An Efficient and Secure Pairing-Based Fair Blind Signature Scheme with Message Recovery

WOEI-JIUNN TSAUR\(^1\) and CHII-JYH GUO\(^2\)

\(^1\)Department of Information Management
Da-Yeh University
112 Shan-Jiau Rd., Da-Tsuen, Changhua 515
TAIWAN

\(^2\)Department of Information Management
Da-Yeh University
112 Shan-Jiau Rd., Da-Tsuen, Changhua 515
TAIWAN

Abstract: - In this paper, we propose a new fair blind signature scheme with message recovery using pairing-based self-certified public key cryptosystems. Preserving the merits inherent in pairing-based cryptosystems, it can possess fewer bits to achieve the higher security level. In addition, our new scheme has the advantage that the authentication of the public key can be accomplished with the verification of the fair blind signature in a logically single step. Furthermore, the fairness of blind signature with message recovery can be actually achieved in our proposed scheme. Based on the proposed security proofs and performance evaluation, we affirm that we not only improve the efficiency of the previously proposed schemes, but also achieve the essential properties of blind signature with provable security.

Key-Words: - Cryptography, Provable security, Fair Blind Signature, Pairing-based public key cryptosystem, Self-certified public key cryptosystem, Electronic payment system

1 Introduction
The blind signature scheme, first proposed by Chaum [1] in 1983, allows users to achieve anonymous property in electronic voting systems and electronic cash payment systems. With the characteristic of the blind signature scheme, a sender can obtain a signature on a message from a signer, but the signer knows nothing about the content of the message, such that the signer cannot link the signature and sender. Unfortunately, this characteristic may be used to pervert the ability of the scheme. Therefore, in 1999 Lee and Kim [8] proposed the fair blind signature scheme with message recovery to withstand the misapplication of financial crime in electronic cash payment systems. However, in 2000 Hsien et al. [6] proposed an attack on Lee and Kim's scheme. They proved that the sender can generate an untraceable signature, which cannot be recovered by the system authority (the trusted entity). In 2002, Chung [2] improved the checking way of the revocation key in Lee and Kim's scheme such that the sender cannot create a pretended revocation key to satisfy the fair requirement. Regrettably, Chung's proposed scheme, which was based on modular exponentiation, is inefficient. Thus, in order to gain much efficiency in saving both the communicational cost and the computational effort, Tsaur and Chou [12] proposed an ECC-based Fair Blind Signature Scheme with Message Recovery. However, Tsaur and Chou’s method does not give security proofs on the blindness and non-forgeability properties of the proposed blind signature scheme.

In recent years, Zhang et al. [15, 16] proposed several kinds of ID-based blind signature schemes using the bilinear pairings. Although the ID-based cryptosystems [11] have the advantage of simple procedure in managing the public key list, a secure channel is required for the key generation center to deliver private keys to corresponding users. Also, the key generation center is a single point of failure in the systems. If the private key of the key generation center is compromised, the security of the entire scheme will be removed. Moreover, a dishonest key generation center may impersonate each user in the
systems, because each user's private key is generated by it. Thus there exist many drawbacks in identity-based public key cryptosystems. In 1991, Girault [4] proposed the self-certified public key cryptosystem, which can implicitly verify public keys without accompanying additional certificates. Self-certified public key cryptosystems can allow a user to generate the secret key by himself/herself (i.e. the secret key needn't be transmitted through a secure channel). Thus the system authority cannot obtain the user's secret key from communications with the user [14]. Moreover, the user and the system authority cooperatively generate the user's public key, and the user can verify the public key by himself/herself when the system authority delivers the public key to him/her. Consequently, the system authority cannot impersonate any user by generating false guarantees, and all frauds of the system authority are detectable.

In this paper, the proposed cryptosystems are first constructed by using the pairing-based cryptosystems [13] instead of modular exponentiation, and further integrating the identity-based self-certified public key cryptosystems. Furthermore, we employ the integrated cryptosystems to design a fair blind signature scheme with message recovery to improve the drawback on Lee and Kim's scheme, and give security proofs on the proposed blind signature scheme.

The remainder of this paper is organized as follows. In Section 2, we first propose the self-certified pairing-based public key cryptosystems, and further develop a fair blind signature scheme with message recovery. We then prove the security of the proposed blind signature scheme in Section 3, and analyze the computational complexity of the proposed scheme in Section 4. Finally, some concluding remarks are presented in Section 5.

2 Proposed Fair Blind Signature Scheme with Message Recovery

In this section, we propose a public key cryptosystem by integrating the pairing-based cryptosystems with the identity-based self-certified public key cryptosystems. In addition, we further employ the integrated cryptosystems to design a fair blind signature scheme with message recovery to efficiently achieve the essential properties of blind signature. The proposed scheme is described as follows.

2.1 Initialization

The entities in the system are a certification authority (CA) and users (U). Assume that the system authority CA is responsible for key generation and user registration. We define notations used in the proposed scheme as follows:

- \( E(F_r) \) : a supersingular elliptic curve \( E: y^2=x^3-x+1 (mod_{3^m}) \), where the characteristic is 3, and the security multiplier is 6.
- \( G_1 \) : an additive group of the elliptic curve \( E \) whose order is a large prime \( q \). We also write \( G_1^* = G_1 - \{O\} \), and \( O \) is the point at infinity.
- \( B \) : a base point of \( G_1 \) whose order is \( q \).
- \( G_2 \) : a multiplicative group of order \( q \) on the elliptic curve \( E \).
- \( e \) : a bilinear pairing map where \( e: G_1 \times G_1 \rightarrow G_2 \).
- \( H_1 \) : a one-way hash function, where \( H_1: \{0,1\}^n \rightarrow G_1^* \).
- \( H_2 \) : a one-way hash function, where \( H_2: \{0,1\}^n \rightarrow \mathbb{Z}_q^* \).
- \( H_3 \) : a one-way hash function \( H_3: G_2 \rightarrow \{0,1\}^n \), where \( n \in N \) denotes the size of message.

2.2 The Integrated Public Key Cryptosystems

The operational procedure of the proposed integrated public key cryptosystems is divided into two phases: system setup and key generation.

[System Setup]

CA creates a system public key and some public parameters in this phase, and then SA releases these parameters.

CA randomly chooses a number \( s_{CA} \in \mathbb{Z}_q^* \) and keeps it secret. Then CA computes the system public key \( P_{CA} = s_{CA} \cdot B \). Accordingly, the public parameters in the system are \( \langle E,q,G_1,G_2,e,B,P_{CA},H_1,H_2,H_3,H_4 \rangle \), and the private key of CA is \( s_{CA} \).

[Key Generation]

Suppose that a user \( U_i \) wants to generate keys with CA, he/she performs the following steps to register to CA, and obtains the corresponding public key. He/She also computes his/her private key in this phase.

Setp 1. \( U_i \) chooses a random number \( k_i \in \mathbb{Z}_q^* \). Then he/she computes \( K_i = k_i \cdot B \), and transmits his/her own \( K_i \) and identity \( ID_i \in \{0,1\}^n \) to the CA.

Step 2. After receiving \( ID_i \) and \( K_i \), CA calculates \( Q = H_1(ID_i) \in G_1^* \), and randomly chooses an integer \( x_i \in \mathbb{Z}_q^* \) to compute \( X_i = x_i \cdot B \). Then CA generates \( U_i \)'s Public key \( P_i = K_i + X_i \) and the witness of the public key \( W_i = S_{CA}(P_i + X_i) + x_i(P_{CA} + Q_i) \). Finally, CA sends \( \{P_i, W_i\} \) to \( U_i \).

Step 3. Upon receiving \( \{P_i, W_i\} \), \( U_i \) calculates his/her own private key \( S_i = W_i + k_i Q_i \), and he/she can
verify the public key by performing the following formula:
\[ e(S, B) = e(P_i, P_s) e(Q_i, P) \]
If the result is correct, then \( U_i \)'s private key is \( S_i \); otherwise, it means that the public key \( P_i \) is altered in the transmission.

### 2.3 The Fair Self-certified Blind Signature Scheme

In this section, we will present a fair self-certified blind signature scheme with message recovery. Our proposed scheme is constructed based on bilinear pairings instead of modular exponentiation for the consideration of efficiency. We define notations used in the proposed scheme as follows:

#### [Notations]
- \( S_{CA} \): CA's secret key, where \( S_{CA} \in \mathbb{Z}_q^* \).
- \( P_{CA} \): CA's public key, where \( P_{CA} = S_{CA} \cdot B \).
- \( h() \): a one-way hash function that accepts variable-length input and produces a fixed-length output value, and its length is 160 bits.
- \( x(P) \): the x-coordinate value of point \( P \).
- \( M \): message.
- \( || \): a symbol denoting concatenation.
- \( \in_R \): a symbol denoting the uniform random selection.
- \( \oplus \): bitwise exclusive-or operator

#### [Registration]

In this phase, user \( U_i \) registers to derive the revocation keys \( \alpha \) and \( \beta \) from \( CA \).

**Step 1. Requesting for registration:**
- User \( U_i \) computes \( \Lambda = \lambda \cdot B \), where \( \lambda \in \mathbb{Z}_q^* \) is a random number. \( U_i \) submits \( \Lambda \) and his/her identity information \( ID_{U_i} \) to \( CA \) under a secret channel.

**Step 2. Registering:**
- After receiving \( \Lambda \) and \( ID_{U_i} \), \( CA \) generates the revocation keys \( \alpha, \beta \in \mathbb{Z}_q^* \), where \( \alpha \) and \( \beta \) are prime. Then, \( CA \) randomly chooses \( \gamma \in \mathbb{Z}_q^* \) and computes \( F = \gamma \cdot B \). \( CA \) uses a one-way hash function \( h() \) to compute \( g = h(x(\Lambda)\alpha || x(\Lambda)\beta || x(F)) \), generates \( d = S_{CA} \cdot g + \gamma \), and then returns \( (x(\Lambda)\alpha, x(\Lambda)\beta, d, g) \) to \( U_i \).
- Moreover, \( CA \) computes \( H = H_i(g) \) and \( D = \alpha \cdot B \). Finally, \( CA \) saves \( (\alpha, \beta, ID_{U_i}, H, D) \) in \( CA \)'s database.

**Step 3. Verifying registration:**
- After receiving \( (x(\Lambda)\alpha, x(\Lambda)\beta, d, g) \) sent from \( CA \), \( U_i \) computes \( F' = d \cdot B - g \cdot P_{CA} \) and \( g' = h(x(\Lambda)\alpha || x(\Lambda)\beta || x(F')) \), and verifies whether \( g = g' \). If \( g' \) is equal to \( g \), \( U_i \) can confirm that the message \( (x(\Lambda)\alpha, x(\Lambda)\beta, d, g) \) sent from \( CA \) is correct.

#### [Blind Signature Issuing Protocol]

In this phase, user \( U_i \) wants to get a blind signature from the signer (\( sg \)), and verifies the message recovery blind signature.

**Step 1. Initial oblivious transformation:**
- First \( U_i \) computes \( H = H_i(g) \), \( \phi = \alpha \beta \cdot B \) and \( \phi = H - \alpha \beta \cdot B \). Then \( U_i \) submits \( \phi \) and \( \phi' \) to the signer.

**Step 2. Generating fair blind factors:**
- The signer computes \( H = \phi + \phi' \) by using the message \( (\phi, \phi') \) from user, and checks whether the value \( H \) has been stored in \( CA \)'s database. If \( H \) is \( CA \)'s database, the signer obtains the values \( D \) from \( CA \)'s database and verify \( \phi = D \) furthermore. Right after that, the signer randomly chooses \( r \in \mathbb{Z}_q^* \), and computes \( U = r \cdot P_{sg} \) and \( \delta = r \cdot \phi \), where \( P_{sg} \) is the signer's public key. Finally, he/she sends the blind factors \( (U, \delta) \) to \( U_i \).

**Step 3. Blinding the message:**
- After receiving \( (U, \delta) \), \( U_i \) verifies the following formula:
\[ e(\alpha \beta \cdot U, B) = e(P_{sg}, \delta) \]
If it is valid, \( U_i \) computes \( U' = \alpha U + \alpha \beta P_{sg} \) and \( U' = H_i\left(e(U, P_{CA} - Q_{sg})\right) \oplus M \). Then, \( U_i \) generates \( h = \alpha^{-1} H_i(U') + \beta \). Finally, \( U_i \) submits \( h \) to the signer.

**Step 4. Generating a blind signature:**
- The signer sends back \( V \), where \( V = (r + h) S_{sg} \).
- And, \( U_i \) computes \( V' = \alpha V \), and outputs \( (M, U', V') \). Then \( (U', V') \) is the blind signature of message \( M \).

#### [Verifying the Fair Blind Signature with Message recovery]

\( U_i \) accepts the signature when the following equation holds:
\[ M = H_i\left(e(V', B) e(P_{sg}, P_{CA} - Q_{sg})^{-u(V')})\right) \oplus U' \]
If the check is correct, then \( (U', V') \) is the blind signature of message \( M \).
3 Security Proofs

3.1 Blindness Property
To prove the blindness, we show that given a valid signature \( (M, U', V') \) and any view \( (\phi, \delta, h, V') \), there always exists a unique pair of blind factors \( \alpha, \beta \in \mathbb{Z}_q^* \). Since the blind factors \( \alpha, \beta \in \mathbb{Z}_q^* \) are chosen randomly, the blindness of the signature scheme are naturally satisfied.

Given a valid signature \( (M, U', V') \) and any view \( (\phi, \delta, h, V') \), then the following equations must hold for \( \alpha, \beta \in \mathbb{Z}_q^* \):
\[
U' = aU + a\beta P_{eg} \\
U' = H_1(e(U, P_{CA} - Q_{sg})) \oplus M \\
h = a\alpha H_1(U') + \beta \\
V' = \alpha V
\]

(1) (2) (3) (4)

It is obvious that \( \alpha \in \mathbb{Z}_q^* \) exists uniquely from Eq.(4) denoted by \( \log_e V' \). So we can get \( \beta = h - (\log_e V')^{-1} H_1(U') \) from Eq. (3), and it is unique in \( \mathbb{Z}_q^* \). Furthermore, we show that such \( \alpha \) and \( \beta \) satisfy Eq. (1). Apparently, due to the non-degenerate of the bilinear pairing, we have \( U' = aU + a\beta P_{eg} \Leftrightarrow e(U, P_{CA}) = e(aU + a\beta P_{eg}, P_{CA}) \)

Just we need to show that such \( \alpha \) and \( \beta \) satisfy
\[
e(U', P_{CA} - Q_{sg}) = e(aU + a\beta P_{eg}, P_{CA} - Q_{sg}).
\]

(5)

We have
\[
e(aU + a\beta P_{eg}, P_{CA} - Q_{sg}) \\
= e \left( \log_e V', U + \log_e V' \cdot h - (\log_e V')^{-1} H_1(U') \right), P_{Ca} - Q_{sg} \\
= e \left( \log_e V' \cdot (r + h) P_{eg}, P_{CA} - Q_{sg} \right) e \left( V', B \right)^{-1} e \left( U', P_{CA} - Q_{sg} \right) \\
= e \left( \log_e V', V' \cdot B \right)^{-1} e \left( U', P_{CA} - Q_{sg} \right) \\
= e(U', P_{CA} - Q_{sg})
\]

Since \( \alpha \) and \( \beta \) satisfy Eq. (5), we can show that such \( \alpha \) and \( \beta \) also satisfy Eq. (2).Thus there always exist the blind factors to lead to the same relation defined in the blind signature issuing protocol.

3.2 Non-forgeability
Let \( \mathcal{A} \) be the attacker who controls the sender. \( \mathcal{A} \) can forge valid blind signatures once gets the signer’s secret key. We consider four lemmas as follows.

**Lemma 1** The advantage of \( \mathcal{A} \) in revealing the signer’s secret key \( S_{sg} \) from \( e(P_{sg}, P_{CA} - Q_{sg}) = e(S_{sg}, B) \) by interacting the signer’s ID is negligible.

**Proof:**
The proof of this case is by contradiction. We assume that \( \mathcal{A} \) successful produces a valid message-signature pair \( (m, \sigma(m)) \) with a non-negligible probability \( \varepsilon \). Then the attacker \( \mathcal{A} \) constructs a simulator \( \mathcal{S} \) to solve the Computational Diffie-Hellman (CDH) problem which is a version of Diffie-Hellman problems [3]. In other words, \( \mathcal{S} \) successful solve the CDH problem with a non-negligible probability \( \varepsilon \).

Let \( q_H \) be the maximum number of queries asked from \( \mathcal{A} \) to \( \mathcal{S} \), it is limited by a polynomial in \( k \). The attacker \( \mathcal{A} \) gets public parameters \( \text{PARAMS}(G_1, G_2, q, e, B, P_{CA}, Q_{sg}) \) and wants to find \( S_{sg} \in G_1 \) from \( e(P_{sg}, P_{CA} - Q_{sg}) = e(S_{sg}, B) \). We describe the process of simulator \( \mathcal{S} \) as follows:

1. The simulator \( \mathcal{S} \) randomly chooses \( i \in \{1, \ldots, q_H\} \).
2. For \( \mathcal{A} \)'s \( i \)-th query to \( \mathcal{S} \), if \( i = 1 \), the attacker \( \mathcal{A} \) randomly chooses \( k_{sg} \in \mathbb{Z}_q^* \), and sends \( \{k_{sg} = k_{sg} \cdot B, ID_{sg}\} \) to the simulator \( \mathcal{S} \). The simulator \( \mathcal{S} \) outputs \( P_{eg} \).
3. If \( i \neq 1 \), \( \mathcal{A} \) randomly chooses a number \( r \in \mathbb{Z}_q^* \) and outputs \( r \) to the simulator \( \mathcal{S} \). The simulator \( \mathcal{S} \) outputs \( U = r \cdot P_{eg} \).
4. The simulator \( \mathcal{S} \) returns \( \{P_{eg}, U\} \) to \( \mathcal{A} \), then \( \mathcal{A} \) outputs a valid message-signature pair \( (m, \sigma(m)) \). Now \( \mathcal{A} \) wants to use \( P_{eg} \) (from \( \mathcal{S} \)) to get \( S_{sg} \) from
\[
e(P_{sg}, P_{CA} - Q_{sg}) = e(S_{sg}, B).
\]

Let \( Q_{sg} = H_1(ID_{sg}) = s \cdot B \), where \( s \in \mathbb{Z}_q^* \), then
\[
e(P_{sg}, P_{CA} - Q_{sg}) = e(B, B)^{k_{sg} + x u_{CA}}
\]

(6)

Let
\[
\begin{align*}
t &= k_{sg} + x u_{CA} \\
u &= S_{CA} - s
\end{align*}
\]

Therefore
\[
e(B, B)^{k_{sg} + x u_{CA}} = e(B, B)^{\omega} = e(S_{sg}, B).
\]

From Eq. (6) We can know that the advantage of \( \mathcal{A} \) in getting \( S_{sg} \) from \( e(P_{sg}, P_{CA} - Q_{sg}) = e(S_{sg}, B) \) is
\[
\text{Adv}_{\mathcal{A}, \mathcal{S}_{sg}} = \Pr_{\omega \in \mathbb{Z}_q^*} \left[ \frac{1}{\mathcal{A}(B, t, B, uB, tBP)} = 1 \right] = \varepsilon.
\]

By the CDH assumption, for every probabilistic, polynomial-time, \( 0/1 \)-valued algorithm \( \mathcal{A} \), \( \text{Adv}_{\mathcal{A}, \mathcal{S}_{sg}} \) is
negligible. This is a contradiction, because the advantage of \( A \) in solving the CDH problem in \( G_1 \) is negligible. In other words, the success probability of the forgery in this attack is negligible.

**Theorem 1:** An attacker can not reveal the signer's secret key \( S_{sg} \) from \( e(P_{sg}, P_{CA} - Q_{sg}) = e(S_{sg}, B) \) by interacting the signer's ID.

**Proof:**
By Lemma 1, we have completed the proof.

**Lemma 2:** The advantage of \( A \) in revealing the signer's secret key \( S_{sg} \) from \( M = H_1(e(V, B)e(P_{sg}, P_{CA} - Q_{sg})^{-1} \oplus U') \) by interacting the signer's ID is negligible.

**Proof:**
Assuming that \( A \) successful produces a valid message-signature pair \( (m, \sigma (m)) \) with a non-negligible probability \( \varepsilon \). Then the attacker \( A \) constructs a simulator \( S \) to solve the CDH problem. In other words, \( S \) successful solve the CDH problem with a non-negligible probability \( \varepsilon \).

Let \( q_M \) be the maximum number of queries asked from \( A \) to \( S \), it is limited by a polynomial in \( k \). The attacker \( A \) gets public parameters \( PARAMS(G_1, G_2, q, e, B, P_{CA}, Q_{sg}) \) and wants to find \( S_{sg} \in G_1 \) from \( M = H_1(e(V, B)e(P_{sg}, P_{CA} - Q_{sg})^{-1} \oplus U') \). The process of simulator \( S \) is the same with the simulator in Lemma 1. And now, \( A \) wants to use \( P_{sg} \) to get \( S_{sg} \) from \( M = H_1(e(V, B)e(P_{sg}, P_{CA} - Q_{sg})^{-1} \oplus U') \).

Since
\[
\begin{align*}
M &= H_1(e(V, B)e(P_{sg}, P_{CA} - Q_{sg})^{-1} \oplus U') \\
&= H_1(e(U', P_{CA} - Q_{sg}) \oplus M \oplus H_1(e(U', P_{CA} - Q_{sg})),
\end{align*}
\]
\( A \) reveals \( S_{sg} \) from \( e(U', P_{CA} - Q_{sg}) \) and
\[
U' = \alpha U + \alpha \beta P_{sg} = P_{sg} (ar + \alpha \beta),
\]
and we can get
\[
e(U', P_{CA} - Q_{sg}) = e(P_{sg}, P_{CA} - Q_{sg})^{(ar+\alpha \beta)} \tag{7}
\]
According to Lemma 1, the advantage of \( A \) in revealing the signer's secret key \( S_{sg} \) from Eq. (7) by interacting the signer's ID is negligible. In other words, the success probability of the forgery in this attack is negligible.

**Theorem 2:** An attacker can not reveal the signer's secret key \( S_{sg} \) from \( M = H_1(e(V, B)e(P_{sg}, P_{CA} - Q_{sg})^{-\alpha \beta} \oplus U') \) by interacting the signer's ID.

**Proof:**
By Lemma 2, we have completed the proof.

### 4 Performance Evaluation
In this section, we discuss the computational complexity of the proposed fair blind signature scheme with message recovery (FBSMR).

The following notations are used for measuring the performance of the proposed systems.

- \( T_{MM}/T_{EXP}/T_{MC} \): the time for computing a modular multiplication/exponentiation/addition
- \( T_{INV} \): the time for computing modular inversion
- \( T_{EM} \): the time for computing the multiplication of a number and an elliptic curve point
- \( T_{E} \): the time for computing the addition of two points on an elliptic curve
- \( T_{H} \): the time for computing the one-way has function \( h \)

According to the paper proposed by Koblitz et al. [7], the above time complexities have the following relationship:

- \( T_{EM} \approx 29T_{MM} \)
- \( T_{E} \approx 0.12T_{MM} \)
- \( T_{EXP} \approx 240T_{MM} \)
- \( T_{MC} \) and \( T_{H} \) are negligible as compared to the above complexities measures.

In Table 1, we can see that our proposed scheme is more efficient than Lee-Kim's [8] in computational complexity. Although our scheme is one \( T_{EM} \) more than Tsaur-Chou's scheme in the steps of verifying the fair blind signature with message recovery, the computational complexity in the step of generating fair blind factors is half of Tsaur-Chou's [12] scheme.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Registration</td>
<td>962 ( T_{MM} + T_{INV} )</td>
<td>147.12 ( T_{MM} )</td>
<td>145.12 ( T_{MM} )</td>
</tr>
<tr>
<td>Signature</td>
<td>2837 ( T_{MM} + 5T_{INV} )</td>
<td>471.36 ( T_{MM} )</td>
<td>435.72 ( T_{MM} )</td>
</tr>
</tbody>
</table>

### 5 Conclusions
In this paper, we propose a public key cryptosystem by integrating the paring-based cryptosystems with the ID-based self-certified public key cryptosystems,
and further employ the integrated cryptosystems to design a fair blind signature scheme with message recovery. Based on the proposed security proofs and performance evaluation, we affirm that we not only improve the efficiency of Lee and Kim’s scheme, but also achieve the essential properties of blind signature with provable security.

References: