Security Analysis on "Secure Untraceable Off-line Electronic Cash System"

Feng Wang[1,2], Chin-Chen Chang[2], and Changlu Lin[3]
(Corresponding author: Chin-Chen Chang)

College of Mathematics and Physics, Fujian University of Technology[1]
Fuzhou, Fujian, 350118, China
Department of Information Engineering and Computer Science, Feng Chia University[2]
100 Wenhwa Rd., Seatwen, Taichung 40724, Taiwan
College of Mathematics and Computer Science, Fujian Normal University[3]
Fuzhou, Fujian, 350117, China
(Email: alan3c@gmail.com)

(Received Aug. 29, 2013; revised and accepted Feb. 18 & Aug. 12, 2014)

Abstract

In 2013, Baseri et al. proposed an untraceable off-line electronic cash scheme from the RSA cryptosystem. They used a method that injects the expiration date and the spenders identity onto the coin to prevent double spending. The authors claimed that the scheme provides the properties of anonymity, unforgeability, double spending detection, and date attachability. Unfortunately, we find that there are security flaws in terms of verifiability, unreuseablity, and unforgeability. First, the verifiable method of e-cash in their scheme is not correct according to Euler's Theorem. Second, malicious spenders can inject a false identity in the withdrawal phase due to the homomorphic property of modular operation. Therefore, coins can be doubly spent without being detected. Finally, a malicious spender or merchant can forge valid coins using existing coins.

Keywords: Electronic cash, off-line, unforgeability, unreuseablity, verifiability

1 Introduction

Electronic cash (e-cash) is a more convenient method of payment in electronic commerce compared with traditional paper cash. In general, there are three entities involved in an e-cash scheme: the bank, the spender, and the merchant. The spender withdraws the e-cash from his/her account and then pays it to the merchant for some goods or services. The merchant sends his/her e-cash to the bank and deposits it in his/her account. An e-cash scheme should have the properties of unforgeability, untraceable, verifiability, unreuseability [12] and might have properties such as divisibility, transferability, anonymity revocation, and so on. Depending on whether the bank attends to the transaction between the spender and the merchant, the e-cash scheme can be classified into two categories: on-line and off-line. In an on-line scheme [2, 8, 10], the bank must attend the transaction to detect double spending; thus, it exhausts the most resources of the bank. An off-line scheme is more efficient, because the bank does not attend the transaction. Therefore, the off-line e-cash scheme is a more attractive research area.

Since Chaum et al. [3] proposed the first off-line e-cash scheme, numerous such schemes have been presented [1, 4, 6, 7, 13]. Unfortunately, there is not a globally acceptable off-line scheme, and many existing schemes have security flaws [1, 5, 9]. For example, in 2013, Baseri et al. [1] found that Eslami and Talebi's scheme [6] had three faults: attacking double spender detection, forging the expiration, and cheating on exchange protocol. They proposed a secure untraceable off-line electronic cash scheme from the RSA cryptosystem. In order to prevent double spending, they injected the expiration date and the spender's identity onto the coin in withdrawal phase. In [1], it was claimed that the author's scheme provides the properties of anonymity, unforgeability, double spending detection, and date attachability. However, we find that Baseri et al.'s scheme has some security flaws such as verifiability, double spending detection, and unforgeability. First, the verifiable method of e-cash in this scheme is not correct according to Euler's Theorem [11], and we revise the flaw in Subsection 3.1. Second, malicious spenders can withdraw coins without injected their actual identities using the homomorphic property of modular operation. Therefore, the bank cannot detect their identities when double spending occurs. Finally, a malicious spender or merchant can forge valid coins using existing coins due to the homomorphic property of modular operation.

The rest of the paper is organized as follows. In Section 2, we review Baseri et al.'s scheme [1]. We describe the flaws of Baseri et al.'s scheme in detail in Section 3, and the conclusion is given in Section 4.

2 Review of Baseri et al.'s Scheme

Baseri et al.'s [1] scheme has four participants: a central authority, the bank, the spender, and the merchant. It also has six phases: initialization, opening an account, withdrawal, payment, deposit, and exchange. We describe them as follows, excluding the exchange phase for simplification.

2.1 Initialization

The central authority (CA) selects two distinct large primes p and q, computes $n = p \cdot q$ and $\varphi(n) = (p-1)\cdot(q-1)$, picks two random numbers $g_1, g_2 \in_R Z_n^*$ with the same large prime order l. Next, he/she picks two random numbers $e'_B, e_B \in_R Z_{\varphi(n)}^*$ such that $e'_B < e_B$ and $\gcd(e'_B, \varphi(n)) = \gcd(e_B, \varphi(n)) = 1$, computes $1/e'_B$ and $1/e_B$ such that $e'_B \cdot (1 \setminus e'_B) = e_B \cdot (1 \setminus e_B) = 1(\bmod \varphi(n))$. Then, CA selects a one-way hash function H, publishes $(g_1, g_2, n, e'_B, e_B, H)$ and keeps $(1/e'_B, 1/e_B, \varphi(n))$ secretly.

2.2 Opening an Account

Step 1. Spender → Bank: ID_C and a zero knowledge proof that he/she knows u.

The spender selects $u \in_R Z_{e_B}^*$, computes $ID_C = g_1^u (\bmod\ n)$ such that $g_1^u \cdot g_2 \neq 1 (\bmod\ n)$, and generates a zero knowledge proof that he/she knows the discrete logarithm of ID_C.

Step 2. Bank → Spender: O_1.

The bank checks the zero knowledge proof and computes $A = ID_C \cdot g_2 (\bmod\ n)$, and $O_1 = A^{1/e_B} (\bmod\ n)$.

2.3 Withdrawal

Step 1. Spender → Bank: (ω_1, ω_2, t).

The spender chooses $x_1, x_2 \in_R Z_{e'_B}^*$ and $s, b_1, b_2 \in_R Z_n^*$. Then, the spender computes $A' = A^s (\bmod\ n)$, $B = g_1^{x_1} \cdot g_2^{x_2} (\bmod\ n)$, $\omega_1 = B \cdot b_1^{e'_B} (\bmod n)$, and $\omega_2 = (A' + B) \cdot b_2^{(e_B * t)} (\bmod n)$.

Step 2. Bank → Spender: (O_2, O_3), where $O_2 = \omega_1^{1/e'_B} (\bmod n)$, and $O_3 = \omega_2^{1/(e_B * t)} (\bmod n)$.

Step 3. The spender stores $(A', B, s_1, s_2, s_3, t)$ as his/her *Coin*, where $s_1 = O_1^s (\bmod n)$, $s_2 = O_2 / b_1 (\bmod n)$, and $s_3 = O_3 / b_2 (\bmod n)$.

Note that s_1, s_2, and s_3 are signatures of A', B, and $A' + B$, respectively, with the private keys $1/e_B$, $1/e'_B$, and $1/(e_B * t)$, respectively.

2.4 Payment

Step 1. Spender → Merchant: *Coin*, where $Coin = (A', B, s_1, s_2, s_3, t)$.

Step 2. Merchant →Spender: d.

The merchant checks whether $A' \neq 0$, and s_1, s_2, and s_3 are valid signatures of A', B, and $A' + B$, respectively, and computes $d = H(A', B, ID_M, date||time)$.

Step 3. Spender → Merchant: (r_1, r_2), where $r_1 = d \cdot u \cdot s + x_1 (\bmod e_B)$, and $r_2 = d \cdot s + x_2 (\bmod e_B)$.

Step 4. The merchant accepts the *Coin* if equation $g_1^{r_1} \cdot g_2^{r_2} = (A')^d \cdot B (\bmod n)$ holds.

2.5 Deposit

Step 1. Merchant → Bank: $(Coin, r_1, r_2)$.

Step 2. The bank checks the merchant's identity and the validity of the *Coin*. If valid, he/she checks whether the *Coin* is in the bank's database; if not, he/she stores the *Coin*. If there is another $(Coin, r'_1, r'_2)$ in the bank's database, then the bank can detect the identity of the malicious spender by $u = \frac{r_1 - r'_1}{r_2 - r'_2} (\bmod e_B)$, and $ID_C = g_1^u (\bmod\ n)$.

3 Security Analysis of Baseri et al.'s Scheme

3.1 Dissatisfying Verifiability

In Steps 3 and 4 of the payment phase, the spender sends $r_1 = d \cdot u \cdot s + x_1 (\bmod e_B)$, and $r_2 = d \cdot s + x_2 (\bmod e_B)$ to the merchant, who accepts the *Coin* if $g_1^{r_1} \cdot g_2^{r_2} = (A')^d \cdot B (\bmod n)$ holds. In order to ensure that $g_1^{r_1} \cdot g_2^{r_2} = (A')^d \cdot B (\bmod n)$ holds, the r_1 and r_2 should be revised as $r_1 = d \cdot u \cdot s + x_1 (\bmod \varphi(n))$ and $r_2 = d \cdot s + x_2 (\bmod \varphi(n))$, respectively, according to Euler's theorem [11], which states that if n and a are coprime positive integers, then $a^{\varphi(n)} = 1 (\bmod n)$. However, the spender does not know the value $\varphi(n)$. Thus, we can only adopt an inefficient method to revise them into $r_1 = d \cdot u \cdot s + x_1$, $r_2 = d \cdot s + x_2$. Furthermore, the value in Step 2 of the deposit phase must be modified as $u = (\frac{r_1 - r'_1}{r_2 - r'_2} (\bmod \varphi(n))) (\bmod e_B)$.

3.2 Attacking Double Spending Detection

A malicious spender can forge one identity in the withdrawal phase to avoid being detected when he/she doubly spends the e-cash. Suppose that the malicious spender changes his/her identity ID_C into ID^*_C. When double spending occurs, the bank cannot find who doubly spends the e-cash even if he/she has computed ID^*_C. There are two methods to forge the spender's identity, as described in the following.

3.2.1 Forging Identity Independently

We first describe how to forge one identity without any help from a third party.

Step 1. The malicious spender executes the withdrawal phase similar to Baseri et al.'s scheme except that he/she changes ω_1 and ω_2 into $\omega_1 = g_1 \cdot b_1^{e'_B} (\bmod n)$, and $\omega_2 = g_1 \cdot b_2^{(e_B * t)} (\bmod n)$. Of course, the coin that he/she obtained is not a valid coin, but he/she can obtain $s_2 = O_2/b_1 (\bmod n) = g_1^{1/e'_B}$, $s_3 = O_3/b_2 (\bmod n) = g_1^{1/(e_B*t)}$. Then, he/she computes $g_1^{1/e_B} = s_3^t (\bmod n)$, and denotes $\alpha_1 = g_1^{1/e_B} (\bmod n)$, $\beta_1 = g_1^{1/e'_B} (\bmod n)$, and $\gamma_1 = g_1^{1/(e_B*t)} (\bmod n)$. Similarly, he/she can obtain $\alpha_2 = g_2^{1/e_B} (\bmod n)$, $\beta_2 = g_2^{1/e'_B} (\bmod n)$, and $\gamma_2 = g_2^{1/(e_B*t)} (\bmod n)$.

Step 2. The spender executes the withdrawal phase similar to Baseri et al.'s scheme except that he/she changes A' into $A'^* = g_1^{a_1} \cdot g_2^{a_2} (\bmod n)$, where $a_1, a_2 \in_R Z^*_{e'_B}$. Then, he/she computes $s_1^* = \alpha_1^{a_1} \cdot \alpha_2^{a_2} (\bmod n)$. This is a valid signature of A'^*, because $(s_1^*)^{e_B} = (\alpha_1^{a_1} \cdot \alpha_2^{a_2})^{e_B} = ((g_1^{1/e_B})^{a_1} \cdot (g_2^{1/e_B})^{a_2})^{e_B} = g_1^{a_1} \cdot g_2^{a_2} = A'^* (\bmod n)$. Furthermore, he/she can change r_1 and r_2 into $r_1^* = d \cdot a_1 + x_1$ and $r_2^* = d \cdot a_2 + x_2$, respectively, in the payment phase and can pass all verifications, because $g_1^{r_1^*} \cdot g_2^{r_2^*} = g_1^{d \cdot a_1 + x_1} \cdot g_2^{d \cdot a_2 + x_2} = (g_1^{a_1} \cdot g_2^{a_2})^d \cdot g_1^{x_1} \cdot g_2^{x_2} = (A'^*)^d \cdot B (\bmod n)$.

Thus, the spender withdraws a valid coin $(A'^*, B, s_1^*, s_2, s_3)$ without being injected his/her identity. If he/she doubly spends the coin, the bank can obtain (r_1^*, r_2^*) and $(r_1^{*\prime}, r_2^{*\prime})$ with identical coin $(A'^*, B, s_1^*, s_2, s_3)$ and compute $u^* = \frac{r_1^* - r_1^{*\prime}}{r_2^* - r_2^{*\prime}} = (\frac{a_1}{a_2} \bmod \varphi(n)))(\bmod e_B)$. However, the coin does not contain the identity of spender, and the bank cannot find out who doubly spends the coin with the value u^*.

3.2.2 Forging Identity Jointly

We give a method for constructing one forging identity if two malicious spenders collaborate. The attack succeeds due to the homomorphic property of modular operation.

Step 1. Two malicious spenders C_1 and C_2 execute the opening an account phase. At the end of this phase, they have their identities $ID_{C_1} = g_1^{u_1} (\bmod n)$, and $ID_{C_2} = g_1^{u_2} (\bmod n)$, respectively, and the values $A_1, A_2, O_{C_1,1}, O_{C_2,1}$.

Step 2. The spenders execute the withdrawal phase similar to Baseri et al.'s scheme, except that they work together to compute $A'^* = A_1^{a_1} \cdot A_2^{a_2} (\bmod n)$ instead of A' for some $a_1, a_2 \in_R Z^*_n$, and $s_1^* = O^{a_1}_{C_1,1} \cdot O^{a_2}_{C_2,1} (\bmod n)$ instead of s_1. Here, s_1^* is a valid signature of A'^*, because $(s_1^*)^{e_B} = (O^{a_1}_{C_1,1} \cdot O^{a_2}_{C_2,1})^{e_B} = (O^{e_B}_{C_1,1})^{a_1} \cdot (O^{e_B}_{C_2,1})^{a_2} = A_1^{a_1} \cdot A_2^{a_2} = A'^* (\bmod n)$. Furthermore, they work together to compute $r_1^* = d \cdot (a_1 \cdot u_1 + a_2 \cdot u_2) + x_1$ and $r_2^* = d \cdot (a_1 + a_2) + x_2$ instead of r_1 and r_2 in the payment phase and can pass all verifications since $g_1^{r_1^*} \cdot g_2^{r_2^*} = g_1^{d \cdot (a_1 \cdot u_1 + a_2 \cdot u_2) + x_1} \cdot g_2^{d \cdot (a_1 + a_2) + x_2} = ((g_1^{u_1} \cdot g_2)^{a_1} (g_1^{u_2} \cdot g_2)^{a_2})^d \cdot g_1^{x_1} \cdot g_2^{x_2} = (A'^*)^d \cdot B (\bmod n)$.

Thus, they can withdraw a valid coin $(A'^*, B, s_1^*, s_2, s_3)$ without being injected their actual identities. If they doubly spend the coin, the bank can obtain (r_1^*, r_2^*) and $(r_1^{*\prime}, r_2^{*\prime})$ with identical coin $(A'^*, B, s_1^*, s_2, s_3)$ and compute $u^* = \frac{r_1^* - r_1^{*\prime}}{r_2^* - r_2^{*\prime}} = (\frac{a_1 \cdot u_1 + a_2 \cdot u_2}{a_1 + a_2} \bmod \varphi(n)))(\bmod e_B)$. However, the coin does not contain the identity of spender, and the bank cannot find who doubly spend the coin with the value u^*.

3.3 Attacking Unforgeability

Suppose a malicious spender has a valid coin (A', B, s_1, s_2, s_3). He/she can forge valid coins independently. Furthermore, a malicious merchant can forge a valid coin by cheating the spender or with the help of the spender.

3.3.1 Forging Coins Independently

We first forge a valid coin by a malicious spender independently.

Step 1. This is identical to Step 1 in Subsection 3.2.1.

Step 2. With coin (A', B, s_1, s_2, s_3), the spender picks two random values $a_1, a_2 \in_R Z^*_{e'_B}$ and computes $A'^* = A' \cdot g_1^{a_1} \cdot g_2^{a_2} (\bmod n)$, $B^* = B \cdot g_1^{a_1} \cdot g_2^{a_2} (\bmod n)$, $s_1^* = s_1 \cdot \alpha_1^{a_1} \cdot \alpha_2^{a_2} (\bmod n)$, $s_2^* = s_2 \cdot \beta_1^{a_1} \cdot \beta_2^{a_2} (\bmod n)$, and $s_3^* = s_3 \cdot \gamma_1^{a_1} \cdot \gamma_2^{a_2} (\bmod n)$. In the payment phase, the spender computes $r_1^* = d \cdot (u \cdot s + a_1) + x_1 + a_1$, and $r_2^* = d \cdot (s + a_2) + x_2 + a_2$ because he/she knows the value of (u, s, x_1, x_2).

Obviously, s_1^*, s_2^*, and s_3^* are valid signatures of A'^*, B^*, and $A'^* + B^*$, respectively. We can verify by the following equations: $(s_1^*)^{e_B} = (s_1 \cdot \alpha_1^{a_1} \cdot \alpha_2^{a_2})^{e_B} = (s_1 \cdot (g_1^{1/e_B})^{a_1} \cdot (g_2^{1/e_B})^{a_2})^{e_B} = A' \cdot g_1^{a_1} \cdot g_2^{a_2} = A'^* (\bmod n)$, $(s_2^*)^{e'_B} = (s_2 \cdot \beta_1^{a_1} \cdot \beta_2^{a_2})^{e'_B} = B \cdot g_1^{a_1} \cdot g_2^{a_2} = (s_2 \cdot (g_1^{1/e'_B})^{a_1} \cdot (g_2^{1/e'_B})^{a_2})^{e'_B} = B \cdot g_1^{a_1} \cdot g_2^{a_2} = B^* (\bmod n)$, and $(s_3^*)^{(e_B*t)} = (s_3 \cdot \gamma_1^{a_1} \cdot \gamma_2^{a_2})^{(e_B*t)} = (s_3 \cdot (g_1^{1/(e_B*t)})^{a_1} \cdot (g_2^{1/(e_B*t)})^{a_2})^{(e_B*t)} = (A' + B) \cdot g_1^{a_1} \cdot g_2^{a_2} = (A'^* + B^*) (\bmod n)$. Furthermore, the spender can compute (r_1^*, r_2^*) which satisfies $g_1^{r_1^*} \cdot g_2^{r_2^*} = g_1^{d \cdot (u \cdot s + a_1) + x_1 + a_1} g_2^{d \cdot (s + a_2) + x_2 + a_2} = ((g_1^{u \cdot s} \cdot g_2^s) \cdot (g_1^{a_1} \cdot g_2^{a_2}))^d \cdot ((g_1^{x_1} \cdot g_2^{x_2}) \cdot (g_1^{a_1} \cdot g_2^{a_2})) =$

$(A' \cdot g_1^{a_1} \cdot g_2^{a_2})^d \cdot (B \cdot g_1^{a_1} \cdot g_2^{a_2}) = (A'^*)^d \cdot B^* (\mathrm{mod} n)$. Thus, the coin $(A'^*, B^*, s_1^*, s_2^*, s_3^*)$ is valid and can be spent with any merchant.

3.3.2 Forging Coins by Cheating the Spender

The malicious merchant also can generate a valid coin by cheating the spender. We describe the detailed process as follows.

Step 1. This is same as Step 1 in Subsection 3.2.1.

Step 2. In the payment phase, when the malicious merchant obtains a valid coin (A', B, s_1, s_2, s_3), he/she computes $A'^* = A' \cdot g_1^{a_1} \cdot g_2^{a_2} (\mathrm{mod} n)$, $B^* = B \cdot g_1^{a_1} \cdot g_2^{a_2} (\mathrm{mod} n)$, $d^* = H(A'^*, B^*, ID_M, date||time)$, and sends d^* to the spender. The spender sends $r_1 = d^* \cdot u \cdot s + x_1 (\mathrm{mod} e_B)$, and $r_2 = d^* \cdot s + x_2 (\mathrm{mod} e_B)$ to the merchant. Then, it is claimed by the merchant that the value d^* is not correct or there is a network fault and stops the transaction. Next, the merchant computes $r_1^* = r_1 + d \cdot a_1 + a_1$, and $r_2^* = r_2 + d \cdot a_2 + a_2$. If the spender is honest, the (r_1^*, r_2^*) is valid, because $g_1^{r_1^*} \cdot g_2^{r_2^*} = g_1^{r_1 + d^* \cdot a_1 + a_1} \cdot g_2^{r_2 + d^* \cdot a_2 + a_2} = (g_1^{r_1} \cdot g_2^{r_2}) \cdot (g_1^{a_1} \cdot g_2^{a_2})^{d^*} \cdot g_1^{a_1} \cdot g_2^{a_2} = (A' \cdot g_1^{a_1} \cdot g_2^{a_2})^{d^*} (B \cdot g_1^{a_1} \cdot g_2^{a_2}) = (A'^*)^{d^*} \cdot B^* (\mathrm{mod} n)$. Then, he/she computes $s_1^* = s_1 \cdot \alpha_1^{a_1} \cdot \alpha_2^{a_2} (\mathrm{mod} n)$, $s_2^* = s_2 \cdot \beta_1^{a_1} \cdot \beta_2^{a_2} (\mathrm{mod} n)$, and $s_3^* = s_3 \cdot \gamma_1^{a_1} \cdot \gamma_2^{a_2} (\mathrm{mod} n)$.

According to Step 2 in Subsection 3.3.1, s_1^*, s_2^*, and s_3^* are valid signatures of A'^*, B^*, and $A'^* + B^*$ respectively. Thus, the coin $(A'^*, B^*, s_1^*, s_2^*, s_3^*)$ with (r_1^*, r_2^*) is valid for the merchant and can be later deposited in the bank. Furthermore, the spender cannot detect the cheating, because that he/she can spend his/her coin (A', B, s_1, s_2, s_3) normally.

3.3.3 Forging Coins with the Help of the Spender

If there is a malicious merchant colluding with a malicious spender, they also can forge a valid coin by following the same process described in Subsection 3.3.2.

4 Conclusions

We review Baseri et al.'s [1] off-line electronic cash scheme and find three flaws. First, the scheme cannot satisfy verifiability. Second, malicious spenders can withdraw coins independently or jointly without being injected his/her identity, therefore, the bank cannot detect the coins owner when double spending occurs. This violates the property of unreuseability. Finally, a malicious spender can forge valid coins using existing coins, and a malicious merchant can forge valid coins by cheating or colluding with a spender. This violates the property of unforgeability. From above, the Baseri et al.'s scheme is not secure. Presently, we are investigating how Baseri et al.'s method can be greatly modified to defend all possible kinds of attacks.

References

[1] Y. Baseri, B. Takhtaei, and J. Mohajeri, "Secure untraceable off-line electronic cash system," *Scientia Iranica*, vol. 20, no. 3, pp. 637–646, June 2013.

[2] D. Chaum, "Blind signatures for untraceable payments," in *Proceedings of Advances in Cryptology (CRYPTO'82)*, pp. 199–203, Santa Barbara, California, USA, 1982.

[3] D. Chaum, A. Fiat, and M. Naor, "Untraceable electronic cash," in *Proceedings of Advances in Cryptology (CRYPTO'88)*, pp. 319–327, Santa Barbara, California, USA, 1988.

[4] X. F. Chen, F. G. Zhang, and S. L. Liu, "ID-based restrictive partially blind signatures and applications," *Journal of Systems and Software*, vol. 80, no. 2, pp. 164–171, Feb. 2007.

[5] Y. Chen, and J. S. Chou, "On the Privacy of 'User efficient recoverable off-line e-cash scheme with fast anonymity revoking'," *International Journal of Network Security*, vol. 17, no. 6, pp. 708–711, Nov. 2015.

[6] Z. Eslami, and M. Talebi, "A new untraceable off-line electronic cash system," *Electronic Commerce Research and Applications*, vol. 10, no. 1, pp. 59–66, Feb. 2011.

[7] C. I. Fan, S. V. Huang, and Y. C. Yu, "User efficient recoverable off-line e-cash scheme with fast anonymity revoking," *Mathematical and Computer Modelling*, vol. 58, no. 1-2, pp. 227–237, Jul. 2013.

[8] C. I. Fan, B. W. Lin, and S. M. Huang, "Customer efficient electronic cash protocol," *Journal of Organizational Computing and Electronic Commerce*, vol. 17, no. 3, pp. 259–281, Dec. 2007.

[9] X. M. Hu, and S. T. Huang, "Analysis of id-based restrictive partially blind signatures and applications," *Journal of Systems and Software*, vol. 81, no. 11, pp. 1951–1954, Nov. 2008.

[10] W. S. Juang, and H. T. Liaw, "A practical anonymous multi-authority e-cash scheme," *Applied Mathematics and Computation*, vol. 147, no. 3, pp. 699–711, Jan. 2004.

[11] W. Mao, *Modern Cryptography: Theory and Practice*, Ch. 6, pp. 186, Publishing House of Electronics Industry, 2004.

[12] Z. W. Tan, "An off-line electronic cash scheme based on proxy blind signature," *The Computer Journal*, vol. 54, no. 4, pp. 505–512, Apr. 2011.

[13] H. Wang, J. L. Cao, and Y. C. Zhang, "A flexible payment scheme and its role-based access control," *IEEE Transactions on Knowledge and Data Engineering*, vol. 17, no. 3, pp. 425–436, Mar. 2005.

Feng Wang was born in Shandong province, China, in 1978. He received his B.S. degree in Mathematics from Yantai Normal University (now named Ludong University), Yantai, in 2000 and the M.S. degree in Applied Mathematics from the Guangzhou University, Guangzhou, in 2006. Currently, he is a Lecturer in the College of Mathematics and Physics at Fujian University of Technology and a visiting scholar in Department of Information Engineering and Computer Science at Feng Chia University. His research interests include computer cryptography and information security.

Chin-Chen Chang received his Ph.D. degree in computer engineering from National Chiao Tung University. His first degree is Bachelor of Science in Applied Mathematics and master degree is Master of Science in computer and decision sciences. Both were awarded in National Tsing Hua University. Dr. Chang served in National Chung Cheng University from 1989 to 2005. His current title is Chair Professor in Department of Information Engineering and Computer Science, Feng Chia University, from Feb. 2005. Prior to joining Feng Chia University, Professor Chang was an associate professor in Chiao Tung University, professor in National Chung Hsing University, chair professor in National Chung Cheng University. He had also been Visiting Researcher and Visiting Scientist to Tokyo University and Kyoto University, Japan. During his service in Chung Cheng, Professor Chang served as Chairman of the Institute of Computer Science and Information Engineering, Dean of College of Engineering, Provost and then Acting President of Chung Cheng University and Director of Advisory Office in Ministry of Education, Taiwan. Professor Chang has won many research awards and honorary positions by and in prestigious organizations both nationally and internationally. He is currently a Fellow of IEEE and a Fellow of IEE, UK. On numerous occasions, he was invited to serve as Visiting Professor, Chair Professor, Honorary Professor, Honorary Director, Honorary Chairman, Distinguished Alumnus, Distinguished Researcher, Research Fellow by universities and research institutes. His current research interests include database design, computer cryptography, image compression and data structures.

Changlu Lin received the BS degree and MS degree in mathematics from the Fujian Normal University, P.R. China, in 2002 and in 2005, respectively, and received the Ph.D degree in information security from the state key laboratory of information security, Graduate University of Chinese Academy of Sciences, P.R. China, in 2010. He works currently for the College of Mathematics and Computer Science, and the Fujian Provincial Key Laboratory of Network Security and Cryptology, Fujian Normal University. He is interested in cryptography and network security, and has conducted research in diverse areas, including secret sharing, public key cryptography and their applications.