{t}_2}{h}}$$

where $g_{ID_\nu} = \hat{g}^{I_\nu} g_1$, and back patching the hash value $H_2(m, ID_\pi, R, T_0, T_1, T_2, t'_1, t'_2, U)$ to h. These values satisfy equation (1) as

$$\begin{aligned} R^h &= e(T_0 T_1^{t'_1} T_2^{t'_2}, d_{ID_\nu})^h \\ &= e({g_{ID_\pi}}^{t' + t'_1 \tilde{t}_1 + t'_2 \tilde{t}_2} {g_{ID_\nu}}^{-1}, d_{ID_\nu}) \\ &= e(g_{ID_\pi}, d_{ID_\nu})^{t' + t'_1 \tilde{t}_1 + t'_2 \tilde{t}_2} \cdot e(\hat{g}, \hat{g})^{-1} \\ &= e(U \hat{g}^t, g_{ID_\pi}) \cdot e(\hat{g}, \hat{g})^{-1} \end{aligned}$$

and they are valid ciphertext tuples as the distribution of the simulated ciphertexts is the same as the one in the real protocol. The ciphertext $\phi = (U, T_0, T_1, T_2, t'_1, t'_2, t,\ (m||ID_\pi) \oplus H_3(R, T_1, T_2, U))$ is returned.

4. *De-signcryption Oracle:* On input a ciphertext $\phi = (U, T_0, T_1, T_2, t'_1, t'_2, t, c)$ for identity (receiver) $ID_r = ID_\nu$, we assume that $\nu = \pi$ because otherwise $\mathcal{B}$ knows the receiver's private key $d_{ID_\nu} = -\tilde{H}_\nu$ and can normally run the decryption algorithm.
$\mathcal{B}$ extracts (t, T_1, T_2, U) from the ciphertext ϕ and searches through the list L_2 for entries of the form $\{h_{2,i}, (., ., ., T_0, T_1, T_2, t'_1, t'_2, U)\}$ indexed by $i \in \{1, \ldots, q_2\}$. If none is found, ϕ is rejected. Otherwise, each one of them is further examined: for the corresponding indexes, $\mathcal{B}$ extracts the values $(h_{2,i}, m_i, ID_i, R_i)$ from that row entry corresponding to $(T_0, T_1, T_2, t'_1, t'_2, U)$ and checks if
$$R_i^{h_{2,i}} = e(\hat{g}^t U, \hat{g}^{I_i} g_1) \cdot e(\hat{g}, \hat{g})^{-1} \tag{2}$$
meaning that equation (1) is satisfied. If the unique $i \in \{1, \ldots, q_2\}$ satisfying equation (2) is detected, the matching pair (m_i, σ_i, ID_i) is returned where $\sigma_i = (R_i, t, U, T_0, T_1, T_2, t'_1, t'_2)$. Otherwise ϕ is rejected.

Challenge: $\mathcal{A}$ outputs messages (m_0, m_1) and identities ID^* for which it never obtained ID^*'s private key. If $ID^* \neq ID_\pi$, $\mathcal{B}$ aborts. Otherwise it randomly selects $t, t'_1, t'_2, \tilde{t}_0, \tilde{t}_1, \tilde{t}_2 \in_R \mathbb{Z}_q^*$, $c \in_R \{0,1\}^{n_m}$ and $U \in_R \mathbb{G}$. Computes $T_0 = \hat{g}^{\tilde{t}_0}, T_1 = \hat{g}^{\tilde{t}_1}, T_2 = \hat{g}^{\tilde{t}_2}$ to return the challenge ciphertext $\phi^* = (U, t, T_0, T_1, T_2, t'_1, t'_2, c)$. Let $\xi = \tilde{t}_0 + t'_1 \tilde{t}_1 + t'_2 \tilde{t}_2$ and $T = \hat{g}^{-\xi}$. Since $x = -\alpha - I_\pi$, we let $\rho = \frac{\xi}{\alpha(I_\mu - \alpha - I_\pi)} = -\frac{\xi}{(I_\pi + x)(I_\mu + x)}$, we can check that

$$\begin{aligned} T = \hat{g}^{-\xi} &= \hat{g}^{-\alpha(I_\mu - \alpha - I_\pi)\rho} \\ &= \hat{g}^{(I_\pi + x)(I_\mu + x)\rho} \\ &= \hat{g}^{(I_\pi I_\mu + (I_\mu + I_\pi)x + x^2)\rho} \end{aligned}$$

$\mathcal{A}$ cannot recognize that ϕ^* is not a proper ciphertext unless it queries H_2 or H_3 on $e(\hat{g}^{I_\mu} g_1, \hat{g})^\rho$. Along the guess stage, its view is simulated as before and its output is ignored. Standard arguments can show that a successful $\mathcal{A}$ is very likely to query H_2 or H_3 on the input $e(g_{ID_\mu}, \hat{g})^\rho$ if the simulation is indistinguishable from a real attack environment.

Output Calculation: $\mathcal{B}$ fetches a random entry $(m, R, T_0, T_1, T_2, t'_1, t'_2, U, h_2)$ or $(R, T_1, T_2, U, \cdot)$ from the lists L_2 or L_3. With probability $1/(2q_2 + q_3)$, the chosen entry will contain the right element

$$R = e(g_{ID_\mu}, \hat{g})^\rho = e(\hat{g}, \hat{g})^{-\xi/(I_\pi + x)} = e(g, g)^{f(\alpha)^2 \xi/\alpha}$$

where $f(z) = \sum_{i=0}^{\ell-1} c_i z^i$ is the polynomial for which $\hat{g} = g^{f(\alpha)}$. The $(\ell+1)$-BDHI solution can be extracted by computing

$$\left(\frac{R^{1/\xi}}{e\left(g^{\sum_{i=0}^{\ell-2} c_{i+1} \alpha^i}, g^{c_0}\right) e\left(g^{\sum_{j=0}^{\ell-2} c_{j+1} \alpha^j}, \hat{g}\right)} \right)^{1/c_0^2}$$

$$
\begin{aligned}
&= \left(\frac{e(g,g)^{f(\alpha)^2/\alpha}}{e(g,g)^{c_0(c_1+c_2\alpha+c_3\alpha^2+\ldots c_{\ell-1}\alpha^{\ell-2})}e(g,g)^{f(\alpha)(c_1+c_2\alpha+c_3\alpha^2+\ldots c_{\ell-1}\alpha^{\ell-2})}} \right)^{1/c_0^2} \\
&= \left(\frac{e(g,g)^{f(\alpha)^2/\alpha}}{e(g,g)^{\frac{c_0(c_1\alpha+c_2\alpha^2+\ldots c_{\ell-1}\alpha^{\ell-1})+f(\alpha)(c_1\alpha+c_2\alpha^2+\ldots c_{\ell-1}\alpha^{\ell-1})}{\alpha}}} \right)^{1/c_0^2} \\
&= e(g,g)^{\frac{f(\alpha)^2-(c_1\alpha+c_2\alpha^2+\ldots c_{\ell-1}\alpha^{\ell-1})(c_0+f(\alpha))}{{c_0}^2\alpha}} \\
&= e(g,g)^{\frac{{c_0}^2}{{c_0}^2\alpha}} \\
&= e(g,g)^{1/\alpha}
\end{aligned}
$$

Probability Analysis: $\mathcal{B}$ only fails in providing a consistent simulation because one of the following independent events happen:

- E_1 : $\mathcal{A}$ does not choose to be challenged on ID_π.
- E_2 : A key extraction query is made on ID_π.
- E_3 : $\mathcal{B}$ aborts in a Signcryption query because of a collision on H_2.
- E_4 : $\mathcal{B}$ rejects a valid ciphertext at some point of the game.

We have $\Pr[\neg E_1] = 1/q_1$ and $\neg E_1$ implies $\neg E_2$. Also observe that $\Pr[E_3] \leq q_s(q_s + q_2)/q$ and $\Pr[E_4] \leq q_d/q$. Combining together, the overall successful probability $\Pr[\neg E_1 \wedge \neg E_3 \wedge \neg E_4]$ is at least

$$\frac{1}{q_1}\Big(1 - q_s\frac{q_s+q_2}{q}\Big)\Big(1-\frac{q_d}{q}\Big)$$

□

Theorem 2 (Unforgeability). *If there is an SC-UEF-CMA adversary $\mathcal{A}$ of the proposed scheme in Section 3 that succeeds with probability ϵ, then there is a simulator $\mathcal{B}$ running in polynomial time that solves the $\ell+1$-SDH problem with probability at least*

$$\epsilon^2 \cdot \frac{1}{q_1 q_2}\Big(1 - q_s\frac{q_s+q_2}{q}\Big)\Big(1-\frac{q_d}{q}\Big)$$

where q_1, q_2, q_3, q_s, q_d are the number of queries allowed to the random oracle H_1, H_2, H_3, signcryption oracle and de-signcryption oracle respectively and we assume $q_1 = \ell$.

Proof. Setup: Suppose $\mathcal{B}$ is given a random instance of the (ℓ+1)-SDH problem $(g, g^\alpha, g^{\alpha^2}, \ldots, g^{\alpha^\ell}, g^{\alpha^{\ell+1}})$, $\mathcal{B}$ runs $\mathcal{A}$ as a subroutine to output $(c, g^{\frac{1}{c+\alpha}})$. $\mathcal{B}$ sets up a simulated environment for $\mathcal{A}$ as follow.

$\mathcal{B}$ first randomly selects $\pi \in_R \{1, \ldots, q_1\}$, $I_\pi \in_R \mathbb{Z}_q^*$ and $w_1, \ldots,\ w_{\pi-1},\ w_{\pi+1}, \ldots, w_\ell \in_R \mathbb{Z}_q^*$. Construct a polynomial with degree $\ell - 1$ as

$$f(z) = \prod_{i=1, i\neq\pi}^{\ell} (z + w_i)$$

to obtain $c_0, \ldots, c_{\ell-1} \in \mathbb{Z}_q^*$ such that $f(z) = \sum_{i=0}^{\ell-1} c_i z^i$. Then it sets generator $\hat{g} = g^{\sum_{i=0}^{\ell-1} c_i\alpha^i} = g^{f(\alpha)}$.

For $i \in \{1, \ldots, \ell\}\backslash\{\pi\}$, $\mathcal{B}$ expands $f_i(z) = f(z)/(z+w_i) = \sum_{j=0}^{\ell-2} d_{i,j}z^j$ to obtain $d_{i,1}, \ldots, d_{i,\ell-2} \in \mathbb{Z}_q^*$ and sets

$$\tilde{H}_i = g^{\sum_{j=0}^{\ell-2} d_{i,j}\alpha^j} = g^{f_i(\alpha)} = g^{\frac{f(\alpha)}{\alpha+w_i}} = \hat{g}^{\frac{1}{\alpha+w_i}}$$

It computes the public key g_1 and g_2 as

$$g_1 = \hat{g}^\alpha \qquad g_2 = \hat{g}^{\alpha^2}$$

where $\hat{g}^{\alpha} = g^{\sum_{i=0}^{\ell-1} c_i \alpha^{i+1}}$ and $\hat{g}^{\alpha^2} = g^{\sum_{i=0}^{\ell-1} c_i \alpha^{i+2}}$ so that its unknown private key is implicitly set to $x = \alpha$. For all $i \in \{1, \ldots, \ell\} \setminus \{\pi\}$, we have $(I_i, \tilde{H}_i) = (w_i, \hat{g}^{\frac{1}{w_i+\alpha}})$.

Oracle Queries are answered in the same way as in Theorem 1.

Output Calculation: $\mathcal{A}$ has produced a forged ciphertext $(U, T_0, T_1, T_2, t_1', t_2', t_1, c)$ and a receiver identity ID_r with its secret key D_{ID_r}. $\mathcal{B}$ uses D_{ID_r} to decrypt and gets R, ID_s and m^*. If $ID_s \neq ID_\pi$, $\mathcal{B}$ aborts. We denote h_1 for the reply of the H_2 query on $(m^*, ID_\pi, R, T_0, T_1, T_2, t_1', t_2', U)$. $\mathcal{B}$ rewinded to the point just before making this particular query. This time $\mathcal{B}$ supplies to a different value $h_2 \neq h_1$ to this query. $\mathcal{A}$ produced another forged ciphertext $(U, \hat{T}_0, T_1, T_2, \hat{t_1'}, \hat{t_2'}, \hat{t}, \hat{c})$ based on h_2. Note that (U, T_1, T_2) are the same in both ciphertext as they are inputs to the H_2 query. R and m^* are also the same, as they are also one of the inputs to the H_2 query. By rewinding to the point just before making this particular query does not change the input values, but only the output values. If both forgeries satisfy equation (1), we obtain the relations

$$e(S_1, g_{ID_\pi})^{\frac{1}{h_1}} e(\hat{g}, \hat{g})^{-\frac{1}{h_1}} = e(S_2, g_{ID_\pi})^{\frac{1}{h_2}} e(\hat{g}, \hat{g})^{-\frac{1}{h_2}}$$

where $S_1 = \hat{g}^t U$, $S_2 = \hat{g}^{\hat{t}} U$ and $g_{ID_\pi} = \hat{g}^{H_1(ID_\pi)} g_1 = \hat{g}^{I_\pi + \alpha}$. Then, it comes that

$$e\Big(S_1{}^{\frac{h_2}{h_2-h_1}} S_2{}^{-\frac{h_1}{h_2-h_1}}, g_{ID_\pi}\Big) = e(\hat{g}, \hat{g})$$

Let $\tilde{T} = S_1{}^{\frac{h_2}{h_2-h_1}} S_2{}^{-\frac{h_1}{h_2-h_1}} = \hat{g}^{\frac{1}{I_\pi+\alpha}}$. From $\tilde{T}$, $\mathcal{B}$ can proceed as in [4] to extract $\sigma^* = g^{\frac{1}{I_\pi+\alpha}}$: it first obtains $\gamma_{-1}, \gamma_0, \ldots, \gamma_{q-2} \in \mathbb{Z}_q^*$ for which $f(z)/(z + I_\pi) = \frac{\gamma_{-1}}{z+I_\pi} + \sum_{i=1}^{\ell-2} \gamma_i z^*$ and computes

$$\sigma^* = \left(\tilde{T} g^{-\sum_{i=0}^{\ell-2} \gamma_i \alpha^i}\right)^{\frac{1}{\gamma_{-1}}} = g^{\frac{1}{I_\pi+\alpha}}$$

and returns the pair (I_π, σ^*) as a result.

Probability Analysis is similar to the one in Theorem 1. In addition, there is a rewind here, with successful probability ϵ/q_2. Combine together, the overall successful probability is at least

$$\epsilon^2 \cdot \frac{1}{q_1 q_2}\Big(1 - q_s \frac{q_s + q_2}{q}\Big)\Big(1 - \frac{q_d}{q}\Big)$$

□

Theorem 3 (Ciphertext (Sender) Anonymity). *If there is a SC-ANON-CCA adversary $\mathcal{A}$ of the proposed scheme in Section 3 that succeeds with probability ϵ, then there is a simulator $\mathcal{B}$ running in polynomial time that solves the $(\ell+1)$-BDHI problem with probability at least*

$$\epsilon \cdot \frac{1}{q_1}\Big(1 - q_s \frac{q_s + q_2}{q}\Big)\Big(1 - \frac{q_d}{q}\Big)$$

where q_1, q_2, q_3, q_s, q_d are the number of queries allowed to the random oracle H_1, H_2, H_3, signcryption oracle and de-signcryption oracle respectively and we assume $q_1 = \ell$.

The proof is similar to the proof of theorem 1. We omit here.

4 Performance Analysis

The performance of our scheme is comparable to previous non-identity based online/offline signcryption scheme, such as [1, 20]. Yet they need to fix the receiver's public key in the offline stage but we allow it to be known only in the online stage.

In terms of functionality, our scheme can be replaced by an online/offline identity-based encryption (OOIBE) (such as [11, 14]) plus an online/offline identity-based signature (OOIBS) (such as [10]) to obtain the same features. However, the efficiency of our scheme highly surpasses the combination of an OOIBE and OOIBS. The advantages are shown in the following table. In the comparison, we assume that $|\mathbb{G}| = 160$ bits, $|q| = 160$ bits, $|\mathbb{G}_T| = 1024$ bits and $|\mathcal{M}| = |q| = 160$ bits. We denote by E the point multiplication in $\mathbb{G}$ or $\mathbb{G}_T$, ME the multi-point multiplication in $\mathbb{G}$ or $\mathbb{G}_T$ (which costs about 1.3 times more than a single point multiplication), M the point addition in $\mathbb{G}$ or $\mathbb{G}_T$ and m_c the modular computation in $\mathbb{Z}_q$.

	GMC-1 + OOIBS	GMC-2 + OOIBS	LZ + OOIBS	Our scheme
Offline computation	$6E + 2ME$	$5E + 2ME$	$5E + 1ME$	$4E + 1ME$
Online computation	$1M + 3m_c$	$1M + 3m_c$	$4m_c$	$3m_c$
Offline storage (bits)	2944	5376	2944	2624
Ciphertext length (bits)	3104	7424	2080	1280
Number of pairing for decryption + verification	9	4	5	2
Security model	selective ID	standard	random oracle	random oracle

Table 1. Comparison of computation cost and size

Currently there are just 3 OOIBE schemes that allow the intended receiver's identity to be unknown in the offline stage. We use GMC-1 and GMC-2 to denote the first two in [11] and LZ to denote the one in [14]. For OOIBS, there is only one concrete scheme by Xu *et al.* [18]. However it was proven insecure by Li *et al.* [12] later. We use the generic construction by Galindo *et al.* [10]. The generic construction requires one public key based signature scheme and one online/offline signature scheme. The underlying signature schemes we use are from [6] (random oracle) and [4] (without random oracle) and the underlying online/offline signature scheme we use is from [5]. All these signature schemes are the most efficient one in the state of the art within their respective security model.

From the above table, we can see that our scheme achieves the least computation and the smallest size in both offline and online stage, when compare to the combinations of OOIBE and OOIBS.

5 Conclusion

In this paper, we re-defined the notion "online/offline ID-based signcryption" and provided a scheme that realizes it. Our construction is very efficient in a sense that it does not require any pairing operation in offline and online signcryption stages. Furthermore, we do *not* require the receiver's information (in our case, identity) in the offline signcryption stage. It is the first in the literature to remove such requirement. Without this restriction, our scheme is more flexible and practical. Our scheme is particularly suitable to provide authentication and confidentiality to power-constrained communication devices. We believe our proposed scheme may provide a practical solution in secure and authenticated transaction for smart cards or mobile devices such as smart phone.

References

1. J. H. An, Y. Dodis, and T. Rabin. On the Security of Joint Signature and Encryption. In *Proc. EUROCRYPT 2002*, volume 2332 of *Lecture Notes in Computer Science*, pages 83–107. Springer-Verlag, 2002.
2. P. Barreto, B. Libert, N. McCullagh, and J. Quisquater. Efficient and provabley-secure identity-based signature and signcryption from bilinear maps. In *AsiaCrypt 2005*, volume 3788 of *Lecture Notes in Computer Science*, pages 515–532. Springer-Verlag, 2005.

3. D. Boneh and X. Boyen. Efficient Selective-ID Secure Identity-Based Encryption Without Random Oracles. In *Proc. EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 223–238. Springer-Verlag, 2004.
4. D. Boneh and X. Boyen. Short Signatures Without Random Oracles. In *Proc. EUROCRYPT 2004*, volume 3027 of *Lecture Notes in Computer Science*, pages 56–73. Springer-Verlag, 2004.
5. D. Boneh and X. Boyen. Short signatures without random oracles and the sdh assumption in bilinear groups. *J. Cryptology*, 21(2):149–177, 2008.
6. D. Boneh, B. Lynn, and H. Shacham. Short Signatures from the Weil Pairing. In *Proc. ASIACRYPT 2001*, volume 2248 of *Lecture Notes in Computer Science*, pages 514–532. Springer-Verlag, 2001.
7. X. Boyen. Multipurpose Identity-Based Signcryption (A Swiss Army Knife for Identity-Based Cryptography). In *Proc. CRYPTO 2003*, volume 2729 of *Lecture Notes in Computer Science*, pages 383–399. Springer-Verlag, 2003.
8. L. Chen and J. Malone-Lee. Improved Identity-Based Signcryption. In *Proc. PKC 2005*, volume 3386 of *Lecture Notes in Computer Science*, pages 362–379. Springer-Verlag, 2005.
9. S. Even, O. Goldreich, and S. Micali. On-line/offline digital signatures. In *Proc. CRYPTO 89*, volume 2442 of *Lecture Notes in Computer Science*, pages 263–277. Springer-Verlag, 1989.
10. D. Galindo, J. Herranz, and E. Kiltz. On the generic construction of identity-based signatures with additional properties. In *ASIACRYPT*, volume 4284 of *Lecture Notes in Computer Science*, pages 178–193. Springer, 2006.
11. F. Guo, Y. Mu, and Z. Chen. Identity-based online/offline encryption. In *Financial Cryptography and Data Security '08*, volume 5143 of *Lecture Notes in Computer Science*, pages 247–261. Springer-Verlag, 2008.
12. F. Li, M. Shirase, and T. Takagi. On the security of online/offline signatures and multisignatures from acisp'06. In *CANS '08*, volume 5339 of *Lecture Notes in Computer Science*, pages 108–119. Springer-Verlag, 2008.
13. B. Libert and J.-J. Quisquater. New Identity Based Signcryption Schemes from Pairings. IEEE Information Theory Workshop 2003, pages 155-158, 2003.
14. J. K. Liu and J. Zhou. An efficient identity-based online/offline encryption scheme. In *ACNS*, volume 5536 of *Lecture Notes in Computer Science*, pages 156–167. Springer, 2009.
15. J. Malone-Lee. Identity-Based Signcryption. Cryptology ePrint Archive, Report 2002/098, 2002. http://eprint.iacr.org/.
16. A. Shamir. Identity-Based Cryptosystems and Signature Schemes. In *Proc. CRYPTO 84*, volume 196 of *Lecture Notes in Computer Science*, pages 47–53. Springer-Verlag, 1984.
17. D. Sun, Y. Mu, and W. Susilo. A generic construction of identity-based online/offline signcryption. In *ISPA*, pages 707–712. IEEE, 2008.
18. S. Xu, Y. Mu, and W. Susilo. Online/offline signatures and multisignatures for AVOD and DSR routing security. In *ACISP '06*, volume 4058 of *Lecture Notes in Computer Science*, pages 99–110. Springer-Verlag, 2006.
19. Z. Xu, G. Dai, and D. Yang. An efficient online/offline signcryption scheme for MANET. In *AINA Workshop '07*, pages 171–176. IEEE Computer Society, 2007.
20. F. Zhang, Y. Mu, and W. Susilo. Reducing security overhead for mobile networks. In *AINA Workshop '05*, pages 398–403. IEEE Computer Society, 2005.
21. Y. Zheng. Digital Signcryption or How to Achieve Cost(Signature & Encryption) << Cost(Signature) + Cost(Encryption). In *Proc. CRYPTO 97*, volume 1294 of *Lecture Notes in Computer Science*, pages 165–179. Springer-Verlag, 1997.