ITC 4/46
Journal of Information Technology and Control
Vol. 46 / No. 4 / 2017
pp. 530-545
DOI 10.5755/j01.itc.46.4.15819

PCMAE: A Proxy Convertible Multi-AE Scheme and Its Variant

Received 2016/07/31 | Accepted after revision 2017/11/14

http://dx.doi.org/10.5755/j01.itc.46.4.15819

PCMAE: A Proxy Convertible Multi-AE Scheme and Its Variant

Han-Yu Lin

Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, Taiwan
e-mail: lin.hanyu@msa.hinet.net

Corresponding author: lin.hanyu@msa.hinet.net

This paper presents a novel proxy convertible multi-authenticated encryption (multi-AE) scheme and its variant with message linkages. The proposed scheme allows two or more original signers to cooperatively delegate their signing power to an authorized proxy signer, such that the proxy signer can generate a valid authenticated ciphertext on behalf of the original signing group and only a designated recipient is capable of decrypting the ciphertext and verifying its embedded proxy multi-signature. Its variant with message linkages further benefits the encryption of a large message by dividing it into many smaller message blocks. The proposed proxy convertible multi-AE scheme and its variant can simultaneously fulfill the security requirements of confidentiality and authenticity. Thus, they are applicable to those group-oriented confidential applications with proxy delegation, e.g., proxy on-line auction, proxy contract signing and so on. In case of a later dispute over repudiation, our proposed scheme also allows a designated recipient to convert the ciphertext into an original proxy multi-signature for public verification. In addition, the security 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 the random oracle model.

KEYWORDS: proxy multi-signature, convertible, authenticated encryption, message linkage.

1. Introduction

With the rapid development of electronic commerce (eCommerce), the security of on-line transactions has received the great attention. Generally speaking, cryptographic techniques can be adopted to protect the communication content over the Internet. Public key encryptions [4] and digital signature schemes [5, 30] are two fundamental cryptographic mechanisms which primarily aim for providing confidentiality [9, 15] and authenticity [30], respectively. The digital signature scheme can further satisfy the requirement of non-repudiation [31] to prevent the signer's dishonesty.

Some applications, however, like the contract signing, the savings withdrawal, on-line auctions and credit card transactions require all the above security re-

quirements simultaneously be achieved. A straightforward way would be sign-then-encrypt [36]. Yet, the approach is costly in terms of computation efforts and communication overheads. In some special circumstances, a proxy might be properly delegated to conduct these confidential transactions, e.g., proxy auctions and the contract signing by an authorized proxy signer. Consider group-oriented applications such as the joint account owned by two or more individuals. To withdraw money from such account, all owners must cooperatively sign the withdrawal form which can only be verified by the bank teller. In case that account owners are unable to sign personally, they can delegate their signing power to a proxy signer who can legitimately conduct transactions on behalf of them. It thus can be seen that the design of efficient and provably secure cryptographic schemes fulfilling such requirements is crucial and benefits to the practical implementation.

1.2. Related Work

In 1996, Mambo *et al.* [25, 26] extended the concept of digital signature and introduced the notion of proxy signatures. A proxy signature scheme allows the original signer to delegate his signing power to an authorized person called proxy signer, such that the proxy signer can generate a valid proxy signature on behalf of the original one. As to the proxy delegation, it can be categorized into four different kinds as follows:

(i). *Full delegation* [25, 26]: The proxy signer uses the private key which is the same as the one of the original signer so that all (proxy) signatures are signed with the same private key. Consequently, it is difficult for a verifier to identify the real signer from a given signature. That is to say, it cannot provide secure mechanisms to protect any one of the original and the proxy signers from being framed by the other.

(ii). *Partial delegation* [25, 26]: The proxy private key is further derived from the original signer's one based on some cryptographic assumptions, e.g., the factorization and the discrete logarithm problems. It is infeasible to compute the original signer's private key from the proxy one. Yet, it needs an additional revocation protocol as no information (e.g., the period of validity) is bonded to the delegation. Moreover, a malicious original signer can easily impersonate the proxy signer by deriving the corresponding proxy private key.

(iii). *Delegation by warrant* [28, 37]: A warrant which contains necessary proxy information, e.g., the period of validity and the identifiers of the original and the proxy signers, could be regarded as the delegation authorization. The warrant is then delivered to the proxy signer for convincing anyone. However, transmitting and verifying the certificate will incur extra computational and communicational costs.

(iv). *Partial delegation with warrant* [16]: This approach integrates the merits of partial delegation and delegation by warrant. It is also computationally infeasible for a proxy signer to derive the original signer's private key from the proxy one. Besides, to certify the warrant and validate the signature can be simultaneously carried out within a single step, which helps reducing the computational and communicational costs.

Obviously, the fourth approach, partial delegation with warrant, is more flexible and secure as compared to the first three. Because of its efficiency and security compared with the others, we also adopt partial delegation with warrant to implement the proposed scheme. Up to the present, lots of variations of proxy signatures have been proposed [10, 12-14, 16, 23, 33, 34, 36].

In 1994, Horster *et al.* [8] proposed an authenticated encryption (AE) scheme which further provides digital signature schemes with the property of confidentiality and only the designated recipient can verify the signature instead of everyone. Since only the designated recipient has the ability to decrypt the ciphertext and verify the corresponding signature, there might be a potential drawback that the signer repudiates his signature. In such a circumstance, it is even difficult for an arbitrator to judge who is lying.

To deal with the case of a later dispute over repudiation, Araki *et al.* [1] proposed a convertible limited verifier signature scheme. However, the signature conversion of their scheme requires the assistance of the signer and incurs additional computation efforts, which is considered to be inefficient and unworkable if the signer is unwilling to cooperate with. Moreover, Zhang and Kim [48] also pointed out that their scheme can not withstand a universal forgery attack on an arbitrary chosen message.

In 2002, Wu and Hsu [40] proposed a convertible authenticated encryption (CAE) scheme, in which the

signature conversion is rather simple and can be solely done by the recipient without any computation effort or communication overhead. Huang and Chang [11] proposed an enhanced scheme in the next year. However, both the Wu-Hsu and the Huang-Chang schemes cannot fulfill the security requirement of confidentiality, i.e., the ciphertext is computationally distinguishable with respect to two candidate messages. To eliminate such a security weakness, Lv *et al.* [24] proposed a secure and practical solution. In 2005, Wu *et al.* [41] proposed generalized CAE schemes and adapted these schemes based on elliptic curves [17, 25] for facilitating gradually popular applications like smart cards [7], mobile phones and PDAs. Since then, lots of related works [6, 21, 35, 39, 43] have been proposed.

In 2008, Chien [3] proposed a selectively CAE scheme allowing either the signer or the designated recipient to perform the signature conversion. In the next year, Lee *et al.* [19] addressed a CAE scheme based on the ElGamal cryptosystem. Considering the RSA cryptosystem, Wu and Lin [44] also presented a CAE scheme based on RSA in 2009. Nevertheless, these schemes are not suitable for the environment of multi-user setting. To fulfill group-oriented application requirements, Wu *et al.* [42] proposed a convertible multi-authenticated encryption (CMAE) scheme which enables a signing group composed of multiple signers to generate a valid authenticated ciphertext. In 2010, Tsai *et al.* [34] removed the necessity of using one-way hash functions. Based on Wu *et al.*'s scheme, Chang [2] addressed another variant with shared verification of multiple designated recipients. In 2012, Lu *et al.* [22] presented a provably CMAE scheme for generalized group communications. Later, Wu *et al.* [46] addressed a publicly verifiable PCAE scheme for confidential applications with proxy delegation. In 2014, Wu and Lin [45] proposed a proxy CAE scheme based on RSA. In 2015, a revocable CAE scheme [20] is also introduced. Yet, none of the above group-oriented CAE schemes can deal with the issue of proxy delegation. Although Lai and Singh [18] had proposed a similar scheme called ID-based multi-proxy multi-signcryption could solve the same problem, their scheme incurred time-consuming bilinear pairing operations and required extra key escrow mechanism.

1.3. Our Contribution

In this paper, we elaborate on the merits of CAE schemes and proxy multi-signature schemes to propose a novel proxy convertible multi-AE scheme and its variant with message linkages. The proposed scheme allows a delegated proxy signer to generate a valid authenticated ciphertext on behalf of the original signing group, such that only a designated recipient can recover the message and verify its embedded proxy multi-signature. When the case of a later dispute over repudiation occurs, a designated recipient can solely convert the authenticated ciphertext into a publicly verifiable proxy multi-signature without extra computation or communication cost. Besides, we further propose a variant with message linkages to benefit the encryption of a large message. We also prove that the proposed scheme achieves 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) in the random oracle model. As compared with related works, the proposed scheme not only provides better functionalities, but also has the provable security.

2. Preliminaries

In this section, we first describe the parties participating in the proposed scheme and define the composed algorithms.

2.1. Parties

Without loss of generality, there are $(n + 2)$ parties participating in a PCMAE scheme including a signing group (composed of n original signers), an authorized proxy signer and an intended recipient. All parties act as probabilistic polynomial-time Turing machines (PPTM). The original signers will deliver proxy credentials to the proxy signer. The latter is responsible for producing an authenticated ciphertext on behalf of the former. Yet, a dishonest proxy signer might repudiate his ciphertext. Finally, the intended recipient has the ability to decrypt the ciphertext and verify the embedded proxy multi-signature. A PCMAE scheme is said to be correct if the authorized proxy signer can generate a valid authenticated ciphertext and only the intended recipient is capable of decrypting it and verifying the proxy multi-signature.

2.2. Algorithms

The proposed PCMAE scheme has the following four algorithms:

- **Setup:** Initially, the Setup algorithm will generate public parameters utilized in the system. Let k be a security parameter. Taking as input 1^k, the algorithm outputs the parameter *params*.
- **Credential-gen (CG):** The CG algorithm is used for producing the proxy credential for an authorized proxy signer. The input information of it is composed of the identity of proxy signer along with the private keys of all original signers. The resulted output is the corresponding proxy credentials.
- **Proxy-sign (PS):** With the PS algorithm, an authorized proxy signer is able to generate a valid ciphertext on behalf of the original signing group. Thus, the input parameter includes a message m, n proxy credentials, the intended recipient's public key and the proxy signer's private key. The corresponding output is an authenticated ciphertext δ.
- **Uncover-verify (UV):** The UV algorithm is used for decrypting authenticated ciphertexts and checking the validity of embedded proxy multi-signature. It takes as input a ciphertext δ, a decryption key and all public keys of original and proxy signers. If the inputted ciphertext δ is valid, it returns the signed message m and its converted proxy multi-signature Ω which is publicly verifiable. Otherwise, an error symbol ¶ is returned instead.

3. The Proposed PCMAE Schemes

We give a concrete construction of our PCMAE scheme in this section. Also, a variant with message linkage for manipulating a large message will be presented. The used notations are stated as Table 1. Detailed steps for each algorithm are shown as follows:

3.1. Basic Construction

- **Setup:** Taking as input a security parameter k, the system authority (SA) selects two large primes (p, q) and a generator g of order q, where $|q| = k$ and $q \mid (p - 1)$. Let h_1: $\{0, 1\}^k \times Z_p^* \rightarrow Z_q$, h_2: $\{0, 1\}^k \times Z_p^* \times Z_p^* \times Z_p^* \rightarrow Z_q$ and h_3: $Z_p^* \rightarrow \{0, 1\}^k$ be collision resistant hash functions. The system's public parameters $params = \{p, q, g, h_1, h_2, h_3\}$. Each user U_i chooses his private key $x_i \in Z_q$ and computes the public key as $y_i = g^{x_i} \bmod p$.
- **Credential-gen (CG):** Let $O = \{U_1, U_2, ..., U_n\}$ be the group of n original users delegating their signing power to the proxy signer U_p. With the following steps, $U_i \in O$ distributes the proxy share to U_p:

Step 1 $U_i \in O$ first chooses $t_i \in_R Z_q$ to compute

$$T_i = g^{t_i} \bmod p, \tag{1}$$

and then sends T_i to U_p and $U_j \in O$, for $j \in i$.

Step 2 Upon receiving all T_j's, U_i computes

$$T = \prod_{j=1}^{n} T_j \bmod p \tag{2}$$

$$\sigma_i = t_i - x_i\, h_1(m_w, T) \bmod q, \tag{3}$$

where m_w is the warrant consisting of the identifier of the original and proxy signers, the delegation duration and so on. (σ_i, m_w, T) is then sent to U_p.

Step 3 Upon receiving (σ_i, m_w, T), U_p verifies

Table 1
The used notations

Z_p	integers modulo p
Z_p^*	multiplicative group of integers modulo p
$x \in Z_p^*$	element x in set Z_p^*
$x \leftarrow Z_p^*$	sampling element x uniformly in set Z_p^*
$a \bmod b$	modulo operation: reminder of a divided by b
$a \mid b$	integer b is divisible by integer a
$\lvert x \rvert$	bit-length of integer x, also absolute value of x
$\sum_{i=1}^{n} v_i, \sum_{i \in S} v_i$	sum of values v_i for $i = 1, 2, ..., n$, or for $i \in S$
$\prod_{i=1}^{n} v_i, \prod_{i \in S} v_i$	product of values v_i for $i = 1, 2, ..., n$, or for $i \in S$
$\otimes$	logical operation XOR
$\neg$	logical operation NOT
$\wedge$	logical operation AND
$\vee$	logical operation OR
$\forall$	for all
$\Pr[E]$	probability of event E occurring

$$T_i = g^{\sigma_i} y_i^{h_1(m_w, T)} (\mathrm{mod}\ p). \quad (4)$$

If it does not hold, (σ_i, m_w, T) is requested to be sent again.

We show that the verification of Eq. (4) works correctly. From the right-hand side of Eq. (4), we have

$$\begin{aligned} & g^{\sigma_i} y_i^{h_1(m_w, T)} \\ &= g^{t_i - x_i h_1(x_w, T)} y_i^{h_1(m_w, T)} \\ &= g^{t_i} \\ &= T_i \ (\mathrm{mod}\ p) \end{aligned}$$ by Eq. (3)

which leads to the left-hand side of Eq. (4).

- **Proxy-sign (PS):** For signing a message $m \in_R \{0, 1\}^k$ on behalf of the original signing group O, U_p chooses $r \in_R Z_q$ to compute

$$R = g^r \bmod p, \quad (5)$$

$$\sigma = \sum_{i=1}^{n} \sigma_i, \quad (6)$$

$$C = \prod_{i=1}^{n} y_i^{h_1(m_w, T)} \bmod p, \quad (7)$$

$$K = y_v^{\sigma} \bmod p, \quad (8)$$

$$S = r + (\sigma - x_p h_2(m, C, K, R)) \bmod q, \quad (9)$$

$$Q = h_3(K) \oplus m, \quad (10)$$

and then delivers the warrant m_w and the authenticated ciphertext $\delta = (Q, S, R, T)$ to the designated recipient U_v.

- **Uncover-verify (UV):** Upon receiving δ, U_v first computes C as Eq. (7) and derives K as

$$K = (TC^{-1})^{x_v} \bmod p. \quad (11)$$

He then recovers the message as

$$m = Q \oplus h_3(K), \quad (12)$$

and checks the redundancy embedded in m. U_v can further verify the proxy multi-signature by checking if

$$RT = g^S y_p^{h_2(m, C, K, R)} C \ (\mathrm{mod}\ p). \quad (13)$$

The correctness of Eqs. (12) and (13) can be easily confirmed. From the right-hand side of Eq. (12), we have

$$\begin{aligned} & Q \oplus h_3(K) \\ &= Q \oplus h_3((TC^{-1})^{x_v} \bmod p) && \text{by Eq. (11)} \\ &= Q \oplus h_3((g^{\sigma})^{x_v} \bmod p) && \text{by Eqs. (4), (6) and (7)} \\ &= m && \text{by Eq. (8) and (10)} \end{aligned}$$

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

If the authenticated ciphertext (Q, S, R, T) is correctly generated, it will pass the test of Eq. (13). From the right-hand side of Eq. (13), we have

$$\begin{aligned} & g^S y_p^{h_2(m, C, K, R)} C \\ &= g^{r + \sigma - x_p h_2(m, C, K, R)} y_p^{h_2(m, C, K, R)} C && \text{by Eq. (9)} \\ &= R g^{\sigma} C && \text{by Eq. (5)} \\ &= RT \ (\mathrm{mod}\ p) && \text{by Eqs. (2), (4) and (6)} \end{aligned}$$

which leads to the left-hand side of Eq. (13).

When a later dispute over repudiation occurs, U_v can reveal the converted proxy multi-signature $\Omega = (S, R, T, K)$, the warrant m_w and the original message m to prove the proxy signer's dishonesty without any additional cost. Thus, anyone can verify the converted proxy multi-signature with the assistance of Eqs. (7) and (13).

3.2. Variant with Message Linkages

Consider the practical implementation that the original message may be large. It therefore will cause the difficulty in encryption. In the subsection, we propose a variant with message linkages to benefit the encryption of a large message by dividing it into lots of small message blocks. The construction is similar as that in Section 3.1. We only describe the different parts as follows:

- **Proxy-sign (PS):** For signing a large message m on behalf of the original signing group O, U_p first divides the message m into n pieces, i.e., $m = m_1 \|$

$m_2 \,||\, ... \,||\, m_n$, m_i's $\in$GF(p), and then chooses $r \in_R Z_q$ and $w_0 = 0$ to compute R, σ, C, K and S as Eqs. (5) to (9). U_p further computes

$$w_i = m_i \cdot h_3(w_{i-1} \oplus h_3(K)) \bmod p, \quad \text{for } i = 1, 2, \ldots, n, \tag{10*}$$

and delivers the warrant m_w along with $\delta = (S, R, T, w_1, w_2, ..., w_n)$ to the designated recipient U_v.

– **Uncover-verify (UV):** Upon receiving it, U_v first derives C and K as Eqs. (7) and (11), respectively. He then computes

$$m_i = w_i \cdot h_3(w_{i-1} \oplus h_3(K))^{-1} \bmod p, \quad \text{for } i = 1, 2, \ldots, n, \tag{12*}$$

and recovers the original message m as $m_1 \,||\, m_2 \,||\, ... \,||\, m_n$. U_v can further verify the proxy multi-signature by checking Eq. (13).

We show that with the authenticated ciphertext $(S, R, T, w_1, w_2, ..., w_n)$ and the warrant m_w, the designated recipient U_v can recover the message m and check its validity with Eq. (12*). From the right-hand side of Eq. (12*), we have

$$\begin{aligned} & w_i \cdot h_3(w_{i-1} \oplus h_3(K))^{-1} \\ & m_i \cdot h_3(w_{i-1} \oplus h_3(K)) \cdot h_3(w_{i-1} \oplus h_3(K))^{-1} \quad \text{by Eq. (10*)} \\ & = m_i \pmod p \end{aligned}$$

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

4. Security Proof and Comparison

In this section, we briefly review the security notions, state the security model and prove the security of our proposed scheme. Some comparisons with related schemes are also made.

4.1. Security Notions

Discrete Logarithm Problem; DLP

Let p and q be two large primes satisfying $q \mid p-1$, and g a generator of order q over GF(p). The discrete logarithm problem is, given an instance (y, p, q, g), where $y = g^x \bmod p$ for some $x \in Z_q$, to derive x.

Discrete Logarithm (DL) Assumption

Let $I_k = \{(p, q, g) \in I \mid |p| = k\}$ with $k \in N$, where I is the universe of all instances and $|p|$ represents the bit-length of p. For every probabilistic polynomial-time algorithm $\mathcal{A}$, every positive polynomial $P(\times)$ and all sufficiently large k, the algorithm $\mathcal{A}$ can solve the DLP with the advantage at most $1/P(k)$, i.e.,

$$\Pr[\mathcal{A}(y, p, q, g) = \mathrm{Log}_{p,q,g}(y), (p, q, g) \leftarrow I_k, y \leftarrow Z_p^*] \le 1/P(k).$$

The probability is taken over the uniformly and independently chosen instance with a given security parameter k and over the random choices of $\mathcal{A}$.

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

Computational Diffie-Hellman Problem; CDHP

Let p and q be two large primes satisfying that $q \mid p-1$, and g a generator of order q over GF(p). The computational Diffie-Hellman problem is, given an instance (p, q, g, g^a, g^b) for some $a, b \in Z_q$, to derive $g^{ab} \bmod p$.

Computational Diffie-Hellman (CDH) Assumption

Let $I_k = \{(p, q, g) \in I \mid |p| = k\}$ with $k \in N$, where I is the universe of all instances and $|p|$ represents the bit-length of p. For every probabilistic polynomial-time algorithm $\mathcal{A}$, every positive polynomial $P(\times)$ and all sufficiently large k, the algorithm $\mathcal{A}$ can solve the CDHP with the advantage at most $1/P(k)$, i.e.,

$$\Pr[\mathcal{A}(p, q, g, g^a, g^b) = g^{ab}, (p, q, g) \leftarrow I_k, a, b \leftarrow Z_q] \le 1/P(k).$$

The probability is taken over the uniformly and independently chosen instance with a given security parameter k and over the random choices of $\mathcal{A}$.

Definition 2. *The (t, ε)-CDH assumption holds if there is no polynomial-time adversary that can solve the CDHP in time at most t and with the advantage ε.*

4.2. Security Model

The security requirements of the proposed PCMAE scheme and its variant are message confidentiality and unforgeability. The widely accepted notion for the security of message confidentiality comes from the definition of indistinguishability-based security, i.e., the adversary attempts to distinguish a target ci-

phertext with respect to two candidate messages. We define these notions as follows:

Definition 3. (Confidentiality) *A PCMAE 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:

- *Credential-gen (CG) queries:* $\mathcal{A}$ makes a CG query with respect to the identity of target proxy signer. $\mathcal{B}$ returns the corresponding proxy credentials.
- *Proxy-sign (PS) queries:* $\mathcal{A}$ chooses a message m and then gives it to $\mathcal{B}$ who will return a corresponding authenticated ciphertext δ with the warrant m_w.
- *Uncover-verify (UV) queries:* $\mathcal{A}$ submits an authenticated ciphertext δ along with the warrant m_w to $\mathcal{B}$. If δ is valid, $\mathcal{B}$ returns the recovered message m and its converted proxy multi-signature Ω; else, the error symbol ¶ is outputted as a result.

Challenge: The adversary $\mathcal{A}$ produces two messages, m_0 and m_1, of the same length. The challenger $\mathcal{B}$ flips a coin $\lambda \leftarrow \{0, 1\}$ and generates an authenticated ciphertext δ^* for m_λ. The ciphertext 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 UV query for the target ciphertext.

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 $Adv(\mathcal{A}) = |\Pr[\lambda' = \lambda] - 1/2|$.

Definition 4. (Unforgeability) *A PCMAE scheme is said to achieve the security requirement of unforgeability against existential forgery under adaptive chosen-message attacks (EF-CMA) 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: $\mathcal{B}$ 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 makes CG and PS queries as those in Phase 1 of Definition 3.

Forgery: Finally, $\mathcal{A}$ produces an authenticated ciphertext δ^* which is not outputted by the PS query. The adversary $\mathcal{A}$ wins if δ^* is valid.

4.3. Security Proofs

We prove the security of our proposed scheme in the random oracle model as Theorems 1 and 2, respectively. The security proofs can be also applied to its variant with message linkages, since they have almost the same structure.

Theorem 1. (Proof of Confidentiality) *The proposed scheme is $(t, q_{h_1}, q_{h_2}, q_{h_3}, q_{CG}, q_{PS}, q_{UV}, \varepsilon)$-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 CDHP, where*

$$\varepsilon' \geq (q_{h_2} + q_{h_3})^{-1}\left(2\varepsilon - \frac{q_{UV}(q_{h_2} + q_{h_3} + 1)}{2^k}\right),$$

$$t' \approx t + t\lambda(2q_{CG} + 4q_{PS} + 3q_{UV}).$$

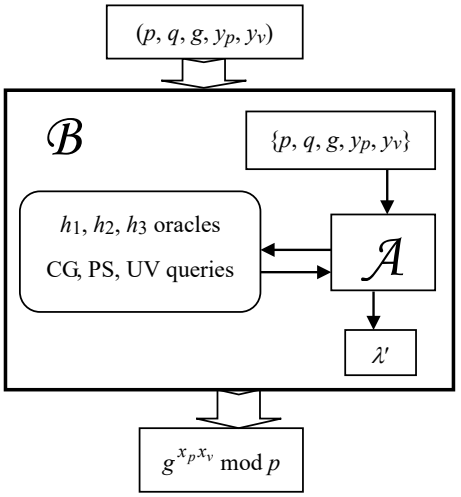
Here t_λ is the time for performing a modular exponentiation over a finite field.

Proof: Fig. 1 depicts the proof structure of this Theorem. Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can $(t, q_{h_1}, q_{h_2}, q_{h_3}, q_{CG}, q_{PS}, q_{UV}, \varepsilon)$-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_{h_i} h_i random oracle (for i = 1 to 3), q_{CG} CG, q_{PS} PS and q_{UV} UV queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the CDHP by taking $\mathcal{A}$ as a subroutine. Let all involved parties and parameters be defined the same as those in Section 3.1. The objective of $\mathcal{B}$ is to obtain $(g^{x_p x_v} \bmod p)$ by taking (p, q, g, y_p, y_v) 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 and sends the system's public parameters *params* = $\{p, q, g, y_o, y_p, y_v\}$ to the adversary $\mathcal{A}$.

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

Figure 1
The proof structure of confidentiality in Theorem 1



- *h_1 oracle:* When $\mathcal{A}$ makes an h_1 oracle of (m_w, T), $\mathcal{B}$ returns O-**Sim_h_1**(m_w, T). The simulated random oracle O-**Sim_h_1** operates as Fig. 2. Note that the function **insert**(N, b) will insert the value b into the array N.
- *h_2 oracle:* When $\mathcal{A}$ makes an h_2 oracle of (m, C, K, R), $\mathcal{B}$ returns O-**Sim_h_2**(m, C, K, R). The simulated random oracle O-**Sim_h_2** operates as Fig. 3.
- *h_3 oracle:* When $\mathcal{A}$ makes an h_3 oracle of K, $\mathcal{B}$ returns O-**Sim_h_3**(K). The simulated random oracle O-**Sim_h_3** operates as Fig. 4.
- *CG queries:* When $\mathcal{A}$ makes a CG query, $\mathcal{B}$ chooses a proper m_w and then returns $(m_w, O\text{-}\mathbf{Sim\_}\boldsymbol{CG}(m_w))$ as the result. The simulated CG oracle O-**Sim_*CG*** operates as Fig. 5. Note that the function **check**(N, b) will return a Boolean value depending on whether the value b is stored in the array N.
- *PS queries:* When $\mathcal{A}$ makes a PS query for some message m, $\mathcal{B}$ returns O-**Sim_*PS***(m) as the result. The simulated PS oracle O-**Sim_*PS*** operates as Fig. 6.
- *UV queries:* When $\mathcal{A}$ makes a UV query for some authenticated ciphertext δ with the warrant m_w, $\mathcal{B}$ returns O-**Sim_*UV***(δ, m_w) as the result. The simulated UV oracle O-**Sim_*UV*** operates as Fig. 7.

Figure 2
Algorithm of the simulated random oracle O-**Sim_h_1**

oracle O-Sim_h_1 (m_W, T)
1: for $i = 0$ to $q_{h_1} - 1$
2: if $(Q\_h_1[i][0] = m_W)$ and $(Q\_h_1[i][1] = T)$ then exit for; // It is an old query.
3: else if $(Q\_h_1[i][0] = \text{“”})$ then // It is a new query.
4: **insert**$(Q\_h_1, (m_W, T))$;
5: $A\_h_1[i] \leftarrow v_1 \in_R Z_q$; exit for;
6: end if
7: next i
8: return $A\_h_1[i]$;

Figure 3
Algorithm of the simulated random oracle O-**Sim_h_2**

oracle O-Sim_h_2 (m, C, K, R)
1: for $i = 0$ to $q_{h_2} - 1$
2: if $(Q\_h_2[i] = (m, C, K, R))$ then
3: exit for; // It is an old query.
4: else if $(Q\_h_2[i][0] = \text{“”})$ then
5: **insert**$(Q\_h_2, (m, C, K, R))$;
6: $A\_h_2[i] \leftarrow v_2 \in_R Z_q$; exit for;
7: end if
8: next i
9: return $A\_h_2[i]$;

Figure 4
Algorithm of the simulated random oracle O-**Sim_h_3**

oracle O-Sim_h_3 (K)
1: for $i = 0$ to $q_{h_3} - 1$
2: if $(Q\_h_3[i] = K)$ then
3: exit for; // It is an old query.
4: else if $(Q\_h_3[i] = \text{“”})$ then
5: $Q\_h_3[i] \leftarrow K$;
6: $A\_h_3[i] \leftarrow v_3 \in_R \{0, 1\}^k$;
exit for;
8: end if
9: next i
10: return $A\_h_3[i]$;

Challenge: $\mathcal{A}$ generates two messages, m_0 and m_1, of the same length. The challenger $\mathcal{B}$ flips a coin $\lambda \leftarrow \{0, 1\}$ and produces an authenticated ciphertext $\delta^* = (Q^*, S^*, R^*, T^*)$ for m_λ by running the simulated **Sim_Chal-**

lenge(m_λ). The algorithm of **Sim_Challenge** operates as Fig. 8.

Phase 2: $\mathcal{A}$ makes new queries as those stated in Phase 1 except the UV query for the target ciphertext δ^*.

Analysis of the game: Consider the above simulations of CG and PS queries. One can see that the simulated proxy credentials σ_i's and authenticated ciphertext δ are computationally indistinguishable from those generated by the real scheme. We refer the

Figure 5
Algorithm of the simulated CG oracle $\mathcal{O}$-**Sim_*CG***

oracle $\mathcal{O}$-Sim_*CG*(m_w)
1: do
2: Choose σ_i's, $v_1 \in_R Z_q$;
3: $T_i = g^{\sigma_i} y_i^{v_1} \bmod p$;
4: $T = \prod_{i=1}^{n} T_i \bmod p$;
5: while (**check**(Q_h_1, (m_w, T)) = true)
6: **insert**(Q_h_1, (m_w, T));
7: **insert**(A_h_1, v_1);
// define $h_1(m_w, T) = v_1$
8: return (T, T_1, T_2, ..., T_n, σ_1, σ_2, ..., σ_n);

Figure 6
Algorithm of the simulated PS oracle $\mathcal{O}$-**Sim_*PS***

oracle $\mathcal{O}$-Sim_*PS*(m)
1: Choose a proper m_w;
2: (T, T_1, T_2, ..., T_n, σ_1, σ_2, ..., σ_n) ← $\mathcal{O}$-**Sim_*PS***(m_w);
3: Compute $v_1 \leftarrow \mathcal{O}$-**Sim_*h*1**($m_w$, T); $\sigma = \sum_{i=1}^{n} \sigma_i$;
4: Compute $C = \prod_{i=1}^{n} y_i^{v_1} \bmod p$; $K = y_v^{\sigma} \bmod p$;
5: do
6: Choose S, $v_2 \in_R Z_q$;
7: $R = g^S y_p^{v_2} CT^{-1} \bmod p$;
8: while (**check**(Q_h_2, (m, C, K, R)) = true)
9: **insert**(Q_h_2, (m, C, K, R));
10: **insert**(A_h_2, v_2); // define $h_2(m, C, K, R) = v_2$
11: $Q = \mathcal{O}$-**Sim_*h*3**(K) $\oplus$ m;
12: return $\delta = (Q, S, R, T)$ and m_w;

Figure 7
Algorithm of the simulated UV oracle $\mathcal{O}$-**Sim_*UV***

oracle $\mathcal{O}$-Sim_*UV*(δ, m_w) // $\delta = (Q, S, R, T)$
1: $v_1 = \mathcal{O}$-**Sim_*h*1**(m_w, T); $C = \prod_{i=1}^{n} y_i^{v_1} \bmod p$;
2: if (**check**(Q_h_2, (*, C, *, R)) = true) then
//h_2(*, C, *, R) has ever been queried.
3: for j = 0 to $q_{h_2} - 1$
4: if (Q_h_2[j][1] = C) and (Q_h_2[j][3] = R) then
5: m = Q_h_2[j][0];
6: K = Q_h_2[j][2];
7: v_2 = A_h_2[j]; exit for;
8: end if
9: next j
10: $v_3 = \mathcal{O}$-**Sim_*h*3**(K);
11: if ($m = Q \oplus v_3$) and ($RT = g^S y_p^{v_2} C \bmod p$)) then
12: return (m, R, S, T, K) and m_w;
13: else
14: return ¶;
15: end if
16: else // h_2(*, C, *, R) has never been queried.
17: return ¶;
18: end if

Figure 8
Algorithm of the simulated **Sim_Challenge**

algorithm Sim_Challenge($m\lambda$)
1: Choose a proper m_w; $v_3 \in_R \{0, 1\}^k$ and S^*, σ, v_1, $v_2 \in_R Z_q$;
2: $C = \prod_{i=1}^{n} y_i^{v_1} \bmod p$; $T^* = y_p^{\sigma} C \bmod p$;
3: **insert**(Q_h_1, (m_w, T^*)); **insert**(A_h_1, v_1);
// define $h_1(m_w, T^*) = v_1$
4: $R^* = g^{S^*} y_p^{v_2} CT^{*-1} \bmod p$;
5: **insert**(Q_h_2, ($m\lambda$, C, null, R^*));
6: **insert**(A_h_2, v_2);
// Implicitly define $h_2(m\lambda, C, K^*, R^*) = v_2$, where $K^* = (y_v^{\sigma})^{x_v} \bmod p$ and $\mathcal{B}$ does not know it.
7: $Q^* = v_3 \oplus m$; // Implicitly define $h_3(K^*) = v_3$
8: return $\delta = (Q^*, S^*, R^*, T^*)$ and m_w;

simulations of CG and PS queries to be perfect. Then we evaluate the simulation of UV queries. From the algorithms of $\mathcal{O}$-**Sim_*UV***, we find out that it is possible for an UV query of some valid $\delta = (Q, S, R, T)$ to return the error symbol ¶ on condition that $\mathcal{A}$ has the ability to produce δ without asking the corresponding $h_2(m_\lambda, C, K, R)$ or $h_3(K)$ random oracles in advance. Let UV_ERR be the event that an UV query returns the error symbol ¶ for some valid δ during the entire game, AC-V an event that the authenticated ciphertext δ of a UV query made by $\mathcal{A}$ is valid. QH_2 and QH_3 separately denote the events that $\mathcal{A}$ has ever asked $h_2(m_\lambda, C, K, R)$ and $h_3(K)$ random oracles beforehand. Then we can express the error probability of any UV query as

$$\begin{aligned}
&\Pr[\text{AC-V} \mid (\neg QH_3 \vee \neg QH_2)] \\
&\le \Pr[\text{AC-V} \mid \neg QH_3] + \Pr[\text{AC-V} \wedge QH_3 \mid \neg QH_2] \\
&= \Pr[\text{AC-V} \wedge QH_2 \mid \neg QH_3] \\
&\quad + \Pr[\text{AC-V} \wedge \neg QH_2 \mid \neg QH_3] \\
&\quad + \Pr[\text{AC-V} \wedge QH_3 \mid \neg QH_2] \\
&\le \frac{q_{h_2}}{2^k} + \frac{1}{2^k} + \frac{q_{h_3}}{2^k} \\
&= \frac{q_{h_2} + q_{h_3} + 1}{2^k}.
\end{aligned}$$

Since $\mathcal{A}$ can make at most q_{UV} UV queries, we can further express the probability of UV_ERR as

$$\Pr[\text{UV\_ERR}] \le \frac{q_{UV}(q_{h_2} + q_{h_3} + 1)}{2^k}. \tag{14}$$

Additionally, in the challenge phase, $\mathcal{B}$ has returned a simulated authenticated ciphertext $\delta^* = (Q^*, S^*, R^*, T^*)$ where $T^* = y_p^{\sigma} C \bmod p$, which implies the shared secret K^* is implicitly defined as $(y_v^{\sigma})^{x_v} \bmod p$. Let GP be the event that the entire simulation game does not abort. Obviously, if the adversary $\mathcal{A}$ never asks $h_2(m_\lambda, C, K^*, R^*)$ or $h_3(K^*)$ random oracles in Phase 2, the entire simulation game could be normally terminated. We denote the two events that $\mathcal{A}$ does make an $h_2(m_\lambda, C, K^*, R^*)$ and $h_3(K^*)$ query in Phase 2 by $QH_2{}^*$ and $QH_3{}^*$. When the entire simulation game does not abort, it can be seen $\mathcal{A}$ gains no advantage in guessing λ due to the randomness of the output of the random oracle, i.e.,

$$\Pr[\lambda' = \lambda \mid \text{GP}] = 1/2. \tag{15}$$

Rewriting the expression of $\Pr[\lambda' = \lambda]$, we have

$$\begin{aligned}
\Pr[\lambda' = \lambda] &= \Pr[\lambda' = \lambda \mid \text{GP}] \Pr[\text{GP}] \\
&\quad + \Pr[\lambda' = \lambda \mid \neg\text{GP}] \Pr[\neg\text{GP}] \\
&\le (1/2)\Pr[\text{GP}] + \Pr[\neg\text{GP}] \qquad \text{by Eq. (15)} \\
&= (1/2)(1 - \Pr[\neg\text{GP}]) + \Pr[\neg\text{GP}] \\
&= (1/2) + (1/2)\Pr[\neg\text{GP}]. \qquad (16)
\end{aligned}$$

On the other hand, we can also derive that

$$\begin{aligned}
\Pr[\lambda' = \lambda] &\ge \Pr[\lambda' = \lambda \mid \text{GP}] \Pr[\text{GP}] \\
&= (1/2)(1 - \Pr[\neg\text{GP}]) \qquad (17) \\
&= (1/2) - (1/2)\Pr[\neg\text{GP}].
\end{aligned}$$

With inequalities (16) and (17), we know that

$$|\Pr[\lambda' = \lambda] - 1/2| \le (1/2)\Pr[\neg\text{GP}]. \tag{18}$$

Recall that in Definition 3, $\mathcal{A}$*'s* advantage is defined as $Adv(\mathcal{A}) = |\Pr[\lambda' = \lambda] - 1/2|$. By assumption, $\mathcal{A}$ has non-negligible probability ε to break the proposed scheme. We therefore have

$$\begin{aligned}
\varepsilon &= |\Pr[\lambda' = \lambda] - 1/2| \\
&\le (1/2)\Pr[\neg\text{GP}] \\
&= (1/2)(\Pr[QH_2{}^* \vee QH_3{}^* \vee \text{UV_ERR}]) \\
&\le (1/2)(\Pr[QH_2{}^*] + \Pr[QH_3{}^*] \\
&\quad + \Pr[\text{UV_ERR}]). \qquad \text{by Eq. (18)}
\end{aligned}$$

Combining Eq. (14) and rewriting the above inequality, we get′

$$\begin{aligned}
(\Pr[QH_2{}^*] + \Pr[QH_3{}^*]) &\ge 2\varepsilon - \Pr[\text{UV_ERR}] \\
&\ge 2\varepsilon - \frac{q_{UV}(q_{h_2} + q_{h_3} + 1)}{2^k}.
\end{aligned}$$

If the event $(QH_2{}^* \vee QH_3{}^*)$ happens, we claim that $K^* = (y_v^{\sigma})^{x_v} \bmod p$ will be stored in some entry of the Q_h_2 or the Q_h_3 array. Consequently, $\mathcal{B}$ has non-negligible probability

$$\varepsilon' \ge (q_{h_2} + q_{h_3})^{-1}\left(2\varepsilon - \frac{q_{UV}(q_{h_2} + q_{h_3} + 1)}{2^k}\right)$$

to output $K^{*\sigma^{-1}} = g^{x_p x_v}$ and solve the CDHP. The computational time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(2q_{CG} + 4q_{PS} + 3q_{UV})$.

Q.E.D.

In 2000, Pointcheval and Stern introduced the Forking lemma [29] to prove the security for generic digital signature schemes in the random oracle model. If we apply their techniques to prove our scheme, we can also obtain the generic result as follows.

(The Forking Lemma) *In the random oracle model, let* $(\mathcal{G}, \Sigma, \mathcal{V})$ *be a generic signature scheme and* $\mathcal{A}$ *a probabilistic polynomial-time Turing machine whose input only consists of public data. We denote respectively by* N_1 *and* N_2 *the number of queries that* $\mathcal{A}$ *can ask to the random oracle and the number of queries that* $\mathcal{A}$ *can ask to the signer. Assume that, within a time bound T,* $\mathcal{A}$ *produces, with probability* $\varepsilon \geq 10(N_2+1)(N_2+N_1)/2^k$, *a valid signature* $(m, \sigma_1, h, \sigma_2)$ *where* $\sigma_1 = (m_w, R, T, K)$, $h = (h_2(m, C, K, R), h_1(m_w, T))$ *and* $\sigma_2 = S$. *If the triples* (σ_1, h, σ_2) *can be simulated without knowing the private key with an indistinguishable distribution probability, then there is another machine which has control over the machine obtained from* $\mathcal{A}$ *replacing interaction with the signer by simulation and produces two valid signatures* $(m, \sigma_1, h, \sigma_2)$ *and* $(m, \sigma_1, h', \sigma_2')$ *such that* $h_2(m, C, K, R) \neq h'_2(m, C, K, R)$ *in the expected time* $T' \leq 120686T/\varepsilon$.

More concretely, in our scheme, we can first obtain two equations below:

$$RT = g^S y_p^{h_2(m, C, K, R)} C \bmod p,$$

$$RT = g^{S'} y_p^{h'_2(m, C, K, R)} C \bmod p.$$

By combining the above two equalities, we can further derive the private key x_p as

$$x_p = (S - S')/(h'_2(m, C, K, R) - h_2(m, C, K, R)).$$

Still, to give a tight reduction from the hardness of DLP to our proposed scheme, we present another more detailed security proof and the advantage analysis as Theorem 2.

Theorem 2. (Proof of Unforgeability) *The proposed scheme is* $(t, q_{h_1}, q_{h_2}, q_{h_3}, q_{CG}, q_{PS}, \varepsilon)$*-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 DLP, where*

$$\varepsilon' \geq 4^{-1}(\varepsilon - 2^{-2k})^3 (q_{h_2}^{-1}),$$

$$t' \approx t + t_\lambda(4q_{CG} + 8q_{PS}).$$

Here t_λ *is the time for performing a modular exponentiation over a finite field.*

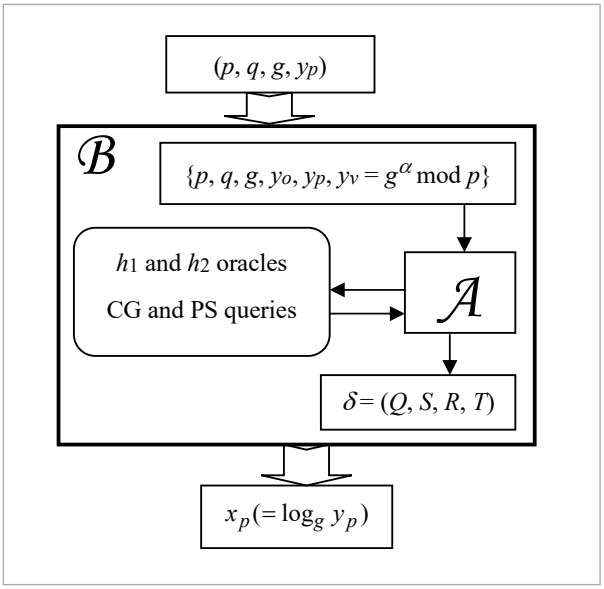
Proof: Fig. 9 depicts the proof structure of this Theorem. Suppose that $\mathcal{A}$ is a probabilistic polynomial-time adversary $\mathcal{A}$ can $(t, q_{h_1}, q_{h_2}, q_{h_3}, q_{CG}, q_{PS}, \varepsilon)$-break the proposed scheme with non-negligible advantage ε under the adaptive chosen-message attack after running in time at most t and asking at most q_{h_i} h_i random oracle (for i = 1 to 3), q_{CG} CG and q_{PS} PS queries. Then we can construct another algorithm $\mathcal{B}$ that (t', ε')-breaks the DLP by taking $\mathcal{A}$ as a subroutine. Let all involved parties and notations be defined the same as those in Section 3.1, h_3 a collision resistant hash function and (h_1, h_2) random oracles. The objective of $\mathcal{B}$ is to obtain $x_p (= \log_g y_p)$ by taking (p, q, g, y_p) 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 to obtain the system's public parameters *params* = {p, q, g} and comes up with a random tape composed of a long sequence of random bits. Then $\mathcal{B}$ simulates two runs of the proposed scheme to the adversary $\mathcal{A}$ on input *params*, y_o, y_p, $y_v = g^\alpha \bmod p$ where $\alpha \in_R Z_q$, and the random tape.

Phase 1: $\mathcal{A}$ adaptively asks h_1 and h_2 random oracles, CG and PS queries as those defined in Theorem 1.

Analysis of the game: According to the analyses of Theorem 1, the simulations of CG and PS queries are perfect. Namely, the adversary $\mathcal{A}$ can not distinguish

Figure 9
The proof structure of unforgeability in Theorem 2



whether he is playing in either a simulation or a real scheme. Let AC-V be the event that $\mathcal{A}$ forges a valid authenticated ciphertext $\delta = (Q, S, R, T)$ for his arbitrarily chosen message m. Since $\mathcal{A}$ has non-negligible probability ε to break the proposed scheme under the adaptive chosen-message attack by the initial assumption, we know that

$\Pr[\text{AC-V}] = \varepsilon$.

Now we further consider the situation where $\mathcal{A}$ is able to output a valid δ without asking h_1 and h_2 random oracles in advance. Let NR be the event that $\mathcal{A}$ guesses correct output values of $h_1(m_w, T)$ and $h_2(m, C, K, R)$ without asking the random oracles, i.e., $\Pr[\text{NH}] \leq 2^{-2k}$. Then, we can express the probability that $\mathcal{A}$ outputs a valid forgery $\delta = (Q, S, R, T)$ after asking the corresponding random oracles as

$\Pr[\text{AC-V} \wedge \neg\text{NH}] \geq (\varepsilon - 2^{-2k})$.

With the initially selected private key α, $\mathcal{B}$ can recovers m and obtain the multi-proxy signature (S, R, T, K) along with m_w.

Then $\mathcal{B}$ launches the second simulation. He again runs $\mathcal{A}$ on input *params*, y_o, y_p, $y_v = g^{\alpha} \bmod p$ where $\alpha \in_R Z_q$, and the same random tape. Since the adversary $\mathcal{A}$ is given the same sequence of random bits, we can anticipate that the i-th random query $\mathcal{A}$ asks will always be the same as the one in the first simulation. In the second simulation, $\mathcal{B}$ returns identical results as those he responds in the first time until $\mathcal{A}$ makes the $h_2(m, C, K, R)$ query. At this time, $\mathcal{B}$ directly gives another answer $v_2^* \in_R Z_q$ rather than original v_2. Meanwhile, $\mathcal{A}$ is then supplied with a different random tape which also consists of a long sequence of random bits. From the statement of "Forking lemma", we can learn that when $\mathcal{A}$ finally makes another valid forgery $\delta^* = (Q^*, S^*, R, T^*)$ where $h_2(m, C, K, R) \neq h_2^*(m, C, K, R)$, $\mathcal{B}$ could solve the DLP with non-negligible probability. To analyze $\mathcal{B}$'s success probability, we use the "Splitting lemma" [29] described below:

Let X and Y be the sets of possible sequences of random bits and random function values provided to $\mathcal{A}$ before and after the $h_2(m, C, K, R)$ query is issued, respectively. It follows that on inputting a random value $(x \,||\, y)$ for any $x \in X$ and $y \in Y$, $\mathcal{A}$ returns a valid forgery with the non-negligible probability ε, i.e.,

$\Pr_{x \in X, y \in Y}[\text{AC-V}] = \varepsilon$.

By the "Splitting lemma", there exists a subset $D \in X$ such that

(a) $\Pr[x \in D] = |D| \times |X|^{-1} \geq 2^{-1}\varepsilon$.

(b) $\forall x \in D, \Pr_{y \in Y}[\text{AC-V}] \geq 2^{-1}\varepsilon$.

If we let $\rho \in D$ and $y' \in Y$ separately be the supplied sequences of random bits and random function values before and after $\mathcal{A}$ makes the $h_2(m, C, K, R)$ query, $\mathcal{A}$ is able to make a valid forgery in the second simulation with the probability of at least $(2^{-1}\varepsilon)^2 = 4^{-1}\varepsilon^2$, i.e.,

$\Pr_{\rho \in D, y' \in Y}[\text{AC-V}] \geq 4^{-1}\varepsilon^2$.

Since we have known that $\mathcal{A}$ eventually returns another valid $\delta^* = (Q^*, S^*, R, T^*)$ with $h_2(m, C, K, R) \neq h_2^*(m, C, K, R)$ is $q_{h_2}^{-1}$, the probability of $\mathcal{B}$ to solve the DLP in the second simulation can be represented as

$\varepsilon' \geq (\varepsilon - 2^{-2k})(4^{-1}(\varepsilon - 2^{-2k})^2)(q_{h_2}^{-1})$

$4^{-1}(\varepsilon - 2^{-2k})^3(q_{h_2}^{-1})$.

Moreover, the computational time required for $\mathcal{B}$ in one simulation is

$t' \approx t + t_{\lambda}(4q_{CG} + 8q_{PS})$.

Q.E.D.

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

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

4.4. Comparisons

We compare the proposed scheme with some related works including Lv *et al.*'s (LWK for short) [24], Tso *et al.*' (TOO for short) [35], Araki *et al.*' (AUI for short) [1], the Wu-Hsu (WH for short) [40], Wu *et al.*'s (WHT for short) [42], Chang's (Cha for short) [2], Tsai's (Tsa for short) [34] and the Lin-Yeh (LY for short) [22] schemes in terms of functionalities and security proofs. Detailed comparisons are demonstrated as Table 2. Since WHT and Cha schemes also have provable security, we further compare our work with them in terms of computational efforts which is evaluated by the number of required modular exponentiation operations. The performance comparison is demonstrated as Table 3. From these tables, it can be seen that the proposed scheme provides not only better functionalities, but also lower computational costs.

Table 2
Comparisons in terms of functionalities and security proofs

Item \ Scheme	LWK	TOO WH	AUI	WHT Cha	Tsa LY	Ours
Multi-User Environment	No	No	No	Yes	Yes	Yes
Proxy Delegation	No	No	No	No	No	Yes
Message Linkages	Yes	No	No	No	No	Yes
Signature Conversion	Yes	Yes	Yes	Yes	Yes	Yes
No Conversion Cost	Yes	Yes	No	Yes	Yes	Yes
Proof of Confidentiality	No	No	No	Yes	No	Yes
Proof of Unforgeability	No	No	No	Yes	No	Yes

Table 3
Comparisons in number of required modular exponentiation operations

Item \ Scheme	WHT	Cha	Ours
Computational Costs*	$3n^2 - n + 5$	$3n^2 - n + 5$	$3n + 7$

Remark *: Let n be the size of original signing group. The computational costs include those executed by each original signer, proxy signer and the designated recipient.

5. Conclusions

In this paper, we have proposed a novel PCMAE scheme to solve the group-oriented delegation problem for confidential transactions. The proposed scheme allows the proxy signer to produce an authenticated ciphertext on behalf of the original signing group and only the designated recipient is capable of recovering the message and verifying its proxy multi-signature for guaranteeing the confidentiality. Its variant with message linkages further benefits the transmission of a large message by dividing it into many smaller message blocks. It is not necessary to establish a session key in advance between a proxy signer and a designated recipient. Without revealing the private key, a designated recipient can independently convert the authenticated ciphertext into an ordinary proxy multi-signature for the public arbitration in case of a later repudiation. Since the converted proxy multi-signature is obtained during the message recovery and signature verification process, the signature conversion process requires no extra computation efforts and communication overheads. In addition, we also proved that the proposed scheme achieves 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) in the random oracle model. As compared with related works, ours not only provides better functionalities, but also has provable security.

Acknowledgements

The author would like to thank anonymous referees for their valuable suggestions. This work was supported in part by the Ministry of Science and Technology of Republic of China under the contract number MOST 106-2221-E-019-008.

References

1. Araki, S., Uehara, S., Imamura, K. The Limited Verifier Signature and Its Application. IEICE Transactions on Fundamentals, 1999, E82-A(1), 63-68.
2. Chang, T. Y. A Convertible Multi-Authenticated Encryption Scheme for Group Communications. Information Sciences, 2008, 178(17), 2008, 3426-3434.
3. Chien, H. Y. Selectively Convertible Authenticated Encryption in the Random Oracle Model. The Computer Journal, 2008, 51(4), 2008, 419-434.
4. Diffie, W., Hellman, M. New Directions in Cryptography. IEEE Transactions on Information Theory, 1976, IT-22(6), 644-654. https://doi.org/10.1109/TIT.1976.1055638
5. ElGamal, T. A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms. IEEE Transactions on Information Theory, 1985, IT-31(4), 469-472. https://doi.org/10.1109/TIT.1985.1057074
6. Elkamshoushy, D. H., AbouAlsoud, A. K., Madkour, M. New Proxy Signcryption Scheme with DSA Verifier. Proceedings of the 23th National Radio Science Conference (NRSC 2006), 2006, 1-8. https://doi.org/10.1109/NRSC.2006.386345
7. Hendry, M. Smart Card Security and Applications, Artech House, Inc., 1997.
8. Horster, P., Michel, M., Peterson, H. Authenticated Encryption Schemes with Low Communication Costs. Electronics letters, 1994, 30(15), 1212-1213. https://doi.org/10.1049/el:19940856
9. Hou, F., Wang, Z., Tang, Y., Liu, Z. Protecting Integrity and Confidentiality for Data Communication. Proceedings of the 9th International Symposium on Computers and Communications (ISCC), 2004, 1(28), 357-362.
10. Hsu, C. L., Wu, T. S., Wu, T. C. New Nonrepudiable Threshold Proxy Signature Scheme with Known Signers. The Journal of Systems and Software, 2001, 58(2), 119-124. https://doi.org/10.1016/S0164-1212(01)00032-2
11. Huang, H. F., Chang, C. C. An Efficient Convertible Authenticated Encryption Scheme and Its Variant. Proceedings of the 5th International Conference on Information and Communications Security (ICICS2003), Springer-Verlag, Berlin, 2003, 382-392. https://doi.org/10.1007/978-3-540-39927-8_35
12. Hwang, S. J., Chen, C. C. A New Multi-Proxy Multisignature Scheme, 2001 National Computer Symposium, 2001, 19-26.
13. Hwang, M. S., Lin, I. C., Eric Lu, J. L. A Secure Nonrepudiable Threshold Proxy Signature Scheme with Known Signers. International Journal of Informatica, 2000, 11(2), 1-8.
14. Hwang, S. J., Shi, C. H. A Simple Multi-Proxy Signature Scheme. Proceedings of the 10th National Conference on Information Security, 2000, 134-138.
15. Jacob, J. A Uniform Presentation of Confidentiality Properties. IEEE Transactions on Software Engineering, 1991, 17(11), 1186-1194. https://doi.org/10.1109/32.106973
16. Kim, S., Park, S., Won, D. Proxy Signatures, Revisited. ICICS'97, Springer-Verlag, 223-232, 1997.
17. Koblitz, N. Elliptic Curve Cryptosystems. Mathematics of Computation, 1987, 48(177), 203-209. https://doi.org/10.1090/S0025-5718-1987-0866109-5
18. Lal, S., Singh, T. New ID Based Multi-Proxy Multi-Signcryption Scheme from Pairings. Computing Research Repository, Cryptography and Security, arXiv:cs/0701044, 2007.
19. Lee, C. C., Hwang, M. S., Tzeng, S. F. A New Convertible Authenticated Encryption Scheme Based on the ElGamal Cryptosystem. International Journal of Foundations of Computer Science, 2009, 20(2), 351-359. https://doi.org/10.1142/S0129054109006607
20. Lin, H. Y. RPCAE: A Novel Revocable Proxy Convertible Authenticated Encryption Scheme. International Journal of Information Security, 2015, 14(5), 431-441. https://doi.org/10.1007/s10207-014-0269-2
21. Lin, H. Y., Wu, T. S. Bilinear Pairings Based Convertible Authenticated Encryption Scheme with Provable Recipient. Proceedings of 2008 International Computer Symposium (ICS 2008), Taipei, Taiwan, November 2008.
22. Lu, C. F., Hsu, C. L., Lin, H. Y. Provably Convertible Multi-Authenticated Encryption Scheme for Generalized Group Communications. Information Sciences, 2012, 199(15), 154-166. https://doi.org/10.1016/j.ins.2012.02.051
23. Lu, R., He, D., Wang, C. On the Security of an Identity-Based Threshold Proxy Signature Scheme with Known Signers. Proceedings of the 3rd International Conference on Natural Computation 2007 (ICNC 2007), IEEE Press, Piscataway, U.S.A., 2007, 3, 210-214. https://doi.org/10.1109/ICNC.2007.515
24. Lv, J., Wang, X., Kim, K. Practical Convertible Authenticated Encryption Schemes Using Self-Certified

Public Keys. Applied Mathematics and Computation, 2005, 169(2), 1285-1297. https://doi.org/10.1016/j.amc.2004.10.057

25. Mambo, M., Usuda, K., Okamoto, E. Proxy Signature for Delegating Signature Operation. Proceedings of the 3rd ACM Conference on Computer and Communications Security, ACM Press, 1996, 48-57.
26. Mambo, M., Usuda, K., Okamoto, E. Proxy Signatures: Delegation of the Power to Sign Messages. IEICE Transactions on Fundamentals of Electronic Communications and Computer Science, 1996, E79-A(9), 1338-1354.
27. Miller, V. Use of Elliptic Curves in Cryptography. Advances in Cryptology – CRYPTO'85, Springer-Verlag, 1985, 417-426.
28. Neuman B. C. Proxy-Based Authentication and Accounting for Distributed Systems. Proceedings of the 13th International Conference on Distributed Computing Systems, 1993, 283-291.
29. Pointcheval, D., Stern, J. Security Arguments for Digital Signatures and Blind Signatures. Journal of CRYPTOLOGY, 2000, 13, 361-369. https://doi.org/10.1007/s001450010003
30. Rivest, R., Shamir, A., Adleman, L. A Method for Obtaining Digital Signatures and Public-Key Cryptosystems. Communications of the ACM, 1978, 21(2), 120-126. https://doi.org/10.1145/359340.359342
31. Schneider, S. Formal Analysis of a Non-Repudiation Protocol. Proceedings of 11th IEEE Computer Security Foundations Workshop, IEEE Press, Piscataway, USA, 1998, 54-65. https://doi.org/10.1109/CSFW.1998.683155
32. Stallings, W. Cryptography and Network Security: Principles and Practices, 4th Ed., Pearson, 2005.
33. Sun, H. M., Lee, N. Y., Hwang, T. Threshold Proxy Signatures. IEE Proceedings of Computers & Digital Techniques, 1999, 146(5), 259-263. https://doi.org/10.1049/ip-cdt:19990647
34. Tsai, J. L., Wu, T. S., Lin, H. Y., Lee, J. E. Efficient Convertible Multi-Authenticated Encryption Scheme Without Message Redundancy or One-Way Hash Function. International Journal of Innovative Computing, Information and Control, 2010, 6(9), 3843-3852.
35. Tso, R., Okamoto, T., Okamoto, E. An Improved Signcryption Scheme and Its Variation. Proceedings of the 4th International Conference on Information Technology (ITNG '07), 2007, 772-778. https://doi.org/10.1109/ITNG.2007.34
36. Tzeng, S. F., Yang, C. Y., Hwang, M. S. A Nonrepudiable Threshold Multi-Proxy Multisignature Scheme with Shared Verification. Future Generation Computer Systems, 2004, 20(5), 887-893. https://doi.org/10.1016/j.future.2004.01.002
37. Varadharajan, V., Allen, P., Black, S. An Analysis of the Proxy Problem in Distributed System. Proceedings of 1991 IEEE Computer Society Symposium on Research in Security and Privacy, 1991, 255-277. https://doi.org/10.1109/RISP.1991.130793
38. VISA and MasterCard Inc. Secure Electronic Transaction (SET) Specification, Version 1.0, 1997.
39. Wang, Q., Cao, Z. Efficient ID-Based Proxy Signature and Proxy Signcryption from Bilinear Pairings. Computational Intelligence and Security, Springer-Verlag, 2005, 3802, 167-172. https://doi.org/10.1007/11596981_25
40. Wu, T. S., Hsu, C. L. Convertible Authenticated Encryption Scheme. The Journal of Systems and Software, 2002, 62(3), 205-209. https://doi.org/10.1016/S0164-1212(01)00143-1
41. Wu, T. S., Hsu, C. L., Lin, H. Y. Efficient Convertible Authenticated Encryption Schemes for Smart Card Applications in Network Environments. Proceedings of the 9th World Multi-Conference on Systemics, Cybernetics and Informatics (WMSCI 2005), Orlando, Florida, U.S.A., July 2005.
42. Wu, T. S., Hsu, C. L., Tsai, K. Y., Lin, H. Y., Wu, T. C. Convertible Multi-Authenticated Encryption Scheme. Information Sciences, 2008, 178(1), 256-263. https://doi.org/10.1016/j.ins.2007.06.011
43. Wu, T. S., Lin, H. Y. ECC Based Convertible Authenticated Encryption Scheme Using Self-Certified Public Key Systems. International Journal of Algebra, 2008, 2(3), 109-117.
44. Wu, T. S., Lin, H. Y. Secure Convertible Authenticated Encryption Scheme Based on RSA. Informatica, 2009, 33(4), 481-486.
45. Wu, T. S., Lin, H. Y. Provably Secure Proxy Convertible Authenticated Encryption Scheme Based on RSA. Information Sciences, 2014, 10, 577-587. https://doi.org/10.1016/j.ins.2014.03.075
46. Wu, T. S., Lin, H. Y., Ting, P. Y. A Publicly Verifiable PCAE Scheme for Confidential Applications with Proxy Delegation. Transactions on Emerging Telecommunications Technologies, 2012, 23(2), 172-185. https://doi.org/10.1002/ett.1522
47. Xue, Q., Cao, Z. A Nonrepudiable Multi-Proxy Multisignature Scheme. Proceedings of 1st Joint Workshop on

Mobile Future & Symposium on Trends in Communications (SympoTIC'04), IEEE Press, Piscataway, USA, 2004, 102-105.

48. Zhang, F., Kim, K. A Universal Forgery on Araki et al.'s Convertible Limited Verifier Signature Scheme. IEICE Transactions on Fundamentals, 2003, E86-A(2), 2003, 515-516.

Summary / Santrauka

This paper presents a novel proxy convertible multi-authenticated encryption (multi-AE) scheme and its variant with message linkages. The proposed scheme allows two or more original signers to cooperatively delegate their signing power to an authorized proxy signer, such that the proxy signer can generate a valid authenticated ciphertext on behalf of the original signing group and only a designated recipient is capable of decrypting the ciphertext and verifying its embedded proxy multi-signature. Its variant with message linkages further benefits the encryption of a large message by dividing it into many smaller message blocks. The proposed proxy convertible multi-AE scheme and its variant can simultaneously fulfill the security requirements of confidentiality and authenticity. Thus, they are applicable to those group-oriented confidential applications with proxy delegation, e.g., proxy on-line auction, proxy contract signing and so on. In case of a later dispute over repudiation, our proposed scheme also allows a designated recipient to convert the ciphertext into an original proxy multi-signature for public verification. In addition, the security 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 the random oracle model.

Straipsnyje pristatoma nauja konvertuojama tarpinio serverio multi-autentifikuota šifravimo (multi-AE) schema ir jos variantas su pranešimų ryšiais. Siūloma schema leidžia dviem ar daugiau pirminių pasirašiusiųjų bendrai perduoti įgaliojimą autorizuotam tarpinio serverio įgaliotiniui pasirašyti. Tokiu būdu, tarpinio serverio įgaliotinis gali pirminės pasirašymo grupės vardu sukurti pagrįstą autentifikuotą šifruotą tekstą ir tik paskirtasis gavėjas gali iššifruoti šifro tekstą bei patikrinti jame esančius daugiapakopius tarpinio serverio parašus. Schemos variantas su pranešimų sąsajomis dar labiau pagerina didelės žinutės šifravimą, dalindamas ją į daugybę mažesnių pranešimo blokų. Siūloma konvertuojama multi-AE tarpinio serverio schema ir jos variantas vienu metu gali atitikti ir konfidencialumo ir autentiškumo saugumo reikalavimus. Taigi, jie gali būti panaudojami į grupes orientuotuose konfidencialiuose taikymo atvejuose su tarpinio serverio įgaliojimu, pavyzdžiui, internetiniuose aukcionuose ar sutarčių pasirašyme tarpiniuose serveriuose ir pan. Jei kyla ginčas dėl atsisakymo, autorių siūloma schema paskirtam gavėjui leidžia konvertuoti šifro tekstą į originalų daugiapakopį tarpinio serverio parašą viešam patvirtinimui. Atsitiktinio orakulo modelyje įrodyta konfidencialumo apsauga nuo neatpažįstamumo adaptyvių pasirinktų šifruotų tekstų atakų (IND-CCA2) ir nuo neatskiriamumo adaptyvių pasirinktų žinučių atakų (EF-CMA) metu.