International Journal of Security and Its Applications
Vol. 10, No. 2 (2016), pp.237-300
http://dx.doi.org/10.14257/ijsia.2016.10.2.26

# An Electronic Cash System Based on Certificateless Group Signature

Liang Yan, Zhang Xiao* and Zheng Zhi-ming

*Key Laboratory of Mathematics, Informatics and Behavioral Semantics, Ministry of Education, and School of Mathematics and Systems Science, Beihang University, Beijing 100191, China*
*Zhang Xiao 09621@buaa.edu.cn*

***Abstract***

*Currently, electronic cash system is one of the most important ways of electronic payment businesses, which requires quick payment and countermeasures against cyber-crime such as falsification of information, double-spending and money-tampering. In this paper, we proposed an efficient elliptic certificateless group signature with forward security and member revocation based on ACJT group signature. Compared to ACJT, our scheme has a shorter length of signature with less parameters and only involves a smaller amount of computation, leading to a higher efficiency. We also proposed a fair off-line multi-bank e-cash system based on the newly-proposed group signature scheme. The new e-cash system is considered to have high security and efficiency. And we proved its unforgeability, and prevention against double-spending and money-tampering actions. Our scheme has a significant advantage over the other schemes.*

***Keywords****: elliptic curve, certificateless signature, forward security, member revocation, group signature, electronic cash.*

## 1. Introduction

With the development of internet technology and e-commerce, electronic cash system is becoming an increasingly important payment method in modern life. The main objective of e-cash is to construct an off-line fair multi-bank system. Besides, how to operate the system efficiently for quick payment and prevent the system vulnerabilities like falsification of information, double-spending and money-tampering are the development bottleneck and hot research area of the e-cash system. E-cash system is mainly founded by cryptographic protocols. Digital signature provides technical support and guarantee for the system. The key to construct efficient and secure e-cash system is the well-designed cryptographic protocols. The cryptographic protocols should be computationally efficient and could realize the practicality of e-cash system.

The first e-cash system is proposed by David Chaum in 1982 using group blind signature [1]. Since then various e-cash systems were proposed with different digital signature schemes. The famous ACJT group signature [2] scheme was put forward in 2000 [3]. After that, e-cash system based on group signature had a fully development. With the good properties such as anonymity, revocable anonymity, unlinkability, and unforgeability of group signature, the practical and provable secure ACJT scheme is widely used in e-cash system and other areas. There are lots of e-cash systems based on ACJT group signature were presented such as [4]-[8]. However, they all have some shortcomings. For example, some of them are inefficient and couldn't prevent money-tampering. How to construct an efficient and secure e-cash system based on group signature is still the research hotspot.

The security and efficiency of e-cash system are rely mainly on the digital signature. In order to construct a better system, we should improve the signature scheme used in it.

ISSN: 1738-9976 IJSIA

With some good properties, ACJT has great research value and significance in improving e-cash system. Scholars proposed many improvement schemes of ACJT. As for the main defect that ACJT lacks member revocation algorithm, scholars try to construct revocable ACJT group signature schemes via various methods. In Song et al.'s two member revocation schemes [9], the calculation is large and verification algorithm depends linearly on the number of members. Also it is inefficient. Camenisch et al. proposed a revocation algorithm used dynamic accumulator [10]. This algorithm has some shortcomings too. It is not practical that the signature would be invalid if a member was revoked. Chen et al. used zero-knowledge proof of coprime product to construct a new revocation scheme [11], but the certificate used in it depends linearly on the number of members. It increases the computation and doesn't fit large groups. In order to improve the security of ACJT, Zhang et al. designed a new revocable forward secure group signature [12]. Although it is efficient, it proves to be insecure later [13]. Xiao et al. improved ACJT in exculpability [14]. However, the scheme is inefficient.

In this paper, we proposed an elliptic forward secure revocable certificateless group signature scheme based on improvement schemes in recent years. We also presented an e-cash system on the basis of this group signature scheme. It is a fair off-line multi-bank system with high efficiency and security. The system also possesses unforgeability, and could avoids double-spending and money-tampering. Section 2 in this paper introduces the group signature scheme. Section 3 describes the e-cash system. Then we introduce the design principles of the group signature scheme in section 4. Section 5 is about the analyses of the group signature scheme and the e-cash system. Some conclusions are given in section 6.

## 2. Efficient Certificateless Group Signature Scheme

Our group signature scheme comprises the following steps: setup, group manager and group member's partial key extraction, member joining, signing, verifying, opening, updating and member revocation.

The scheme consists three main bodies: key generation center KGC, group manager GM and group member A.

The specific processes are as follows:

Let $F_q$ be the finite field with order $q$. The elliptic curve E is $y^2 = x^3 + ax + b$ where $a, b \in F_q$ and $\Delta = 4a^3 + 27b^2 \neq 0$. $P \in E(F_q)$ is the generator of $E(F_q)$. The order of element in $E(F_q)$ is $n, n > 2^{160}$.

Let the system security parameter be $k \in N$ and the secure Hash is $H : \{0,1\}^* \to Z_p$.

### 2.1. Setup

① KGC chooses $x \in Z_q^*$ secretly and computes $P_{pub} = xP$.

$P_{pub}$ denotes the public key of KGC . The system master key is $x$ and the system parameters are $(F_q, E / F_q, q, P, P_{pub}, H)$.

### 2.2. Partial key extraction for group manager GM

The partial key of group manager GM is generated by KGC. Let $ID_{GM}$ be the identity information of GM. The interaction between KGC and GM is as follows:

①GM chooses his secret value $x_{GM} \in Z_q^*$ randomly and computes $P_{GM} = x_{GM} P$.

 

Then he sends $(ID_{GM}, P_{GM})$ to KGC.

②KGC chooses $r_{GM} \in Z_q^*$ randomly and calculates $R_{GM} = r_{GM}P, s_{GM} = r_{GM} + xH(ID_{GM} \| P_{GM} \| R_{GM})$. Then he sends GM $(R_{GM}, s_{GM})$ through the secure channel.

③GM verifies the equation $s_{GM}P = R_{GM} + P_{pub}H(ID_{GM} \| P_{GM} \| R_{GM})$. If the equality doesn't hold, the process would get back to the last step. If the equality holds, GM would accept $s_{GM}$ as part of his partial key. Then KGC stores GM's information $(ID_{GM}, P_{GM}, S_{GM}, s_{GM})$ and put the public key of GM in the open public key list. After these steps, GM gets his partial key $SK_{GM} = (x_{GM}, s_{GM})$, and the corresponding public key is $PK_{GM} = (x_{GM}P, s_{GM}P) = (P_{GM}, S_{GM})$.

We named this process as $Ext(GM)$.

### 2.3. Partial key extraction for group member A

Group member A's partial key is also generated by KGC. And his identity information is $ID_A$. The interaction between KGC and A is as follows:

①A chooses his secret value $x_A \in Z_q^*$ randomly and computes $P_A = x_AP$. Then he sends $(ID_A, P_A)$ to KGC.

②KGC chooses $r_A \in Z_q^*$ randomly and calculates $R_A = r_AP, s_A = r_A + xH(ID_A \| P_A \| R_A)$. Then he sends A $(R_A, s_A)$ through the secure channel.

③A verifies the equation $s_AP = R_A + P_{pub}H(ID_A \| P_A \| R_A)$. If the equality doesn't hold, the process would get back to the last step. If the equality holds, A would accept $s_A$ as part of his partial key. Then KGC stores A's information $(ID_A, P_A, S_A, s_A)$.

After these steps, A gets his partial key $SK_A = (x_A, s_A)$, and the corresponding public key is $PK_A = (x_AP, s_AP) = (P_A, S_A)$.

We named this process as $Ext(A)$.

### 2.4. Member Joining

If a member A wants to join the group, GM would generate validity duration of signature $T$ for A. We suppose that the current time period is $i$. The interaction between GM and A follows these steps:

①A first chooses $x_{A,i} \in Z_q^*$ and computes $Y_{A,i} = x_{A,i}P$. He then chooses $u \in Z_q^*$ and calculates $t = uP, h_{A,i} = H(ID_A \| PK_A \| Y_{A,i} \| t \| T), s_{A,i} = u - h_{A,i}SK_A = u - h_{A,i}(x_A + s_A)$. Then A sends $(ID_A, Y_{A,i}, h_{A,i}, s_{A,i})$ to GM through the secure channel.

②In order to get A's public key, GM sends $ID_A$ to KGC. KGC then sends $PK_A = (P_A, S_A)$ to GM. GM computes $t' = s_{A,i}P + h_{A,i}PK_A = s_{A,i}P + h_{A,i}(P_A + S_A)$ and verifies the equality of $h'_{A,i} = H(ID_A \| PK_A \| Y_{A,i} \| t' \| T) = h_{A,i}$. If the equality holds, GM generates certificate for A. If not, the process would get back to the last step. GM chooses $e_A \in Z_q^*$ and calculates $E_{A,i} = (SK_{GM} + Y_{A,i}) \cdot e_A^{-1}$. Then he sends

$(E_{A,i}, e_A, PK_{GM})$ to A and stores $(ID_A, PK_A, Y_{A,i}, E_{A,i}, e_A, h_{A,i}, s_{A,i})$ in his member information list.

③A first verifies whether there exists the $PK_{GM}$ in the open public key list which KGC published before. Then he would verify the equation $E_{A,i} \cdot e_A \cdot P = PK_{GM} + P \cdot Y_{A,i}$.

We named this process as $Join_{GM}(A)$.

### 2.5. Signing

①A chooses $w \in Z_q^*$ randomly and computes
$T_{A,i} = P \cdot E_{A,i} + w \cdot P \cdot PK_{GM}, T_2 = wP$.

②Then A chooses $r_1, r_2, r_3 \in Z_q^*$ and calculates

$$d_1 = r_1 \cdot T_{A,i} - r_2 \cdot P \cdot PK_{GM}, d_2 = r_1 T_2 - r_2 P, d_3 = r_3 P, d_4 = r_4 T_2$$
$$c = H(Y_{A,i} \| PK_{GM} \| T_{A,i} \| T_2 \| d_1 \| d_2 \| d_3 \| d_4 \| m \| i)$$
$$s_1 = r_1 - ce_A, s_2 = r_2 - ce_A w, s_3 = r_3 - cw, s_4 = r_4 - cx_{A,i} w^{-1}$$

③A outputs the signature that is $(c, s_1, s_2, s_3, s_4, T_{A,i}, T_2, i)$.

We named this process as $Sig_A(m)$.

### 2.6. Verification

The verifier should calculates
$c' = H(Y_{A,i} \| PK_{GM} \| T_{A,i} \| T_2 \| c(PK_{GM} + P \cdot Y_{A,i}) + s_1 T_{A,i} - s_2 \cdot P \cdot PK_{GM} \| s_1 T_2 - s_2 P \| cT_2 + s_3 P \| cY_{A,i} + s_4 T_2 \| m \| i)$.

And he accepts the signature if and only if $c = c'$.

We named this process as $Verify_A(m)$.

### 2.7. Opening

①A sends $T'_{A,i} = E_{A,i} + w \cdot PK_{GM}$ to GM.

②In order to get $E_{A,i}$, GM should computes $E_{A,i} = T'_{A,i} - SK_{GM} T_2$ by his partial key and prove that $PK_{GM} / P = (T'_{A,i} - E_{A,i}) / T_2$。

We named this process as $Open(A)$.

### 2.8. Updating

When the system gets into the period (i+1) from period i, A would update his sign key by the following steps:

①A computes $x_{A,i+1} = x_{A,i}^r \bmod(q-1), r \in Z_q^*$.

②GM updates the member information he stored before to $(ID_A, PK_A, Y_{A,i+1}, E_{A,i+1}, e_A, h_{A,i+1}, s_{A,i+1})$.

We named this process as $Update(A)$.

### 2.9. Member revocation

If a member A in period j need to be revoked, GM and KGC would update their

 

stored member information by the following steps:

①GM stores A's information $(ID_A, PK_A, Y_{A,j}, E_{A,j}, e_A, h_{A,j}, s_{A,j})$ in his internal revocation list and put some information $(PK_A, Y_{A,j}, T_{A,j}, T_2, j)$ to the open revocation list. Then GM sends the revocation signal to KGC.

②KGC stores A's information into his internal revocation list.

We named this process as $\mathrm{Re}voke(A)$.

There are two figures in our group signature scheme. The first one is the member joining process and the second one is the signing and verifying process.

**Member joining**

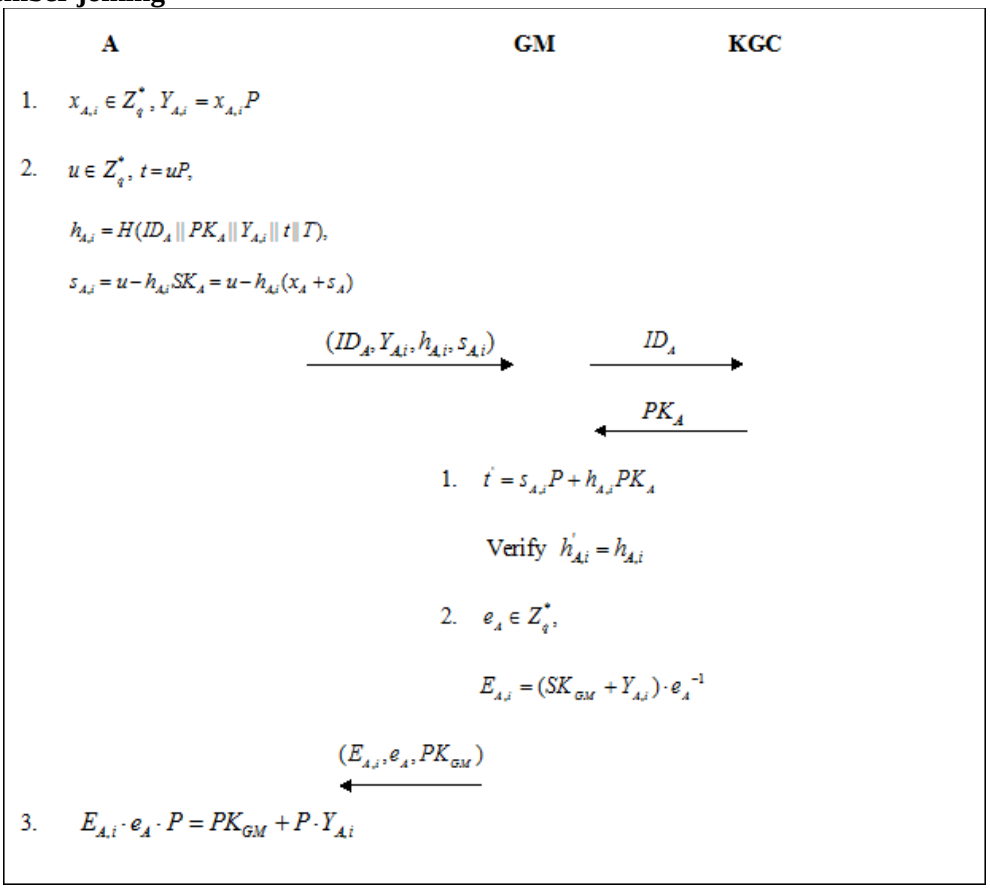


**Figure 1. Member Joining**

**Signing and verification**

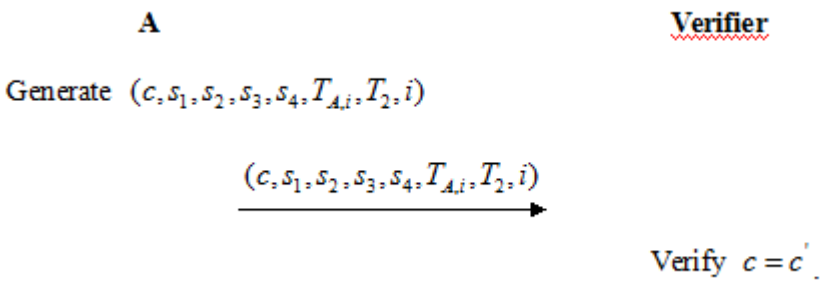


Figure 2. Signing and verification

## 3. Fair off-line Multi-bank e-cash System

Our e-cash system is constructed on the basis of the new group signature scheme. The specific processes are as follows:

This e-cash system consists of two groups: ① The first group $G_1$ is the user group. It contains users and merchants. And we call them $U_i$. The manager of this group $GM_1$ is a trusted third party or $TTP$ for short. ②The second group $G_2$ is the bank group. This group includes many branches $B_i$. And the central bank $B$ is the group manager $GM_2$.

Figure 3 is the structure of our e-cash system.

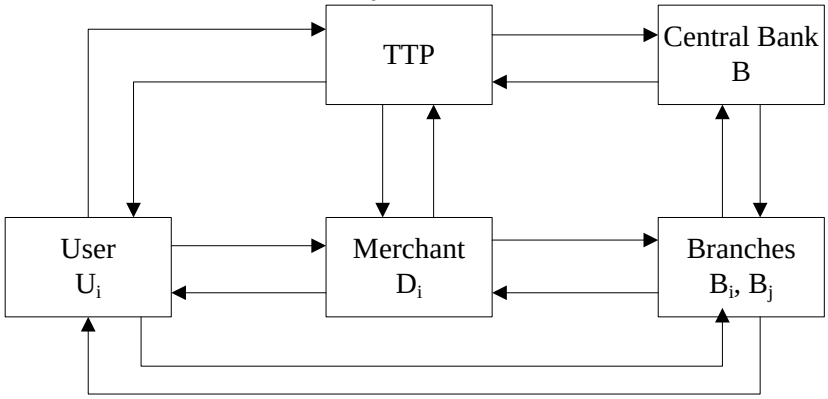


**Figure 3. E-cash System**

### 3.1. Setup

KGC consists of TTP and B.

**Setup of User group $G_1$, and $G_2$**

①Group manager $GM_1$ chooses his master key $x_1 \in Z_q^*$ randomly and computes his public key $P_{pub1} = x_1 P$.

②User $U_i$ chooses his secret value $x_U \in Z_q^*$ and calculates $Y_U = x_U P$. Then he generates his partial key $SK_U = (x_U, s_U)$ using $Ext(U_i)$ algorithm. The corresponding public key is $PK_U = (P_U, S_U)$. Then TTP stores user's information $(ID_U, P_U, S_U, s_U)$.

③Group manager $GM_2$ randomly chooses $x_2 \in Z_q^*$ as his master key. Then he calculates $P_{pub2} = x_2 P$ as his public key.

④Branch $B_i$ chooses his secret value $x_{Bi} \in Z_q^*$ and computes $Y_{Bi} = x_{Bi} P$. $B_i$ extracts his partial key $SK_{Bi} = (x_{Bi}, s_{Bi})$ by $Ext(B_i)$ algorithm. The corresponding public key of $B_i$ is $PK_{Bi} = (P_{Bi}, S_{Bi})$. Then $B$ stores information of $B_i$ which contains $(ID_{Bi}, P_{Bi}, S_{Bi}, s_{Bi})$ and publishes $PK_{Bi} = (P_{Bi}, S_{Bi})$.

### 3.2. Registration

The issuing bank $B_i$ generates validity duration $T$. The interaction between $U_i$ and $B_i$ is as follows:

 

①We suppose that current period is $i$. User $U_i$ randomly chooses $x_{U,i} \in Z_q^*$ and computes $Y_{U,i} = x_{U,i}P$.

②$U_i$ gets his certificate $(E_{U,i}, e_U)$ using $Join_{Bi}(U_i)$ algorithm. The issuing bank $B_i$ stores $(ID_U, PK_U, Y_{U,i}, E_{U,i}, e_U, h_{U,i}, s_{U,i})$.

### 3.3. Withdrawal Protocol

In this process, $U_i$ would contact with the bank again. For the sake of preventing user's extortion or deceitful actions, the information of users needs to be verified again. $U_i$ would make a group signature for denomination m.

①By the $Sig_{Ui}(m)$ algorithm, $U_i$ generates the group signature $(c, s_1, s_2, s_3, s_4, T_{U,i}, T_2, i)$. Then he computes $T_{U,i}^{'} = E_{U,i} + w \cdot PK_{Bi}$. After that, $U_i$ sends withdraw request $req$ and $T_{U,i}^{'}, (c, s_1, s_2, s_3, s_4, T_{U,i}, T_2, i), T_{U,i}^{'}, Y_{U,i}, PK_{Bi}, m$ to the transaction bank $B_j$.

② $B_j$ first examines whether there exists the $PK_{Bi}$ in the open public key list which was announced by central bank $B$ before. Then he verifies the validity of the signature using $Verify_{Ui}(m)$ algorithm.

③Branch $B_j$ generates $m-time$ which is the mark of denomination m. Let $M = (m, m-time, T_{U,i}, T_2)$. $B_j$ chooses $v \in Z_q^*$ randomly and calculates $t_2 = vP, h_{i2} = H(PK_{Bj} \| M \| t_2 \| T \| i), s_{i2} = v - h_{i2}SK_{Bj}$. Then he sends $(m, m-time, h_{i2}, s_{i2}, PK_{Bj})$ to $U_i$ via the secure channel.

④$U_i$ computes $t_2^{'} = s_{i2}P + h_{i2}PK_{Bj}$ and verifies the equality of $h_{i2}^{'} = H(PK_{Bj} \| M \| t_2^{'} \| T \| i) = h_{i2}$.

### 3.4. Payment Protocol

①We suppose that the current time period is $j$. $U_i$ sends $(c, s_1, s_2, s_3, s_4, T_{U,i}, T_2, i), h_{i2}, s_{i2}, T_{U,i}^{'}, m^{'}, Y_{U,i}, PK_{Bi}, PK_{Bj}$ to the merchant $D_i$.

② $D_i$ would verify the validity duration of e-cash and whether $U_i$ has been revoked or not. Then he would examine the correctness of the group signature by $Verify_{Ui}(m)$ algorithm and the validity of the authenticated signature of e-cash made by the issuing bank. He could compute $t_2^{'} = s_{i2}P + h_{i2}PK_{Bj}$ and check the equality of $h_{i2}^{'} = H(PK_{Bj} \| M \| t_2^{'} \| T \| i) = h_{i2}$ to determine the correctness of issuing bank's signature. If all above holds, he would generate the payment time. We named it $pay-time$.

### 3.5. Deposit Protocol

①We assume the current time period is $k$. The merchant $D_i$ sends all information $(c, s_1, s_2, s_3, s_4, T_{U,i}, T_2, i), h_{i2}, s_{i2}, T_{U,i}^{'}, m^{'}, Y_{U,i}, PK_{Bi}, PK_{Bj}, pay-time$ to the branch $B_l$.

② $B_l$ would verify the validity duration of the e-cash and whether $U_i$ and $D_i$ have been revoked before. He would also examine whether $D_i$ has illegal actions or not.

③ $B_l$ deposits m in $D_i$ 's account.

### 3.6. Updating Protocol

The system updates by $Update(D_i), Update(U_i)$ and $Update(B_i)$ algorithms.

### 3.7. Member Revocation

If $U_i$ in time period $i$ needs to be revoked, then the system would use $Revoke(U_i)$ algorithm. After that $B_i$ and KGC update their member information respectively.

### 3.8. Tracing Protocol

If there was one verification of the transaction failed, the merchants or branches could send $PK_{Bi}$ in member information to the central bank $B$. Then $B$ would trace the corresponding issuing bank and the issuing bank could open user's identity by his group signature. Finally, the issuing bank would mark the e-cash and put it in the illegal e-cash library. Besides, the issuing bank wouldn't deduct m in user's account.

## 4. Design Principles

In this paper, we combine the group signature with e-cash in order to present an efficient and secure system. By analyses of improvement measures, we proposed a new group signature scheme and applied it to the design of e-cash system. This group signature scheme has the following features:

① The scheme has established an elliptic curve model and has chosen specific elliptic curves. We operate this scheme using point multiplication and addition in elliptic curves thanks to the efficiency. Elliptic curve discrete logarithm problem is more complex than standard discrete logarithm problem. Based on the known solutions of encryption, the security of 160-bit elliptic curve cryptography is equal to the 1024-bit RSA. That's why ECC can achieve high security with short key length. The new scheme in this paper uses different parameters. Compared to ACJT which has complex modular multiplication, exponentiation and large computation, our scheme has less parameters, shorter signature length and could be operated in a shorter time. Table 1 is a comparison between ACJT and our scheme.

**Table 1. Comparison between ACJT and our Scheme**

| Name of schemes | Parameters in group public key | Group public key length | Parameters in signing algorithm | Signature length | Time of signing, verification and opening |
|---|---|---|---|---|---|
| **ACJT** | 6 | 6144 bits | 13 | 8192 bits | 45840 ms |
| **Our scheme** | 2 | 320 bits | 11 | 1136 bits | 43.21 ms |

② In order to simplify the member joining process, we apply certificateless signature into group signature and design a certificateless signature member joining algorithm. As a result, group member A doesn't need to face certificate distribution problem and key escrow problem when he joins the group. At the beginning of the member joining process, A generates a signature with his partial key and sends it to group manager GM, resulting

 

in one mutual authentication between A and GM. The partial key of A contains the secret value of A and the key extracted from KGC. KGC generates the keys corresponding to A by calculating $R_A = r_A P, s_A = r_A + xH(ID_A \| P_A \| R_A)$. Then A extracts $s_A$ and thus forms his partial key with his secret value. During the member joining process, A generates a knowledge signature with his partial key. After that A calculates $t = uP, h_{A,i} = H(ID_A \| PK_A \| Y_{A,i} \| t \| T), s_{A,i} = u - h_{A,i} SK_A$ and sends $(ID_A, Y_{A,i}, h_{A,i}, s_{A,i})$ to GM. GM gets public key of A from KGC rather than A, and then verifies the signature sent by A before. This algorithm proves the validity of A's identity and could resist against public key substitution attack. After the verification process, GM sends A his certificate.

③ Our scheme provides forward security, so the scheme could avoid key exposure attack and is more secure than the others. In this scheme, we divide the system time into segments. In certain time period i, A's sign key $x_{A,i}$ can be updated by $x_{A,i+1} = x_{A,i}^r \bmod(q-1)$. Without loss of generality, calculating $x_{A,i-1}$ from $x_{A,i}$, requires finding the r-th root module q-1. The difficulty of this problem is equal to the factorization. Hence, the update algorithm is secure. All the sign public key, member certificate and member signature would be updated with the sign key. This update algorithm leads to renew of the system information and doesn't increase the computation. It also helps to keep the stability and reduce the burden of the system. We put the time period i into the signature $(c, s_1, s_2, s_3, s_4, T_{A,i}, T_2, i)$, so the signature has real time ability and is easy to verify.

④ Our scheme has the member revocation algorithm, while ACJT has not. In our scheme, KGC and GM revoke group members by keeping some member revocation lists. The member information renews with time, and thus the information stored in KGC and GM would also change. If A needs to be revoked at some period, KGC and GM would store A's information into the corresponding revocation lists. It is convenient for KGC and GM to verify the identity of A later. At the same time, GM would publish a revocation list which covers A's partial information. The verifier could prove the validity of A's identity through the public revocation list. This revocation algorithm has only simple computation, thus increases the system efficiency.

## 5. Scheme Analyses

This section is mainly about the security and efficiency analyses of group signature scheme and e-cash system.

### 5.1. Group Signature Scheme Analysis

This section aims to analyse the security and efficiency of our group signature scheme. Security analysis is mainly about unforgeability and forward security. The efficiency analysis of our scheme is concluded from comparing with other schemes.

#### 5.1.1. Unforgeability

There are two types of adversaries in certificateless signature. We named them adversary Ⅰ and adversary Ⅱ. Type Ⅰ attack is the public key substitution attack. Type Ⅱ attack is the malicious KGC attack.

**Type Ⅰ attack**

An adversary $A_1$ has access to the system information except for member A's secret

value $x_A$. $A_1$ may replace A's public keys, extract A's partial key and generate encrypted texts. Now we will prove it is unsuccessful.

$A_1$ simulates the interaction between member A and KGC firstly. He could get $s_A$. Then he replaces A's secret value with his own secret value $x_{A1}$. $A_1$ gets his partial key $(x_{A1}, s_A)$. And he replaces A's partial public key $(P_A, S_A)$ with his $(P_A, S_{A1})$. Secondly, $A_1$ simulates the interaction between A and GM. In the member joining process, $A_1$ generates sign key $x_{A1,i}$ which respects to the time period. Then he signs for $Y_{A1,i}$ using $(x_{A1}, s_A)$ and calculates $t_1 = uP, h_{A1,i} = H(ID_A \| PK_{A1} \| Y_{A1,i} \| t_1 \| T), s_{A1,i} = u - h_{A1,i} SK_{A1}$. $A_1$ sends $(ID_A, h_{A1,i}, s_{A1,i})$ to GM. GM needs to contact with KGC to get member's public key. GM will calculate $t_1^{'} = s_{A1,i} P + h_{A1,i} PK_A$ and verity the equation $h_{A1,i}^{'} = H(ID_A \| PK_A \| Y_{A1,i} \| t_1^{'} \| T)$. It is clear that $h_{A1,i}^{'} \neq H(ID_A \| PK_{A1} \| Y_{A1,i} \| t_1^{'} \| T)$. Hence, the type Ⅰ attack is unsuccessful. Our scheme could avoid the Type Ⅰ attack.

**Type Ⅱ attack**

An adversary $A_2$ may have access to the system master key of KGC, but he couldn't replace member A's partial public key. $A_2$ may forge some information of A and generate some encrypted texts. It is unsuccessful. We prove it as follows:

$A_2$ has access to A's partial public key $(P_A, S_A)$. It is impossible for him to calculate A's secret value because of the difficulty of ECDLP. If $A_2$ forges A's partial key $(x_{A2}, s_A)$, he then calculates $t_2 = uP, h_{A2,i} = H(ID_A \| PK_A \| Y_{A2,i} \| t_2 \| T), s_{A2,i} = u - h_{A2,i} SK_{A2}$. GM would compute $t_2^{'} = s_{A2,i} P + h_{A2,i} PK_A$ and verify the equation $h_{A2,i}^{'} = H(ID_A \| PK_A \| Y_{A2,i} \| t_2^{'} \| T)$. It is obvious that $h_{A2,i}^{'} \neq H(ID_A \| PK_A \| Y_{A2,i} \| t_2 \| T)$. Thus, the type Ⅱ attack is unsuccessful. Our scheme could avoid the Type Ⅱ attack.

### 5.1.2. Forward Security

During the updating process, all the public key, member certificate and signature change with A's sign key. The system information renews in different time and the update algorithm doesn't increase the system computation. Besides, the signature of former time periods would still be valid after the member revocation. As a consequence, our scheme provides forward security. Suppose that member A is in period j and his sign key is $x_{A,j}$.

If adversary A' gets A's sign key $x_{A,j}$. It is impossible for him to forge the sign key of period j-1. This conclusion comes in the update algorithm $x_{A,i+1} = x_{A,i}^r \bmod(q-1)$. If A' wants to get $x_{A,j-1}$ from this equation, he would solve the problem of r-th root module q-1. The difficulty of this problem is equal to that of factorization problem. Due to the lack of $x_{A,j-1}$, it is hard for A' to forge member certificate and thus he couldn't generate valid signature.

If A' gets $x_{A,j+1}$ through the update algorithm, it is still hard for him to forge

 

information of A. The reason is that A would sign for $Y_{A,j+1}$ with his partial key $SK_A$ when he joins the group. But A' doesn't know $SK_A$ and he couldn't finish the verification in member joining process. As a result, A' wouldn't get member certificate and forge a valid signature.

### 5.1.3. Efficiency Analysis

The proposed scheme is constructed on elliptic curve model. It achieves high security with shorter key length. It is known that the security of 160-bit ECC is equal to that of 1024-bit RSA.

The computation complexity of modular multiplication and exponentiation in finite fields is high. The low computation cost algorithms like addition in finite fields could be ignored. We estimate the scheme by high computation cost algorithms. Table 2 is the performance comparison among our scheme, ACJT [3], Song et al.'s scheme [9] and Shi et al.'s scheme [20].

**Table 2. Performance Comparison among Several Schemes**

| Name of schemes | Modulus size | Signing Computation | Verification Computation | Opening Computation |
|---|---|---|---|---|
| **ACJT** | ≥1024 bits | 12E+11M | 11E+7M | 1E+1M |
| **Song Ⅰ** | ≥1024 bits | 20E+17M | (22+k)E+13M | 2E+1M |
| **Shi** | ≥1024 bits | 12E+11M | 11E+7M | 1E+1M |
| **Our scheme** | ≥160 bits | 17C+7J | 11C+6J | 1C+1J |
| **Name of schemes** | **Signature length** | **Group public key length** | **Member revocation** | **Forward security** |
| **ACJT** | 8192 bits | 6144 bits | × | × |
| **Song Ⅰ** | 7168 bits | 6144 bits | √ | √ |
| **Shi** | 9232 bits | 6144 bits | √ | √ |
| **Our scheme** | 1136 bits | 320 bits | √ | √ |

E stands for modular multiplication in finite fields.
M stands for modular exponentiation in finite fields.
C stands for point multiplication in elliptic curves.
J stands for point addition in elliptic curves.
T stands for the number of system time periods.
k stands for the number of revoked members.
"×" stands for the lack of the process.
"√" stands for the possession of the process.

Suppose that the element size in finite field G is 1024 bits, the element size in ECC is 160 bits and the time length is 16 bits.

According to the method used in [26], we suppose our CPU is 1.6GHz and RAM is 2.0G. The language we used is C++. We need 1910ms to finish the exponent operation in average. It costs 1.49ms to realize the point multiplication in groups with 160-bit key length. Deducing from the computation, we conclude that the total time of signing, verifying and opening processes in ACJT [3], Song [9], Shi [20] are respectively 45840ms, (84080+1910k) ms and 45840ms while our scheme only needs 43.21ms. It is clear that our scheme has high efficiency.

### 5.2. E-cash System Analysis

This section makes an analysis of some practical properties of our e-cash system and compares it with other systems.

#### 5.2.1. Fair off-line Multi-bank System

In our system, banks needn't online all the time. The system could trace afterwards and thus reduces the bank cost and the communication cost. Besides, it is convenient for users to deposit and withdrawal in any banks in our system.

#### 5.2.2. No Money-tampering and Double-spending

In the withdrawal process, the issuing bank distributes e-cash which contains denomination, e-cash number and the verification signature generated by itself. For the sake of judging whether the user has tampered money or spent money for many times, the verifier could verify the information in e-cash or e-cash libraries. In the interaction process between users and merchants, the merchants would generate a transaction time named pay-time. Pay-time is another mark of e-cash. One can check whether there exists a repeat transaction time to verify the double-spending actions.

#### 5.2.3. Performance Comparison

Table 3 is a comparison among some e-cash systems based on ACJT. The systems are Zhang et al.'s system [7], FEI et al.'s system [8] and our system.

**Table 3. Performance Comparison among Several E-cash Systems**

| Name of systems | Theoretical basis | Modulus size | Member revocation |
|---|---|---|---|
| **Zhang** | DLP | ≥1024 bits | √ |
| **FEI** | DLP | ≥1024 bits | ╳ |
| **Our scheme** | ECDLP | ≥160 bits | √ |
| **Name of systems** | **Forward security** | **Money-tampering** | **Double-spending** |
| **Zhang** | ╳ | √ | √ |
| **FEI** | ╳ | ╳ | ╳ |
| **Our scheme** | √ | ╳ | ╳ |

"√" stands for the possession of the process.
"╳" stands for the lack of the process.

This e-cash system is constructed on the basis of our new group signature scheme. It has high security and efficiency. By comparison, our system founded an elliptic curve model and the modulus size is ≥160 bits while other systems are ≥1024 bits. We realize high security with shorter key length. There is a member revocation algorithm in our system. Group members could join and revoke from the group freely. Our system also provides forward security. The member information updates with time and thus avoids the key exposure attack. It makes the system more secure. For the sake of preventing cyber crime, we design a new algorithm and the system could avoid money-tampering and double-spending actions etc..

## 6. Conclusion

In order to improve the efficiency of e-cash system to achieve quick payment and avoid cyber-crime like forge of information, double-spending and money-tampering, we combine group signature with e-cash system. In this paper, we construct a new e-cash system based on an efficient and secure group signature scheme. The security and efficiency are enhanced compared with the other systems. Besides, we prove that our system could prevent forge of information, double-spending and money-tampering

 

actions.

Our new group signature scheme is designed based on the fact that solving the ECDLP is harder than DLP. We establish an elliptic curve model and choose the specific elliptic curves. The whole scheme is operated by using point multiplication and addition in elliptic curves. Our scheme has lower computation complexity compared to modular multiplication and exponentiation in finite fields used in ACJT. We have less parameters, and the signature length is reduced by 86.13%. The main processes in our scheme cost only ≥40ms while ACJT cost ≥40000ms. Therefore, our scheme is more efficient. We realize a system of high security with shorter key length. We also apply the certificateless signature into our group signature scheme. We design a new member joining algorithm used certificateless signature. In this process, members generate a signature directly using their partial key extracted from KGC, and then the group manager GM verifies them, which reduces the interactions between members and GM and simplifies the member joining process. Also, it decreases the storage pressure of GM. The scheme has strong exculpability and could avoid the combined attacks. In our scheme, member's sign key changes with time through the update algorithm. This provides forward security and prevents the key exposure attacks. There also exists a member revocation algorithm in our scheme, which makes our scheme more practical compared to ACJT without this function. Our scheme proves to be secure and efficient over the other schemes.

## Acknowledgement

This work is supported by Major Program of National Natural Science Foundation of China(11290141), NSFC(61402030), Fundamental Research of Civil Aircraft no.MJ-F-2012-04.

## References

[1] Chaum D, Blind Signature for Untraceable Payments. Proc of Advances in Cryptology- CRYPTO'82. Plenum Press, **(1983)** 199-203; New York.
[2] Chaum D, Heyst F, Group signature. Proceedings of EUROCRYPT'91, LNCS, Springer- Verlag, 257-265 **(1991)**.
[3] Ateniese G, Camenisch J, Joye M, Tsudik G, A practical and provably secure coalition-resistant group signature scheme. Advances in Cryptology-CRYPTO'00, LNCS 1880, Springer-Verlag, 255-270 **(2000)**.
[4] Maitland G, Boyd C, Fair Electronic Cash Based on a Group Signature Scheme. Proc of ICICS'01, Beijing: Journal of Software, 461-465 **(2001)**.
[5] Constantin P, An Off-line Electronic Cash System with Revokable Anonymity. Proc of the 12th IEEE Mediterranean Electro-technical Conference, **(2004)** Dubrovnik, Croatia.
[6] HOU Xiao-song, TAN C H, A new electronic cash model. Information Technology : Coding and Computing, 1 (4-6): 374-379 **(2005)**.
[7] Jingliang Zhang, Lizhen Ma, Yumin Wang, Fair E-cash System without Trustees for Multiple Banks. International Conference on Computational Intelligence and Security Workshops, **(2007)**.
[8] FEI Xiong-wei, LI Qiao-liang, New electronic cash system with higher security and efficiency. Application Research of Computers, 1001-3695 **(2008)** 05-1543-03.
[9] D X Song, Practical forward secure group signature schemes. Proc of the 8th ACM Conf on Computer and Communications Security, ACM Press, **(2001)** 225-234; New York.
[10] J Camenisch, A Lysyanskaya, Dynamic accumulators and application to efficient revocation of anonymous credentials. Proc of Crypto 2002, Springer-Verlag, **(2002)** 61-76; Berlin.
[11] Chen ZW, Wang JL, Huang JW, Wang YM, Huang DR, An efficient Revocation Algorithm in ACJT Group Signature. Journal of Software, 16(1): 151-157 **(2005)**. (in Chinese)
[12] J Zhang, Q Wu, Y Wang, A novel efficient group signature scheme with forward security. Proc of Int'l Conf on Information and Communications Security(ICICS'03), Springer-Verlag, **(2003)** 292-300; Berlin.
[13] G Wang, On the security of a group signature scheme with forward security. Proc of Int'l Conf on Information Security and Cryptology-ICISC 2003, Springer-Verlag, **(2003)** 27-39; Berlin.
[14] Xiao Pingan, Yang Ling, Study on identity authentication scheme based group signature. Lanzhou University, 6-9 **(2013)**. (in Chinese)
[15] Li Rupeng, Yu Jia, Li Guowen, Li Daxing, Forward Secure Group Signature Schemes with Efficient Revocation. Journal of Computer Research and Development, 44(7): 1219-1226 **(2007)**. (in Chinese)

 

[16] Camenish J, Stadler M, Efficient Group Signature Schemes for Large Groups. Advances in Cryptology-CRPTO'97, LNCS 1294, Springer-Verlag, **(1997)** 410-424; Berlin.
[17] Ateniese G, Song D, Tsudik G, Quasi-efficient revocation in group signature. Financial Crypotography(FC'02), LNCS 2357, Springer-Verlag, **(2002)** 183-197; Berlin.
[18] Sattam S, Al-Riyami, Kenneth G, Paterson, Certificateless Public Key Cryptography. International Association for Cryptologic Research. LNCS 2894, pp. 452–473 **(2003)**.
[19] CHEN Shao-Zhen, LI Da-Xing, An Efficient Revocable Group Signature Schemes with Forward Security. Chinese Journal of Computers, June **(2006)**: Vol. 29 No. 6. (in Chinese)
[20] SHI Rong-hua, LONG Cheng-sheng, ACJT group signature scheme with forward security. Computer Engineering and Application, 44(7): 126-128 **(2008)**. (in Chinese)
[21] LI Mao-tang, YANG Xiao-yuan, HAN Yi-liang, et al, New ACJT based generalized group signcryption. Computer Engineering and Applications, 44(31): 128-131 **(2008)**. (in Chinese)
[22] ZHANG Xing-lan, Efficient group signature scheme. Application Research of Computers, November, Vol. 26 No. 11 **(2009)**. (in Chinese)
[23] Han Xiaohua, Li Qiaoliang, Yuan Yuqing, An Electronic Cash System with Multiple Banks Based on ECC Group Signature Scheme. Journal of Computer Research and Development, 46(Suppl.): 306-310 **(2009)**. (in Chinese)
[24] D.He, J Chen, R.Zhang, An efficient and provably-secure certificateless signature scheme without bilinear pairings. International Journal of Communication Systems,25 **(2012)**:1432-1442.
[25] Sujata Mohanty, Bansidhar Majhi, Subhalaxmi Das, A secure electronic cash based on a certificateless group signcryption scheme. Mathematical and Computer Modelling. 58 **(2013)**: 186–195.
[26] Xin Xu, Ping Zhu, Qiaoyan Wen, Zhengping Jin, Hua Zhang, Lian He, A Secure and Efficient Authentication and Key Agreement Scheme Based on ECC for Telecare Medicine Information Systems. Springer Science+Business Media New York 2013, J Med Syst **(2014)** 38: 9994.

## Authors

**Liang Yan**, female, master in reading and the member of Key Laboratory of Mathematics, Informatics and Behavioral Semantics, Ministry of Education. Her research interests include information security and cryptography.

**Zhang Xiao**, female, associate professor of Mathematics at Beihang University and the member of Key Laboratory of
Mathematics, Informatics and Behavioral Semantics, Ministry of Education. Her research interests include cryptography, information security and complex informationsystem.

**Zhiming Zheng**, male, professor of Mathematics at Beihang University and the director of Key Laboratory of Mathematics, Informatics and Behavioral Semantics, Ministry of Education. His research interests include information security, complex infor mation system and dynamic system.

 