Computers and Mathematics with Applications 60 (2010) 1850–1858

Contents lists available at ScienceDirect

Computers and Mathematics with Applications

journal homepage: www.elsevier.com/locate/camwa

Efficient proxy signcryption scheme with provable CCA and CMA security

Han-Yu Lin [a], Tzong-Sun Wu [b,*], Shih-Kun Huang [a], Yi-Shiung Yeh [a]

[a] Department of Computer Science, National Chiao Tung University, Hsinchu, 300, Taiwan

[b] Department of Computer Science and Engineering, National Taiwan Ocean University, 2, Beining Road, Keelung, 202, Taiwan

ARTICLE INFO

Article history:
Received 25 December 2009
Received in revised form 24 May 2010
Accepted 12 July 2010

Keywords:
Proxy
Signcryption
Bilinear pairing
Warrant
Public key encryption

ABSTRACT

For facilitating the confidential transaction with delegation such as on-line proxy auction and business contract signing by an authorized proxy, we propose an efficient proxy signcryption scheme from pairings. Our scheme allows an original signer to delegate his signing power to a proxy one such that the latter can signcrypt a plaintext on behalf of the former. The signcrypted message can only be decrypted by a designated recipient who is also responsible for verifying the recovered proxy signature. To deal with a later dispute over repudiation, the designated recipient can easily announce the ordinary proxy signature for public verification without extra computational efforts. To guarantee the realistic applicability, we demonstrate that our scheme outperforms previous works in terms of functionalities and computational efficiency. Moreover, the security requirement of confidentiality against indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA2) and that of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) are proved in random oracle models.

1. Introduction

The first public key cryptosystem was proposed by Diffie and Hellman [1] in 1976. Since then, public key systems have been widely used in many kinds of fields. In essence, the public key encryption and the digital signature scheme [2,3] are two commonly applied techniques for assuring the communication security. Nevertheless, with the coming of gradually complex business applications, such as the proxy delegation, the on-line credit card transaction, the contract signing, etc., traditional cryptographic techniques are not sufficient to deal with these specific application requirements.

In 1996, Mambo et al. [4,5] introduced the notion of proxy signature. A proxy signature scheme allows the proxy signer authorized by the original signer to generate a proxy signature on behalf on the latter such that everyone can verify the proxy signature. It can be seen that proxy signature schemes effectively solve the problem of proxy delegation in an organization. Generally speaking, the proxy delegation can be categorized into four sorts including the full delegation [4,5], the partial delegation [4,5], the delegation by warrant [6,7] and the partial delegation by warrant [8]. Among these delegations, it is believed that the last one is a better alternative, since it inherits the merits of partial delegation and delegation by warrant. Besides, certifying the warrant and verifying the signature can be simultaneously carried out within one step.

Consider the applications where we have to simultaneously fulfill the security requirements of confidentiality, integrity, authenticity and non-repudiation [9,10], such as the on-line credit card transaction and the contract signing. In 1997, Zheng [11] proposed a so-called signcryption scheme which is suitable for these applications. A signcryption scheme only allows the designated recipient to verify the signer's signature instead of everyone for the purpose of confidentiality.

* Corresponding author. Tel.: +886 2 2462 2192x6622; fax: +886 2 2462 3249.
E-mail address: ibox456@gmail.com (T.-S. Wu).

0898-1221/$ – see front matter © 2010 Elsevier Ltd. All rights reserved.
doi:10.1016/j.camwa.2010.07.015

In 1998, Petersen and Michels [12] also proposed another signcryption variant modified from an authenticated encryption scheme. Yet, He and Wu [13] pointed out that their scheme is vulnerable to the forgery attack. To deal with the later dispute that the signer repudiates his generated signature, Zheng [14] introduced an arbitration mechanism by using the zero-knowledge protocol [15,16]. However, the arbitration mechanism is inefficient for that it will increase extra computation efforts and communication overheads. In 1998, Bao and Deng [17] addressed an efficient way to handle the repudiation dispute. Their scheme enables the designated recipient to convert the signcrypted message into an ordinary signature for the public verification without imposing extra burdens on the computation and the communication cost. In 2002, Baek et al. [18] introduced the formal security proof model for a signcryption scheme in the random oracle model. The next year, Boyen [19] proposed a provably secure identity-based signcryption scheme with ciphertext anonymity. In 2005, Hwang et al. [20] proposed an elliptic curve based signcryption scheme with forward secrecy for facilitating the gradually widely used mobile applications.

Recently, bilinear pairing cryptosystems from elliptic curves have received great attention in cryptography [21–25]. Many researchers [26–33] also dedicate themselves to the construction of pairing-based signcryption schemes. Some [28–30,33] of them are constructed to handle the issue of proxy delegation. Such schemes have realistic applicability and are suitable for the confidential transactions, e.g., on-line proxy auctions or contract signing by an authorized proxy signer. Consider the application such as a bank account owned by a busy boss. To withdraw money from his saving account, the boss must sign a withdrawal slip which can only be verified by the bank teller. In case that this boss is unable to sign personally, he can delegate his signing power to a proxy signer who can legitimately conduct transactions on behalf of him. However, the above mentioned proxy signcryption schemes cannot provide strong and complete security proofs in either random oracle or standard model.

1.1. Our contribution

Elaborating on the merits of signcryption schemes and proxy signature schemes, we adopt the partial delegation with warrant to propose an efficient proxy signcryption scheme based on bilinear pairings in this paper. The proposed scheme only requires four bilinear pairing operations for the entire protocol, which benefits to practical implementation. Consider the realistic situation that an original signer might delegate his signing power to different proxy signers for various transactions. In this case, our scheme with optimal computational efficiency for the original signer would be a better alternative, since our delegation process involves no bilinear pairing computation which is regarded as the most time-consuming operation. When the case of a later dispute over repudiation occurs, the designated recipient is capable of announcing the ordinary proxy signature to convince anyone of the proxy signer's dishonesty. Note that the conversion takes no extra computational efforts, since the ordinary proxy signature will be derived during the decryption and verification process. Compared with related works, the proposed scheme not only has lower computational costs, but also provides better functionalities. Additionally, we also prove that the proposed scheme achieves the IND-CCA2 and the EF-CMA security in the random oracle model.

2. Preliminaries

For facilitating the reader with the following description, in this section, we first state involved parties and then review some security notions.

2.1. Involved parties

A proxy signcryption scheme mainly has three involved parties: an original signer, a proxy signer and a designated recipient. All parties are probabilistic polynomial-time Turing machines (PPTM). The original signer delegates his signing power to the proxy signer by issuing the proxy credential. After that, the latter can generate a signcrypted message on behalf of the former, and sends it to the designated recipient. Finally, the designated recipient decrypts the message and verifies the proxy signature.

2.2. Security notions

Correctness

A proxy signcryption scheme is correct if the proxy signer can generate a valid signcrypted message on behalf of the original signer and only the designated recipient is capable of decrypting it and verifying the proxy signature.

Bilinear pairings [34]

Let $(\boldsymbol{G}_1, +)$ and $(\boldsymbol{G}_2, \times)$ denote two groups of the same prime order q and $e : \boldsymbol{G}_1 \times \boldsymbol{G}_1 \to \boldsymbol{G}_2$ be a bilinear map which satisfies the following properties:

(i) *Bilinearity*:

$$e(P_1 + P_2, Q) = e(P_1, Q)e(P_2, Q);$$
$$e(P, Q_1 + Q_2) = e(P, Q_1)e(P, Q_2).$$

(ii) *Non-degeneracy*: If P is a generator of $\boldsymbol{G}_1$, then $e(P, P)$ is a generator of $\boldsymbol{G}_2$.
(iii) *Computability*: Given $P, Q \in \boldsymbol{G}_1$, the value of $e(P, Q)$ can be efficiently computed by a polynomial-time algorithm.

Bilinear Diffie–Hellman Problem; BDHP

The BDHP is, given an instance $(P, A, B, C) \in \boldsymbol{G}_1^4$ where P is a generator, $A = aP$, $B = bP$ and $C = cP$ for some $a, b, c \in Z_q^*$, to compute $e(P, P)^{abc} \in \boldsymbol{G}_2$.

Bilinear Diffie–Hellman (BDH) Assumption

For every probabilistic polynomial-time algorithm $\mathcal{A}$, every positive polynomial $Q(\cdot)$ and all sufficiently large k, the algorithm $\mathcal{A}$ can solve the BDHP with an advantage at most $1/Q(k)$, i.e.,

$$\Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc};\ a, b, c \leftarrow Z_q^*,\ (P, aP, bP, cP) \leftarrow \boldsymbol{G}_1^4] \leq 1/Q(k).$$

The probability is taken over the uniformly and independently chosen instance and over the random choices of $\mathcal{A}$.

Definition 1. The (t, ε)-BDH assumption holds if there exists no polynomial-time adversary that can solve the BDHP in time at most t and with the advantage ε.

3. Formal model of the proposed scheme

This section addresses the formal model of our proposed proxy signcryption scheme and its security model.

3.1. Proxy signcryption scheme

The proposed scheme consists of the following algorithms:

- *Setup:* Taking as input 1^k where k is a security parameter, the algorithm generates the system's public parameters *params*.
- *Proxy-Credential-Generation (PCG):* The PCG algorithm takes as input the private key of original signer and outputs a corresponding proxy credential for the proxy signer.
- *Signcrypted-Message-Generation (SMG):* The SMG algorithm takes as input a plaintext m, a proxy credential, the public key of designated recipient and the private key of proxy signer. It generates a corresponding signcrypted message δ.
- *Signature-Recovery-and-Verification (SRV):* The SRV algorithm takes as input a signcrypted message δ, the private key of designated recipient and the public keys of original and proxy signers. It outputs a plaintext m and its converted ordinary proxy signature Ω if the signcrypted message δ is valid. Otherwise, an error symbol ¶ is returned.

3.2. Security model

Two crucial security requirements of the proposed scheme are message confidentiality and unforgeability. We define two security models for these notions as follows:

Definition 2 (*Confidentiality*)**.** A proxy signcryption scheme is said to achieve the security requirement of confidentiality against indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA2) if there is no probabilistic polynomial-time adversary $\mathcal{A}$ with non-negligible advantage in the following game played with a challenger $\mathcal{B}$:

Setup: The challenger $\mathcal{B}$ first runs the Setup (1^k) algorithm and sends the system's public parameters *params* to the adversary $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ can issue several kinds of queries adaptively, i.e., each query might be based on the result of previous queries:

- *Proxy-Credential-Generation (PCG) queries:* $\mathcal{A}$ issues a PCG query with respect to the target proxy signer. $\mathcal{B}$ returns the corresponding warrant and its proxy credential.
- *Signcrypted-Message-Generation (SMG) queries:* $\mathcal{A}$ chooses a plaintext m and then $\mathcal{B}$ outputs the corresponding signcrypted message δ to $\mathcal{A}$.
- *Signature-Recovery-and-Verification (SRV) queries:* On receiving a signcrypted message δ with its warrant sent by $\mathcal{A}$, $\mathcal{B}$ returns a plaintext m and its converted proxy signature Ω if the signcrypted message δ is valid. Otherwise, an error symbol ¶ is returned.

Challenge: The adversary $\mathcal{A}$ produces two plaintexts, m_0 and m_1, of the same length. $\mathcal{B}$ flips a coin $\lambda \leftarrow \{0, 1\}$ and generates a signcrypted message δ^* for m_λ. The signcrypted message δ^* is then delivered to $\mathcal{A}$ as a target challenge.

Phase 2: The adversary $\mathcal{A}$ can issue new queries as those in Phase 1, except the SRV query for the target challenge δ^*.

Guess: At the end of the game, $\mathcal{A}$ outputs a bit λ'. The adversary $\mathcal{A}$ wins this game if $\lambda' = \lambda$. We define $\mathcal{A}$'s advantage as $\mathrm{Adv}(\mathcal{A}) = |\Pr[\lambda' = \lambda] - 1/2|$.

Definition 3 (*Unforgeability*)**.** A proxy signcryption scheme is said to achieve the security requirement of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) if there exists no probabilistic polynomial-time adversary $\mathcal{A}$ with non-negligible advantage in the following game played with a challenger $\mathcal{B}$:

Setup: $\mathcal{B}$ first runs the Setup (1^k) algorithm and sends the system's public parameters *params* to the adversary $\mathcal{A}$.

Phase 1: The adversary $\mathcal{A}$ adaptively issues PCG and SMG queries as those in Phase 1 of Definition 2.

Forgery: Finally, $\mathcal{A}$ arbitrarily chooses a plaintext m and produces a signcrypted message δ^* which is not outputted by the SMG query. The adversary $\mathcal{A}$ wins if δ^* is valid.

4. The proposed scheme

We propose an efficient proxy signcryption scheme based on BDHP in this section. Our scheme has optimal computational efficiency for the original signer, since the delegation process involves no bilinear pairing operation which is considered the most time-consuming computation. As for the entire protocol, only four pairing operations are required. Details of each algorithm are described below:

- *Setup*: Taking as input 1^k, the system authority (SA) selects two groups $(\boldsymbol{G}_1, +)$ and $(\boldsymbol{G}_2, \times)$ of the same prime order q with $|q| = k$. Let P be a generator of order q over $\boldsymbol{G}_1$, $e : \boldsymbol{G}_1 \times \boldsymbol{G}_1 \to \boldsymbol{G}_2$ a bilinear pairing and $h_1 : \{0,1\}^k \times \boldsymbol{G}_1 \to Z_q$, $h_2 : \boldsymbol{G}_1 \to \boldsymbol{G}_1$ and $h_3 : \boldsymbol{G}_2 \times \boldsymbol{G}_1 \to \{0,1\}^k$ collision resistant hash functions. The system publishes $params = \{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e, h_1, h_2, h_3\}$. Each user U_i chooses his private key $x_i \in_R Z_q$ and computes the corresponding public one as $Y_i = x_i P$.
- *Proxy-Credential-Generation (PCG)*: Let U_o be an original signer delegating his signing power to a proxy signer U_p. U_o first chooses an integer $d \in Z_q$ to compute

$$N = \mathrm{d}P, \tag{1}$$

$$\sigma = x_o + d(m_w) \bmod q, \tag{2}$$

  where m_w is the warrant consisting of the identifiers of original signer, proxy signer and designated recipient, the delegation duration and so on. The proxy credential (σ, N, m_w) are then sent to U_p. Upon receiving (σ, m_w, N), U_p first checks its validity by verifying whether

$$\sigma P = Y_o + m_w N. \tag{3}$$

  If it does not hold, (σ, m_w, N) is requested to be sent again.
- *Signcrypted-Message-Generation (SMG)*: For signcrypting a chosen plaintext $m \in_R \{0,1\}^k$ on behalf of the original signer U_o, U_p chooses $r \in_R Z_q$ to compute

$$R = rP, \tag{4}$$

$$S = r(h_1(m, R) + x_p + \sigma)^{-1} P, \tag{5}$$

$$V = e(h_2(\sigma Y_v), x_p Y_v), \tag{6}$$

$$X = E_V(S), \tag{7}$$

$$Y = h_3(V, R) \oplus m, \tag{8}$$

  and then delivers the warrant m_w and the signcrypted message $\delta = (R, X, Y, N)$ to the designated recipient U_v, where E_V denotes the symmetric encryption function with key V.
- *Signature-Recovery-and-Verification (SRV)*: Upon receiving (R, X, Y, N), U_v first computes

$$V = e(h_2(x_v(Y_o + m_w N)), x_v Y_p), \tag{9}$$

  to recover the plaintext m as

$$m = h_3(V, R) \oplus Y \tag{10}$$

  and checks the redundancy embedded in m. U_v further computes S as

$$S = D_V(X) \tag{11}$$

  and verifies the proxy signature by checking if

$$e(h_1(m, R)P + Y_p + Y_o + m_w N, S) = e(P, R). \tag{12}$$

  Note that D_V is the symmetric decryption function with key V.

Since the converted proxy signature $\Omega = (S, R, N)$ is derived during the verification process, the designated recipient U_v can easily announce it together with (m, m_w) in case of a later dispute over repudiation. Accordingly, any third party can check Eq. (12) to realize the proxy signer's dishonesty.

5. Security proof and comparison

In this section, we first analyze the security of our proposed scheme and then make a comparison with some previous works.

5.1. Security proof

We demonstrate that the proposed scheme is correct and achieves the IND-CCA2 and the EF-CMA security in the random oracle model. We first show that the verification of Eq. (3) works correctly. From the left-hand side of Eq. (3), we have

$$\begin{aligned}\sigma P &= (x_o + d(m_w))P \quad \text{(by Eq. (2))}\\ &= x_o P + d(m_w)P\\ &= Y_o + m_w N \quad \text{(by Eq. (1))}\end{aligned}$$

which leads to the right-hand side of Eq. (3).

Upon receiving (R, X, Y, N) with the warrant m_w, the designated recipient can correctly recover the plaintext and verify the embedded proxy signature with Eq. (12). From the left-hand side of Eq. (12), we have

$$\begin{aligned}e(h_1(m,R)P + Y_p + Y_o + m_w N, S) &= e(h_1(m,R)P + Y_p + Y_o + m_w N, r(h_1(m,R) + x_p + \sigma)^{-1}P) \quad \text{(by Eq. (5))}\\ &= e((h_1(m,R) + x_p + x_o + d(m_w))P, r(h_1(m,R) + x_p\\ &\quad + x_o + d(m_w))^{-1}P) \quad \text{(by Eqs. (1) and (2))}\\ &= e(P, rP)\\ &= e(P, R) \quad \text{(by Eq. (4))}\end{aligned}$$

which leads to the right-hand side of Eq. (12).

We then prove that the proposed scheme achieves the IND-CCA2 and the EF-CMA security in the random oracle model as Theorems 1 and 2, respectively.

Theorem 1 (*Proof of Confidentiality*). *The proposed scheme is* $(t, q_{h1}, q_{h2}, q_{h3}, q_{\text{PCG}}, q_{\text{SMG}}, q_{\text{SRV}}, \varepsilon)$*-secure against indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA2) in the random oracle model if there is no probabilistic polynomial-time adversary that can* (t', ε')*-break the BDHP, where*

$$\varepsilon' \geq (q_{h3}^{-1})(2\varepsilon - q_{\text{SRV}}(2^{-k})),$$
$$t' \approx t + t_\lambda(q_{\text{SMG}} + 2q_{\text{SRV}}).$$

Here t_λ *is the time for performing one bilinear pairing operation.*

Proof. Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can $(t, q_{h1}, q_{h2}, q_{h3}, q_{\text{PCG}}, q_{\text{SMG}}, q_{\text{SRV}}, \varepsilon)$-break the proposed scheme with non-negligible advantage ε under the adaptive chosen-ciphertext attack after running in time at most t and asking at most $q_{hi}h_i$ random oracle (for $i = 1$ to 3), q_{PCG} PCG queries, q_{SMG} SMG queries and q_{SRV} SRV queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the BDHP by taking $\mathcal{A}$ as a subroutine. Let all involved parties and parameters be defined the same as those in Section 4. The objective of $\mathcal{B}$ is to obtain $e(P, P)^{abc}$ by taking (P, aP, bP, cP) as inputs. In this proof, $\mathcal{B}$ simulates a challenger to $\mathcal{A}$ in the following game.

Setup: The challenger $\mathcal{B}$ runs the Setup (1^k) algorithm, sets $(Y_p = aP, Y_v = bP, Y_o = wP$ where $w \in_R Z_q)$ and sends the system's public parameters $params = \{\boldsymbol{G}_1, \boldsymbol{G}_2, q, P, e\}$ along with (Y_o, Y_p, Y_v) to the adversary $\mathcal{A}$

Phase 1: $\mathcal{A}$ issues the following kinds of queries adaptively:

- h_1 *query*: When $\mathcal{A}$ makes an h_1 query of $h_1(m, R)$, $\mathcal{B}$ first checks the h_1-list for a matched entry. Otherwise, $\mathcal{B}$ randomly chooses $v_1 \in Z_q$ and stores the entry (m, R, v_1) into h_1-list. Finally, $\mathcal{B}$ returns v_1 as a result.
- h_2 *query*: When $\mathcal{A}$ makes an h_2 query of $h_2(\sigma Y_v)$, $\mathcal{B}$ first checks the h_2-list for a matched entry. Otherwise, $\mathcal{B}$ randomly chooses an integer $v_2 \in Z_q$ to compute $V_2 = v_2 P$, stores the entry $(\sigma Y_v, v_2, V_2)$ and returns V_2 as a result.
- h_3 *query:* When $\mathcal{A}$ makes an h_3 query of $h_3(V, R)$, $\mathcal{B}$ first checks the h_3-list for a matched entry. Otherwise, $\mathcal{B}$ randomly chooses $v_3 \in \{0, 1\}^k$, stores the entry (V, R, v_3) and returns v_3 as a result.
- *PCG queries*: When $\mathcal{A}$ makes a PCG query for the proxy signer U_p, $\mathcal{B}$ directly runs the PCG algorithm with his selected private key w and returns the corresponding (σ, m_w, N) as a result.
- *SMG queries*: When $\mathcal{A}$ makes an SMG query for a plaintext m, $\mathcal{B}$ first obtains the corresponding proxy credential (σ, m_w, N) by making the PCG query, randomly chooses fresh s, $v_1 \in Z_q$ to compute

  $$S = sP,$$
  $$R = sv_1 P + s(aP) + s(wP) + sm_w N,$$

  such that $h_1(m, R)$ has never been queried before and then defines $v_1 = h_1(m, R)$. $\mathcal{B}$ further makes an $h_2(\sigma(bP))$ query to get (v_2, V_2), computes $V = e(v_2(aP), (bP))$ and (X, Y) as Eqs. (7) and (8), respectively. The signcrypted message $\delta = (R, X, Y, N)$ and m_w are returned as a result.

– *SRV queries*: When $\mathcal{A}$ submits a signcrypted message $\delta = (R, X, Y, N)$ with m_w, $\mathcal{B}$ first searches the h_3-list for possible V using R as an index and computes $S = D_V(X)$ and $m = v_3 \oplus Y$. If one satisfies $e(h_1(m, R)P + Y_p + Y_o + m_w N, S) = e(P, R)$, $\mathcal{B}$ returns m and its converted proxy signature $\Omega = (S, R, N)$. Otherwise, $\mathcal{B}$ directly returns the symbol ¶ as a result to signal that δ is invalid.

Challenge: $\mathcal{A}$ generates two plaintexts, m_0 and m_1, of the same length and sends them to the challenger $\mathcal{B}$. Then $\mathcal{B}$ flips a coin $\lambda \leftarrow \{0, 1\}$ and generates a signcrypted message $\delta^* = (R^*, X^*, Y^*, N^*)$ for m_λ as follows:

Step 1 Make a PCG query to obtain (σ^*, m_w^*, N^*);
Step 2 Choose $X^* \in_R \boldsymbol{G}_1$, $v_3^* \in_R \{0, 1\}^k$ and s, $v_1^*, z \in_R Z_q$;
Step 3 Compute $S^* = sP$ and $R^* = sv_1^* P + s(aP) + s(wP) + sm_w^* N^*$;
Step 4 Store the entry (m_λ, R^*, v_1^*) into h_1-list, i.e., implicitly define $h_1(m_\lambda, R^*) = v_1^*$;
Step 5 Store the entry $(\sigma^* Y_v, \text{null}, z(cP))$ into h_2-list, i.e., implicitly define $h_2(\sigma^* Y_v) = z(cP)$;
Step 6 Compute $Y^* = v_3^* \oplus m_\lambda$, i.e., implicitly define $h_3(V^*, R^*) = v_3^*$ where $V^* = e(z(cP), a(bP))$ and $\mathcal{B}$ does not know it.

Phase 2: $\mathcal{A}$ issues new queries as those stated in Phase 1. It is not allowed to request an SRV query for the target challenge δ^*.

Analysis of the game: Since $\mathcal{B}$ has set $Y_o = wP$ where $w \in_R Z_q$, he can always return a valid proxy credential for $\mathcal{A}$'s PCG query. For each SMG query, $\mathcal{B}$ also simulates a computationally indistinguishable signcrypted message by manipulating the h_1 random oracle. Therefore, we refer the simulations of PCG and ACG queries to be perfect. Then we evaluate the simulation of SRV queries. It is possible that an SRV query returns the error symbol ¶ for some valid δ if the corresponding h_3 random oracle has never been asked before. Let SRV_ERR be the event that an SRV query returns ¶ for a valid δ during the entire game, SM-V an event that a signcrypted message δ submitted by $\mathcal{A}$ is valid, and QH_3 the event that the corresponding h_3 oracle has ever been asked before. Then we can express the error probability of any SRV query as

$$\Pr[\text{SM-V} | \neg \text{QH}_3] \leq 2^{-k}.$$

Since $\mathcal{A}$ is allowed to make at most q_{SRV} SRV queries, we can further express the probability of SRV_ERR as

$$\Pr[\text{SRV\_ERR}] \leq q_{\text{SRV}}(2^{-k}). \tag{13}$$

Moreover, in the challenge phase, $\mathcal{B}$ has returned a simulated $\delta^* = (R^*, X^*, Y^*, N^*)$ where $h_2(R^*) = z(cP)$, which implies the shared secret V^* is implicitly defined as $V^* = e(z(cP), a(bP))$. Let GP be the event that the entire simulation game is perfect. In Phase 2, if the adversary $\mathcal{A}$ never makes the query of $h_3(V^*, R^*)$, the entire simulation game could be perfect. We denote the event that $\mathcal{A}$ does ask an $h_3(V^*, R^*)$ random oracle in Phase 2 by QH_3^*. When the entire simulation game is perfect, $\mathcal{A}$ gains no advantage in guessing λ due to the randomness of the output of the random oracle, i.e.,

$$\Pr[\lambda' = \lambda | GP] = 1/2. \tag{14}$$

From the expression of $\Pr[\lambda' = \lambda]$, we can derive that

$$\begin{aligned}
\Pr[\lambda' = \lambda] &= \Pr[\lambda' = \lambda | \text{GP}] \Pr[\text{GP}] + \Pr[\lambda' = \lambda | \neg \text{GP}] \Pr[\neg \text{GP}] \\
&\leq (1/2) \Pr[\text{GP}] + \Pr[\neg \text{GP}] \quad \text{(by Eq. (14))} \\
&= (1/2)(1 - \Pr[\neg \text{GP}]) + \Pr[\neg \text{GP}] \\
&= (1/2) + (1/2) \Pr[\neg \text{GP}].
\end{aligned} \tag{15}$$

Besides, we can also derive that

$$\begin{aligned}
\Pr[\lambda' = \lambda] &\geq \Pr[\lambda' = \lambda | \text{GP}] \Pr[\text{GP}] \\
&= (1/2)(1 - \Pr[\neg \text{GP}]) \\
&= (1/2) - (1/2) \Pr[\neg \text{GP}].
\end{aligned} \tag{16}$$

By combining inequalities (15) and (16), we obtain that

$$|\Pr[\lambda' = \lambda] - 1/2| \leq (1/2) \Pr[\neg \text{GP}]. \tag{17}$$

Recall that in Definition 2, $\mathcal{A}$'s advantage is defined as $\text{Adv}(A) = |\Pr[\lambda' = \lambda] - 1/2|$. By assumption, $\mathcal{A}$ has non-negligible probability ε to break the proposed scheme. We therefore have

$$\begin{aligned}
\varepsilon &= |\Pr[\lambda' = \lambda] - 1/2| \\
&\leq (1/2) \Pr[\neg \text{GP}] \quad \text{(by Eq. (17))} \\
&= (1/2)(\Pr[\text{QH}_3{}^* \vee \text{SRV_ERR}]) \\
&\leq (1/2)(\Pr[\text{QH}_3{}^*] + \Pr[\text{SRV_ERR}]).
\end{aligned}$$

Rewriting the above inequality, we get

$$\begin{aligned}\Pr[\mathrm{QH}_3{}^*] &\geq 2\varepsilon - \Pr[\mathrm{SRV\_ERR}] \\ &\geq 2\varepsilon - q_{\mathrm{SRV}}(2^{-k}) \quad \text{(by Eq. (13))}.\end{aligned}$$

When the event QH_3^* happens, we claim that $V^* = e(z(cP), a(bP))$ will be left in some entry of the h_3-list. Consequently, $\mathcal{B}$ would have non-negligible probability

$$\varepsilon' \geq (q_{h3}^{-1})(2\varepsilon - q_{\mathrm{SRV}}(2^{-k}))$$

to solve the BDHP by outputting $V^{*z^{-1}}$. The computational time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(q_{\mathrm{SMG}} + 2q_{\mathrm{SRV}})$. □

Theorem 2 (*Proof of Unforgeability*)**.** *The proposed scheme is $(t, q_{h1}, q_{h2}, q_{h3}, q_{\mathrm{PCG}}, q_{\mathrm{SMG}}, \varepsilon)$-secure against existential forgery under adaptive chosen-message attacks (EF-CMA) in the random oracle model if there is no probabilistic polynomial-time adversary that can (t', ε')-break the BDHP, where*

$$\varepsilon' \geq (\varepsilon - (q_{h2} + 1)/2^k)/(q_{h2}q_{h3}),$$
$$t' \approx t + t_\lambda(q_{\mathrm{SMG}}).$$

Here t_λ is the time for performing one bilinear pairing operation.

Proof. Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can $(t, q_{h1}, q_{h2}, q_{h3}, q_{\mathrm{PCG}}, q_{\mathrm{SMG}}, \varepsilon)$-break the proposed scheme with non-negligible advantage ε under the adaptive chosen-message attack after running in time at most t and asking at most $q_{hi}h_i$ random oracle (for $i = 1$–3), q_{PCG} PCG and q_{SMG} SMG queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the BDHP by taking $\mathcal{A}$ as a subroutine. Let all involved parties and notations be defined the same as those in Section 4. The objective of $\mathcal{B}$ is to obtain $e(P, P)^{abc}$ by taking (P, aP, bP, cP) as inputs. In this proof, $\mathcal{B}$ simulates a challenger to $\mathcal{A}$ in the following game.

Setup: The challenger $\mathcal{B}$ runs the Setup (1^k) algorithm, sets $(Y_p = aP, Y_v = bP, Y_o = wP$ where $w \in_R Z_q)$ and sends the system's public parameters $params = \{G_1, G_2, q, P, e\}$ along with (Y_o, Y_p, Y_v) to the adversary $\mathcal{A}$.

Phase 1: $\mathcal{A}$ adaptively asks h_1, h_2, h_3 random oracles, PCG and SMG queries as those defined in Theorem 1. Note that in the jth h_2 random oracle, $\mathcal{B}$ directly returns $z(cP)$ where $z \in_R Z_q$.

Forgery: Finally, $\mathcal{A}$ outputs a signcrypted message $\delta^* = (R^*, X^*, Y^*, N^*)$ and m_w^* for his arbitrarily chosen plaintext m^*. If δ^* is valid, $\mathcal{A}$ wins the game.

Analysis of the game: According to the analyses of Theorem 1, we know that the simulation of each PCG or ACG query is regarded as perfect. Furthermore, $\mathcal{B}$ answers each h_i random oracle with a computationally indistinguishable value without collision. Let SM-V be the event that the forged δ^* is valid. QH_2 and QH_3 separately denote that $\mathcal{A}$ has ever asks the corresponding h_2 and h_3 random oracles. The probability that $\mathcal{A}$ can guess the correct random value without asking h_2 or h_3 random oracle is not greater than 2^{-k}. Since $\mathcal{A}$ has non-negligible advantage ε to break the proposed scheme under adaptive chosen-message attacks, we have

$$\begin{aligned}\varepsilon &= \Pr[\text{SM-V}] \\ &= \Pr[\text{SM-V}|\mathrm{QH}_2] + \Pr[\text{SM-V}|\neg\mathrm{QH}_2] \\ &\leq \Pr[\text{SM-V}|\mathrm{QH}_2] + 2^{-k} \\ &= \Pr[\text{SM-V}|(\mathrm{QH}_2 \wedge \mathrm{QH}_3)] + \Pr[\text{SM-V}|(\mathrm{QH}_2 \wedge \neg\mathrm{QH}_3)] + 2^{-k} \\ &\leq \Pr[\text{SM-V}|(\mathrm{QH}_2 \wedge \mathrm{QH}_3)] + q_{h2}(2^{-k}) + 2^{-k}.\end{aligned}$$

Further writing the above inequality, we can also obtain

$$\Pr[\text{SM-V}|(\mathrm{QH}_2 \wedge \mathrm{QH}_3)] \geq \varepsilon - (q_{h2} + 1)/2^k.$$

Seeing that in the jth h_2 random oracle, $\mathcal{B}$ directly returned $z(cP)$ as the result, we claim that when the event $(\text{SM-V}|(\mathrm{QH}_2 \wedge \mathrm{QH}_3))$ occurs, $\mathcal{B}$ would have the probability of $(q_{h2}q_{h3})^{-1}$ to output

$$V^{*z^{-1}} = e(P, P)^{abc}$$

from some entry of h_3-list. Therefore, we can express the probability of $\mathcal{B}$ to solve the BDHP as

$$\varepsilon' \geq (\varepsilon - (q_{h2} + 1)/2^k)/(q_{h2}q_{h3}).$$

The running time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(q_{\mathrm{SMG}})$. □

According to Theorem 2, the proposed scheme is secure against existential forgery attacks. That is, the signcrypted message cannot be forged and the delegated proxy signer cannot repudiate having generated his ciphertext. Hence, we obtain the following corollary.

Corollary 1. *The proposed scheme satisfies the security requirement of non-repudiation.*

Table 1
Comparisons of functionalities and security proofs.

Item	Scheme					
	EA	DCZ	LC	WC	DC	Ours
Pairing-based scheme	O	O	O	O	O	O
Resist key-compromised attack	×	×	O	O	O	O
Proxy delegation	O	O	O	O	×	O
Partial delegation with warrant	×	O	O	O	×	O
Public verifiability	×	O	O	O	O	O
No conversion cost	×	O	O	O	O	O
Complete proof of confidentiality	×	×	×	×	O	O
Complete proof of unforgeability	×	×	×	×	O	O

Table 2
Comparisons of computational costs in number of bilinear pairing operations.

Item	Scheme				
	EA	DCZ	LC	WC	Ours
#Bilinear pairing for PCG phase	3	3	3	2	0
#Bilinear pairing for SMG phase	2	2	2	1	1
#Bilinear pairing for SRV phase	7	4	8	3	3
Total costs for the entire scheme	12	9	13	6	4

5.2. Comparison

We compare the proposed scheme with some previous works including the Elkamchouchi–Abouelseoud [29] (EA for short), Duan et al.'s (DCZ for short) [28], the Li–Chen (LC for short) [30], the Wang–Cao (WC for short) [33] and the Duan–Cao (DC for short) [27] schemes. Table 1 summarizes the comparison in terms of functionalities and security proofs. Note that the Elkamchouchi–Abouelseoud and Duan et al.'s schemes are vulnerable to the key-compromised attack, i.e., once the private key of proxy signer is compromised, an attacker can easily recover the plaintext without the knowledge of designated recipient's private key. From this table, it can be seen that the proposed scheme not only provides better functionalities, but also has provable security.

Table 2 further summarizes the comparison of computational costs in number of the most time-consuming operations, i.e., the bilinear pairing computations. In practice, assume that the elliptic curve $E/F_{3^{163}}$ defined by the equation $y^2 = x^3 - x + 1$ is adopted in the compared schemes. According to the best result in [34], one pairing operation still requires about 11 110 multiplications in $F_{3^{163}}$. To obtain fair comparison results, the Duan–Cao scheme is excluded from Table 2, since their scheme does not have the property of proxy delegation. From the comparison results shown in Table 2, one can see that the proposed scheme outperforms compared ones and hence is more suitable for practical implementation.

6. Conclusions

To extend the application of traditional signcryption schemes, in this paper, we adopt the partial delegation with warrant to propose an efficient proxy signcryption scheme based on pairings. Preserving the property of traditional signcryption schemes that only the designated recipient can decrypt the signcrypted message and verify the signature, the proposed scheme further provides the designated recipient with the ability to easily reveal the ordinary proxy signature for public verification if necessary. Furthermore, the proofs of security requirement for confidentiality against indistinguishability under adaptive chosen-ciphertext attacks (IND-CCA2) and that for unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) are given in the random oracle model. As compared with related works, the proposed scheme earns more computational efficiency and provides better functionalities.

References

[1] W. Diffie, M. Hellman, New directions in cryptography, IEEE Transactions on Information Theory IT-22 (6) (1976) 644–654.
[2] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, IEEE Transactions on Information Theory IT-31 (4) (1985) 469–472.
[3] R. Rivest, A. Shamir, L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, Communications of the ACM 21 (2) (1978) 120–126.
[4] M. Mambo, K. Usuda, E. Okamoto, Proxy signature for delegating signature operation, in: Proceedings of the 3rd ACM Conference on Computer and Communications Security, ACM Press, 1996, pp. 48–57.
[5] M. Mambo, K. Usuda, E. Okamoto, Proxy signatures: delegation of the power to sign messages, IEICE Transactions on Fundamentals of Electronic Communications and Computer Science E79-A (9) (1996) 1338–1354.
[6] B.C. Neuman, Proxy-based authentication and accounting for distributed systems, in: Proceedings of the 13th International Conference on Distributed Computing Systems, 1993, pp. 283–291.
[7] V. Varadharajan, P. Allen, S. Black, An analysis of the proxy problem in distributed system, in: Proceedings of 1991 IEEE Computer Society Symposium on Research in Security and Privacy, 1991, pp. 255–277.

[8] S. Kim, S. Park, D. Won, Proxy signatures, revisited, in: ICICS'97, Springer-Verlag, 1997, pp. 223–232.
[9] B. Meng, S. Wang, Q. Xiong, A fair non-repudiation protocol, in: Proceedings of the 7th International Conference on Computer Supported Cooperative Work in Design, Rio de Janeiro, Brazil, 2002, pp. 68–73.
[10] W. Stallings, Cryptography and Network Security: Principles and Practices, 4th ed., Pearson, 2005.
[11] Y. Zheng, Digital signcryption or how to achieve cost(signature & encryption) $\ll$ cost(signature) + cost(encryption), in: Advances in Cryptology, CRYPTO'97, Springer-Verlag, 1997, pp. 165–179.
[12] H. Petersen, M. Michels, Cryptanalysis and improvement of signcryption schemes, IEE Proceedings—Computers and Digital Techniques 145 (2) (1998) 149–151.
[13] W.H. He, T.C. Wu, Cryptanalysis and improvement of Petersen-Michels signcryption scheme, IEE Proceedings—Computers and Digital Techniques 146 (2) (1999) 123–124.
[14] Y. Zheng, Signcryption and its applications in efficient public key solutions, in: Information Security Workshop, Springer-Verlag, 1997, pp. 291–312.
[15] M. Bellare, M. Jakobsson, M. Yung, Round-optimal zero-knowledge arguments based on any one-way hash function, in: Advances in Cryptology, EUROCRYPT'97, Springer-Verlag, 1997, pp. 280–305.
[16] D. Chaum, Zero-knowledge undeniable signatures, in: Advances in Cryptology, EUROCRYPT'90, Springer-Verlag, 1990, pp. 458–464.
[17] F. Bao, R.H. Deng, A signcryption scheme with signature directly verifiable by public key, in: Workshop on Public Key Cryptography, Springer-Verlag, 1998, pp. 55–59.
[18] J. Baek, R. Steinfeld, Y. Zheng, Formal proofs for the security of signcryption, in: Public Key Cryptography, PKC'02, Springer-Verlag, 2002, pp. 80–98.
[19] X. Boyen, Multipurpose identity-based signcryption: a Swiss army knife for identity-based cryptography, in: Advances in Cryptology, CRYPTO'03, Springer-Verlag, 2003, pp. 383–399.
[20] R.J. Hwang, C.H. Lai, F.F. Su, An efficient signcryption scheme with forward secrecy based on elliptic curve, Applied Mathematics and Computation 167 (2) (2005) 870–881.
[21] P.S.L.M. Barreto, H.Y. Kim, B. Lynn, M. Scott, Efficient algorithms for pairing-based cryptosystems, in: Advances in Cryptology, CRYPTO 2002, Springer-Verlag, 2002, pp. 354–368.
[22] D. Boneh, M. Franklin, Identity-based encryption from the Weil pairing, in: Advances in Cryptology, CRYPTO 2001, Springer-Verlag, 2001, pp. 213–229.
[23] D. Boneh, B. Lynn, H. Shacham, Short signature from the Weil pairing, in: Advances in Cryptology, ASIACRYPT 2001, Springer-Verlag, 2001, pp. 514–532.
[24] C. Gentry, A. Silverberg, Hierarchical ID based cryptography, in: Advances in Cryptology, ASIACRYPT 2002, Springer-Verlag, 2002, pp. 548–566.
[25] F. Zhang, K. Kim, ID-based blind signature and ring signature from pairings, in: Advances in Cryptology, ASIACRYPT 2002, Springer-Verlag, 2002, pp. 533–547.
[26] S. Chow, S.M. Yiu, L. Hui, K.P. Chow, Efficient forward and provably secure ID-based signcryption scheme with public verifiability and public ciphertext authenticity, in: The 6th Annual International Conference on Information Security and Cryptology, ICISC 2003, Springer-Verlag, 2003, pp. 352–369.
[27] S. Duan, Z. Cao, Efficient and provably secure multi-receiver identity-based signcryption, in: Information Security and Privacy, Springer-Verlag, 2006, pp. 195–206.
[28] S. Duan, Z. Cao, Y. Zhou, Secure delegation-by-warrant ID-based proxy signcryption scheme, in: Proceedings of Computational Intelligence and Security Conference, CIS 2005, in: LNAI, vol. 3802, Springer-Verlag, 2005, pp. 445–450.
[29] H. Elkamchouchi, Y. Abouelseoud, A new proxy identity-based signcryption scheme for partial delegation of signing rights, Cryptology ePrint Archive, Report 2008/041, 2008. http://eprint.iacr.org/.
[30] X. Li, K. Chen, Identity based proxy-signcryption scheme from pairings, in: Proceedings of the 2004 IEEE International Conference on Services Computing, IEEE Computer Society, 2004, pp. 494–497.
[31] B. Libert, J.J. Quisquater, New identity based signcryption schemes from pairings, in: IEEE Information Theory Workshop, Paris, France, 2003, pp. 155–158.
[32] J. Malone-Lee, Identity-based signcryption, Cryptology ePrint Archive, Report 2002/098, 2002. http://eprint.iacr.org/.
[33] Q. Wang, Z. Cao, Efficient ID-based proxy signature and proxy signcryption from bilinear pairings, in: Computational Intelligence and Security, vol. 3802, Springer-Verlag, 2005, pp. 167–172.
[34] P.S.L.M. Barreto, B. Lynn, M. Scott, On the selection of pairing-friendly groups, in: Selected Areas in Cryptography, SAC 2003, Springer-Verlag, 2003.