Identity-Based Hybrid Signcryption

Fagen Li[1,2,3], Masaaki Shirase[3], and Tsuyoshi Takagi[3]
[1]School of Computer Science and Engineering, University of Electronic Science and Technology of China, Chengdu 610054, China
[2]Key Laboratory of Computer Networks and Information Security, Xidian University, Xi'an 710071, China
[3]School of Systems Information Science, Future University-Hakodate, Hakodate 041-8655, Japan
Email: {fagenli,shirase,takagi}fun.ac.jp

***Abstract*—Signcryption is a cryptographic primitive that fulfills both the functions of digital signature and public key encryption simultaneously, at a cost significantly lower than that required by the traditional signature-then-encryption approach. In this paper, we address a question whether it is possible to construct a hybrid signcryption scheme in identity-based setting. This question seems to have never been addressed in the literature. We answer the question positively in this paper. In particular, we extend the concept of signcryption key encapsulation mechanism to the identity-based setting. We show that an identity-based signcryption scheme can be constructed by combining an identity-based signcryption key encapsulation mechanism with a data encapsulation mechanism. We also give an example of identity-based signcryption key encapsulation mechanism.**

I. Introduction

Identity-based (ID-based) cryptography was introduced by Shamir in 1984 [23]. The distinguishing property of ID-based cryptography is that a user's public key can be any binary string, such as an email address that can identify the user. This removes the need for senders to look up the recipient's public key before sending out an encrypted message. ID-based cryptography is supposed to provide a more convenient alternative to conventional public key infrastructure. Several practical ID-based signature schemes have been devised since 1984 [15], [16] but a satisfying ID-based encryption scheme only appeared in 2001 [7]. It was devised by Boneh and Franklin and cleverly uses bilinear maps (the Weil or Tate pairing) over supersingular elliptic curves.

Confidentiality, integrity, non-repudiation and authentication are the important requirements for many cryptographic applications. A traditional approach to achieve these requirements is to sign-then-encrypt the message. Signcryption, first proposed by Zheng [27], is a cryptographic primitive that fulfills both the functions of digital signature and public key encryption simultaneously, at a cost significantly lower than that required by the traditional signature-then-encryption approach. The original scheme in [27] is based on the discrete logarithm problem but no security proof is given. Zheng's original construction [27] was only proven secure by Baek, Steinfeld, and Zheng [3] who described a formal security model in a multi-user setting. A recent direction is to merge the concepts of ID-based cryptography and signcryption to design efficient ID-based signcryption schemes. Several ID-based signcryption schemes have been proposed so far, e.g. [4], [8], [9], [11], [20], [21].

The practical way to perform secrecy communication for large messages is to use hybrid encryption that separates the encryption into two parts: one part uses public key techniques to encrypt a one-time symmetric key; the other part uses the symmetric key to encrypt the actual message. In such a construction, the public key part of the algorithm is known as the key encapsulation mechanism (KEM) while the symmetric key part is known as the data encapsulation mechanism (DEM). A formal treatment of this paradigm originates in the work of Cramer and Shoup [12]. The resulting KEM-DEM hybrid encryption paradigm has received much attention in recent years [1], [18], [19]. It is very attractive as it gives a clear separation between the various parts of the cipher allowing for modular design. In [1], Abe, Gennaro, and Kurosawa introduced tag-KEM which takes as input a tag in KEM. Bentahar et al's [5] extended KEM into identity-based setting and proposed several efficient constructions of ID-based KEM (ID-KEM). Chen et al. [10] proposed an efficient ID-KEM based on the Sakai-Kasahara key construction [22]. Kiltz and Galindo [17] proposed a direct construction of ID-KEM in the standard model, based on Waters's ID-based encryption scheme [26].

The use of hybrid techniques to build signcryption schemes has been studied by Dent [13], [14]. He generalized KEM to signcryption KEM which includes an authentication in KEM. However, he only consider the insider security for authenticity. That is, if the sender's private key is exposed, an attacker is able to recover the key generated by signcryption KEM. The full insider security [2] means that (a) if the sender's private key is exposed, an attacker is still not able to recover the message from the ciphertext and (b) if the receiver's private key is exposed, an attacker is still not able to forge a ciphertext. In 2006, Bjørstad and Dent [6] showed how to built signcryption schemes using tag-KEM. However, they also only consider the insider security for authenticity and not for confidentiality. In 2008, Tan [25] proposed full insider secure signcryption KEM and tag-KEM without random oracles (in the standard model). Tan's schemes are insider secure for both authenticity and confidentiality.

All the above hybrid signcryption schemes [13], [14], [6], [25] is not ID-based. In this paper, we address a question whether it is possible to construct a hybrid signcryption scheme in ID-based setting. This question seems to have never been addressed in the literature. We answer the question

This is the full version of a paper published in the Fourth International Conference on Availability, Reliability and Security (ARES 2009), IEEE Computer Society, Fukuoka, Japan, 2009, pp. 534–539.

positively in this paper. In particular, we extend the concept of signcryption KEM to the ID-based setting. We show that an ID-based signcryption scheme can be constructed by combining an ID-based signcryption KEM (IDSC-KEM) with a DEM. We also give an example of ID-based signcryption KEM. Our schemes are insider secure for both authenticity and confidentiality.

The rest of this paper is organized as follows. We introduce the preliminary work in Section II. We give the formal model of ID-based signcryption KEM in Section III. We show how to construct an ID-based signcryption scheme using an ID-based signcryption KEM and a DEM in Section IV. An example of ID-based signcryption KEM is described in Section V. Finally, the conclusions are given in Section VI.

II. Preliminaries

A. ID-Based Signcryption (IDSC)

A generic ID-based signcryption scheme consists of the following four algorithms.

- $\texttt{Setup}$: is a probabilistic polynomial-time (PPT) algorithm run by a private key generator (PKG) that takes as input 1^k and outputs a master public key mpk and a master secret key msk. Here k is a security parameter.
- $\texttt{Extract}$: is a key generation algorithm run by the PKG that takes as input the master secret key msk and an identity $ID \in \{0,1\}^*$, and outputs the corresponding private key S_{ID}.
- $\texttt{Signcrypt}$: is a PPT algorithm that takes as input a plaintext message m, a receiver's identity ID_r, and a sender's private key S_{ID_S}, and outputs a ciphertext $\sigma \leftarrow \texttt{Signcrypt}(m, S_{ID_s}, ID_r)$.
- $\texttt{Unsigncrypt}$: is a deterministic algorithm that takes as input a ciphertext σ, the receiver's private key S_{ID_r} and the sender's identity ID_s, and outputs the original message m or the symbol $\perp$ if σ is an invalid ciphertext between identities ID_s and ID_r.

We make the consistency constraint that if

$$\sigma \leftarrow \texttt{Signcrypt}(m, S_{ID_s}, ID_r),$$

then

$$m \leftarrow \texttt{Unsigncrypt}(\sigma, S_{ID_r}, ID_s).$$

Malone-Lee [21] defines the security notions for ID-based signcryption schemes. These notions are semantic security (i.e. indistinguishability against adaptive chosen ciphertext attacks (IND-CCA2) and existential unforgeability against adaptive chosen messages attacks (UF-CMA)). For the stronger notion of insider security, we use the notion of strong existential unforgeability (sUF-CMA). The strong existential unforgeability means that an adversary wins if it outputs a valid message/signcryption pair (m, σ) for identities ID_s and ID_r and the signcryption σ was not returned by the signcryption oracle when queried on the message m.

For the confidentiality, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

- $\texttt{Initial}$: The challenger $\mathcal{C}$ runs $(mpk, msk) \leftarrow \texttt{Setup}(1^k)$ and runs $\mathcal{A}$ on input $(1^k, mpk)$.
- $\texttt{Phase1}$: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries).
  - Key extraction queries: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $S_{ID} \leftarrow \texttt{Extract}(ID)$ and sends S_{ID} to $\mathcal{A}$.
  - Signcryption queries: $\mathcal{A}$ produces a sender's identity ID_s, a receiver's identity ID_r and a plaintext m. $\mathcal{C}$ computes the private key $S_{ID_s} \leftarrow \texttt{Extract}(ID_s)$ and $\sigma \leftarrow \texttt{Signcrypt}(m, S_{ID_s}, ID_r)$ and sends σ to $\mathcal{A}$.
  - Unsigncryption queries: $\mathcal{A}$ chooses a sender's identity ID_s, a receiver's identity ID_r, and a ciphertext σ. $\mathcal{C}$ generates the private key $S_{ID_r} \leftarrow \texttt{Extract}(ID_r)$ and sends the result of $\texttt{Unsigncrypt}(\sigma, S_{ID_r}, ID_s)$ to $\mathcal{A}$.
- $\texttt{Challenge}$: The adversary $\mathcal{A}$ decides when Phase 1 ends. $\mathcal{A}$ generates two equal length plaintexts m_0, m_1, a sender's identity ID_s^*, and a receiver's identity ID_r^* on which it wishes to be challenged. The identity ID_r^* should not appear in any key extraction queries in Phase 1. The challenger $\mathcal{C}$ picks a random bit δ from $\{0,1\}$, computes $\sigma^* \leftarrow \texttt{Signcrypt}(m_\delta, S_{ID_s^*}, ID_r^*)$, and returns σ to $\mathcal{A}$.
- $\texttt{Phase2}$: The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the restriction that it cannot make a key extraction query on ID_r^* and cannot make an unsigncryption query on σ^* to obtain the corresponding plaintext.
- $\texttt{Guess}$: The adversary $\mathcal{A}$ produces a bit δ' and wins the game if $\delta' = \delta$.

  The advantage of $\mathcal{A}$ is defined to be

$$Adv_{\text{IDSC}}^{\text{IND-CCA2}}(\mathcal{A}) = |2\Pr[\delta' = \delta] - 1|,$$

where $\Pr[\delta' = \delta]$ denotes the probability that $\delta' = \delta$.

Definition 1: An ID-based signcryption scheme is considered to be IND-CCA2 secure, if for all PPT adversaries $\mathcal{A}$, the advantage in the IND-CCA2 game is a negligible function of the security parameter k.

Notice that the adversary is allowed to make a key extraction query on identity ID_s^* in the above definition. This condition corresponds to the stringent requirement of insider security for confidentiality of signcryption [2]. On the other hand, it ensures the forward security of the scheme, i.e. confidentiality is preserved in case the sender's private key becomes compromised.

For the strong existential unforgeability, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{F}$.

- $\texttt{Initial}$: The challenger $\mathcal{C}$ runs $(mpk, msk) \leftarrow \texttt{Setup}(1^k)$ and runs $\mathcal{F}$ on input $(1^k, mpk)$.
- $\texttt{Attack}$: The adversary $\mathcal{F}$ performs a polynomially bounded number of queries just like in the confidentiality game.
- $\texttt{Forgery}$: $\mathcal{F}$ produces a quaternion $(m^*, \sigma^*, ID_s^*, ID_r^*)$, where the private key of ID_s^* was not asked and σ^*

was not returned by the signcryption oracle on the input (m^*, ID_s^*, ID_r^*) during $\texttt{Attack}$ stage. $\mathcal{F}$ wins the game if the result of $\texttt{Unsigncrypt}(\sigma^*, S_{ID_r^*}, ID_s^*)$ is not the $\perp$ symbol.

The advantage of $\mathcal{F}$ is defined as the probability that it wins.

Definition 2: An ID-based signcryption scheme is considered to be sUF-CMA secure, if for all PPT adversaries $\mathcal{F}$, the advantage in the sUF-CMA game is a negligible function of the security parameter k.

Note that the adversary is allowed to make a key extraction query on the identity ID_r^* in the above definition. Again, this condition corresponds to the stringent requirement of insider security for authenticity of signcryption [2].

B. Date Encapsulation Mechanism (DEM)

A DEM is a symmetric encryption scheme which consists of the following two algorithms.

- $\texttt{Enc}$: is a deterministic encryption algorithm which takes as input 1^k, a key K and a message $m \in \{0,1\}^*$, and outputs a ciphertext $c \in \{0,1\}^*$, where $K \in \mathcal{K}_{\text{DEM}}$ is a key in the given key space, and m is a bit string of arbitrary length. We denote this as $c \leftarrow \texttt{Enc}(K, m)$.
- $\texttt{Dec}$: is a deterministic decryption algorithm which takes as input a key K and a ciphertext c, and outputs the message $m \in \{0,1\}^*$ or a symbol $\perp$ to indicate that the ciphertext is invalid.

For the purposes of this paper, it is only required that a DEM is secure with respect to indistinguishability against passive attackers (IND-PA). Formally, this security notion is captured by the following game played between a PPT adversary $\mathcal{A}$ and a challenger $\mathcal{C}$.

- $\texttt{Initial}$: $\mathcal{A}$ runs on input 1^k and submits two equal length messages, m_0 and m_1.
- $\texttt{Challenge}$: $\mathcal{C}$ chooses a random key $K \in \mathcal{K}_{\text{DEM}}$ as well as a random bit $\lambda \in \{0,1\}$, and sends $c^* \leftarrow \texttt{Enc}(K, m_\lambda)$ to $\mathcal{A}$ as a challenge ciphertext.
- $\texttt{Guess}$: The adversary $\mathcal{A}$ produces a bit λ' and wins the game if $\lambda' = \lambda$.

The advantage of $\mathcal{A}$ is defined to be

$$Adv_{\text{DEM}}^{\text{IND-PA}}(\mathcal{A}) = |2\Pr[\lambda' = \lambda] - 1|,$$

where $\Pr[\lambda' = \lambda]$ denotes the probability that $\lambda' = \lambda$.

Definition 3: A DEM is considered to be IND-PA secure, if for all PPT adversaries $\mathcal{A}$, the advantage in the above game is a negligible function of the security parameter k.

III. ID-Based Signcryption KEM

In this section, we give the formal definition for ID-based signcryption KEM.

A. Generic Scheme

A generic ID-based signcryption KEM consists of the following four algorithms.

- $\texttt{Setup}$: is a PPT algorithm which takes as input 1^k and outputs the master public key mpk and the master secret key msk. Here k is a security parameter.
- $\texttt{Extract}$: is a key generation algorithm which takes as input msk and an identity $ID \in \{0,1\}^*$, and outputs the corresponding private key S_{ID}.
- $\texttt{Encap}$: is a PPT key encapsulation algorithm which takes as input a plaintext message m, a receiver's identity ID_r, and a sender's private key S_{ID_s}, and outputs an encapsulation key pair (K, ψ), where $K \in \mathcal{K}_{\text{IDSC-KEM}}$ is a key in the space of possible session keys at a given security level, and $\psi \in \mathcal{E}_{\text{IDSC-KEM}}$ is the encapsulation of that key. We denote this as $(K, \psi) \leftarrow \texttt{Encap}(m, S_{ID_s}, ID_r)$.
- $\texttt{Decap}$: is a deterministic key decapsulation algorithm which takes as input the sender's identity ID_s, the receiver's private key S_{ID_r} and the encapsulation of that key ψ, and outputs the corresponding key K or the error symbol $\perp$. We denote this as $K \leftarrow \texttt{Decap}(\psi, S_{ID_r}, ID_s)$.
- $\texttt{Verify}$: is a deterministic verification algorithm which takes as input a sender's identity ID_s, a receiver's private key S_{ID_r}, a message m, and an encapsulation ψ, and outputs $\top$ for "true" or $\perp$ for "false". Note that the verification algorithm does not need to take the symmetric key K as input as it can be easily computed from the encapsulation ψ using the deterministic decapsulation algorithm.

  We make the consistency constraint that if

$$(K, \psi) \leftarrow \texttt{Encap}(m, S_{ID_s}, ID_r),$$

then

$$K \leftarrow \texttt{Decap}(\psi, S_{ID_r}, ID_s) \text{and} \top \leftarrow \texttt{Verify}(\psi, m, S_{ID_r}, ID_s).$$

B. Security Notions

An ID-based signcryption KEM should satisfy confidentiality and unforgeability. For the confidentiality, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{A}$.

- $\texttt{Initial}$: The challenger $\mathcal{C}$ runs $(mpk, msk) \leftarrow \texttt{Setup}(1^k)$ and runs $\mathcal{A}$ on input $(1^k, mpk)$.
- $\texttt{Phase1}$: The adversary $\mathcal{A}$ can perform a polynomially bounded number of queries in an adaptive manner (that is, each query may depend on the responses to the previous queries).
  - Key extraction queries: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $S_{ID} \leftarrow \texttt{Extract}(ID)$ and sends S_{ID} to $\mathcal{A}$.
  - Key encapsulation queries: $\mathcal{A}$ produces a sender's identity ID_s, a receiver's identity ID_r and a plaintext m. $\mathcal{C}$ computes the private key $S_{ID_s} \leftarrow \texttt{Extract}(ID_s)$ and $(K, \psi) \leftarrow \texttt{Encap}(m, S_{ID_s}, ID_r)$ and sends (K, ψ) to $\mathcal{A}$.
  - Key decapsulation queries: $\mathcal{A}$ chooses a sender's identity ID_s, a receiver's identity ID_r, and an encapsulation ψ. $\mathcal{C}$ generates the private key $S_{ID_r} \leftarrow \texttt{Extract}(ID_r)$ and sends the result of $\texttt{Decap}(\psi, S_{ID_r}, ID_s)$ to $\mathcal{A}$.
  - Verification queries: $\mathcal{A}$ chooses a sender's identity ID_s, a receiver's identity ID_r, a message m, and an encapsulation ψ. $\mathcal{C}$ generates the private key $S_{ID_r} \leftarrow \texttt{Extract}(ID_r)$ and sends the result of $\texttt{Verify}(\psi, m, S_{ID_r}, ID_s)$ to $\mathcal{A}$.

- `Challenge` : The adversary $\mathcal{A}$ decides when Phase 1 ends. $\mathcal{A}$ generates a message m^*, a sender's identity ID_s^*, and a receiver's identity ID_r^* on which it wishes to be challenged. The identity ID_r^* should not appear in any key extraction queries in Phase 1. $\mathcal{C}$ then runs $(K_1, \psi^*) \leftarrow \texttt{Encap}(m, S_{ID_s^*}, ID_r^*)$ and randomly chooses $K_0 \leftarrow \mathcal{K}_{\mathrm{IDSC-KEM}}$. $\mathcal{C}$ also chooses a random bit $b \in \{0,1\}$ and sends (K_b, ψ^*) to $\mathcal{A}$ as a challenge encapsulation key pair.
- `Phase2` : The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the restriction that it cannot make a key extraction query on ID_r^* and cannot make a decapsulation query on (K_b, ψ^*) to obtain the corresponding key.
- `Guess` : The adversary $\mathcal{A}$ produces a bit b' and wins the game if $b' = b$.

  The advantage of $\mathcal{A}$ is defined to be

$$Adv_{\mathrm{IDSC-KEM}}^{\mathrm{IND-CCA2}}(\mathcal{A}) = |2\Pr[b' = b] - 1|,$$

where $\Pr[b' = b]$ denotes the probability that $b' = b$.

Definition 4: An ID-based signcryption KEM is considered to be IND-CCA2 secure, if for all PPT adversaries $\mathcal{A}$, the advantage in the IND-CCA2 game is a negligible function of the security parameter k.

For the unforgeability, we consider the following game played between a challenger $\mathcal{C}$ and an adversary $\mathcal{F}$.

- `Initial` : The challenger $\mathcal{C}$ runs $(mpk, msk) \leftarrow \texttt{Setup}(1^k)$ and runs $\mathcal{F}$ on input $(1^k, mpk)$.
- `Attack` : The adversary $\mathcal{F}$ performs a polynomially bounded number of queries just like in the confidentiality game.
- `Forgery` : $\mathcal{F}$ produces a quaternion $(m^*, \psi^*, ID_s^*, ID_r^*)$, where the private key of ID_s^* was not asked and ψ^* was not returned by the key encapsulation oracle on the input (m^*, ID_s^*, ID_r^*) during `Attack` stage. $\mathcal{F}$ wins the game if the result of $\texttt{Verify}(\psi^*, m^*, S_{ID_r^*}, ID_s^*)$ is not the $\perp$ symbol.

The advantage of $\mathcal{F}$ is defined as the probability that it wins.

Definition 5: An ID-based signcryption KEM is considered to be sUF-CMA secure, if for all PPT adversaries $\mathcal{F}$, the advantage in the sUF-CMA game is a negligible function of the security parameter k.

IV. Identity-Based Hybrid Signcryption

We can combine an ID-based signcryption KEM with a DEM to form an ID-based hybrid signcryption scheme. We describe it in Figure 1.

We give the security results for ID-based hybrid signcryption in Theorems 1 and 2

Theorem 1: Let IDSC be an ID-based hybrid signcryption scheme constructed from an ID-based signcryption KEM and a DEM. If the ID-based signcryption KEM is IND-CCA2 secure and the DEM is IND-PA secure, then IDSC is IND-CCA2 secure. In particular, we have

$$Adv_{\mathrm{IDSC}}^{\mathrm{IND-CCA2}}(\mathcal{A}) \leq 2Adv_{\mathrm{IDSC-KEM}}^{\mathrm{IND-CCA2}}(\mathcal{B}_1) + Adv_{\mathrm{DEM}}^{\mathrm{IND-PA}}(\mathcal{B}_2).$$

`Setup` : On input 1^k:

1. $(mpk, msk) \leftarrow \texttt{IDSC}-\texttt{KEM.Setup}(1^k)$

2. Output the master public key mpk and the master secret key msk

`Extract` : On input the master secret key msk and an identity $ID \in \{0,1\}^*$:

1. $S_{ID} \leftarrow \texttt{IDSC}-\texttt{KEM.Extract}(msk, ID)$

2. Output the private key S_{ID} of the identity ID

`Signcrypt` : On input the sender's private key S_{ID_s}, the receiver's identity ID_r, and a message $m \in \{0,1\}^*$:

1. $(K, \psi) \leftarrow \texttt{IDSC}-\texttt{KEM.Encap}(m, S_{ID_s}, ID_r)$

2. $c \leftarrow \texttt{DEM.Enc}(K, m)$

3. Output the ciphertext $\sigma \leftarrow (\psi, c)$

`Unsigncrypt` : On input the sender's identity ID_s, the receiver private key S_{ID_r} and the ciphertext σ:

1. $K \leftarrow \texttt{IDSC}-\texttt{KEM.Decap}(\psi, S_{ID_r}, ID_s)$

2. If $K = \perp$, then output $\perp$ and stop

3. $m \leftarrow \texttt{DEM.Dec}(K, c)$

4. If $\top \leftarrow \texttt{IDSC}-\texttt{KEM.Verify}(\psi, m, S_{ID_r}, ID_s)$, output m. Otherwise output $\perp$.

Fig. 1. Identity-based hybrid signcryption

Proof: See the appendix A. □

Theorem 2: Let IDSC be an ID-based hybrid signcryption scheme constructed from an ID-based signcryption KEM and a DEM. If the ID-based signcryption KEM is sUF-CMA secure, then IDSC is sUF-CMA secure. In particular, we have

$$Adv_{\mathrm{IDSC}}^{\mathrm{sUF-CMA}}(\mathcal{F}) \leq Adv_{\mathrm{IDSC-KEM}}^{\mathrm{sUF-CMA}}(\mathcal{B}),$$

where $Adv_{\mathrm{IDSC}}^{\mathrm{sUF-CMA}}(\mathcal{F})$ is the advantage of the sUF-CMA adversary against IDSC, and $Adv_{\mathrm{IDSC-KEM}}^{\mathrm{sUF-CMA}}(\mathcal{B})$ is the advantage of the resulting sUF-CMA adversary against ID-based signcryption KEM.

Proof: See the appendix B. □

V. An Example of ID-Based Signcryption KEM

Most of ID-based signcryption schemes [4], [8], [9], [11], [20], [21] fit the new generic framework. Here we give an example of ID-based signcryption KEM based on Barreto et al.'s scheme [4]. Barreto et al.'s scheme is the fastest ID-based signcryption scheme so far. If we combine the ID-based signcryption KEM with a DEM as Figure 1, we can get a scheme that is very similar to Barreto et al.'s original scheme.

Since Barreto et al.'s scheme uses the bilinear pairings, we first describe the basic definition and properties of the bilinear pairings.

A. Bilinear Pairings

Let G_1, G_2 and G_T be three cyclic groups of prime order q. Let P, Q be generators of G_1 and G_2, respectively. A bilinear pairing is a map $\hat{e} : G_1 \times G_2 \rightarrow G_T$ with the following properties:

1) Bilinearity: $\forall (S,T) \in G_1 \times G_2, \forall a,b \in Z_q, \hat{e}(aS,bT) = \hat{e}(S,T)^{ab}$.
2) Non-degeneracy: $\forall S \in G_1$, $\hat{e}(S,T) = 1$ for all $T \in G_2$ iff $S = \mathcal{O}$.
3) Computability: $\forall (S,T) \in G_1 \times G_2$, $\hat{e}(S,T)$ is efficiently computable.
4) There exists an efficient, publicly computable (but not necessarily invertible) isomorphism $\varphi : G_2 \rightarrow G_1$ such that $\varphi(Q) = P$.

The security of Barreto et al.'s scheme relies on the hardness of the following problems.

Definition 6: Define G_1, G_2, G_T and $\hat{e}$ as in this section. The l-Strong Diffie-Hellman problem (l-SDHP) in the groups (G_1, G_2) is to find a pair $(c, \frac{1}{c+\alpha}P)$ with $c \in Z_q^*$ given a $(l+2)$-tuple $(P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^l Q)$.

Definition 7: Define G_1, G_2, G_T and $\hat{e}$ as in this section. The l-Bilinear Diffie-Hellman Inversion problem (l-BDHIP) in the groups (G_1, G_2, G_T) is to compute $\hat{e}(P,Q)^{1/\alpha} \in G_T$ given $(P, Q, \alpha Q, \alpha^2 Q, \ldots, \alpha^l Q)$.

B. ID-Based Signcryption KEM

- `Setup` : Define G_1, G_2, G_T and $\hat{e}$ as in previous subsection. Let H_1, H_2 and H_3 be three cryptographic hash functions where $H_1 : \{0,1\}^* \rightarrow Z_q^*$, $H_2 : \{0,1\}^* \times G_T \rightarrow Z_q^*$ and $H_3 : G_T \rightarrow \{0,1\}^n$. Here n is the key length of a DEM. Let $Q \in G_2$, $P = \varphi(Q) \in G_1$ be generators of G_2 and G_1, respectively and $g = \hat{e}(P,Q) \in G_T$. The PKG chooses a master secret key $s \in Z_q^*$ randomly and computes $Q_{pub} \leftarrow sQ \in G_2$. The PKG publishes system parameters $\{G_1, G_2, G_T, P, Q, g, Q_{pub}, \hat{e}, \varphi, H_1, H_2, H_3\}$ and keeps the master key s secret.
- `Extract` : Given an identity ID, the PKG computes the private key $S_{ID} \leftarrow \frac{1}{H_1(ID)+s}Q \in G_2$. Then PKG sends the private key to its owner in a secure way.
- `Encap` : Given a message m, a receiver's identity ID_r and a sender's private key S_{ID_s}, this algorithm works as follows.
  1) Choose $x \in Z_q^*$ randomly and compute $r \leftarrow g^x$.
  2) Compute $K \leftarrow H_3(r)$.
  3) Compute $h \leftarrow H_2(m, r)$.
  4) Compute $S \leftarrow (x+h)\varphi(S_{ID_s})$.
  5) Compute $T \leftarrow x(H_1(ID_r)P + \varphi(Q_{pub}))$.
  6) Set $\psi \leftarrow (S, T)$.
  7) Output (K, ψ).
- `Decap` : Given the sender's identity ID_s, the receiver's private key S_{ID_r}, and an encapsulation ψ, this algorithm works as follows.
  1) Compute $r \leftarrow \hat{e}(T, S_{ID_r})$.
  2) Compute $K \leftarrow H_3(r)$.
  3) Output K.
- `Verify` : Given the sender's identity ID_s, the receiver's private key S_{ID_r}, a message m, and an encapsulation ψ, this algorithm works as follows.
  1) Compute $r \leftarrow \hat{e}(T, S_{ID_r})$.
  2) Compute $h \leftarrow H_2(m, r)$.
  3) If $r = \hat{e}(S, H_1(ID_s)Q + Q_{pub})g^{-h}$, output symbol $\top$. Otherwise, output symbol $\perp$.

In a real implementation of this algorithm, we can store the value of r computed by the decapsulation algorithm and use it again in the verification algorithm. Such an implementation would be functionally identical to the above algorithm and would therefore be just as secure. We choose to separate the decapsulation and verification algorithms so that they can be studied independently. The security proof is similar to that of [4]. We omit it.

VI. Conclusions

In this paper, we extended the concept of signcryption KEM to the identity-based setting. We showed that an ID-based signcryption scheme can be constructed by combining an ID-based signcryption KEM with a DEM. To show that our framework is reasonable, we also gave an example of ID-based signcryption KEM based on Barreto et al.'s ID-based signcryption scheme.

References

[1] M. Abe, R. Gennaro, and K. Kurosawa. Tag-KEM/DEM: a new framework for hybrid encryption. *Journal of Cryptology*, Vol. 21, No. 1, pp. 97–130, 2008.

[2] J.H. An, Y. Dodis, T. Rabin. On the security of joint signature and encryption. In *Advances in Cryptology-EUROCRYPT 2002*, LNCS 2332, pp. 83–107, Springer-Verlag, 2002.

[3] J. Baek, R. Steinfeld, and Y. Zheng. Formal proofs for the security of signcryption. *Journal of Cryptology*, Vol. 20, No 2, pp. 203–235, 2007.

[4] P.S.L.M. Barreto, B. Libert, N. McCullagh, and J.J. Quisquater. Efficient and provably-secure identity-based signatures and signcryption from bilinear maps. In *Advances in Cryptology-ASIACRYPT 2005*, LNCS 3788, pp. 515–532, Springer-Verlag, 2005.

[5] K. Bentahar, P. Farshim, J. Malone-Lee, and N.P. Smart. Generic constructions of identity-based and certificateless KEMs. *Journal of Cryptology*, Vol. 21, No 2, pp. 178–199, 2008.

[6] T.E. Bjørstad and A.W. Dent. Building better signcryption schemes with tag-KEMs. In *Public Key Cryptography-PKC 2006*, LNCS 3958, pp. 491–507, Springer-Verlag, 2006.

[7] D. Boneh and M. Franklin. Identity-based encryption from the weil pairing. In *Advances in Cryptology-CRYPTO 2001*, LNCS 2139, pp. 213–229, Springer-Verlag, 2001.

[8] X. Boyen. Multipurpose identity-based signcryption: a swiss army knife for identity-based cryptography. In *Advances in Cryptology-CRYPTO 2003*, LNCS 2729, pp. 383–399, Springer-Verlag, 2003.

[9] L. Chen and J. Malone-Lee. Improved identity-based signcryption. In *Public Key Cryptography-PKC 2005*, LNCS 3386, pp. 362–379, Springer-Verlag, 2005.

[10] L. Chen, Z. Cheng, J. Malone-Lee, N.P. Smart. Efficient ID-KEM based on the Sakai-Kasahara key construction. *IEE Proceedings-Information Security*, Vol. 153, No 1, pp. 19–26, 2006.

[11] S.S.M. Chow, S.M. Yiu, L.C.K. Hui, and K.P. Chow. Efficient forward and provably secure ID-based signcryption scheme with public verifiability and public ciphertext authenticity. In *Information Security and Cryptology-ICISC 2003*, LNCS 2971, pp. 352–369, Springer-Verlag, 2004.

[12] R. Cramer and V. Shoup. Design and analysis of practical public-key encryption schemes secure against adaptive chosen ciphertext attack. *SIAM Journal on Computing*, Vol. 33, No. 1, pp. 167–226, 2003.
[13] A.W. Dent. Hybrid signcryption schemes with outsider security. In *Information Security-ISC 2005*, LNCS 3650, pp. 203–217, Springer-Verlag, 2005.
[14] A.W. Dent. Hybrid signcryption schemes with insider security. In *Information Security and Privacy-ACISP 2005*, LNCS 3574, pp. 253–266, Springer-Verlag, 2005.
[15] A. Fiat and A. Shamir. How to prove yourself: practical solutions to identification and signature problems. In *Advances in Cryptology-CRYPTO'86*, LNCS 263, pp. 186–194, Springer-Verlag, 1986.
[16] L. Guillou and J.J. Quisquater. A "Paradoxical" Identity-based signature scheme resulting from zero-knowledge. In *Advances in Cryptology-CRYPTO'88*, LNCS 403, pp. 216–231, Springer-Verlag, 1988.
[17] E. Kiltz and D. Galindo. Direct chosen-ciphertext secure identity-based key encapsulation without random oracles. In *Information Security and Privacy-ACISP 2006*, LNCS 4058, pp. 336–347, Springer-Verlag, 2006.
[18] E. Kiltz. Chosen-ciphertext secure key-encapsulation based on gap hashed diffie-hellman. In *Public Key Cryptography-PKC 2007*, LNCS 4450, pp. 282–297, Springer-Verlag, 2007.
[19] K. Kurosawa and Y. Desmedt. A new paradigm of hybrid encryption scheme. In *Advances in Cryptology-CRYPTO 2004*, LNCS 3152, pp. 426–442, Springer-Verlag, 2004.
[20] B. Libert and J.J. Quisquater. A new identity based signcryption schemes from pairings. In *2003 IEEE Information Theory Workshop*, pp. 155–158, Paris, France, 2003.
[21] J. Malone-Lee. Identity based signcryption. *Cryptology ePrint Archive*, Report 2002/098, 2002.
[22] R. Sakai and M. Kasahara. ID based cryptosystems with pairing on elliptic curve. *Cryptology ePrint Archive*, Report 2003/054, 2003.
[23] A. Shamir. Identity-based cryptosystems and signature schemes. In *Advances in Cryptology-CRYPTO'84*, LNCS 196, pp. 47–53, Springer-Verlag, 1984.
[24] V. Shoup. OAEP reconsidered. In *Advances in Cryptology-CRYPTO 2001*, LNCS 2139, pp. 239–259, Springer-Verlag, 2001.
[25] C.H. Tan. Insider-secure signcryption KEM/tag-KEM schemes without random oracles. In *The Third International Conference on Availability, Reliability and Security-ARES 2008*, pp. 1275–1281, Barcelona, Spain, 2008.
[26] B. Waters. Efficient identity-based encryption without random oracles. In *Advances in Cryptology-EUROCRYPT 2005*, LNCS 3494, pp. 114–127, Springer-Verlag, 2005,
[27] Y. Zheng. Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost (signature) + cost(encryption). In *Advances in Cryptology-CRYPTO'97*, LNCS 1294, pp. 165–179, Springer-Verlag, 1997.

APPENDIX A
PROOF OF THEOREM 1

Proof: Our proof strategy is as follows. We define a sequence $\texttt{Game}_0$, $\texttt{Game}_1$, $\texttt{Game}_2$ of modified attack games. The only difference between games is how the environment responds to $\mathcal{A}$'s oracle queries.

Let $\sigma^* \leftarrow (\psi^*, c^*)$ be the challenge ciphertext submitted to $\mathcal{A}$ by its challenge oracle that encrypts either m_0 or m_1 according to a bit b. Let K^* denote the symmetric key used by the challenge oracle in the generation of the challenge ciphertext, or alternatively, the decapsulation of ψ^* using the identities ID_s^* and ID_r^* that are chosen by the adversary. For any $i = 0, 1, 2$, we let S_i be the event that $\delta' = \delta$ in game $\texttt{Game}_\texttt{i}$, where δ is the bit chosen by $\mathcal{A}$'s challenge oracle and δ' is the bit output by $\mathcal{A}$. This probability is taken over the random choices of $\mathcal{A}$ and those of $\mathcal{A}$'s oracles.

We will use the following useful lemma from [24].

Lemma 1: Let E, E', and F be events defined on a probability space such that $\Pr[E \wedge \neg F] = \Pr[E' \wedge \neg F]$. Then we have

$$|\Pr[E] - \Pr[E']| \leq \Pr[F].$$

$\texttt{Game}_0$: We simulate the view of the adversary in a real attack by running the key generation algorithm and using the resulting keys to respond to $\mathcal{A}$'s queries. So the view of $\mathcal{A}$ is the same as it would be in a real attack. Therefore, we have

$$|\Pr[S_0] - \frac{1}{2}| = \frac{1}{2} Adv_{\mathrm{IDSC}}^{\mathrm{IND-CCA2}}(\mathcal{A}).$$

$\texttt{Game}_1$: In this game, we slightly modify how the unsigncryption oracle responds to queries from $\mathcal{A}$. When a sender' identity ID_s, a receiver's identity ID_r, and (ψ, c) is presented to the unsigncryption oracle after the invocation of the challenge signcryption oracle, if $ID_s = ID_s^*$, $ID_r = ID_r^*$ and $\psi = \psi^*$, then the unsigncryption oracle does not use the genuine unsigncryption procedure for the hybrid scheme, instead it uses the key K^* to decrypt c and returns the result to the adversary $\mathcal{A}$.

Clearly this change has no impact on the adversary and so

$$\Pr[S_1] = \Pr[S_0].$$

$\texttt{Game}_2$: In this game, we modify Game_1 by replacing K^* with a random key K' from $\mathcal{K}_{\mathrm{DEM}}$. The result then follows from the following two lemmas. □

Lemma 2: There exists a PPT algorithm $\mathcal{B}_1$, whose running time is essentially the same as that of $\mathcal{A}$, such that

$$|\Pr[S_2] - \Pr[S_1]| = Adv_{\mathrm{IDSC-KEM}}^{\mathrm{IND-CCA2}}(\mathcal{B}_1).$$

Proof: To prove this we demonstrate how to construct an adversary $\mathcal{B}_1$ of the ID-based signcryption KEM to violate the assumed security against adaptive chosen ciphertext attack.

- `Initial` : Given $(1^k, mpk)$, $\mathcal{B}_1$ sends it to $\mathcal{A}$.
- `Phase1` : When $\mathcal{A}$ make a key extraction query on identity ID, $\mathcal{B}_1$ makes a key extraction query to its own key extraction oracle and forwards the answer to $\mathcal{A}$. When $\mathcal{A}$ make a signcryption query with a sender's identity ID_s, a receiver's identity ID_r and a plaintext m, $\mathcal{B}_1$ follows the steps below.
  1) Make a key encapsulation query on (m, ID_s, ID_r) to its own key encapsulation oracle to obtain (K, ψ).
  2) Compute $c \leftarrow \texttt{DEM.Enc}(K, m)$.
  3) Return the ciphertext $\sigma \leftarrow (\psi, c)$ to $\mathcal{A}$.

  When $\mathcal{A}$ make a unsigncryption query with a sender's identity ID_s, a receiver's identity ID_r and a ciphertext $\sigma \leftarrow (\psi, c)$, $\mathcal{B}_1$ follows the steps below.
  1) Make a key decapsulation query on (ψ, ID_s, ID_r) to its own key decapsulation oracle to obtain K.
  2) If $K = \perp$, return $\perp$ and stop.
  3) Compute $m \leftarrow \texttt{DEM.Dec}(K, c)$.
  4) Make a verification query on (ψ, m, ID_s, ID_r) to its own verification oracle to obtain $\top$ or $\perp$. If $\top$ is obtained, $\mathcal{B}_1$ output m. Otherwise output $\perp$.
- `Challenge` : $\mathcal{A}$ generates two equal length plaintexts m_0, m_1, a sender's identity ID_s^*, and a receiver's identity

ID_r^* on which it wishes to be challenged. $\mathcal{B}_1$ follows the steps below.

1) Submit ID_s^* and ID_r^* to its challenger to obtain (K_b, ψ^*), where $b \in \{0,1\}$.
2) Pick a random bit δ from $\{0,1\}$.
3) Compute $c^* \leftarrow \texttt{DEM.Enc}(K_b, m_\delta)$.
4) Return the ciphertext $\sigma^* \leftarrow (\psi^*, c^*)$ to $\mathcal{A}$.

- `Phase2` : The adversary $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in Phase 1 with the restriction that it cannot make a key extraction query on ID_r^* and it cannot make an unsigncryption query on $\sigma^* \leftarrow (\psi^*, c^*)$ to obtain the corresponding plaintext.
- `Guess` : $\mathcal{A}$ outputs δ'. If $\delta' = \delta$, $\mathcal{B}_1$ outputs $b' = 1$ indicating K_b is the real key; otherwise it outputs $b' = 0$ indicating K_b is a random key.

When K_b is the real key, $\mathcal{A}$ is run exactly as it would be run in $\texttt{Game}_1$. This means that

$$\Pr[S_1] = \Pr[\delta' = \delta | b = 1] = \Pr[b' = 1 | b = 1].$$

When K_b is the random key, $\mathcal{A}$ is run exactly as it would be in $\texttt{Game}_2$. This means that

$$\Pr[S_2] = \Pr[\delta' = \delta | b = 0] = \Pr[b' = 1 | b = 0].$$

From the definition of security for ID-based signcryption KEM, we have

$$\begin{aligned} Adv_{\mathrm{IDSC-KEM}}^{\mathrm{IND-CCA2}}(\mathcal{B}_1) &= |2\Pr[b' = b] - 1| \\ &= |\Pr[b' = 1 | b = 1] - \Pr[b' = 1 | b = 0]| \end{aligned}$$

So the result holds.

□

Lemma 3: There exists a PPT algorithm $\mathcal{B}_2$, whose running time is essentially the same as that of $\mathcal{A}$, such that

$$|\Pr[S_2] - \frac{1}{2}| = \frac{1}{2} Adv_{\mathrm{DEM}}^{\mathrm{IND-PA}}(\mathcal{B}_2).$$

Proof: To construct such a $\mathcal{B}_2$ we simply run $\mathcal{A}$ as it would be run in game $\texttt{Game}_2$. We run the ID-based key extraction step so we can respond to $\mathcal{A}$'s queries before it calls its challenge signcryption oracle. When $\mathcal{A}$ calls its challenge signcryption oracle with a sender's identity ID_s^*, a receiver's identity ID_r^*, and messages (m_0, m_1), we simply relay (m_0, m_1) to the challenge encryption oracle of $\mathcal{B}_2$ to obtain c^*. We then make a key encapsulation query on (m_0, ID_s^*, ID_r^*) to obtain (K^*, ψ^*). We discard K^* and return (ψ^*, c^*) to $\mathcal{A}$. $\mathcal{B}_2$ continues to answer $\mathcal{A}$'s queries as before. The only exception is that if $\mathcal{A}$ queries the decapsulation oracle on a ciphertext (ψ^*, c), we respond with $\perp$.

In this simulation $\mathcal{A}$ is run by $\mathcal{B}_2$ in exactly the same manner as the former would be run in game $\texttt{Game}_2$; moreover, $\Pr[S_2]$ corresponds exactly to the probability that $\mathcal{B}_2$ correctly determines the hidden bit of its challenge encryption oracle since $\mathcal{B}_2$ outputs whatever $\mathcal{A}$ outputs. The result follows. □

APPENDIX B
PROOF OF THEOREM 2

Proof: Suppose that $\mathcal{F}$ is an adversary that breaks the ID-based signcryption scheme with probability $Adv_{\mathrm{IDSC}}^{\mathrm{sUF-CMA}}(\mathcal{F})$. We use this to construct an algorithm $\mathcal{B}$ that breaks the sUF-CMA game for the ID-based signcryption KEM with probability at least $Adv_{\mathrm{IDSC}}^{\mathrm{sUF-CMA}}(\mathcal{F})$ too. $\mathcal{B}$ runs as follow

- `Initial` : Given $(1^k, mpk)$, $\mathcal{B}$ sends it to $\mathcal{F}$.
- `Attack` : When $\mathcal{F}$ make a key extraction query on identity ID, $\mathcal{B}$ makes a key extraction query to its own key extraction oracle and forwards the answer to $\mathcal{F}$. When $\mathcal{F}$ make a signcryption query with a sender's identity ID_s, a receiver's identity ID_r and a plaintext m, $\mathcal{B}$ follows the steps below.
  1) Make a key encapsulation query on (m, ID_s, ID_r) to its own key encapsulation oracle to obtain (K, ψ).
  2) Compute $c \leftarrow \texttt{DEM.Enc}(K, m)$.
  3) Return the ciphertext $\sigma \leftarrow (\psi, c)$ to $\mathcal{F}$.

  When $\mathcal{F}$ make a unsigncryption query with a sender's identity ID_s, a receiver's identity ID_r and a ciphertext $\sigma \leftarrow (\psi, c)$, $\mathcal{B}$ follows the steps below.
  1) Make a key decapsulation query on (ψ, ID_s, ID_r) to its own key decapsulation oracle to obtain K.
  2) If $K = \perp$, return $\perp$ and stop.
  3) Compute $m \leftarrow \texttt{DEM.Dec}(K, c)$.
  4) Make a verification query on (ψ, m, ID_s, ID_r) to its own verification oracle to obtain $\top$ or $\perp$. If $\top$ is obtained, $\mathcal{B}_1$ output m. Otherwise output $\perp$.
- `Forgery` : $\mathcal{F}$ outputs $(m^*, \sigma^*, ID_s^*, ID_r^*)$, where $(\psi^*, c^*) \leftarrow \sigma^*$. $\mathcal{B}$ outputs $(m^*, \psi^*, ID_s^*, ID_r^*)$.

Clearly, this algorithm perfectly simulates the environment in which $\mathcal{F}$ should be running. If $\mathcal{F}$ wins the sUF-CMA game for ID-based signcryption, $\mathcal{B}$ have the same probability to win the sUF-CMA game for ID-based signcryption KEM. □