An Efficient and Provably Secure ID-Based Threshold Signcryption Scheme

Fagen Li[1,2] and Yong Yu[1]
[1]School of Computer Science and Engineering,
University of Electronic Science and Technology of China, Chengdu 610054, P.R. China
[2]Key Laboratory of Computer Networks and Information Security,
Xidian University, Xi'an 710071, P.R. China
E-mail:fagenli@uestc.edu.cn

***Abstract*— Signcryption is a cryptographic primitive that performs digital signature and public key encryption simultaneously, at a lower computational costs and communication overheads than the signature-then-encryption approach. Recently, two identity-based threshold signcryption schemes [12], [26] have been proposed by combining the concepts of identity-based threshold signature and signcryption together. However, the formal models and security proofs for both schemes are not considered. In this paper, we formalize the concept of identity-based threshold signcryption and give a new scheme based on the bilinear pairings. We prove its confidentiality under the Decisional Bilinear Diffie-Hellman assumption and its unforgeability under the Computational Diffie-Hellman assumption in the random oracle model. Our scheme turns out to be more efficient than the two previously proposed schemes.**

I. Introduction

Identity-based (ID-based) cryptography was introduced by Shamir in 1984 [29]. 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 [13], [15] but a satisfying ID-based encryption scheme only appeared in 2001 [6]. It was devised by Boneh and Franklin and cleverly uses bilinear maps (the Weil or Tate pairing) over supersingular elliptic curves.

Group-oriented cryptography was introduced by Desmedt in 1987 [10]. Elaborating on this concept, Desmedt and Frankel [11] proposed a (t, n) threshold signature scheme based on the RSA system [27]. In such a (t, n) threshold signature scheme, any t out of n signers in the group can collaboratively sign messages on behalf of the group for sharing the signing capability. The first ID-based threshold signature scheme was proposed by Baek and Zheng in 2004 [3].

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 in 1997 [33], is a cryptographic primitive that performs digital signature and public key encryption simultaneously, at lower computational costs and communication overheads than the signature-then-encryption approach. Following [33], various signcryption schemes have been proposed, for instance, signcryption schemes in certificate-based public key setting [25], [4], [30], [14], [28], [34], [23], [20], [31], [24] and signcryption schemes in ID-based public key setting [22], [19], [9], [7], [5], [17], [32], [16]. The original scheme in [33] is based on the discrete logarithm problem but no security proof is given. Zheng's original construction [33] was only proven secure in 2002 by Baek et al. [2] who described a formal security model in a multi-user setting.

In 2004, Duan et al. [12] proposed an ID-based threshold signcryption scheme by combining the concepts of ID-based threshold signature and signcryption together. However, in Duan et al.'s scheme [12], the master-key of the PKG is distributed to a number of other PKGs, which creates a bottleneck on the PKGs. In 2005, Peng and Li [26] proposed an ID-based threshold signcryption scheme based on Libert and Quisquater's ID-based signcryption scheme [19]. However, Peng and Li' scheme [26] does not provide the forward security. That is, anyone who obtains the sender's private key can recover the original message of a signcrypted text. In addition, both Duan et al.'s scheme [12] and Peng and Li's scheme [26] do not consider the formal models and security proofs. Ma et al. [21] also proposed a threshold signcryption scheme using the bilinear pairings. However, Ma et al.'s scheme [21] is not ID-based. Therefore, an interesting question is to find a provably secure ID-based threshold signcryption scheme. The aim of this paper is to answer this question.

A. Related Work

Signcryption in certificate-based public key setting. The non-repudiation procedure of Zheng's original schemes [33] is inefficient since they are based on interactive zero-knowledge proofs. In [25], Petersen and Michels showed that Zheng's idea violates the confidentiality to achieve the non-repudiation. To achieve simple and safe non-repudiation procedure, Bao and Deng [4] introduced a signcryption scheme that can be verified by a sender's public key. However, Shin et al. [30] pointed out that Bao and Deng's scheme [4] is not semantically secure since the signature on the plaintext is visible in the ciphertext. An attacker can distinguish two messages m_0 and m_1 by

This is the full version of a paper published in the 2008 International Conference on Communications, Circuits and Systems (ICCCAS 2008), IEEE Press, Xiamen, China, 2008, pp. 547–551.

verifying the signature. In [14], Gamage et al. modified Bao and Deng's scheme [4] to carry out the signature verification without accessing the plaintext. In [28], based on Gamage et al.'s scheme [14], Seo and Kim proposed a domain-verifiable signcryption scheme which signcrypts n messages to n users. Each user with domain can decrypt just his own message and all users can verify the whole transaction. In [34], Zheng and Imai showed how to construct efficient signcryption schemes on elliptic curves. In [23], Malone-Lee and Mao proposed an efficient signcryption scheme using RSA [27]. In [20], Libert and Quisquater proposed a signcryption scheme using the bilinear pairings which is showed to be insecure against chosen ciphertext attack (not even secure against chosen plaintext attack) by Yang et al. in [31]. In [24], Mu and Varadharajan proposed a distributed signcryption scheme and extended it to a group signcryption scheme.

Signcryption in ID-based public key setting. In 2002, Malone-Lee [22] gave the first ID-based signcryption scheme along with a security model. This model deals with notions of privacy and unforgeability. Libert and Quisquater [19] pointed out that Malone-Lee's scheme [22] is not semantically secure and proposed three provably secure ID-based signcryption schemes. However, the properties of public verifiability and forward security are mutually exclusive in the their schemes. To overcome this weakness, Chow et al. [9] designed an ID-based signcryption scheme that provides both public verifiability and forward security. In [7], Boyen presented an ID-based signcryption scheme that provides not only public verifiability and forward security but also ciphertext unlinkability and anonymity. In [5], Barreto et al. constructed the most efficient ID-based signcryption scheme to date. In [17], Li and Chen proposed an ID-based proxy signcryption scheme. In [32], Yuen and Wei proposed an ID-based blind signcryption scheme. In [16], Huang et al. proposed an ID-based ring signcryption scheme. In [18], Li et al. proposed an ID-based signcryption for multiple private key generators.

B. Our Contribution

In this paper, we present a formal security model for identity-based threshold signcryption and give a new scheme based on the bilinear pairings. We prove its confidentiality under the DBDH assumption and its unforgeability under the CDH assumption in the random oracle model. As compared with two previously proposed schemes (Duan et al.'s scheme [12] and Peng and Li's scheme [26]), our scheme is more efficient.

C. Organization

The rest of this paper is organized as follows. Some preliminary works are given in Section II. The formal model of ID-based threshold signcryption is described in Section III. The proposed ID-based threshold signcryption scheme is given in Section IV. We analyze the proposed scheme in Section V. Finally, the conclusions are given in Section VI.

II. Preliminaries

In this section, we briefly describe the basic definition and properties of the bilinear pairings.

Let G_1 be a cyclic additive group generated by P, whose order is a prime q, and G_2 be a cyclic multiplicative group of the same order q. A bilinear pairing is a map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ with the following properties:

1) Bilinearity: $\hat{e}(aP, bQ) = \hat{e}(P, Q)^{ab}$ for all $P, Q \in G_1$, $a, b \in Z_q$.
2) Non-degeneracy: There exists P and $Q \in G_1$ such that $\hat{e}(P, Q) \neq 1$.
3) Computability: There is an efficient algorithm to compute $\hat{e}(P, Q)$ for all $P, Q \in G_1$.

The modified Weil pairing and the Tate pairing [6] are admissible maps of this kind. The security of our scheme described here relies on the hardness of the following problems.

Definition 1: Given two groups G_1 and G_2 of the same prime order q, a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ and a generator P of G_1, the Decisional Bilinear Diffie-Hellman problem (DBDHP) in $(G_1, G_2, \hat{e})$ is to decide whether $h = \hat{e}(P, P)^{abc}$ given (P, aP, bP, cP) and an element $h \in G_2$. We define the advantage of a distinguisher against the DBDHP like this

$$\begin{aligned} Adv(D) \quad = \quad & |P_{a,b,c,\in_R Z_q, h \in_R G_2}[1 \leftarrow D(aP, bP, cP, h)] \\ & -P_{a,b,c,\in_R Z_q}[1 \leftarrow D(aP, bP, cP, \hat{e}(P, P)^{abc})]|. \end{aligned}$$

Definition 2: Given two groups G_1 and G_2 of the same prime order q, a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$ and a generator P of G_1, the Computational Bilinear Diffie-Hellman problem (CBDHP) in $(G_1, G_2, \hat{e})$ is to compute $h = \hat{e}(P, P)^{abc}$ given (P, aP, bP, cP).

The decisional problem is of course not harder than the computational one. However, no algorithm is known to be able to solve any of them so far.

III. Formal Model of ID-Based Threshold Signcryption

A. Generic Scheme

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

- **Setup:** Given a security parameter k, the private key generator (PKG) generates the system's public parameters *params*. Among the parameters produced by **Setup** is a key P_{pub} that is made public. There is also corresponding master key s that is kept secret.
- **Extract:** Given an identity ID, the PKG computes the corresponding private key S_{ID} and transmits it to its owner in a secure way.
- **Keydis:** Given a private key S_{ID} associated with an identity ID, the number of signcryption members n and a threshold parameter t, this algorithm generates n shares of S_{ID} and provides each one to the signcryption members $M_1, \ldots, M_n$. It also generates a set of verification keys that can be used to check the validity of each

shared private key. We denote the shared private keys and the matching verification keys by $\{S_i\}_{i=1,\ldots,n}$ and $\{y_i\}_{i=1,\ldots,n}$, respectively. Note that each (S_i, y_i) is sent to M_i, then M_i publishes y_i but keeps S_i secret.
- **Signcrypt:** Give a message m, the private keys of t members $\{S_i\}_{i=1,\ldots,t}$ in a sender group U_A with identity ID_A, a receiver's identity ID_B, it outputs an ID-based (t, n) threshold signcryption σ on the message m.
- **Unsigncrypt:** Give a ciphertext σ, the private key of the receiver S_{ID_B}, the identity of the sender group ID_A, it outputs the plaintext m or the symbol $\perp$ if σ is an invalid ciphertext between the group U_A and the receiver.

We make the consistency constraint that if

$$\sigma = \mathbf{Signcrypt}(m, \{S_i\}_{i=1,\ldots,t}, ID_B),$$

then

$$m = \mathbf{Unsigncrypt}(\sigma, ID_A, S_{ID_B}).$$

B. Security Notions

Malone-Lee [22] defines the security notions for ID-based signcryption schemes. These notions are indistinguishability against adaptive chosen ciphertext attacks and unforgeability against adaptive chosen messages attacks. We modify their definitions slightly to adapt for our ID-based threshold signcryption scheme. In addition, an ID-based threshold signcryption scheme should have the robustness.

Definition 3 (Confidentiality): An ID-based threshold signcryption scheme (IDTSC) is said to have the indistinguishability against adaptive chosen ciphertext attacks property (IND-IDTSC-CCA2) if no polynomially bounded adversary has a non-negligible advantage in the following game.

1) The challenger $\mathcal{C}$ runs the **Setup** algorithm with a security parameter k and sends the system parameters to the adversary $\mathcal{A}$.
2) $\mathcal{A}$ performs a polynomially bounded number of queries (these queries may be made adaptively, i.e. each query may depend on the answer to the previous queries).
   - Key extraction queries: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $S_{ID} = \mathbf{Extract}(ID)$ and sends S_{ID} to $\mathcal{A}$.
   - Signcryption queries: $\mathcal{A}$ produces a sender group U_i with identity ID_i, an identity ID_j and a plaintext m. $\mathcal{C}$ computes $S_{ID_i} = \mathbf{Extract}(ID_i)$ and runs **Keydis** to output n shared private keys $\{S_i\}_{i=1,\ldots,n}$. $\mathcal{C}$ sends the result of **Signcrypt**$(m, \{S_i\}_{i=1,\ldots,t}, ID_j)$ to $\mathcal{A}$.
   - Unsigncryption queries: $\mathcal{A}$ produces a sender group U_i with identity ID_i, an identity ID_j, and a ciphertext σ. $\mathcal{C}$ generates the private key $S_{ID_j} = \mathbf{Extract}(ID_j)$ and sends the result of **Unsigncrypt**(σ, ID_i, S_{ID_j}) to $\mathcal{A}$ (this result can be the $\perp$ symbol if σ is an invalid ciphertext)
3) $\mathcal{A}$ generates two equal length plaintexts m_0, m_1, a sender group U_A with identity ID_A, and an identity ID_B on which he wants to be challenged. He cannot have asked the private key corresponding to ID_B in the first stage.
4) $\mathcal{C}$ takes a bit $b \in_R \{0, 1\}$ and runs **Keydis** to output n shared private keys $\{S_i\}_{i=1,\ldots,n}$. $\mathcal{C}$ sends the result of $\sigma = \mathbf{Signcrypt}(m_b, \{S_i\}_{i=1,\ldots,t}, ID_B)$ to $\mathcal{A}$.
5) $\mathcal{A}$ can ask a polynomially bounded number of queries adaptively again as in the first stage. This time, he cannot make a key extraction query on ID_B and cannot make an unsigncryption query on σ to obtain the corresponding plaintext.
6) Finally, $\mathcal{A}$ produces a bit b' and wins the game if $b' = b$.

The advantage of $\mathcal{A}$ is defined as $Adv(\mathcal{A}) = |2P[b' = b] - 1|$, where $P[b' = b]$ denotes the probability that $b' = b$.

Notice that the adversary is allowed to make a key extraction query on identity ID_A in the above definition. This condition corresponds to the stringent requirement of insider security for confidentiality of signcryption [1]. 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.

Definition 4 (Unforgeability): An ID-based threshold signcryption scheme (IDTSC) is said to have the existential unforgeability against adaptive chosen messages attacks (EUF-IDTSC-CMA) if no polynomially bounded adversary has a non-negligible advantage in the following game.

1) The challenger $\mathcal{C}$ runs the **Setup** algorithm with a security parameter k and sends the system parameters to $\mathcal{A}$.
2) $\mathcal{A}$ corrupts $t - 1$ members in the sender group.
3) $\mathcal{A}$ performs a polynomially bounded number of queries (these queries may be made adaptively, i.e. each query may depend on the answer to the previous queries).
   - Key extraction queries: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $S_{ID} = \mathbf{Extract}(ID)$ and sends S_{ID} to $\mathcal{A}$.
   - Private keys queries to the corrupted members: $\mathcal{A}$ chooses an identity ID. $\mathcal{C}$ computes $S_{ID} = \mathbf{Extract}(ID)$ and runs **Keydis** to output n shared private keys $\{S_i\}_{i=1,\ldots,n}$. $\mathcal{C}$ sends S_i for $i = 1, \ldots, t-1$ to $\mathcal{A}$.
   - Signcryption queries: $\mathcal{A}$ produces a sender group U_i with identity ID_i, an identity ID_j and a plaintext m. $\mathcal{C}$ computes $S_{ID_i} = \mathbf{Extract}(ID_i)$ and runs **Keydis** to output n shared private keys $\{S_i\}_{i=1,\ldots,n}$. $\mathcal{C}$ sends the result of **Signcrypt**$(m, \{S_i\}_{i=t,\ldots,n}, ID_j)$ to $\mathcal{A}$.
   - Unsigncryption queries: $\mathcal{A}$ produces a sender group U_i with identity ID_i, an identity ID_j, and a ciphertext σ. $\mathcal{C}$ generates the private key $S_{ID_j} = \mathbf{Extract}(ID_j)$ and sends the result of **Unsigncrypt**(σ, ID_i, S_{ID_j}) to $\mathcal{A}$ (this result can be the $\perp$ symbol if σ is an invalid ciphertext)
4) Finally, $\mathcal{A}$ produces a new triple (ID_A, ID_B, σ)(i.e. a triple that was not produced by the signcryption oracle), where the private key of ID_A was not asked in the

second stage and wins the game if the result of the **Unsigncrypt**(σ, ID_A,S_{ID_B}) is not the $\perp$ symbol.

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

Note that the adversary is allowed to make a key extraction query on the identity ID_B in the above definition. Again, this condition corresponds to the stringent requirement of insider security for signcryption [1].

Definition 5 (Robustness): An ID-based (t,n) threshold signcryption scheme (IDTSC) is said to be robust if it computes a correct output even in the presence of a malicious adversary that makes the $t-1$ corrupted members deviate from the normal execution.

IV. An Efficient ID-Based Threshold Signcryption Scheme

In this section, we present an efficient ID-based threshold signcryption scheme based on the bilinear pairings. The proposed scheme involves four roles: the PKG, a trusted dealer, a sender group $U_A = \{M_1, \ldots, M_n\}$ with identity ID_A, and a receiver Bob with identity ID_B. The following shows the details of our scheme.

- **Setup:** Given a security parameter k, the PKG chooses groups G_1 and G_2 of prime order q (with G_1 additive and G_2 multiplicative), a generator P of G_1, a bilinear map $\hat{e} : G_1 \times G_1 \rightarrow G_2$, a secure symmetric cipher (E, D) and hash functions $H_1 : \{0,1\}^* \rightarrow G_1$, $H_2 : G_2 \rightarrow \{0,1\}^{n_1}$, $H_3 : \{0,1\}^* \rightarrow Z_q^*$. The PKG chooses a master-key $s \in_R Z_q^*$ and computes $P_{pub} = sP$. The PKG publishes system parameters $\{G_1, G_2, n_1, \hat{e}, P, P_{pub}, E, D, H_1, H_2, H_3\}$ and keeps the master-key s secret.
- **Extract:** Given an identity ID, the PKG computes $Q_{ID} = H_1(ID)$ and the private key $S_{ID} = sQ_{ID}$. Then PKG sends the private key to its owner in a secure way.
- **Keydis:** Suppose that a threshold t and n satisfy $1 \leq t \leq n < q$. To share the private key S_{ID_A} among the group U_A, the trusted dealer performs the steps below.
  1) Choose $F_1, \ldots, F_{t-1}$ uniformly at random from G_1^*, construct a polynomial $F(x) = S_{ID_A} + xF_1 + \cdots + x^{t-1}F_{t-1}$ and compute $S_i = F(i)$ for $i = 0, \ldots, n$. Note that $S_0 = S_{ID_A}$.
  2) Send S_i to member M_i for $i = 1, \ldots, n$ secretly. Broadcast $y_0 = \hat{e}(S_{ID_A}, P)$ and $y_j = \hat{e}(F_j, P)$ for $j = 1, \ldots, t-1$.
  3) Each M_i then checks whether his share S_i is valid by computing $\hat{e}(S_i, P) = \prod_{j=0}^{t-1} y_j^{i^j}$. If S_i is not valid, M_i broadcasts an error and requests a valid one.
- **Signcrypt:** Without loss of generality, we assume that $M_1, \ldots, M_t$ are the t members who want to cooperate to signcrypt a message m on behalf of the group U_A. Each $M_i (1 \leq i \leq t)$ uses Cheng et al.'s ID-based signature scheme [8] to generate the partial signature and an appointed clerk C, who is one of the t members, combines the partial signatures to generate the final threshold signcryption.
  1) Each M_i chooses $x_i \in_R Z_q^*$, computes $R_{1i} = x_i P$ and $R_{2i} = x_i P_{pub}$, and sends (R_{1i}, R_{2i}) to the clerk C.
  2) The clerk C computes $R_1 = \sum_{i=1}^t R_{1i}$, $R_2 = \sum_{i=1}^t R_{2i}$, $\tau = \hat{e}(R_2, Q_{ID_B})$, $k = H_2(\tau)$, $c = E_k(m)$, and $h = H_3(m, R_1, k)$. Then the clerk C sends h to M_i for $i = 1, \ldots, t$.
  3) Each M_i computes the partial signature $W_i = x_i P_{pub} + h\eta_i S_i$ and sends it to the clerk C, where $\eta_i = \prod_{j=1, j\neq i}^t -j(i-j)^{-1} \bmod q$.
  4) When receiving M_i's partial signature W_i, the clerk C verifies its correctness by checking if the following equation holds:
  $$\hat{e}(P, W_i) = \hat{e}(R_{1i}, P_{pub})(\prod_{j=0}^{t-1} y_j^{i^j})^{h\eta_i}.$$
  If all partial signatures are verified to be legal, the clerk C computes $W = \sum_{i=1}^t W_i$; otherwise rejects it and requests a valid one. The final threshold signcryption is $\sigma = (c, R_1, W)$.
- **Unsigncrypt:** When receiving σ, Bob follows the steps below.
  1) Compute $\tau = \hat{e}(R_1, S_{ID_B})$ and $k = H_2(\tau)$.
  2) Recover $m = D_k(c)$.
  3) Compute $h = H_3(m, R_1, k)$ and accept σ if and only if the following equation holds:
  $$\hat{e}(P, W) = \hat{e}(P_{pub}, R_1 + hQ_{ID_A}).$$

V. Analysis of the Scheme

A. Correctness

The correctness can be easily verified by the following equations.

$$\begin{aligned}
\hat{e}(R_1, S_{ID_B}) &= \hat{e}(\sum_{i=1}^t R_{1i}, S_{ID_B}) = \hat{e}(\sum_{i=1}^t (x_i P), S_{ID_B}) \\
&= \hat{e}(\sum_{i=1}^t (x_i P_{pub}), Q_{ID_B}) \\
&= \hat{e}(\sum_{i=1}^t R_{2i}, Q_{ID_B}) \\
&= \hat{e}(R_2, Q_{ID_B})
\end{aligned}$$

and

$$\begin{aligned}
\hat{e}(P, W) &= \hat{e}(P, \sum_{i=1}^t W_i) = \hat{e}(P, \sum_{i=1}^t (x_i P_{pub} + h\eta_i S_i)) \\
&= \hat{e}(P, \sum_{i=1}^t (x_i P_{pub}) + \sum_{i=1}^t (h\eta_i S_i)) \\
&= \hat{e}(P, \sum_{i=1}^t (x_i P_{pub}) + hS_{ID_A}) \\
&= \hat{e}(P_{pub}, \sum_{i=1}^t (x_i P) + hQ_{ID_A})
\end{aligned}$$

$$= \quad \hat{e}(P_{pub}, R_1 + hQ_{ID_A})$$

B. Security

Theorem 1 (Confidentiality): In the random oracle model, we assume we have an IND-IDTSC-CCA2 adversary called $\mathcal{A}$ that is able to distinguish ciphertext during the game of Definition 3 with an advantage ϵ when running in a time t and asking at most q_{H_1} identity hashing queries, at most q_{H_2} H_2 queries, at most q_{H_3} H_3 queries, at most q_K key extraction queries, q_S signcryption queries and q_U unsigncryption queries. Then, there exists a distinguisher $\mathcal{C}$ that can solve the Decisional Bilinear Diffie-Hellman problem in a time $O(t + (q_{H_3} q_S + q_S^2 + 3q_U)T_{\hat{e}})$ with an advantage

$$Adv(\mathcal{C})^{DBDH(G_1,P)} > \frac{\epsilon(2^k - q_U) - q_U}{q_{H_1} 2^{k+1}},$$

where $T_{\hat{e}}$ denotes the computation time of the bilinear map.

Proof: We assume the distinguisher $\mathcal{C}$ receives a random instance (P, aP, bP, cP, h) of the Decisional Bilinear Diffie-Hellman problem. His goal is to decide whether $h = \hat{e}(P, P)^{abc}$ or not. $\mathcal{C}$ will run $\mathcal{A}$ as a subroutine and act as $\mathcal{A}$'s challenger in the IND-IDTSC-CCA2 game. During the game, $\mathcal{A}$ will consult $\mathcal{C}$ for answers to the random oracles H_1, H_2 and H_3. Roughly speaking, these answers are randomly generated, but to maintain the consistency and to avoid collision, $\mathcal{C}$ keeps three lists L_1, L_2, L_3 respectively to store the answers. The following assumptions are made.

1) $\mathcal{A}$ will ask for $H_1(ID)$ before ID is used in any key extraction query, signcryption query and unsigncryption query.
2) Ciphertext returned from a signcryption query will not be used by $\mathcal{A}$ in an unsigncryption query.

At the beginning of the game, $\mathcal{C}$ gives $\mathcal{A}$ the system parameters with $P_{pub} = cP$. Note that c is unknown to $\mathcal{C}$. This value simulates the master-key value for the PKG in the game. Then, $\mathcal{C}$ chooses a random number $j \in \{1, 2, \ldots, q_{H_1}\}$. $\mathcal{A}$ asks a polynomially bounded number of H_1 queries on identities of his choice. At the j-th H_1 query, $\mathcal{C}$ answers by $H_1(ID_j) = bP$. For queries $H_1(ID_e)$ with $e \neq j$, $\mathcal{C}$ chooses $b_e \in_R Z_q^*$, puts the pair (ID_e, b_e) in list L_1 and answers $H_1(ID_e) = b_e P$.

We now explain how the other kinds of queries are treated by $\mathcal{C}$.

- H_2 queries: On a $H_2(\tau_e)$ query, $\mathcal{C}$ searches a pair (τ_e, k_e) in the list L_2. If such a pair is found, $\mathcal{C}$ answers k_e, otherwise he answers $\mathcal{A}$ by a random binary sequence $k \in_R \{0,1\}^{n_1}$ such that no entry $(\cdot, k)$ exists in L_2 (in order to avoid collisions on H_2) and puts the pair (τ_e, k) into L_2.
- H_3 queries: On a $H_3(m_e, R_{1_e}, k_e)$ query, $\mathcal{C}$ checks if there exists (m_e, R_{1_e}, k_e, h_e) in L_3. If such a tuple is found, $\mathcal{C}$ answers h_e, otherwise he chooses $h \in_R Z_q^*$, gives it as an answer to the query and puts the tuple (m_e, R_{1_e}, k_e, h) into L_3.
- Key extraction queries: When $\mathcal{A}$ asks a question **Extract**(ID_e), if $ID_e = ID_j$, then $\mathcal{C}$ fails and stops. If $ID_e \neq ID_j$, then the list L_1 must contain a pair (ID_e, b_e) for some b_e (this indicates $\mathcal{C}$ previously answered $H_1(ID_e) = b_e P$ on a H_1 query on ID_e). The private key corresponding to ID_e is then $b_e P_{pub} = cb_e P$. It is computed by $\mathcal{C}$ and returned to $\mathcal{A}$.
- Signcryption queries: At any time, $\mathcal{A}$ can perform a signcryption query for a plaintext m, a sender group U_A with identity ID_A and a receiver with identity ID_B. We have the following three cases to consider.
  - Case 1: $ID_A \neq ID_j$. $\mathcal{C}$ computes the private key S_{ID_A} corresponding to ID_A by running the key extraction query algorithm. Then $\mathcal{C}$ runs **Keydis** to output n shared private keys $\{S_i\}_{i=1,\ldots,n}$. Finally, $\mathcal{C}$ answers the query by a call to **Signcrypt**$(m, \{S_i\}_{i=1,\ldots,t}, Q_{ID_B})$.
  - Case 2: $ID_A = ID_j$ and $ID_B \neq ID_j$. $\mathcal{C}$ chooses $x, h \in_R Z_q^*$ and computes $R_1 = xP - hQ_{ID_A}$, $W = xP_{pub}$, and $\tau = \hat{e}(R_1, S_{ID_B})$($\mathcal{C}$ could obtain S_{ID_B} from the key extraction algorithm because $ID_B \neq ID_j$). $\mathcal{C}$ runs the H_2 simulation algorithm to find $k = H_2(\tau)$ and computes $c = E_k(m)$. $\mathcal{C}$ then checks if L_3 already contains a tuple (m, R_1, k, h') with $h' \neq h$. In this case, $\mathcal{C}$ repeats the process with another random pair (x, h) until finding a tuple (m, R_1, k, h) whose first three elements do not appear in a tuple of the list L_3. This process repeats at most $q_{H_3} + q_S$ times as L_3 contains at most $q_{H_3} + q_S$ entries ($\mathcal{A}$ can issue q_{H_3} H_3 queries and q_S signcryption queries, while each signcryption query contains a single H_3 query). When an appropriate pair (x, h) is found, the ciphertext (c, R_1, W) appears to be valid from $\mathcal{A}$'s viewpoint. $\mathcal{C}$ has to compute one pairing operation for each iteration of the process.
  - Case 3: $ID_A = ID_j$ and $ID_B = ID_j$. $\mathcal{C}$ chooses $x^*, h^* \in_R Z_q^*$, computes $R_1^* = x^*P - h^*Q_{ID_A}$, $W^* = x^*P_{pub}$, and chooses $\tau^* \in_R G_2$ and $k^* \in_R \{0,1\}^{n_1}$ such that no entry $(\cdot, k^*)$ is in L_2 and computes $c^* = E_{k^*}(m)$. $\mathcal{C}$ then checks if L_3 already contains a tuple (m, R_1^*, k^*, h') with $h' \neq h^*$. If not, $\mathcal{C}$ puts the tuple (m, R_1^*, k^*, h^*) into L_3 and (τ^*, k^*) into L_2. Otherwise, $\mathcal{C}$ chooses another random pair (x^*, h^*) and repeats the process as above until he finds a tuple (m, R_1^*, k^*, h^*) whose first three elements do not appear in an entry of L_3. Once an appropriate pair (x^*, h^*) is found, $\mathcal{C}$ gives the ciphertext $\sigma^* = (c^*, R_1^*, W^*)$ to $\mathcal{A}$. As $\mathcal{A}$ will not ask for the unsigncryption of σ^*, he will never see that σ^* is not a valid ciphertext of the plaintext m for identities ID_A and ID_B.
- Unsigncryption queries: For a unsigncryption query on a ciphertext $\sigma' = (c', R_1', W')$ between a sender group with identity ID_A and a receiver with identity ID_B. We have the following two cases to consider.

- Case 1: $ID_B = ID_j$. $\mathcal{C}$ always answers $\mathcal{A}$ that σ' is invalid.
- Case 2: $ID_B \neq ID_j$. $\mathcal{C}$ computes $\tau' = \hat{e}(R_1', S_{ID_B})$ ($\mathcal{C}$ could obtain S_{ID_B} from the key extraction algorithm because $ID_B \neq ID_j$). $\mathcal{C}$ then runs the H_2 simulation algorithm to obtain $k' = H_2(\tau')$ and computes $m' = D_{k'}(c)$. Finally, $\mathcal{C}$ runs the H_3 simulation algorithm to obtain $h' = H_3(m', R_1', k')$ and checks if $\hat{e}(P, W') = \hat{e}(P_{pub}, R_1' + h'Q_{ID_A})$ holds. If the above equation does not hold, $\mathcal{C}$ rejects the ciphertext. Otherwise $\mathcal{C}$ returns m'.

It is easy to see that, for all queries, the probability to reject a valid ciphertext does not exceed $q_U/2^k$.

After the first stage, $\mathcal{A}$ picks a pair of identities on which he wishes to be challenged. Note that $\mathcal{C}$ fails if $\mathcal{A}$ has asked a key extraction query on ID_j during the first stage. We know that the probability for $\mathcal{C}$ not to fail in this stage is $\frac{q_{H_1}-q_K}{q_{H_1}}$. Further, with a probability exactly $\frac{1}{q_{H_1}-q_K}$, $\mathcal{A}$ chooses to be challenged on the pair (ID_i, ID_j) with $i \neq j$. Hence the probability that $\mathcal{A}$'s response is helpful to $\mathcal{C}$ is $\frac{1}{q_{H_1}}$. Note that if $\mathcal{A}$ has submitted a key extraction query on ID_j, then $\mathcal{C}$ fails because he is unable to answer the question. On the other hand, if $\mathcal{A}$ does not choose (ID_i, ID_j) as target identities, $\mathcal{C}$ fails too.

Then $\mathcal{A}$ outputs two plaintexts m_0 and m_1. $\mathcal{C}$ chooses $b \in_R \{0, 1\}$ and signcrypts m_b. To do so, he sets $R_1^* = aP$, obtains $k^* = H_2(h)$(where h is $\mathcal{C}$ candidate for the DBDH problem) from the H_2 simulation algorithm, and computes $c_b = E_{k^*}(m_b)$. Then $\mathcal{C}$ chooses $W^* \in_R G_1$ and sends the ciphertext $\sigma^* = (c_b, R_1^*, W^*)$ to $\mathcal{A}$.

$\mathcal{A}$ then performs a second series of queries which is treated in the same way as the first one. At the end of the simulation, he produces a bit b' for which he believes the relation $\sigma^* = \mathbf{Signcrypt}(m_{b'}, \{S_i\}_{i=1,\dots,t}, ID_j)$ holds. At this moment, if $b = b'$, $\mathcal{C}$ outputs $h = \hat{e}(R_1^*, S_{ID_j}) = \hat{e}(aP, cbP) = \hat{e}(P, P)^{abc}$ as a solution of the DBDH problem, otherwise $\mathcal{C}$ stops and outputs "failure".

Taking into account all the probabilities that $\mathcal{C}$ will not fail its simulation, the probability that $\mathcal{A}$ chooses to be challenged on the pair (ID_i, ID_j), and also the probability that $\mathcal{A}$ wins the IND-IDTSC-CCA2 game, the value of $Adv(\mathcal{C})$ is calculated as follows.

$$Adv(\mathcal{C}) > (\frac{(\epsilon + 1)}{2}(1 - \frac{q_U}{2^k}) - \frac{1}{2})(\frac{1}{q_{H_1}}) = \frac{\epsilon(2^k - q_U) - q_U}{q_{H_1} 2^{k+1}}$$

The bound on $\mathcal{C}$'s computation time derives from the fact that every signcryption query requires at most $q_{H_3} + q_S$ pairing operations and every unsigncryption query requires at most 3 pairing operations. □

Baek and Zheng [3] defined the simulatability of ID-based threshold signature and proved the relationship between the security of ID-based threshold signature and that of ID-based signature. From these results, we can obtain the following Theorem 3.

Definition 6 ([3]): An ID-based threshold signature scheme is said to be simulatable if the following conditions hold.

1) The private key distribution is simulatable: given the system parameters *params* and the identity ID, there exists a simulator which can simulate the view of the adversary on an execution of private key distribution.
2) The threshold signature generation is simulatable: given the system parameters *params*, the identity ID, the message m, the corresponding signature (R_1, W), $t-1$ shares of the private key that matches to ID of the corrupted members, and the corresponding verification keys, there is a simulator which can simulate the view of the adversary on an execution of threshold signature generation.

Theorem 2 ([3]): If an ID-based threshold signature scheme is simulatable and the ID-based signature scheme which is associated with the ID-based threshold signature scheme is secure in the sense of unforgeability, then the ID-based threshold signature scheme is also secure in the sense of unforgeability.

Theorem 3 (Unforgeability): The proposed ID-based threshold signcryption scheme is secure in the sense of unforgeability.

Proof: The proposed scheme uses Cheng et al.'s ID-based signature scheme [8]. Cheng et al.'s scheme has been proved to be secure in the sense of unforgeability under the Computational Diffie-Hellman (CDH) problem assumption in the random oracle model. Therefore, we only need to prove the proposed scheme is simulatable. Our scheme uses Baek and Zheng's private key distribution scheme [3]. Baek and Zheng's proved that their private key distribution scheme is simulatable in [3]. Now, we prove the threshold signature generation is simulatable. Given the system parameters *params*, the identity ID_A, the message m, the encryption key k, the corresponding signature (R_1, W), $t-1$ shares $\{S_i\}_{i=1,\dots,t}$ of the private key S_{ID_A}, and the corresponding verification keys $\{y_j\}_{j=0,\dots,t}$. The adversary computes $h = H_3(m, R_1, k)$ and $W_i = x_i P_{pub} + h\eta_i S_i$ for $i = 1, \dots, t-1$. Let $f(x)$ be a polynomial of degree $t-1$ such that $f(0) = W$ and $f(i) = W_i$ for $i = 1, \dots, t-1$. The adversary can compute $f(i) = W_i$ for $i = t, \dots, n$. So, the proposed scheme is secure in the sense of unforgeability. □

Theorem 4 (Robustness): The proposed ID-based threshold signcryption sch-eme is robust against an adversary which is allowed to corrupt any $t-1$ members, where $n \geq 2t-1$.

Proof: In the **Keydis** phase, each member M_i can validate his private key share S_i using the published verification keys $\{y_j\}_{j=0,\dots,t-1}$. In the **Signcrypt** phase, any $t-1$ or fewer members can not generate a valid signcryption, and only t or more members can generate a valid signcryption. The clerk C first verifies all the partial signatures by $\hat{e}(P, W_i) = \hat{e}(R_{1i}, P_{pub})(\prod_{j=0}^{t-1} y_j^{i^j})^{h\eta_i}$ and then chooses the valid ones to generate a threshold signcryption. Even if having corrupted up to $t-1$ members, the adversary still cannot produce a valid threshold signcryption. While the clerk C can get t valid partial signatures, thus can produce a valid threshold signcryption. □

	Signcrypt			Unsigncrypt			Ciphertext size
	G_1 Mul	G_2 Exp	Pairing	G_1 Mul	G_2 Exp	Pairing	
Duan et al. [12]	$t+3$	0	$3t$	0	0	4	$\|m\|+2\|G_1\|$
Peng and Li [26]	$2t$	$3t$	$3t$	0	2	4	$\|m\|+\|q\|+\|G_1\|$
Our	$4t$	t	$2t+1$	1	0	3	$\|m\|+2\|G_1\|$

Fig. 1. Efficiency comparison

C. *Efficiency*

We compare the major computational costs and communication overheads (the length of the ciphertext) of our scheme with those of Duan et al.'s ID-based threshold signcryption scheme [12] and Peng and Li's ID-based threshold signcryption scheme [26] in Figure 1. We consider the costly operations which include point scalar multiplications in G_1 (G_1 Mul), exponentiations in G_2 (G_2 Exp), and pairing operations (Pairing). From Figure 1, we can see that both Duan et al.'s scheme and Peng and Li's scheme need $3t+4$ pairing computations and our scheme only needs $2t+4$ pairing computations. Since the pairing computation is the most time consuming, the proposed scheme is more efficient than Duan et al.'s scheme and Peng and Li's scheme.

VI. Conclusions

We have proposed an efficient and provably secure ID-based threshold signcryption scheme based on the bilinear pairings. We proved that our scheme satisfies the confidentiality, the unforgeability, and the robustness. As compared with two previously proposed schemes (Duan et al.'s scheme [12] and Peng and Li's scheme [26]) which need $3t+4$ pairing computations, our scheme is more efficient since it only needs $2t+4$ pairing computations.

Acknowledgment

This work is supported by the National Natural Science Foundation of China under contract no. 60673075, the National High Technology Research and Development Program of China (863) under contract no. 2006AA01Z428, the Key Laboratory of Computer Networks and Information Security of Xidian University under contract no. 2008CNIS-02, and Youth Science and Technology Foundation of UESTC.

References

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

[2] J. Baek, R. Steinfeld, and Y. Zheng, "Formal proofs for the security of signcryption", In Proc. Public Key Cryptography-PKC 2002, LNCS 2274, pp. 80–98, Springer-Verlag, 2002.

[3] J. Baek and Y. Zheng, "Identity-based threshold signature scheme from the bilinear pairings", In Proc. International Conference on Information Technology: Coding and Computing-ITCC'04, pp. 124–128, Las Vegas, Nevada, USA, 2004.

[4] F. Bao and R.H. Deng, "A signcryption scheme with signature directly verifiable by public key", In Proc. Public Key Cryptography-PKC'98, LNCS 1431, pp. 55–59, Springer-Verlag, 1998.

[5] 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 Proc. Advances in Cryptology-ASIACRYPT 2005, LNCS 3788, pp. 515–532, Springer-Verlag, 2005.

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

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

[8] X. Cheng, J. Liu, and X. Wang, "An identity-based signature and its threshold version", In Proc. 19th International Conference on Advanced Information Networking and Applications-AINA'05, pp. 973–977, Taipei, Taiwan, 2005.

[9] 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 Proc. Information Security and Cryptology-ICISC 2003, LNCS 2971, pp. 352–369, Springer-Verlag, 2004.

[10] Y. Desmedt, "Society and group oriented cryptography: a new concept", In Proc. Advances in Cryptography-CRYPTO'87, LNCS 293, pp. 120–127, Springer-Verlag, 1987.

[11] Y. Desmedt and Y. Frankel, "Shared generation of authenticators and signatures", In Proc. Advances in Cryptography-CRYPTO'91, LNCS 576, pp. 457–469, Springer-Verlag, 1991.

[12] S. Duan, Z. Cao, and R. Lu, "Robust ID-based threshold signcryption scheme from pairings", In Proc. 2004 International Conference on Information security, pp. 33–37, Shanghai, China, 2004.

[13] A. Fiat and A. Shamir, "How to prove yourself: practical solutions to identification and signature problems" In Proc. Advances in Cryptology-CRYPTO'86, LNCS 263, pp. 186–194, Springer-Verlag, 1986.

[14] C. Gamage, J. Leiwo, and Y. Zheng, "Encrypted message authentication by firewalls", In Proc. Public Key Cryptography-PKC'99, LNCS 1560, pp. 69–81, Springer-Verlag, 1999.

[15] L. Guillou and J.J. Quisquater, A "Paradoxical" Identity-based signature scheme resulting from zero-knowledge", In Proc. Advances in Cryptology-CRYPTO'88, LNCS 403, pp. 216–231, Springer-Verlag, 1988.

[16] X. Huang, W. Susilo, Y. Mu, and F. Zhang, "Identity-based ring signcryption schemes: cryptographic primitives for preserving privacy and authenticity in the ubiquitous world", In Proc. 19th International Conference on Advanced Information Networking and Applications-AINA 2005, pp. 649–654, Taipei, Taiwan, 2005.

[17] X. Li and K. Chen, "Identity based proxy-signcryption scheme from pairings", In Proc. 2004 IEEE International Conference on Services Computing, pp. 494–497, Shanghai, China, 2004.

[18] F. Li, Y. Hu, and C. Zhang, "An identity-based signcryption scheme for multi-domain ad hoc networks", In Proc. Applied Cryptography and Network Security-ACNS 2007, LNCS 4521, pp. 373–384, Springer-Verlag, 2007.

[19] B. Libert and J.J. Quisquater, "A new identity based signcryption schemes from pairings", In Proc. 2003 IEEE information theory workshop, pp. 155–158, Paris, France, 2003.

[20] B. Libert and J.J. Quisquater, "Efficient signcryption with key privacy from gap Diffie-Hellman groups", In Proc. Public Key Cryptography-PKC 2004, LNCS 2947, pp. 187–200, Springer-Verlag, 2004.

[21] C. Ma, K. Chen, D. Zheng, and S. Liu, "Efficient and proactive threshold signcryption", In Proc. Information Security Conference-ISC 2005, LNCS 3650, pp. 233–243, Springer-Verlag, 2005.

[22] J. Malone-Lee, "Identity based signcryption", Cryptology ePrint Archive, Report 2002/098, 2002. Available from: http://eprint.iacr.org/2002/098.

[23] J. Malone-Lee and W. Mao, "Two birds one stone: signcryption using RSA", In Proc. Topics in Cryptology-CT-RSA 2003, LNCS 2612, pp. 211–226, Springer-Verlag, 2003.

[24] Y. Mu and V. Varadharajan, "Distributed Signcryption", In Proc.

Progress in Cryptology-INDOCRYPT 2000, LNCS 1977, pp. 155–164, Springer-Verlag, 2000.

[25] H. Petersen and M. Michels, "Cryptanalysis and improvement of signcryption schemes", IEE Proceedings-Computers and Digital Techniques, Vol.145, No. 2, pp. 149–151, 1998.

[26] C. Peng and X. Li, "An identity-based threshold signcryption scheme with semantic security", In Proc. Computational Intelligence and Security-CIS 2005, LNAI 3802, pp. 173–179, Springer-Verlag, 2005.

[27] R.L. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems", Communications of the ACM, Vol. 21, No. 2, pp. 120–126, 1978.

[28] M. Seo and K. Kim, "Electronic funds transfer protocol using domain-verifiable signcryption scheme", In Proc. Information Security and Cryptology-ICISC'99, LNCS 1787, pp. 269–277, Springer-Verlag, 1999.

[29] A. Shamir, "Identity-based cryptosystems and signature schemes", In Proc. Advances in Cryptology-CRYPTO'84, LNCS 196, pp. 47–53, Springer-Verlag, 1984.

[30] J.B. Shin, K. Lee, and K. Shim, "New DSA-verifiable signcryption schemes", In Proc. Information Security and Cryptology-ICISC 2002, LNCS 2587, pp. 35–47, Springer-Verlag, 2003.

[31] G. Yang, D.S. Wong, and X. Deng, "Analysis and improvement of a signcryption scheme with key privacy", In Proc. Information Security Conference-ISC 2005, LNCS 3650, pp. 218–232, Springer-Verlag, 2005.

[32] T.H. Yuen and V.K. Wei, "Fast and proven secure blind identity-based signcryption from pairings", In Proc. Topics in Cryptology-CT-RSA 2005, LNCS 3376, pp. 305–322, Springer-Verlag, 2005.

[33] Y. Zheng, "Digital signcryption or how to achieve cost (signature & encryption) $\ll$ cost (signature) + cost(encryption)", In Proc. Advances in Cryptology-CRYPTO'97, LNCS 1294, pp. 165–179, Springer-Verlag, 1997.

[34] Y. Zheng and H. Imai, "How to construct efficient signcryption schemes on elliptic curves", Information Processing Letters, Vol. 68, No. 5, pp. 227–233, 1998.