An E-Cash Scheme Based on Proxy Blind Signature from Bilinear Pairings

Zuowen Tan[1,2]
1. Department of Computer Science & Technology, School of Information Technology,
Jiangxi University of Finance and Economics, Nanchang 330032,Jiangxi Province, P.R. China
2. Key Lab of Network Security and Cryptology, School of Mathematics and Computer Science,
Fujian Normal University, Fuzhou 350007,Fujian Province, P.R. China
Email: tanzyw@gmail.com

***Abstract*—In this paper, a proxy blind signature scheme based on bilinear pairing is proposed. The proposed proxy blind signature is existential unforgeable under adaptively chosen warrant attacks and chosen message attacks upon the CDH assumptions and DBDH assumptions in the Random Oracle Model. In order to make all levels of banks issue electronic coin, the proxy blind signature scheme is applied to construct an electronic cash system. The electronic cash system holds unforgeability, unlinkability and efficient traceability. It can provide protection for the honest consumers and the transactions, and prevent the double spending**.

***Index Terms*—Electronic cash; Proxy Blind Signature; Discrete Logarithm; Unlinkability; Random Oracle Model**

I. INTRODUCTION

Since Chaum introduced the concept of electronic cash [1, 2], there has been a lot of e-cash research on it [3,4,5,6,7]. In an electronic cash system, the bank is responsible for issuing electronic cash. When the transaction is completed, electronic cash is deposited to the bank by the merchants. The basic requirement of an e-cash scheme is that withdrawal and payment protocols can not disclose when a specific e-coin (e-cash, hereafter we refer to them as the same meaning) is consumed and who spends the e-coin. During the withdrawal, the private information is kept secret so that the bank can not trace how the money is consumed. However, electronic cash can be easily copied, which will lead to double-spending of an e-coin.

Based on whether the bank is required to be on-line or not during the transaction, the e-cash schemes are classified into on-line e-cash schemes [2, 8, 9] and off-line e-cash schemes [3]. For an on-line e-cash scheme, when the consumers spend e-cash, in order to prevent double-spending the banks are required online. The merchants do not accept an e-coin until the bank verifies its validity. Although this method can provide real-time verification of an e-coin, it is likely to cause the service blockage of the bank servers. In the case of off-line e-cash schemes, when the merchants and consumers conduct transactions, the banks are not required to be online. The merchant first accepts the payment, and then deposits the e-coin to the bank. The latter can effectively avoid the bottleneck problem of the bank server, but it would bring about the double-spending problem.

An electronic cash system should satisfy the following security properties:

-Unforgeability of e-coin: Any probability polynomial time adversary can not forge an e-coin. Only the banks can issue electronic cash.

-Anonymity of an honest consumer: Even if a malicious consumer or the merchant colludes with the bank, they can not obtain the identity information or consumption behavior of honest consumers from e-coins.

-Traceability of the double-spending: If a consumer spends an e-coin twice or more times, then the e-coin will reveal the consumer's identity information.

In some application environments, the privacy protection of the signer is necessary. Group signature can provide anonymity and unlinkability. Therefore, group signature is always applied to build up electronic cash systems [10-13]. However, most of the group signature based electronic cash systems are inefficient. David Chaum introduced the concept of blind signature [1, 2]. A blind signature scheme allows the sender of the message receive its signature, while the signer can not see the message during the signing and afterwards can not link the signature with the message. A secure blind signature scheme requires unforgeability and unlinkability. When the signature requester, such as a customer requires preventing the sender (e.g. the bank) from the linkage of the message with the signature, the blind signature scheme can be used. Blind signatures have already found wide applications in the electronic cash system [1, 2, 4, 6, 7, 14, 15]. The message in the blind signature scheme represents such information as a consumer's identity and e-coin value in the electronic cash system. When the consumer renders the message and its signature issued by the bank, the bank can not determine who the signature is signed to. In fact, the withdrawal participant obtains the blind signatures through the un-blinding operation. The signer (i.e. the bank) can not link the final signature with the blinding message. Therefore, the consumer can withdraw e-coins from the bank and spend anonymously them. In [4], a trusty party is required in order to trace the double spending.

In the real world, each bank system consists of many levels of banks, e.g. the central banks, headquarters,

doi:10.4304/jcp.5.11.1638-1645

branches and sub-branch banks. Most of consumers deal with various local bank departments more but with the central bank. Only the central banks have the authority to issue cash which will bring much inconvenience to the consumers. Consider the ideal situation: there is a great many of banks, monitored by the National Central Bank, where each bank can issue electronic cash. The previous e-cash schemes have not considered the situation. It is necessary for all the levels of banks to issue e-cash. In fact, all the levels of banks can require the authority of the central banks and issue e-cash. Proxy signature can achieve the signature transform function. In proxy signature schemes [16], an original signer delegates a proxy signer to sign message on its behalf. Proxy signature and blind signature can be combined into proxy blind signature [17,18]. Proxy blind signature involves three participants: the original signer, the proxy signer and the signature receiver (signature requesting party). Proxy blind signature has a broad application prospects for the participants to protect their privacy and anonymity. In order to ensure the consumer's privacy in the electronic cash system, a consumer often does not allow the banks link a specific e-coin issued by the bank with the payment behavior of the consumer.

Since proxy blind signature was introduced, people have undertaken extensive research on it [19-21]. There have been a number of proxy blind signature schemes, such as Schnor signature based proxy blind signature [18], braid group based proxy blind signature [22] and identity based proxy blind signature [23]. It is desirable for proxy blind signature schemes to possess the security attributes as claimed [18]. Here, we highlight these security attributes.

(1) *Distinguishability*: Proxy blind signatures are easily distinguishable from ordinary signatures.

(2) *Verifiability*: A proxy blind signature receiver can verify the proxy blind signature in the same manner as verifying an ordinary signature of the original signer.

(3) *Identifiability*: A proxy blind signature includes the original signer's authorization certificate. So the signature receiver can determine the original signer's identity and the corresponding proxy signer's identity.

(4) *Unforgeability*: Only the designated proxy signer is able to generate a proxy blind signature, while any other participants (even the original signer) can not produce a valid proxy blind signature.

(5) *Unlinkability*: For a proxy blind signature, the signer can not link it with the message or the intermediate signature (i.e. the blinding signature).

(6) *Non-repudiation*: Once a proxy blind signature is produced, the proxy signer can not deny the signature. Moreover, the original signer can not deny that the proxy blind signature has been authorized by itself.

Most of the proxy blind signature schemes are not equipped with provable security proofs. In this paper, we first designed a proxy blind signature scheme based on the bilinear pairing. Upon the CDH assumptions and DBDH assumptions, the proposed proxy blind signature scheme is provable secure in the Random Oracle model. Then we apply the secure proxy blind signature scheme to construct an off-line electronic cash system. The new e-cash scheme is unforgeable and has unlinkability in the Random Oracle model. In addition, our e-cash scheme can trace efficiently the double-spending.

The rest of this paper is organized as follows. Some preliminary works are given in Section 2. Section 3 presents a proxy blind signature scheme based on the bilinear pairing. Its security proof will be given in 4. In section 5, we will describe a new e-cash scheme and prove the security of the proposed scheme. Section 6 concludes.

II. PRELIMINARIES

A. Bilinear pairings

Let G_1 and G_2 be two cyclic groups of prime order q, Q be a generator of G_1. Let $\hat{e}$ be an admissible map from $G_1 \times G_1$ to G_2, which satisfies the properties:

• Bilinearity: For any $u, v \in G_1$ and $a, b \in Z_q^*$, $\hat{e}(u^a, v^b) = \hat{e}(u, v)^{ab}$.

• Non-degenerate: $\hat{e}(Q, Q) \neq 1$.

•Computability: There is an efficient algorithm to compute $\hat{e}(u, v)$ for $u, v \in_R G_1$.

B. Cryptographic assumptions

We review some cryptographic assumptions.

Definition 1. (*Discrete logarithm Problem* (*DLP*)) Given the elements (Q , aQ) in a multiplicative cyclic group G_1, solve the exponent a.

The advantage of an algorithm A against DLP is defined as

$$\mathrm{Succ}_{A,G_1}^{DLP} = \Pr[A(Q, aQ) = a].$$

Definition 2. (*Discrete Logarithm (DL) Assumption*) Given (Q , aQ) in G_1, $\mathrm{Succ}_{A,G_1}^{DLP}$ of an algorithm A which solves *DLP* is negligible.

Definition 3. (*Decisional Bilinear Diffie–Hellman (DBDH) Problem*) Given the elements (Q , aQ , bQ, cQ) in an additive cyclic group G_1 for some unknown $a, b, c \in Z_p^*$, and an element $Z \in G_2$, decide whether $Z = \hat{e}(Q, Q)^{abc}$ or not.

The advantage of a distinguisher A against the DBDH problem is *defined as*

$$\mathrm{Succ}_{A,G_1}^{DBDH} = |\Pr[A(Q, aQ, bQ, cQ, \hat{e}(Q, Q)^{abc}) = 1] - \Pr[A(Q, aQ, bQ, cQ, \hat{e}(Q, Q)^{z}) = 1]|.$$

Definition 4. (*Decisional Bilinear Diffie–Hellman (DBDH) Assumption*) Given (Q , aQ , bQ , cQ) for some unknown $a, b, c \in Z_p^*$, and an element $Z = \hat{e}(Q, Q)^z$, $\mathrm{Succ}_{A,G_1}^{DBDH}$ of a distinguisher A which solves the DBDH problem is negligible.

Definition 5. (*Computational Diffie–Hellman (CDH) Problem*) Given (Q , aQ , bQ) in G_1 for some unknown $a,b \in Z_p^*$, compute abQ.

The advantage of a polynomial algorithm A in solving CDH problem is defined as

$$\mathrm{Succ}_{A,G_1}^{CDH} = \Pr[A(aQ,bQ) = abQ, a,b \in Z_q^*].$$

Definition 6. (*Computational Diffie–Hellman (CDH) Assumption*) Given (Q , aQ , bQ) in G_1 for some unknown $a,b \in Z_p^*$, the advantage $\mathrm{Succ}_{A,G_1}^{CDH}$ of a polynomial algorithm *A* in solving CDH problem is negligible.

Definition 7. (*Decisional Diffie–Hellman (DDH) Problem*) Given (Q , aQ , bQ , cQ) in G_1 for some unknown $a,b,c \in Z_p^*$, decide whether $abQ = cQ$.

If there is an admissible map $\hat{e}$: $G_1 \times G_1 \to G_2$, *DDH Problems in* G_1 *can easily be solved by* checking $\hat{e}(aQ,bQ) = \hat{e}(Q,cQ)$.

Definition 8. (*Gap Diffie–Hellman (GDH) Group*) If CDH problem in G_1 is hard, but DDH problem is easy, G_1 is called a Gap Diffie–Hellman (GDH) Group.

III PROXY BLIND SIGNATURE SCHEME BASED ON BILINEAR PAIRING

Based on the short signature scheme [24] (hereinafter referred to as BLS signatures) and Schnor signature scheme [25], we construct a new proxy blind signature scheme. The signature scheme is divided into the following phases.

A. Parameter generation

Let G_1 and G_2 be two groups of prime *q* order. Let Q be a generator of GDH group G_1. $\hat{e}$ is a bilinear pairing: $G_1 \times G_1 \to G_2$. H_1:{0,1}*$\to G_1$ and H_2:{0,1}*$\to Z_q^*$ are two cryptographic secure hash functions. The original signer O and the proxy signer B hold their private/public key pair (x_o, Y_o) and (x_B, Y_B), respectively, where $Y_o = x_o Q$, $Y_B = x_B Q$.

B. Delegation phase

The original signer O generates a warrant m_ω and computes $\overline{X} = x_o H_1(m_\omega)$. Next, O sends $\overline{X}$ and warrant m_ω to the signer B via a public channel.

C. Proxy signature key generation phase

The proxy signer B determines the validity of the warrant and delegation by checking whether the following equality holds:

$$\hat{e}(\overline{X}, Q) = \hat{e}(H_1(m_\omega), Y_o). \quad (1)$$

If the above equality does not hold, B refuses the delegation. Otherwise, B accepts it and computes the proxy signature key

$$X = \overline{X} + x_B H_1(m_\omega). \quad (2)$$

D. Proxy signature generation phase

This phase can be further divided into three sub-phases.

- **Blinding**

B chooses a random integer $k \in_R Z_q^*$, and computes $R = kQ$.

B sends (m_ω ,*R*) to the signature requester U.

U chooses two integers $a \in_R Z_q^*$, $b \in_R Z_q^*$, computes and delivers c^* to B:

$$t = \hat{e}(R + aQ, Y_B) \cdot \hat{e}(bH_1(m_\omega), Y_o + Y_B), \quad (3)$$

$$c^* = H_2(m \| m_\omega \| t) + b. \quad (4)$$

- **Blind Signing**

B computes and sends S^* to U.

$$S^* = c^* X + kY_B. \quad (5)$$

- **Unblinding**

U computes

$$S = S^* - aY_B, \quad c = c^* - b. \quad (6)$$

Then, (m, m_ω, S, c) is a proxy blind signature on message *m*.

E. Signature verification phase

After a verifier receives the proxy blind signature (m, m_ω, S, c), she computes

$$t' = \hat{e}(S,Q) \cdot \hat{e}(-cH_1(m_\omega), Y_o + Y_B). \quad (7)$$

Next, she checks if the equality holds:

$$c = H_2(m \| m_\omega \| t'). \quad (8)$$

If the equality (8) holds, (m, m_ω, S, c) is valid. Otherwise, (m, m_ω, S, c) is invalid.

IV. SECURITY ANALYSIS ON THE PROPOSED PROXY BLIND SIGNATURE SCHEME

Since the proxy blind signature (m, m_ω, S, c) contains the warrant m_ω, any verifier can tell a proxy blind signature from an ordinary signature. Our proxy blind signature is distinguishable. Moreover, the verifier can determine the identity of the original signer and the identity of the proxy signer through the warrant m_ω. So our proxy blind signature scheme has *identifiability*. The validity of a proxy blind signature can be verified by the equation (8). Therefore, our proxy blind signature scheme has *verifiability*.

Next, we show its correctness, unforgeability, unlinkability and non-repudiation.

Theorem 1 *The proposed proxy blind signature scheme is correct.*

Proof: The validity of the proxy blind signature is

verified through the verification equation (8). From the equation (4), we have $c^* = H_2(m \| m_\omega \| t) + b$. From the equation (7), we have $c = c^* - b$. If $t' = t$, the verification equation (8) will hold.

In essence, we have

$$t' = \hat{e}(S,Q) \cdot \hat{e}(-cH_1(m_\omega), Y_o + Y_B), \quad (9)$$

$$\hat{e}(S,Q) = \hat{e}(S^* + aY_B, Q)$$
$$= \hat{e}(c^* X + kY_B + aY_B, Q)$$
$$= \hat{e}((H_2(m \| m_\omega \| t) + b)X + kY_B + aY_B, Q)$$
$$= \hat{e}(H_2(m \| m_\omega \| t)(x_o + x_B)H_1(m_\omega), Q)$$
$$\cdot \hat{e}(b(x_o + x_B)H_1(m_\omega), Q) \cdot \hat{e}(kY_B + aY_B, Q)$$
$$= \hat{e}(cH_1(m_\omega), Y_o + Y_B)$$
$$\cdot \hat{e}(b(x_o + x_B)H_1(m_\omega), Q) \cdot \hat{e}(kY_B + aY_B, Q)$$
$$= \hat{e}(cH_1(m_\omega), Y_o + Y_B)$$
$$\cdot \hat{e}(bH_1(m_\omega), Y_o + Y_B) \cdot \hat{e}(Y_B, R + aQ).$$

Therefore,

$$t' = \hat{e}(S,Q) \cdot \hat{e}(-cH_1(m_\omega), Y_o + Y_B)$$
$$= \hat{e}(bH_1(m_\omega), Y_o + Y_B) \cdot \hat{e}(Y_B, R + aQ) = t. \square$$

Theorem 2 *The proposed proxy blind signature is existential unforgeable under the adaptively chosen warrant attacks and chosen message attacks upon the CDH assumptions and DBDH assumptions in the Random Oracle Model.*

Proof: Motivated by the techniques in [26] and [27], we give the proof. Suppose that an adversary succeeds in attacking the proxy blind signature when the adversary reaches one or both of the following goals: ①forgery of a delegation, ②forgery of a proxy blind signature.

Assume that an adversary generates a valid delegation (m_ω, σ) or a valid proxy signature key/warrant pair (X, m_ω), in other words, the adversary obtains the first goal. Since the pair (m_ω, σ) is valid, it must satisfy $\hat{e}(\sigma, Q) = \hat{e}(H_1(m_\omega), Y_o)$. Then the adversary succeeds in forging a BLS signature (m_ω, σ) on m_ω with the secret key x_o. While the valid proxy signature key/warrant pair (X, m_ω) is valid, it must satisfy the following: $\hat{e}(X,Q) = \hat{e}(H_1(m_\omega), Y_o + Y_B)$.It means that the adversary succeeds in forging a BLS signature (X, m_ω) on m_ω with the secret key $(x_o + x_B)$.

This is in contradiction with the unforgeability of BLS signature [24] upon DBDH assumptions in the Random Oracle Model.

Assume that the adversary achieves the second objective, that is, the adversary succeeds in forging a valid proxy blind signature. We apply the adversary to construct an algorithm ***A*** to solve CDH problems in GDH Group G_1.

A is given an instance of CDH problems in G_1: Given $Y_1 = y_1 Q$, $Y_2 = y_2 Q$, where $y_1 \in Z_q^*$, $y_2 \in Z_q^*$ are unknown, solve $y_1 y_2 Q$. ***A*** works as follows:

By applying the techniques in [26] and [27] (with minor modification), it is easy to construct a simulator ***I*** to simulate the adversary's view during the protocol execution. These views include the outputs of parameter generation, delegation generation, blinding and unblinding, and the hash value $H_1(\cdot)$ of the warrant m_ω. Suppose that the random oracle $H_2(\cdot)$ is queried during the simulation. Otherwise, ***A*** would fail. ***A*** maintains a random table to store the response value of $H_2(\cdot)$-query. During the parameter generation, let $Y_1 = Y_o + Y_B$. ***A*** chooses randomly z in Z_q^* and sets $H_1(m_\omega) = zY_2$. According to fork lemma [28], we can obtain two corresponding tuples (m, m_ω, S_1, c_1) and (m, m_ω, S_2, c_2) on the pair (m, m_ω).

$$\hat{e}(S_1,Q) \cdot \hat{e}(-c_1 H_1(m_\omega), Y_o + Y_B)$$
$$= \hat{e}(S_2,Q) \cdot \hat{e}(-c_2 H_1(m_\omega), Y_o + Y_B).$$
$$\hat{e}(S_1 - S_2, Q) \cdot \hat{e}((c_2 - c_1)H_1(m_\omega), Y_o + Y_B)$$
$$= \hat{e}((S_1 - S_2) + (c_2 - c_1)(x_o + x_B)H_1(m_\omega), Q)$$
$$= \hat{e}(\theta, Q),$$

where θ is the zero element of the GDH Group G_1.

Therefore, we have

$$(x_o + x_B)H_1(m_\omega) = (c_2 - c_1)^{-1}(S_2 - S_1).$$

A can compute

$$y_1 y_2 Q = z^{-1}(c_2 - c_1)^{-1}(S_2 - S_1).$$

Thus, ***A*** solves the CDH problems in GDH Group G_1.□

Theorem 3 *The proposed proxy blind signature scheme satisfies the non-repudiation property.*

Proof: A proxy blind signature contains the warrant m_ω. Moreover, during the signature verification, the public key of the original signer and the public key of the proxy signer must be used. From Theorem 2, the proxy blind signature scheme is not forgeable. Thus, for a valid proxy blind signature, the original signer could not deny its delegation to B and the proxy signer B could not deny its signature. □

Theorem 4 *The proposed proxy blind signature scheme satisfies the unlinkablility property. That is, when the proxy signer receives the proxy blind signature, B could not link it with the view during the actual protocol execution.*

Proof: Without loss of generality, assume that B keeps all the views during the actual protocol execution. For any view $(R_i, m_\omega, c_i^*, S_i^*)$ and any proxy blind signature $(m_j, m_\omega, S_j, c_j)$, if there exists only one blinding factor such that the view $(R_i, m_\omega, c_i^*, S_i^*)$ and the proxy blind signature $(m_j, m_\omega, S_j, c_j)$ satisfies all

the equations in the scheme, then we will complete the proof of the theorem.

Now we look for such a blinding factor. From the equalities (5) and (6), set

$$b_{ij}= c_i^* - c_j,\ V_{ij} = S_j - S_i^* . \quad (10)$$

Thus, we define an integer a_{ij} such that $V_{ij} = a_{ij} Y_B$. Let

$$t_{ij} = \hat{e}(R_i + a_{ij}Q, Y_B)\cdot \hat{e}(b_{ij}H_1(m_\omega), Y_o + Y_B),$$
$$t' = \hat{e}(S_j, Q)\cdot \hat{e}(-c_j H_1(m_\omega), Y_o + Y_B).$$

From (7) and (8), it is enough to verify the equality $t_{ij} = t'$. In fact, we have

$$\begin{aligned} t_{ij} &= \hat{e}(R_i + a_{ij}Q, Y_B)\cdot \hat{e}(b_{ij}H_1(m_\omega), Y_o + Y_B) \\ &= \hat{e}(x_B R_i + a_{ij}Y_B, Q)\cdot \hat{e}(b_{ij}H_1(m_\omega), Y_o + Y_B) \\ &= \hat{e}(r_i Y_B + S_j - S_i^*, Q)\cdot \hat{e}(b_{ij}H_1(m_\omega), Y_o + Y_B) \\ &= \hat{e}(S_j - c_i^* X, Q)\cdot \hat{e}(b_{ij}H_1(m_\omega), Y_o + Y_B) \\ &= \hat{e}(S_j, Q)\cdot \hat{e}((b_{ij} - c_i^*)H_1(m_\omega), Y_o + Y_B) \\ &= \hat{e}(S_j, Q)\cdot \hat{e}(-c_j H_1(m_\omega), Y_o + Y_B) = t'. \end{aligned}$$

It demonstrates that the proxy signer can not link the proxy blind signature with the views. This is because the blinding factor is randomly chosen. Therefore, the proposed proxy blind signature scheme satisfies the unlinkablility property. □

V. E-CASH SCHEME

A. E-cash scheme based on bilinear mapping

Now, we construct an e-cash scheme based on the proposed proxy blind signature scheme. If a dishonest consumer has done double-spending, the trusty third party is not required in the new e-cash scheme. Any participant in the e-cash scheme can trace the identity of the dishonest consumer.

The e-cash scheme involves with four parties: a consumer U, a branch bank B, B's upper bank O and a merchant M. Here, after B is authorized by the upper bank O, B can issue e-cash to the consumer. In the real world, cash is issued only by the central bank. In our electronic cash system, any one branch, even a small bank department, can issue e-cash under the authority of their upper banks. Thus, the electronic cash system can provide more flexibility than in the real world.

The parameters of the e-cash scheme include two group G_1, G_2 of prime order *q*, a generator *Q* of the GDH group G_1, a bilinear pairing $\hat{e}$ from $G_1 \times G_1$ to G_2 and two strong resistant-collision hash functions H_1:{0,1}*→ G_1, H_2: {0,1}*→Z_q^*. The private /public key pair of the central bank O and the bank B are (x_o,Y_o) and (x_B,Y_B), respectively, where Y_o=$x_o Q$, Y_B= $x_B Q$. The consumer U has its identity ID. Let ID $\in G_1$.

The new e-cash scheme consists of the delegation protocol, opening-account protocol, a withdrawal protocol, payment protocol, deposit protocol and tracing protocol.

Delegation protocol:

The protocol is executed between the upper bank O and the bank B. The e-cash issuer B will obtain the authorization of the upper bank O by executing the protocol.

First, O produces a warrant m_ω which includes the identities of the bank B and its upper bank O ,the period of validity, etc. Then O computes $\sigma = x_o H_1(m_\omega)$ and sends (σ, m_ω) to the e-cash issuer B. After B receives the delegation message, B checks its validity through the equality:

$$\hat{e}(\sigma, Q) = \hat{e}(H_1(m_\omega), Y_o). \quad (11)$$

If the above equality does not hold, B refuses the delegation. Otherwise, B computes the signature key $X = \sigma + x_B H_1(m_\omega)$.

Opening-account protocol

The protocol is executed between the bank B and the consumer U.

The consumer U sends the application for opening an account to the bank B. The bank B identifies the consumer U and then opens an account $account_U$ to the consumer U.

Withdrawal protocol:

First, U sends its identity ID and the account information $account_U$ to B. B checks their validity. Next, B and U cooperatively execute the following withdrawal protocol. The withdrawal protocol can be divided into four sub-phases.

- **Blinding:**

B randomly chooses an integer $k \in_R Z_q^*$, and computes $R = kQ$.

Then the bank B sends (m_ω, *R*) to U. The consumer U chooses randomly four integers *a*, *b*, α, β in Z_q^*, determines the withdrawal cash value *value* and computes

$$d_1 = \alpha Y_B, d_2 = \beta Y_B, d = \beta ID, \quad (12)$$
$$t = \hat{e}(R + aQ, Y_B)\cdot \hat{e}(bH_1(m_\omega), Y_o + Y_B), \quad (13)$$
$$c^* = H_2(d \| d_1 \| d_2 \| value \| m_\omega \| t) + b. \quad (14)$$

Finally, U sends c^* to B.

- **Blind signing**: The bank B computes S^* and transmits S^* to U.

$$S^* = c^* X + kY_B. \quad (15)$$

- **Unblinding**: U calculates

$$S = S^* - aY_B,\ c = c^* - b. \quad (16)$$

- **Cash verifying**: U computes:

$$t' = \hat{e}(S, Q)\cdot \hat{e}(-cH_1(m_\omega), Y_o + Y_B). \quad (17)$$

Then, U checks whether the following equality holds:

$$c = H_2(d \| d_1 \| d_2 \| value \| m_\omega \| t'). \quad (18)$$

If the above equality does not hold, (*S*, *c*) is not a valid cash. Otherwise, U has withdrawn a *cash* (*d*, d_1, d_2, m_ω ,*value*, *S*, *c*) issued by the bank B. U stores (*cash*, α , β).

Payment protocol:

U executes the payment protocol with the merchant M like this.

Step 1. U sends *cash* to the merchant M.

Step 2. M checks the validity of the coin. If the coin is valid, M continues the nest step. Otherwise, M refuses the coin.

Step 3. M sends a challenge $cha \in \{0,1\}^*$ to the consumer U.

Step 4. U computes and sends (e, μ) to M.

$$e = H_2(cash \| cha \| d_2), \mu = \beta + \alpha e . \quad (19)$$

Step 5. M verifies the spending record (e, μ) by checking if the following holds:

$$e = H_2(cash \| cha \| \mu Y_B - ed_1), \quad (20)$$

$$c = H_2(d \| d_1 \| \mu Y_B - ed_1 \| value \| m_\omega \| t'). \quad (21)$$

If the equations hold, then M agrees to transact with U and stores the transaction records in a database. Otherwise, M refuses the transaction with U.

Deposit protocol:

The protocol is executed between the bank B and the merchant M. First, M sends the transaction record (*cash*, e, μ) to the bank B. B searches the database to check whether *cash* has existed. If cash is new, B deposits *value* to M's account and stores (*cash*, e, μ) in its database. Otherwise, B traces the double-spending consumer.

B. Security analysis of e-cash scheme

In the following, we will prove that the new e-cash scheme satisfies the security properties: unforgeability of coin, anonymity for honest consumers and traceability against dishonest consumers.

Theorem 5 (*Completeness*) *If the banks O and B, the consumer U and the merchant M follow the protocols, then the consumer* U *will withdraw a valid coin by executing the withdrawal protocol and the merchant M will obtain a valid coin by performing the payment protocol.*

Proof: First, we prove that the *cash* (*d*, d_1, d_2, m_ω ,*value*, *S*, *c*) is valid.

From the equality (16), if t' in the equality (17) equals to *t* in (13), the cash verification equation (18) will hold. Since the details of the proof are the same as the proof of Theorem 1, here we omit it.

Next, we need to prove that if both U and M follow the payment protocol, the equalities (19) and (20) will hold.

It is easily known from the equalities (14) and (16):

$$c = H_2(d \| d_1 \| d_2 \| value \| m_\omega \| t') .$$

Since $d_2 = \beta Y_B$ and $\mu = \beta + \alpha e$, we have

$$d_2 = \mu Y_B - ed_1 .$$

Thus, the equalities (19) and (20) hold. □

Theorem 6 (*Unforgeability*) *The e-coin in our e-cash scheme is existential unforgeable upon the CDH assumptions and DBDH assumptions in the Random Oracle model.*

Proof: In our e-cash scheme, the consumer U obtains an e-coin during the withdrawal. The bank B gives U a proxy blind signature (*S*, *c*) as the e-coin. According to Theorem 3, e-coin is existential unforgeable upon the CDH assumptions and DBDH assumptions in the Random Oracle Model.

Theorem 7 (*Anonymity*) *In our e-cash scheme, any party including the bank B can not trace an honest consumer. In other word, an honest consumer can spend e-coin anonymously.*

Proof: In our e-cash scheme, e-coin is in essence a proxy blind signature (*S*, *c*) on (m_ω , *value*). According to Theorem 2, the proposed proxy blind signature scheme satisfies the unlinkablility property. Thus, when the bank receives the e-coin, B could not link it with the identity of an honest consumer. In addition, the e-cash paid by the consumer U does not contains U's identity in the payment protocol. Therefore, the consumption records (e, μ) also will not disclose the identity of the consumer. □

Theorem 8 (*Traceability*) *In the proposed e-cash scheme, if a dishonest consumer spends the same coin twice, the identity of the dishonest consumer can be traced.*

Proof: Suppose the consumer uses a certain coin *cash* twice. Then there exist the two transaction records $(e', \mu', cash)$ and $(e, \mu, cash)$ about the coin *cash*. From the equality (19), we have

$$e = H_3(cash \| cha \| d_2),$$

$$e' = H_3(cash \| cha' \| d_2) .$$

Notice that $e \neq e'$. From the equality (12) and (19), we have

$$d_2 = uY_B - ed_1, d_2 = u'Y_B - e'd_1 .$$

Thus, any participant who has the transaction records can compute the identity of the honest consumer: $\mathrm{ID} = d \cdot \beta^{-1}$, where $\beta = (u'e - ue')/(e - e')$. □

The electronic cash based on blind multisignature scheme in [4] is more efficient of all the electronic cash schemes in the literature. Assuming the size of the point and the size *q* and *n* are 160 bit in [4] and our proposed electronic cash scheme. The messages of payment phase are 1464 bit in [4] and 1120 bit in our electronic cash system. Furthermore, a trusty party is required when the anonymity of users need being removed in [4]. Thirdly, our electronic cash system is suitable for the requirement all levels of banks can issue e-cash.

VI. CONCLUSION

In this paper, we have designed a proxy blind signature based on bilinear mappings. The proposed proxy blind signature is existential unforgeable under the adaptively chosen warrant attacks and chosen message attacks upon CDH assumptions and DBDH assumptions in the Random Oracle Model. Based on the new proxy blind signature scheme, we construct an off-line fair e-cash scheme. The electronic cash system is secure against existential forgery. The scheme can protect the anonymity of honest consumers and can also provide an efficient traceability function to double-spending. The new electronic cash system is suitable for the requirement all levels of banks can issue e-cash. Compared with electronic cash systems in the literature, our electronic cash system has less communication cost and lower complexity.

ACKNOWLEDGMENT

The author would like to thank the reviewers for their useful suggestions. This work was supported in part by a grant from the National Natural Science Foundation of China (10961013) and The Opening Fund (09A003) of Key Lab of Network Security and Cryptology, School of Mathematics and Computer Science, Fujian Normal University.

REFERENCES

[1] David Chaum. “Blind signatures for untraceable payments,” *Advances in Cryptology-CRYPTO ’82*, New York: Plemum Press, pp. 199-203.

[2] David Chaum. “Blind signature systems,” *Advances in Cryptology-CRYPTO ’83*, New York: Plemum Press, pp.153-166.

[3] David Chaum, Amos Fiat, and Moni Naor. “Untraceable electronic cash,” *Advances in Cryptology-CRYPTO ’88*, Lecture Notes in Computer Science, 403, Springer Verlag, 1988, pp.319–327.

[4] C. Popescu, “A Fair Off-line Electronic Cash System Based on Elliptic Curve Discrete Logarithm Problem,” Studies in Informatics and Control, Volume 14, No. 4, 2005, pp.291-298.

[5] Mihir Bellare and Adriana Palacio. “GQ and Schnorr Identification Schemes: Proofs of Security against Impersonation under Active and Concurrent Attacks,” *Advances in Cryptology-CRYPTO ’02*, Lecture Notes in Computer Science, 2442, Springer Verlag, 2002, pp.162–177.

[6] Jan L. Camenisch, Jean-Marc Piveteau, and Markus A. Stadler. “Blind signatures based on the discrete logaritm problem,” *Advances in Cryptology -EUROCRYPT’94*, Lecture Notes in Computer Science, 950, Springer Verlag Berlin, 1994, pp.428–432.

[7] Markus Stadler, J.-M. Piveteau, and J. Camenisch. “Fair blind signatures,” *Advances in Cryptology-EUROCRYPT’95*, Lecture Notes in Computer Science, 921, Springer Verlag, 1995, pp.209–219.

[8] David Chaum. “Security without identification: Transaction systems to make big brother obsolete,” *Communications of the ACM*, Volume 28, Issue 10, October 1985, pp.1030–1044.

[9] David Chaum. “Online cash checks,” *Advances in Cryptology-EUROCRYPT ’89*, Lecture Notes in Computer Science, 434, Springer Verlag, 1989, pp. 289–3293.

[10] Aggelos Kiayias, Yiannis Tsiounis, and Moti Yung. “Traceable signatures,” *Advances in Cryptology-EUROCRYPT ’04*, Lecture Notes in Computer Science, 3027, Springer, 2004, pp. 571–589.

[11]. A. Lysyanskaya and Z. Ramzan. “Group blind digital signatures: A scalable solution to electronic cash,” *Financial Cryptography: Second International Conference, FC’98*, Lecture Notes in Computer Science, 1465, Springer-Verlag, 1998, pp. 184–197.

[12] Toru Nakanishi, Nobuaki Haruna, and Yuji Sugiyama. “Unlinkable electronic coupon protocol with anonymity control,” *International Workshop on Information Security(ISW’99)*, Lecture Notes in Computer Science, 1729, 1999, pp. 37–46,.

[13] Jacques Traor´e. “Group signatures and their relevance to privacy-protecting off-line electronic cash systems,” *Australasian Conference on Information Security and Privacy (ACISP’99)*, Lecture Notes in Computer Science, 1587, Springer-Verlag, 1999, pp.228–243..

[14] M. Bellare. “Practice-oriented provable-security,” *Proceedings of First International Workshop on Information Security (ISW97)*, Lecture Notes in Computer Science, 1396, Springer-Verlag, 1998, pp.221-231.

[15] Yoshikazu Hanatani1, Yuichi Komano, Kazuo Ohta, and Noboru Kunihiro, “Provably Secure Electronic Cash Based on Blind Multisignature Schemes,” *FC 2006*, Lecture Notes in Computer Science, 4107, Springer-Verlag Berlin Heidelberg, 2006, pp.236–250.

[16]Mambo M, Usuda K,Okamoto E. “Proxy signature:delegation of the power to sign messages,” *IEICE Transactions on Fundamentals*, 1996,E79-A(9) , pp.1338-1353.

[17] Lin W.D., Jan J.K. “Security personal learning tools using a proxy blind signature scheme,” *Proccedings of International Conference on Chinese Language Computing*. Illinois, [S.l.]: KSI, 2000, pp. 273-277.

[18] Tan Z.-W., Liu Z.-J., Tang C.-M. “A proxy blind signature scheme based on DLP,” *Journal of Software*, 2003,Volume 14, Issue 11, pp.1931-1935.

[19] Amit K. Awasthi, Sunder Lal. “Proxy blind signature scheme,” T*ransaction on cryptology*. Volume 2, Issue 1, 2005, pp. 1-4.

[20] H.-M Sun and B.-T Hsieh. “On the security of some proxy blind signature schemes,” *Journal of Systems and Software*, Volume 74, Issue 3,2005, pp.297-302.

[21] Wang Shao-bin, Hong Fan, Cui Guo-hua. “Secure efficient proxy blind signature schemes based DLP,” *Proceedings of the Seventh IEEE International Conference on E-Commerce Technology* (CEC’05), IEEE, 2005, pp.452-455.

[22] Girraj Kumar Verma，“A Proxy Blind Signature Scheme over Braid Groups,” *International Journal of Network Security*, Volume 9, Issue 3, Nov. 2009, pp.214–217.

[23] Banshidhar Majhi, Deepak Kumar Sahu, RamNarayan Subudhi. “An Efficient ID based Proxy Signature, Proxy Blind Signature and Proxy Partial Blind Signature,” *International Conference on Information Technology* 2008, IEEE, pp.19-23.

[24] D.Boneh, B. Lynn and H. Shacham. “Short Signatures from the Weil Pairing,” *Proceedings of Asiacrypt* 2001, Lecture Notes in Computer Science, 2248, Springer-Verlag, 2001, pp. 514-532.

[25] Schnorr, C.P.. “Efficient signature generation by smart cards,” *Journal of Cryptology* 4(3), 1991, pp.161-174.

[26] A.Boldyreva, A. Palacio, B. Warinschi. “Secure Proxy

Signature Schemes for Delegation of Signing Rights," At:http://eprint.iacr.org/2003/096.
[27] Zuowen Tan, Zhuojun Liu, "Provably secure delegation by certification proxy signature schemes," *ACM International Conference Proceeding Series, Proceedings of the 3rd international conference on Information security*, Shanghai, China 2004, pp.38-43.
[28] Pointecheval, D., Stern, J.. "Security of Proofs for Signatures," *Advances in Cryptology -EUROCRYPT'96*, Lecture Notes in Computer Science, 1070, Springer-Verlag, 1996, pp.387-398.

Zuowen Tan, born in Yiyang, Hunan, China, 1967. He received his M.S. degree in Fundamental Mathematics from Xiangtan University in 2002, and his Ph.D. degree in Applied Mathematics from Institute of Systems Science, Academy of Mathematics and System Science, CAS in 2005.

He is currently an associate professor at Department of Computer Science & Technology, School of Information Technology, Jiangxi University of Finance & Economics. He has published over 40 papers on information security in international conferences and journals. His current research interests include e-commerce security, information security and cryptography.

Dr. Tan was committee members of some international conferences and reviewers on international Journals.