Applied Mathematics & Information Sciences
*An International Journal*

http://dx.doi.org/10.12785/amis/080338

# Convertible Authenticated Encryption Scheme with Hierarchical Access Control

*Chien-Lung Hsu*[1] *and Han-Yu Lin*[2,*]

[1] Department of Information Management, Chang Gung University, Tao-Yuan, 333, Taiwan
[2] Department of Computer Science and Engineering, National Taiwan Ocean University, Keelung, 202, Taiwan

Received: 9 Jun. 2013, Revised: 14 Oct. 2013, Accepted: 15 Oct. 2013
Published online: 1 May. 2014

**Abstract:** Convertible authenticated encryption (CAE) scheme with hierarchical access control has crucial benefits to the transmission of digital evidence. Such a scheme allows a judicial policeman to generate an authenticated ciphertext and only a designated investigator of Investigation of Bureau, Ministry of Justice (MJIB) has the ability to decrypt the ciphertext and verify the corresponding signature. The designated investigator can further convert the ciphertext into an ordinary signature and give it to a judge or a prosecutor for the litigation process. A senior manager of MJIB also has the right to take over either one or all ciphertext, i.e., digital evidence, intended for his subordinate. The underlying security assumption of our proposed scheme is based on the bilinear Diffie-Hellman problem (BDHP). We 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. Compared with related works, the proposed scheme not only provides better functionalities, but also has provable security.

**Keywords:** Convertible, authenticated encryption, hierarchical access control, bilinear pairing, random oracle.

## 1 Introduction

In 1976, Diffie and Hellman [3] introduced the first public cryptosystem in which everyone owns a self-chosen private key and the corresponding public one. It is computationally infeasible to derive the private key from its public one due to the intractability of solving the discrete logarithm problem (DLP). By encrypting messages with the recipient's public key, a sender can ensure that only the one who has the corresponding private key can decrypt the ciphertext, as so to fulfill the security requirement of confidentiality. Digital signature [4, 14] is another commonly used mechanism in a digitalized world, which could be regarded as a replacement for hand-written signature. Any signer can not deny his generated signature later, which is referred to as no-repudiation [15].

For facilitating more and more diversified applications such as credit card transactions, contract signings and on-line auctions, in 1994, Horster *et al.* [5] proposed a so-called authenticated encryption (AE) scheme which could simultaneously satisfy the properties of confidentiality [6, 9, 10] and authenticity [11, 13, 18]. In an AE scheme, a signer can generate an authenticated ciphertext such that only the designated recipient is capable of decrypting the ciphertext and then verifying the signature. However, a later dispute might occur if a dishonest signer repudiates his generated ciphertext. In 1999, Araki *et al.* [1] proposed a convertible limited verifier signature scheme to deal with the repudiation dispute. Yet, their arbitration mechanism needs the assistance of original signer to complete, which means that if the dishonest signer is not willing to cooperate with, the mechanism is unworkable. Additionally, Zhang and Kim [23] also found out that Araki *et al.*'s scheme is vulnerable to a universal forgery attack on an arbitrary chosen message.

A convertible authenticated encryption (CAE) scheme was first proposed by Wu and Hsu [19] in 2002, which preserves the merits of AE scheme and Araki *et al.*'s one. In case of a later dispute over repudiation, the designated recipient has the ability to solely convert the ciphertext into an ordinary signature for public verification. Huang and Chang [7] also proposed a CAE scheme with lower computational costs. However, Lv *et al.* [12] pointed out

* Corresponding author e-mail: lin.hanyu@msa.hinet.net

that neither the Wu-Hsu nor the Huang-Chang schemes achieve the semantic security, i.e., the ciphertext is computationally distinguishable with respect to two candidate plaintexts. In 2005, Yang [21] presented formal proofs for CAE scheme. In 2008, Chien [2] proposed a selectively CAE scheme allowing either the signer or the designated recipient to perform the signature conversion. In 2009, Lee *et al.* [16] addressed a CAE scheme based on the ElGamal cryptosystem. Wu and Lin [20] also presented a CAE scheme based on RSA cryptosystem recently.

Considering the application of computer forensics [22], in this paper, we propose a novel CAE scheme with hierarchical access control. In the proposed scheme, a judicial policeman generates an authenticated ciphertext for his collected digital evidence and then delivers it to a designated investigator of Investigation of Bureau, Ministry of Justice (MJIB). The investigator can further convert the ciphertext into an ordinary signature and give it to a judge or a prosecutor for the litigation process. A senior manager of MJIB also has the right to take over either one or all ciphertext, i.e., digital evidence, intended for his subordinate when the designated investigator resigns or just for a routine inspection.

The rest of this paper is organized as follows. Section 2 states some preliminaries. We introduce the proposed scheme in Section 3. The security proofs and some comparisons are detailed in Section 4. Finally, a conclusion is made in Section 5.

## 2 Preliminaries

In this section, we first briefly review security notions and the computational assumption with respect to the proposed scheme.

***Bilinear Pairing***

Let $(G_1, +)$ and $(G_2, \times)$ be groups of the same prime order $q$ and $e : G_1 \times G_1 \rightarrow G_2$ 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 $G_1$, then $e(P, P)$ is a generator of $G_2$.

(iii) ***Computability:***

Given $P, Q \in G_1$, the value of $e(P, Q)$ can be efficiently computed by a polynomial-time algorithm.

***Bilinear Diffie-Hellman Problem; BDHP***

Given an instance $(P, A, B, C) \in G_1^4$ where $P$ is a generator, $A = aP$, $B = bP$ and $C = cP$ for some $a$, $b$, $c$ $\in Z_q^*$ , to compute $e(P, P)^{abc} \in G_2$.

***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 the advantage at most $1/Q(k)$, i.e.,

$$\Pr[\mathcal{A}(P, aP, bP, cP) = e(P, P)^{abc}; a, b, c \leftarrow Z_q^*, P, aP, bP, cP \leftarrow G_1] \leq 1/Q(k).$$

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

## 3 The Proposed Scheme

In this section, we present our proposed scheme from bilinear pairings. We first describe composed algorithms of the proposed scheme and then give a concrete construction.

### *3.1 Involved parties*

A CAE scheme with hierarchical access control mainly has two involved parties: a signer and a designated recipient who belongs to a hierarchical organization consisting of many security clearances (SC). Each is a probabilistic polynomial-time Turing machine (PPTM). The signer can produce an authenticated ciphertext for a designated recipient in $SC_j$. Then, the designated recipient decrypts the ciphertext and verifies the signature. He can also reveal the converted signature for public verification in case of a later dispute. Any senior manager in $SC_i$ where $SC_i \succ SC_j$ also has the ability to decrypt the ciphertext intended for a designated recipient in $SC_j$.

### *3.2 Algorithms*

The proposed CAE scheme consists of the following algorithms:

**Setup($1^k$):** Taking as input $1^k$ where $k$ is a security parameter, the algorithm generates the system's public parameters $params$.

**Reg_U($i$):** The Reg_U algorithm takes as input an index $i$ and then outputs the corresponding private key $x_i$, public key $Y_i$ and the public key certificate $Cert_i$.

**SubKey_Gen($x_{AC}, ID_i, x_i, Q_i$):** The SubKey_Gen algorithm takes as input the private key $x_{AC}$ of authority center (AC), the identity $ID_i$, the private key $x_i$, and the surveillance public key $Q_i$ of user $U_i$. It outputs the corresponding surveillance parameter $SUmsg$.

**Sign_M($m, x_s, Y_v$):** The Sign_M algorithm takes as input a message $m$, the public key $Y_v$ of the designated recipient and the private key $x_s$ of signer. It generates an authenticated ciphertext $\delta$

**Verify_AEC($\delta, x_v, Y_s$):** The Verify_AEC algorithm takes as input an authenticated ciphertext $\delta$, the private key $x_v$ of the designated recipient and the public key $Y_s$ of signer. It outputs a message $m$ and its converted signature $\Omega$ if the authenticated ciphertext $\delta$ is valid. Otherwise, the symbol ¶ is returned as a result.

**Key_Derivation($x_{AC}, xx_{SU}, D_v, f_v(c)$):** The Key_Derivation algorithm takes as input the private key $x_{AC}$ of authority center (AC), the surveillance private key $xx_{SU}$ of senior manager $U_{SU}$ and two surveillance parameters ($D_v$, $f_v(c)$). It outputs the private key $x_v$ with respect to user $U_v$.

**M_Derivation($x_{AC}$, $Y_s$, $SPK_v$, $ssk_v$, $R$, $\sigma$, $r$):** The M_Derivation algorithm takes as input the private key $x_{AC}$ of authority center (AC), the public key $Y_s$ of signer, the surveillance parameter ($SPK_v$, $ssk_v$) and an authenticated ciphertext $(R, \sigma, r)$. It outputs the recovered message $m$.

### 3.3 Concrete Construction

**Setup($1^k$):** Taking as input $1^k$, the System Authority (SA) selects two groups $(G_1, +)$ and $(G_2, \times)$ of the same prime order $q$ where $|q| = k$. Let $P$ be a generator of order $q$ over $G_1$, $e$: $G_1 \times G_1 \to G_2$ a bilinear pairing and $h_1$: $\{0,1\}^k \times G_1 \to Z_q$, $h_2$: $G_1 \times G_1 \times G_2 \to \{0,1\}^k$, $h_3$: $G_1 \to G_1$ and $h_4$: $Z_q \to Z_q$ collision resistant hash functions. The algorithm outputs public parameters $params = \{G_1, G_2, q, P, e, h_1, h_2, h_3, h_4\}$.

**Reg_U($i$):** On input an index $i$, Reg_U algorithm chooses a private key $x_i \in Z_q$, computes the public key $Y_i = x_i P$ and then further generates the public key certificate $Cert_i$ by the X.509 standard [8].

**SubKey_Gen($x_{AC}, ID_i, x_i, Q_i$):** Let AC associated with the key pair $(x_{AC}, Y_{AC} = x_{AC}P)$ be an authority center in the hierarchical organization composed of many security clearances. The diagram of the structure of security clearances is depicted as Figure 1. Each user $i$ of the hierarchical organization first generates his surveillance key pair ($xx_i \in Z_q$, $Q_i = xx_i P$) and then sends ($ID_i$, $x_i$, $Q_i$) to AC via a secure channel. Upon receiving it, the AC chooses $d_i \in Z_q^*$, for $1 \le i \le n$ where $n$ is the number of users in the organization, to compute

$$D_i = d_i P, \tag{1}$$

$$f_i(c) = \prod_j (c - e(d_i h_4(x_{AC} \| S\_data_i) Q_j, d_i Q_j)) + x_i, \tag{2}$$

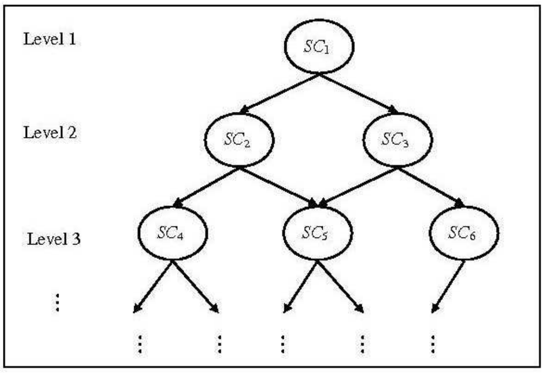


**Fig. 1** Diagram of the structure of security clearances

where $SC_i \prec SC_j$ and $S\_data_i$ is the surveillance information such as the distinguishable identifiers of senior managers, subordinates and surveillance cases.

$$SPK_i = h_4(x_{AC} \| S\_data_i)P, \tag{3}$$

$$ssk_i = x_i - h_4(x_{AC} \| S\_data_i)(SPK_i.x) \bmod q, \tag{4}$$

and then outputs surveillance parameter $SUmsg$ = ($SPK_i$, $ssk_i$, $D_i$, $f_i(c)$).

**Sign_M($m$, $x_s$, $Y_v$):** On input a message $m$, the public key $Y_v$ of the designated recipient and the private key $x_s$ of signer, the algorithm chooses $t \in Z_q^*$ to compute

$$R = tP, \tag{5}$$

$$\sigma = \frac{1}{x_s + h_1(m, R)} R, \tag{6}$$

$$W = h_3(tY_v), \tag{7}$$

$$Z = e(x_s Y_v, W), \tag{8}$$

$$r = m \oplus h_2(R, \sigma, r), \tag{9}$$

and then outputs the authenticated ciphertext $\delta$ = ($R$, $\sigma$, $r$).

**Verify_AEC($\delta$, $x_v$, $Y_s$):** On input an authenticated ciphertext $\delta$ = ($R$, $\sigma$, $r$), the private key $x_v$ of designated recipient and the public key $Y_s$ of signer, the algorithm first computes

$$W = h_3(x_v R), \tag{10}$$

$$Z = e(x_v Y_s, W), \tag{11}$$

to recover the message $m$ as

$$m = r \oplus h_2(R, \sigma, Z), \tag{12}$$

and then checks the redundancy embedded in $m$. The algorithm further verifies the signature by checking if

$$e(\sigma, Y_s + h_1(m, R)P) = e(R, P), \tag{13}$$

If it holds, the message $m$ and its converted signature $\Omega$ = ($R$, $\sigma$) is outputted; else, the error symbol ¶ is returned as a result.

We prove that Eqs. (12) and (13) work correctly. From the right-hand side of Eq. (12), we have

$$\begin{aligned} & r \oplus h_2(R, \sigma, Z) \\ &= r \oplus h_2(R, \sigma, e(x_v Y_s, W)) && \text{(by Eq. (11))} \\ &= r \oplus h_2(R, \sigma, e(x_v Y_s, h_3(x_v R))) && \text{(by Eq. (10))} \\ &= r \oplus h_2(R, \sigma, e(x_s Y_v, h_3(t Y_v))) && \text{(by Eq. (7))} \\ &= r \oplus h_2(R, \sigma, Z) && \text{(by Eq. (8))} \\ &= m && \text{(by Eq. (9))} \end{aligned}$$

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

If an authenticated ciphertext ($R$, $\sigma$, $r$) is correctly generated, it will pass the test of Eq. (13). From the left-hand side of Eq. (13), we have

$$\begin{aligned} & e(\sigma, Y_s + h_1(m, R)P) \\ &= e(\tfrac{1}{x_s + h_1(m,R)} R, Y_s + h_1(m, R)P) && \text{(by Eq. (6))} \\ &= e(\tfrac{1}{x_s + h_1(m,R)} R, (h_1(m, R) + x_s)P) \\ &= e(R, P) \end{aligned}$$

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

**Key_Derivation(**$x_{AC}$, $xx_{SU}$, $D_v$, $f_v(c)$**):** When a senior manager $U_{SU}$ wants to take over all ciphertexts intended for $U_v$ where $SC_v \prec SC_{SU}$, $U_{SU}$ sends a request to the AC. After approving the request, the AC computes

$$ES_v = h_4(x_{AC} \| S\_data_v) D_v, \tag{14}$$

and then returns $ES_v$ to $U_{SU}$ via a secure channel. $U_{SU}$ can derive $U_v$'s private key as

$$x_v = f_v(e(xx_{SU} ES_v, xx_{SU} D_v)). \tag{15}$$

**M_Derivation(**$x_{AC}$, $Y_s$, $SPK_v$, $ssk_v$, $R$, $\sigma$, $r$**):** When a senior manager $U_{SU}$ just wants to take over one ciphertext intended for $U_v$ where $SC_v \prec SC_{SU}$, $U_{SU}$ sends a request to the AC. After approving the request, the AC computes

$$ES_{v,1} = h_4(x_{AC} \| S\_data_v) Y_s, \tag{16}$$

$$ES_{v,2} = h_4(x_{AC} \| S\_data_v) R, \tag{17}$$

and then returns ($ES_{v,1}$, $ES_{v,2}$) to $U_{SU}$ via a secure channel. $U_{SU}$ can further derive

$$W = h_3(ssk_v R + (SPK_{v.x}) ES_{v,2}), \tag{18}$$

$$Z = e(ssk_v Y_s + (SPK_{v.x}) ES_{v,1}, W), \tag{19}$$

and then recover $m$ with Eq. (12).

## 4 Security proof

In this section, we first state the security model of our proposed scheme and prove it in the random oracle model. Then some comparisons to related schemes are also made.

### 4.1 Security model

We define two security models for the proposed scheme in relation to confidentiality and unforgeability as follows:

**Definition 1. (Confidentiality)** *A CAE 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:

- *Reg_U query* $\langle i \rangle$*:* $\mathcal{A}$ makes an Reg_U query $\langle i \rangle$. $\mathcal{B}$ returns the corresponding public key $Y_i$ and the public key certificate $Cert_i$ to $\mathcal{A}$.
- *Sign_M query* $\langle m, Y_s, Y_v \rangle$*:* $\mathcal{A}$ makes an Sign_M query $\langle m, Y_s, Y_v \rangle$. $\mathcal{B}$ returns the corresponding authenticated ciphertext $\delta$ to $\mathcal{A}$.
- *Verify_AEC query* $\langle \delta, Y_s, Y_v \rangle$*:* $\mathcal{A}$ makes a Verify_AEC query $\langle \delta, Y_s, Y_v \rangle$. If $\delta$ is valid, $\mathcal{B}$ returns the recovered message $m$ and its converted signature $\Omega$; 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 $\lambda \longleftarrow \{0, 1\}$ and generates an authenticated ciphertext $\delta^*$ for $m_\lambda$. The ciphertext $\delta^*$ 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 Verify_AEC for the target ciphertext.

**Guess:** At the end of the game, $\mathcal{A}$ outputs a bit $\lambda'$. 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 2. (Unforgeability)** *A CAE 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($1^k$):** $\mathcal{B}$ first runs the Setup($1^k$) algorithm and sends system's public parameters $params$ to the adversary $\mathcal{A}$.

**Phase 1:** The adversary $\mathcal{A}$ adaptively issues Reg_U and Sign_M queries as those defined in Phase 1 of Definition

1.

**Forgery:** Finally, $\mathcal{A}$ produces an authenticated ciphertext $\delta^*$ for some message $m^*$. Note that $\delta^*$ is not outputted by the Sign_M query $\langle m, Y_s, Y_v\rangle$. The adversary $\mathcal{A}$ wins if $\delta^*$ is valid.

## *4.2 Security proofs*

We 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, $q_{h_1}$, $q_{h_2}$, $q_{h_3}$, $q_{h_4}$, $q_{Reg\_U}$, $q_{Sign\_M}$, $q_{Verify\_AEC}$, $\epsilon$)-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 ($t'$, $\epsilon'$)-break the BDHP, where*

$$\epsilon' \geq (2\epsilon - 2^{-k}(q_{Verify\_AEC}))/(q_{h_2}q_{h_3}),$$
$$t' \approx t + t_\lambda(2q_{Verify\_AEC}).$$

*Here $t_\lambda$ is the time for performing one bilinear pairing computation.*

**Proof:** Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can ($t$, $q_{h_1}$, $q_{h_2}$, $q_{h_3}$, $q_{h_4}$, $q_{Reg\_U}$, $q_{Sign\_M}$, $q_{Verify\_AEC}$, $\epsilon$)-break the proposed scheme with non-negligible advantage $\epsilon$ 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 4), $q_{Reg\_U}$ *Reg_U* queries, $q_{Sign\_M}$ *Sign_M* and $q_{Verify\_AEC}$ $Verify\_AEC$ queries. Then we can construct another algorithm $\mathcal{B}$ that ($t'$, $\epsilon'$)-breaks the BDHP by taking $\mathcal{A}$ as a subroutine. 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 and sends public parameters $params$ = $\{G_1, G_2, q, P, e\}$ to the adversary $\mathcal{A}$.

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

- $h_1$ *oracle:* When $\mathcal{A}$ makes an $h_1$ oracle of ($m$, $R$), $\mathcal{B}$ first searches the $h_1$-list for a matched entry. Otherwise, $\mathcal{B}$ chooses $v_1 \in_R Z_q$ and adds the entry ($m$, $R$, $v_1$) into $h_1$-list. Finally, $\mathcal{B}$ returns $v_1$ as a result.

- $h_2$ *oracle:* When $\mathcal{A}$ makes an $h_2$ oracle of ($R$, $\sigma$, $Z$), $\mathcal{B}$ first searches the $h_2$-list for a matched entry. Otherwise, $\mathcal{B}$ chooses $v_2 \in_R \{0,1\}^k$ and adds ($R$, $\sigma$, $Z$, $v_2$) into $h_2$-list. Finally, $\mathcal{B}$ returns $v_2$ as a result.

- $h_3$ *oracle:* When $\mathcal{A}$ makes an $h_3$ oracle of $(tY_v)$, $\mathcal{B}$ first searches the $h_3$-list for a matched entry. Otherwise, $\mathcal{B}$ chooses $v_3 \in_R G_1$ and adds ($tY_v$, $v_3$) into $h_3$-list. Finally, $\mathcal{B}$ returns $v_3$ as a result.

- $h_4$ *oracle:* When $\mathcal{A}$ makes an $h_4$ oracle of $w$, $\mathcal{B}$ first searches the $h_4$-list for a matched entry. Otherwise, $\mathcal{B}$ chooses $v_4 \in_R Z_q$ and adds the entry ($w$, $v_4$) into $h_4$-list. Finally, $\mathcal{B}$ returns $v_4$ as a result.

- *Reg_U query $\langle i\rangle$:* When $\mathcal{A}$ makes an Reg_U query $\langle i\rangle$, $\mathcal{B}$ responds as follows. If $i$ = $s$, $\mathcal{B}$ returns ($Y_s = aP$, $Cert_s$) to $\mathcal{A}$. If $i$ = $v$, $\mathcal{B}$ returs ($Y_v = bP$, $Cert_v$) to $\mathcal{A}$. Otherwise, $\mathcal{B}$ runs Reg_U($i$) and then returns ($Y_i$, $Cert_i$) to $\mathcal{A}$.

- *Sign_M query $\langle m, Y_i, Y_j\rangle$:* When $\mathcal{A}$ makes an Sign_M query for some message $m$ with respect to the public keys ($Y_i$, $Y_j$), $\mathcal{B}$ returns Sign_M($m$, $x_i$, $Y_j$) as a result if $Y_i \neq aP$. When $Y_i = aP$, $\mathcal{B}$ performs the following steps:
**Step 1** Choose $d, v_1 \in_R Z_q$ and $v_2 \in_R \{0,1\}^k$;
**Step 2** Compute $\sigma = dP$, $r$ = $m \oplus v_2$ and $R$ = $d(aP) + v_1dP$;
**Step 3** Add the entry ($m$, $R$, $v_1$) into $h_1$-list, i.e., define $h_1(m, R)$ = $v_1$;
**Step 4** Implicitly define $h_2(R, \sigma, Z)$ = $v_2$ and $\mathcal{B}$ doesn't know $Z$.
The ciphertext $\delta$ = ($R$, $\sigma$, $r$) is then returned to $\mathcal{A}$.

- *Verify_AEC query $\langle\delta, Y_i, Y_j\rangle$:* When $\mathcal{A}$ makes a Verify_AEC query for some authenticated ciphertext $\delta$ = ($R$, $\sigma$, $r$) with respect to the public keys ($Y_i$, $Y_j$), $\mathcal{B}$ performs the following steps:
**Step 1** Search the $h_1$-list for any matched entry ($m^*$, $R^*$, $v_1^*$) where $R^*$ = $R$;
**Step 2** If one satisfies $e(\sigma, Y_i + h_1(m^*, R)P)$ = $e(R, P)$, $\mathcal{B}$ outputs ($m^*$, $R$, $\sigma$); else, $\mathcal{B}$ returns the error symbol ¶.

**Challenge:** $\mathcal{A}$ generates two messages, $m_0$ and $m_1$, of the same length. The challenger $\mathcal{B}$ flips a coin $\lambda \longleftarrow \{0,1\}$ and produces an authenticated ciphertext $\delta^*$ for $m_\lambda$ as follows:
**Step 1** Choose $d, v_1 \in_R Z_q$ and $v_2 \in_R \{0,1\}^k$;
**Step 2** Compute $\sigma^* = dP$, $r^*$ = $m_\lambda \oplus v_2$ and $R^*$ = $d(aP) + v_1dP$;
**Step 3** Add the entry ($m_\lambda$, $R^*$, $v_1$) into $h_1$-list, i.e., define $h_1(m_\lambda, R^*)$ = $v_1$;
**Step 4** Implicitly define $h_2(R^*, \sigma^*, Z^*)$ = $v_2$ and $\mathcal{B}$ doesn't know $Z^*$.
The ciphertext $\delta^*$ = ($R^*$, $\sigma^*$, $r^*$) is then delivered to $\mathcal{A}$ as a target challenge.

**Phase 2:** $\mathcal{A}$ makes new queries as those stated in Phase 1 except the Verify_AEC query for the target ciphertext $\delta^*$. Note that in the $j$-th $h_3$ oracle query, where $1 \leq j \leq q_{h_3}$,

$\mathcal{B}$ directly returns $cP$.

**Analysis of the game:** For each Sign_M query, $\mathcal{B}$ always returns a valid authenticated ciphertext. Hence, the simulated result of Sign_M query is computationally indistinguishable from the one generated by a real scheme. Consider the simulation of Verify_AEC query. One can observe that it is possible for a Verify_AEC query to return the error symbol ¶ for a valid ciphertext $\delta$ on condition that $\mathcal{A}$ is able to generate $\delta$ without asking the corresponding $h_1(m,\ R)$ random oracle. Let VLD, ERR and $\mathrm{QH}_1$ separately be the events that a ciphertext submitted by $\mathcal{A}$ is valid, a Verify_AEC query finally returns the error symbol ¶ for some valid ciphertext during the entire simulation game, and $\mathcal{A}$ has ever asked the corresponding $h_1(m,\ R)$ random oracle for his submitted ciphertext. Then we can express the error probability of any Verify_AEC query as $Pr[\mathrm{VLD} \mid \neg\mathrm{QH}_1] \leq 1/2^k$. Since $\mathcal{A}$ can make at most $q_{Verify\_AEC}$ Verify_AEC queries, we can further express the probability of ERR as

$$Pr[\mathrm{ERR}] \leq 2^{-k}(q_{Verify\_AEC}). \quad (20)$$

In the challenge phase, $\mathcal{B}$ has returned a simulated authenticated ciphertext $\delta^*$ = ($R^*$, $\sigma^*$, $r^*$) where $h_2(R^*,\ \sigma^*,\ Z^*)$ = $v_2$, which implies the shared secret $Z^*$ is implicitly defined as $Z^*$ = $e(b(aP),\ h_3(bR))$. Let NA be the event that the entire simulation game does not abort. It can be seen that if the adversary $\mathcal{A}$ never makes an $h_2(R^*,\ \sigma^*,\ Z^*)$ query in Phase 2, denoted by $\neg\mathrm{QH}_2^*$, the entire simulation game could be normally terminated. When the entire simulation game does not abort, $\mathcal{A}$ gains no advantage in guessing $\lambda$ due to the randomness of the output of the random oracle, i.e.,

$$Pr[\lambda' = \lambda \mid \mathrm{NA}] = 1/2. \quad (21)$$

Derived from the left-hand side of Eq. (21), we have

$$\begin{aligned} Pr[\lambda' = \lambda] &= Pr[\lambda' = \lambda \mid \mathrm{NA}]Pr[\mathrm{NA}] \\ &\quad + Pr[\lambda' = \lambda \mid \neg\mathrm{NA}]Pr[\neg\mathrm{NA}] \\ &\leq (1/2)Pr[\mathrm{NA}] + Pr[\neg\mathrm{NA}] \quad \text{(by Eq. (21))} \\ &= (1/2)(1 - Pr[\neg\mathrm{NA}]) + Pr[\neg\mathrm{NA}] \\ &= (1/2) + (1/2)Pr[\neg\mathrm{NA}]. \end{aligned} \quad (22)$$

On the other hand, we can also derive that

$$\begin{aligned} Pr[\lambda' = \lambda] &\geq Pr[\lambda' = \lambda \mid \mathrm{NA}]Pr[\mathrm{NA}] \\ &= (1/2)(1 - Pr[\neg\mathrm{NA}]) \\ &= (1/2) - (1/2)Pr[\neg\mathrm{NA}]. \end{aligned} \quad (23)$$

Combining inequalities (22) and (23), we can obtain that

$$|Pr[\lambda' = \lambda] - 1/2\ | \leq (1/2)Pr[\neg\mathrm{NA}]. \quad (24)$$

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

$$\begin{aligned} \epsilon &= |Pr[\lambda' = \lambda] - 1/2| \\ &\leq (1/2)Pr[\neg\mathrm{NA}] \qquad (by Eq.(24)) \\ &= (1/2)Pr[\mathrm{QH}_2^* \vee \mathrm{ERR}] \\ &\leq (1/2)(Pr[\mathrm{QH}_2^*] + Pr[\mathrm{ERR}]) \end{aligned}$$

Combining Eq. (20) and rewriting the above inequality, we have

$$\begin{aligned} Pr[\mathrm{QH}_2^*] &\geq 2\epsilon - Pr[\mathrm{ERR}] \\ &\geq 2\epsilon - 2^{-k}(q_{Verify\_AEC}). \end{aligned}$$

As in the $j$-th $h_3$ oracle query, where $j \leq q_{h_3}$, $\mathcal{B}$ directly returns $cP$, if the event $\mathrm{QH}_2^*$ happens, we claim that the value $Z^*$ = $e(b(aP), cP)$ will be stored in some entry of the $h_2$-list. Hence, $\mathcal{B}$ will has non-negligible probability

$$\epsilon' \geq (2\epsilon - 2^{-k}(q_{Verify\_AEC}))/(q_{h_2}q_{h_3})$$

to solve the BDHP. The computational time required for $\mathcal{B}$ is $t' \approx t + t_\lambda(2q_{Verify\_AEC})$.

Q.E.D.

**Theorem 2. (Proof of Unforgeability)** *The proposed scheme is ($t$, $q_{h_1}$, $q_{h_2}$, $q_{h_3}$, $q_{h_4}$, $q_{Reg\_U}$, $q_{Sign\_M}$, $\epsilon$)-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 ($t'$, $\epsilon'$)-break the BDHP problem, where*

$$\begin{aligned} &\epsilon' \geq (\epsilon - 2^{-k})/(q_{h_2}q_{h_3}), \\ &t' \approx t. \end{aligned}$$

**Proof:** Suppose that a probabilistic polynomial-time adversary $\mathcal{A}$ can ($t$, $q_{h_1}$, $q_{h_2}$, $q_{h_3}$, $q_{h_4}$, $q_{Reg\_U}$, $q_{Sign\_M}$, $\epsilon$)-break the proposed scheme with non-negligible advantage $\epsilon$ 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 4), $q_{Reg\_U}$ Reg_U queries and $q_{Sign\_M}$ Sign_M queries. Then we can construct another algorithm $\mathcal{B}$ that($t'$, $\epsilon'$)-breaks the BDHP by taking $\mathcal{A}$ as a subroutine. 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 and sends public parameters $params$ = $\{G_1, G_2, q, P, e\}$ to the adversary $\mathcal{A}$.

**Phase 1:** $\mathcal{A}$ adaptively asks $h_i$ random oracle (for $i$ = 1 to 4), Reg_U and Sign_M queries as those defined in Theorem 1. Note that in the $j$-th $h_3$ oracle query, where $1 \leq j \leq q_{h_3}$, $\mathcal{B}$ directly returns $cP$.

**Forgery:** $\mathcal{A}$ outputs a forged authenticated ciphertext $\delta^*$ = $(R^*, \sigma^*, \gamma^*)$ for his arbitrarily chosen message $m^*$. If $\delta^*$ is valid, $\mathcal{A}$ wins the game.

**Analysis of the game:** For each random oracle query, $\mathcal{B}$ returns with a computationally indistinguishable value without collision. The simulation of Sign_M query could be regarded as perfect, as it always outputs a valid ciphertext without being accidentally terminated. Let VLD and $QH_2$ separately be the events that the ciphertext $\delta^*$ forged by $\mathcal{A}$ is valid and $\mathcal{A}$ has ever asks the corresponding $h_2$ random oracle. The probability that $\mathcal{A}$ can guess the correct random value without querying the random oracle is not greater than $2^{-k}$. Since $\mathcal{A}$ has non-negligible advantage $\epsilon$ to break the proposed scheme under adaptive chosen-message attacks, we can derive

$$\begin{aligned}\epsilon &= Pr[\text{VLD}] \\ &\leq Pr[\text{VLD} \mid \text{QH}_2] + Pr[\text{AC-V} \mid \neg\text{QH}_2] \\ &\leq Pr[\text{VLD} \mid \text{QH}_2] + 2^{-k}.\end{aligned}$$

Writing the above inequality, we can also obtain

$$Pr[\text{VLD} \mid \text{QH}_2] \geq \epsilon - 2^{-k}.$$

Seeing that in the $j$-th $h_3$ random oracle, the challenger $\mathcal{B}$ directly returned $cP$ as the result, we claim that when the event (VLD | $QH_2$) occurs, $\mathcal{B}$ would have the probability of $(q_{h_2} q_{h_3}^{-1})$ to output $Z = e(P, P)^{abc}$ from some entry of the $h_2$-list. Therefore, we can express the probability of $\mathcal{B}$ to solve the BDHP as $\epsilon' \geq (\epsilon - 2^{-k})/(q_{h_2} q_{h_3})$. The running time required for $\mathcal{B}$ is $t' \approx t$.

Q.E.D.

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

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

### *4.3 Comparisons*

We compare the proposed scheme with some previous ones including Araki *et al.*'s (AUI for short) [1], Sekhar's (Sek for short) [17], the Wu-Hsu (WH for short) [19], Lee *et al.*'s (LHT for short) [16], Chien's (Chi for short) [2] and the Wu-Lin (WL for short) [20] schemes. Detailed comparisons in terms of functionalities and security are demonstrated as Table 1. From this table, it can be seen that the proposed scheme not only provides better functionalities, but also has provable security.

**Table 1** Comparisons in terms of functionalities and security

| Item \ Scheme | AUI Sek | WH | LHT | Chi WL | Ours |
|---|---|---|---|---|---|
| Hierarchical architecture | X | X | X | X | V |
| Access control | X | X | X | X | V |
| Signature conversion | V | V | V | V | V |
| No conversion cost | X | V | V | V | V |
| Proof of Confidentiality | V | X | X | V | V |
| Proof of Unforgeability | X | V | X | V | V |

## 5 Conclusion

In this paper, we proposed a novel CAE scheme with hierarchical access control from bilinear pairings. To the best of our knowledge, this is the first CAE scheme combining with hierarchical access control and has crucial benefits to the application of computer forensics. Without the help of signer, the designated recipient is capable of solely revealing the ordinary signature for public verification. If necessary, a senior manager with higher security clearance can take over the ciphertext intended for his subordinates. We also demonstrate 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. Compared with previous related works, ours not only provides better functionalities, but also has provable security.

## Acknowledgement

We would like to thank anonymous referees for their valuable suggestions. We thank Healthy Aging Research Center (HARC) of Chang Gung University for excellent technical assistance. This work was supported in part by the Chang Gung University Grant UARPD3B0061 and in part by the National Science Council of Republic of China under the contract numbers NSC 100-2628-H-182-001-MY3 and NSC 101-2218-E-019-005.

## References

[1] S. Araki, S. Uehara and K. Imamura, The limited verifier signature and its application, *IEICE Transactions on Fundamentals*, **E82-A**, 63-68 (1999)

[2] H.Y. Chien, Selectively convertible authenticated encryption in the random oracle model, *The Computer Journal*, **51**, 419-434 (2008).

[3] W. Diffie and M. Hellman, New directions in cryptography, *IEEE Transactions on Information Theory*, **IT-22**, 644-654 (1976).

[4] T. ElGamal, A public key cryptosystem and a signature scheme based on discrete logarithms, *IEEE Transactions on Information Theory*, **IT-31**, 469-472 (1985).

[5] P. Horster, M. Michel and H. Peterson, Authenticated encryption schemes with low communication costs, *Electronics letters*, **30**, 1212-1213 (1994).

[6] F. Hou, Z. Wang, Y. Tang and Z. Liu, Protecting integrity and confidentiality for data communication, *Proceedings of the 9th International Symposium on Computers and Communications (ISCC)*, **1**, 357-362 (2004).

[7] H. F. Huang and C. C. Chang, An efficient convertible authenticated encryption scheme and its variant, *Proceedings of the 5th International Conference on Information and Communications Security (ICICS2003)*, Springer-Verlag, Berlin, 382-392 (2003).

[8] ISO/IEC 9594-8, Information technology open systems interconnection the directory: public-key and attribute certificate frameworks, International Organization for Standardization, (2001).

[9] J. Jacob, A uniform presentation of confidentiality properties, *IEEE Transactions on Software Engineering*, **17**, 1186-1194 (1991).

[10] V. A. Ustimenko and Y. M. Khmelevsky, Walks on graphs as symmetric or asymmetric tools to encrypt data, The South Pacific Journal of Natural Science, **20**, 34-44 (2002).

[11] M. R. Girgis, T. M. Mahmoud, H. F. Abd El-Hameed and Z. M. El-Saghier. Routing and Capacity Assignment Problem in Computer Networks Using Genetic Algorithm. Information Science Letters, **2**, 13-25 (2013).

[12] J. Lv, X. Wang and K. Kim, Practical convertible authenticated encryption schemes using self-certified public keys, *Applied Mathematics and Computation*, **169**, 1285-1297 (2005).

[13] Bo Zhang and Zhicai Juan. Modeling User Equilibrium and the Day-to-day Traffic Evolution based on Cumulative Prospect Theory. Information Science Letters, **2**, 9-12 (2013).

[14] R. Rivest, A. Shamir and L. Adleman, A method for obtaining digital signatures and public-key cryptosystems, *Communications of the ACM*, **21**, 120-126 (1978).

[15] S. Schneider, Formal analysis of a non-repudiation protocol, *Proceedings of 11th IEEE Computer Security Foundations Workshop*, IEEE Press, Piscataway, USA, 54-65 (1998).

[16] C. C. Lee, M. S. Hwang and S. F. Tzeng, A new convertible authenticated encryption scheme based on the ElGamal cryptosystem, *International Journal of Foundations of Computer Science*, **20**, 351-359 (2009).

[17] M. R. Sekhar, Signatures scheme with message recovery and its applications, *International Journal of Computer Mathematics*, **81**, 285-289 (2004).

[18] W. Stallings, *Cryptography and Network Security: Principles and Practices*, 4th Ed., Pearson, (2005).

[19] T. S. Wu and C. L. Hsu, Convertible authenticated encryption scheme, *The Journal of Systems and Software*, **62**, 205-209 (2002).

[20] T.S. Wu and H.Y. Lin, Secure convertible authenticated encryption scheme based on RSA, *Informatica*, **33**, 481-486 (2009).

[21] F. Y. Yang, A secure scheme for authenticated encryption, *Cryptology ePrint Archive*, Report 2005/456, . http://eprint.iacr.org/2005/456, (2005.)

[22] William Enck. A Study of Android Application Security. USENIX Security Symposium, (2011)

[23] F. Zhang and K. Kim, A universal forgery on Araki et al.'s convertible limited verifier signature scheme, *IEICE Transactions on Fundamentals*, **E86-A**, 515-516 (2003).


**Chien-Lung Hsu** received a B.S. degree in business administration, an M.S. degree in information management, and a Ph.D. degree in information management from the National Taiwan University of Science and Technology, Taiwan in 1995, 1997, and 2002, respectively. He was an Assistant Professor and an Associate Professor in the Department of Information Management, Chang Gung University (CGU), Taiwan from 2004 to 2007 and from 2007 to 2011, respectively. Currently, he is a Professor in the Department of Information Management, Chang Gung University since 2011. He is also the leader of the Ubiquitous Security and Applications Lab, the director of Chinese Cryptology Information Security Association (CCISA, Taiwan), the chair of Education Promotion Committee of CCISA, the member of Academia-Industry Cooperation Committee of CCISA, the chair of Program of RFID Applications in Logistics Supply Chain Management of CGU, the chair of Program of Information Security with Medical Applications of CGU, the director of Division of Instructional Support of Computer Center of CGU, the researcher of Healthy Aging Research Center (HARC) of CGU, the researcher of Elder Industry Development and Research Center (EIDRC) of CGU, and the senior researcher of Taiwan Information Security Center (TWISC). His current research includes cryptography, information security, wireless sensor network, mobile commerce, digital forensics, vehicular system security, healthcare system and user acceptance, smart home system, and etc.

**Han-Yu Lin** received BA degree in economics from the Fu-Jen University, Taiwan in 2001, his MS degree in information management from the Huafan University, Taiwan in 2003, and his Ph.D. degree in computer science and engineering from the National Chiao Tung University, Taiwan in 2010. He has been an Assistant Professor in the Department of Computer Science and Engineering of National Taiwan Ocean University since August 2012. His research interests include Cryptology, Network Security, Digital Forensics, Cloud Computing Security and E-commerce Security.