A Novel E-cash Payment Protocol Using Trapdoor Hash Function on Smart Mobile Devices

Jian-Sen Wang[†], Fuw-Yi Yang[††], and Incheon Paik[†]

[†]Department of Computer and Information Systems,
The University of Aizu, Japan
[††]Department of Computer Science and Information Engineering,
Chaoyang University of Technology, Taiwan, R.O.C.

Summary
Electronic commerce is becoming more important along with fast progress of the Internet and information technology, many customers use e-cash instead of real cash in wide. An e-cash is issued by a bank with a fixed face-value. The face-value of the e-cash is usually not fitted with actual amount, and it is inconvenient for customers to use. Moreover, it consumes huge amounts of computing resources and difficult to engage in electronic commerce with mobile devices for limited computing capability. This paper proposes a novel payment system with smart mobile devices, wherein customers are not limited to purchase e-cash with the fixed face-value. The amount of every transaction is deducted directly from the customer's account, eliminating the inconvenience of fixed face-value of the e-cash, and reducing online computation cost of a bank. Using a technique of trapdoor hash function to mitigate the computational cost, our system can be used with the mobile devices effectively.

Key words:
Electronic commerce, E-cash, Trapdoor hash function, Smart mobile devices.

1. Introduction

Owing to the rapid progress of the Internet and information technology, the electronic commerce has been used widely. Currently, researchers focus on the e-payment system such that electronic cash [1-6], electronic check [7, 8], electronic traveler's check [9, 10] and so on. The advantages of e-payment system are the customer doesn't bring any real money to consume and achieves the property of transference and convenience. D. Chaum [1] proposed an e-cash system based on blind signatures. In the system, the customer had to purchase the e-cash via the Internet or directly from the bank in person, and banks issued the e-cash only in fixed face-value. Due to these restrictions and shortcomings, the customer cannot use e-cash flexibly. Moreover, many researchers proposed the e-cash payment protocol [1, 2, 4, 5, 6], using plenty of computational resources such that exponential operation. It causes the big burden for the system. Chang and Lai [5] proposed a flexible date-attachment scheme on e-cash and Juang [6] proposed the D-cash. In withdraw phase of previous scheme, if the customer wants to apply for e-cash, it will attach date slip on the e-cash. The date slip adapts to request the bank signing the signature on the date of payment. Because of attaching the date slip, it needs signing the signature with the bank to prevent the conspiracy between customers and merchants. This system consumes huge computational cost for using the digital signature twice.

To eliminate the shortcomings above, we proposed a novel e-cash payment protocol using a trapdoor hash function on smart mobile devices. This system would temporarily freeze and deduct the amount of the transaction from customers' accounts. The customer would be unable to exceed the balance of his/her account, thus avoiding the possibility of overspending. To integrate trapdoor hash function into the system, the computational cost on the customer side would require only the integer multiplication and addition; hence, it would reduce computational costs and make the system more suitable for application with mobile devices.

On the other hand, many applications of e-cash is used on smart card, the customers need to bring different smart cards issued by the banks to consume, which is very inconvenient. Moreover, many researchers have proposed systems that do not explain how to transfer the e-cash to the merchant over the transaction. For the current QR-code applications, it usually needs to print the visual graph on the paper. For instances: boarding card, high speed railway and so on. It causes the consumption and pollution. This system packages the payment information from the bank with QR-Code (Quick Response) code and transfers the visual graph to the merchant. The merchant will recognize it by QR-code recognizer and sends it to the bank for verifying the validity of the user and the double-spending problem. In the whole transaction processes, it doesn't need to consume any paper and reduce the cost to achieve the paperless transaction.

The paper is organized as follows. Briefly introduce the concept of the trapdoor hash function in Section 2. In

Manuscript received June 5, 2011
Manuscript revised June 20, 2011

Section 3, a brief description of Chang and Lai's e-cash protocol is shown. In Section 4, we propose a novel payment system. The security analysis is given in Section 5. Section 6 then provides the discussion of the protocol. Finally, the conclusion is given in Section 7.

2. Preliminary on Trapdoor hash function

The hash functions are applied to the technique of digital signature commonly, the digital signature can be divided into three phases: signing key generation, signing, and signature verification. In general, using hash functions to extract the abstract of the document to be signed and signing the digital signature on the abstract.

Collision-resistance is one of the important properties of traditional hash functions. For Chameleon functions [11] or trapdoor hash functions [12-15], the property of collision-resistance is optional; the owner of a trapdoor key can easily find other collided pre-images and produce the same trapdoor hash value. For instance, assuming $TH(\cdot)$ represents a trapdoor hash function and a trapdoor hash value $v = TH(h_1)$, the owner of the trapdoor key can compute h_2; hence, $v = TH(h_2) = TH(h_1)$ when knowing h_1. The online Computational cost of Chameleon function (collided pre-images) requires a multiplication and modulo operation. In the literature [12], only one modulo operation was required for the computation. In the literature [14, 15], the computational cost were reduced to one integer multiplication and addition, it is suitable for the mobile devices with limited computational resources. The techniques mentioned in literature [14, 15] are as follows:

Let p, q, t, P and Q be prime numbers, the compound number n is the product of P and Q; that is, $n = P \cdot Q$, $P = 2 \cdot p \cdot t + 1$, $Q = 2 \cdot q + 1$. $|P|$, $|Q|$, and $|p|$ represent the encoded bit length of P, Q, and p. $|P| = |Q| = 512$, $|p| = l = 160$. The order of $g \in Z_n^*$ is p. Randomly selecting $x \in_R \{0, 1\}^l$; Computing $y = g^x \bmod n$. The trapdoor key is $TK = x$, and the public key is $HK = (g, n, y)$.

If the message $m_1 \in_R \{0, 1\}^l$, the hash operation is to obtain the hash value of the message m_1. The processes are as follows.

1. Randomly Selects: $r_1 \in_R \{0, 1\}^{2 \cdot l + k}$,
2. Computes the trapdoor hash value: $v = TH_{HK}(m_1, r_1)$, i.e., $v = g^{r_1} \cdot y^{m_1} \bmod n$.

After determining r_1, then the hash value of message m_1 is $v = TH_{HK}(m_1, r_1)$. The owner of a trapdoor key can use the trapdoor operation to obtain m_2 and r_2 such that $v = TH_{HK}(m_1, r_1) = TH_{HK}(m_2, r_2)$. The detailed processes of the trapdoor operation are shown as below.

1. Determines the message: $m_2 \in \{0, 1\}^l$,
2. Computes r_2, i.e., $TH_{TK}(m_2) = r_2 = r_1 + (m_1 - m_2) \cdot x$.

Although integer arithmetic is used for computing r_2, the confidential information $(m_1 - m_2) \cdot x$ is still properly hidden behind the random number r_1 because $|r_1| = 2 \cdot l + k$, $k = 80$. Similar research of information hiding techniques can also be seen in the literature [16-18].

3. Review of Chang and Lai's scheme

In this section, we review of the Chang-Lai's scheme [5]. The date-attachment e-cash scheme is for customer to attach the effective date into e-cash. Their system is divided into five phases: initializing phase, withdrawing phase, unblinding phase, date-attaching phase and depositing phase. The brief description is shown as following.

Initializing phase: There are three participants in the protocol: the customer, the merchant and the bank. Firstly, the back generates two pairs of RSA keys. That is, the public keys are (n, e) and (n', e'), and the private keys are (d, p, q) and (d', p', q').

Withdrawing phase: The customer randomly chooses the blinding factor $r_1 \in_R Z_n^*$ and computes $\alpha = r_1^e \cdot H(m) \bmod n$ then sends α to the bank. The bank signs the message α then transmits the signature $t_1 = \alpha^d \bmod n$ back to the customer and deducts w dollars from the customer's account.

Unblinding phase: After receiving t_1, the customer computes $s = r_1^{-1} \cdot t_1 \bmod n$ to obtain the e-cash (w, s). The customer chooses the random blinding factor $r_2 \in_R Z_{n'}^*$ and sends the blind message $\beta = r_2^{e'} \cdot G(s) \bmod n'$ to the bank. After receiving β, then uses the private key to generate the blind signature $t_2 = \beta^{d'} \bmod n'$ and sends back to the customer. The customer computes $\delta = r_2^{-1} \cdot t_2 \bmod n'$, where δ is the date slip.

Date-attaching phase: When the customer decides to purchase, the customer sends (δ, s) and (a, b, c), where is (a, b, c) is the effective date. The bank checks the validation of the date slip with $\delta^{e'} \stackrel{?}{=} G(s) \bmod n'$. If it holds, the bank sends $s' = (G(s \| a \| b \| c))^{d'} \bmod n'$ to the customer.

Depositing phase: If the customer wants to purchase the merchandises, the customer sends the e-cash $(s', m, s, (a, b, c))$ to the merchant for the w dollars. The merchant verifies the validity of the e-cash by $s^e \stackrel{?}{=} H(m) \bmod n$ and $s'^{e'} \stackrel{?}{=} G(s \| a \| b \| c) \bmod n'$. If it holds, the merchant sends the e-cash $(s', m, s, (a, b, c))$ to the bank for checking double-spending problem. If the e-cash is fresh, the bank deposits the amount into the merchant's

account and stores the transaction information in its database.
In their scheme, it adopts the date-attachment on e-cash can be used more flexible, unfortunately, owing to the date-attachment, the bank needs to signing the signature on the date slip. It increases the cost of computation and communication.

4. The proposed scheme

In order to solve the above flaws, this paper introduces the mobile pre-paid system based on trapdoor hash functions. If the customer wants to purchase merchandises through the Internet, after receiving the purchase order, the merchant would sign up the order and send the valid transaction information back to the customer. When payment is required, the customer would use an electronic payment certificate issued by the bank to request payment from the bank. After receiving the certificate from the customer, the bank would confirm whether the customer's balance is enough, electronic payment certificate and order information were valid. If all checked out, the bank would temporarily freeze the amount of the transaction in the customer's account and send the transaction information to the merchant, verifying that the customer can afford it. The merchant would accept the order and deliver the merchandises to the customer. After receiving confirmation that the customer has received the goods already, the bank would deduct the total amount of the purchase from the customer's account and deposit it into the merchant's account.
The mobile pre-paid system consists of three entities: customer, merchant, and bank. This system will be illustrated in Fig. 1.

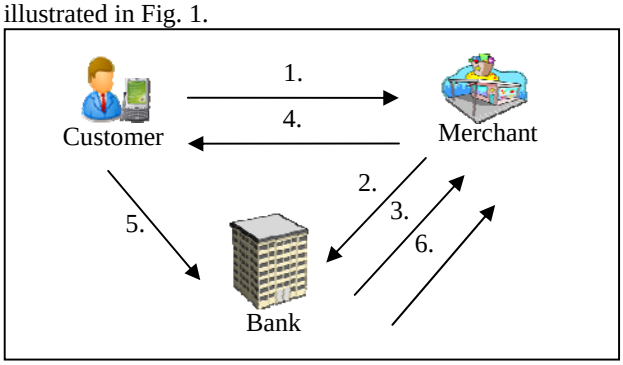


Fig. 1: The mobile pre-paid system

The procedures are as follows:

1. Customer accepts the price. Using the e-cash issued by the bank and packages it as the QR code visual graph and places the order.
2. The merchant verifies the graph with QR code recognizer and delivers it to the bank for verifying.
3. The bank freezes the purchase amount of the customer's account and notifies the merchant to deliver the merchandises to the customer.
4. Merchant delivers the merchandises to the customer.
5. Customer confirms with the bank that they have received the goods already; the bank may proceed with the payment.
6. The bank deposits the payment into the merchant's account.

4.1 Parameters and Notations

The parameters and notations for mobile pre-paid system can be divided into four sections: the public system's parameters, the bank's parameters, the customer's parameters, and the merchant's parameters. There would be described as follows:

The Public System's Parameters

$H(\cdot)$: The collision-resistant one-way hash function and defines $H(\cdot):\{0, 1\}^* \rightarrow Z_n^*$.
k, l: The confidential parameters classified in accordance with level of security. For instance, $k = 80$, $l = 160$.

The Bank's Parameters:

ID_B: The identity of the bank.
P, Q, p, q, t: All large prime numbers and P, Q have the same encoded bit length. $P = 2 \cdot p \cdot t + 1$, $Q = 2 \cdot q + 1$.
n: The product of two large prime numbers, for instance, $n = P \cdot Q$.
g: $g \in_R Z_n^*$ is with the order p, the encoded bit length of p is l, for instance, $|p| = l$.
d, e: The bank's signing key and public verification key, $e \in_R Z_n$ and $d \cdot e = 1 \bmod (P - 1) \cdot (Q - 1)$.

The Customer's Parameters:

ID_i: The identity of the customer i.
x_1: The trapdoor key selected by the customer, for instance, $TK = x_1$, and $x_1 \in_R \{0, 1\}^l$.
HK: The public key corresponding to the trapdoor key, for instance, $HK = (g, n, y_1 = g^{x_1} \bmod n)$
m_1, r_1: The random message and the random number selected by the customer, for instance, $m_1 \in_R \{0, 1\}^l$, $r_1 \in_R \{0, 1\}^{2 \cdot l + k}$.

The Merchant's Parameters:

ID_M: The identity of the merchant.
MP: The price of the purchased goods.
MN: The name of the purchased goods.
TI: The transaction information is included time, date and serial number of the purchase order.
$Sig_M(\cdot)$, $Ver_M(\cdot)$: The merchant uses signing key to sign the message, and verifies the signature with the merchant's public verification key.

$Enc_M(\cdot)$, $Dec_M(\cdot)$: The encryption and decryption function of the merchant.

4.2 The procedures of mobile pre-paid system

The mobile pre-paid system can be divided into three phases: the registration phase, the payment phase, and the deposit phase. Fig. 2, 3, 4 would show the three phases, respectively.

(1) The registration phase

In this phase, if the customer wants to apply for the e-cash from the bank, firstly, he/she needs to open the bank account with the smart mobile devices via the Internet, then selects a random number $x_1 \in_R \{0, 1\}^l$,and calculate $y_1 = g^{x_1} \bmod n$. The public key for the trapdoor hash function is $HK = (g, n, y_1)$ and the trapdoor key is x_1. In order to apply for an electronic payment certificate σ, the customer calculates the trapdoor hash value and sends his/her identity ID_i and the hash value A to the bank. The process is shown as follows.

1. Randomly generates message $m_1 \in_R \{0, 1\}^l$ and number $r_1 \in_R \{0, 1\}^{2 \cdot l + k}$.
2. Computes the trapdoor hash value $A = TH_{HK}(m_1, r_1, y_1) = g^{r_1} \cdot y_1^{m_1} \bmod n$.
3. Transmits identity message ID_i, and the trapdoor hash value A to the bank.

After receiving the registration message, the bank uses its signing key d to sign the message and generate an electronic payment certificate σ. The bank will send the e-payment certificate σ to the customer. The process is shown as follows.

1. Computes $\sigma = H(ID_i, A)^d \bmod n$.
2. Transmits the electronic payment certificate σ to the customer.

After receiving the certificate σ, the customer verifies if σ is a valid signature on the registration message signed by the bank, then check $\sigma^e \stackrel{?}{=} H(ID_i, A) \bmod n$. If it holds, the customer stores the random pair (m_1, r_1, x_1) and the e-payment certificate σ into the smart mobile devices. Then compute another pair (m_2, r_2) in the payment phase such that both pairs generate the same trapdoor hash value. Fig. 2 shows the process of the registration phase as below.

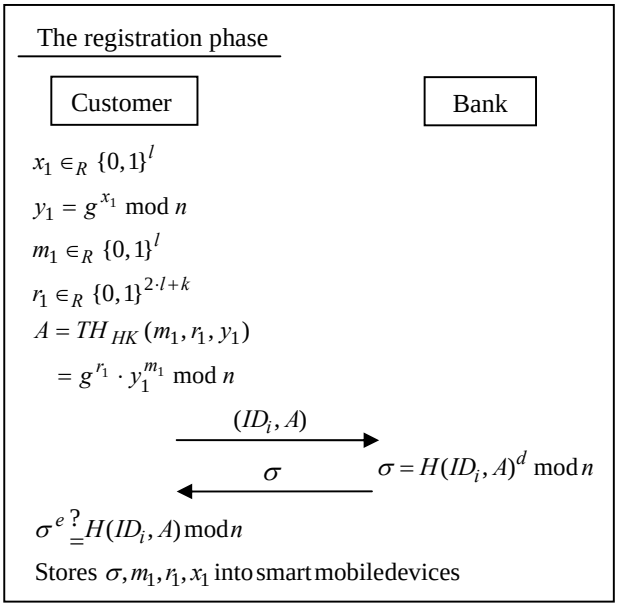


Fig. 2: The registration phase

(2) The payment phase

1. When the customer agrees the price of the merchandises and would like to purchase, the customer will determine the document $m_2 = H(ID_i, ID_M, ID_B, MN, MP)$. The contents of the document include: the identity of the customer, merchant and bank, name and price of the purchased goods.
2. The customer retrieves the stored information (m_1, r_1, x_1) from the smart mobile devices.
3. Firstly, selects a new trapdoor key $x_2 \in_R \{0, 1\}^l$ and compute $y_2 = g^{x_2} \bmod n$. Then execute the trapdoor operation $TH_{TK}(m_2) = r_2 = r_1 + x_1 \cdot m_1 - x_2 \cdot m_2$. For $r_1 + x_1 \cdot m_1 = r_2 + x_2 \cdot m_2$, it implies that $A = g^{r1} \cdot y_1^{m_1} = g^{r2} \cdot y_2^{m_2} \bmod n$.
4. Let $\alpha = Enc_M(\sigma, A, ID_i, ID_M, ID_B, MN, MP, r_2, y_2)$ forms the payment information. The customer utilizes QR Code generator to package the payment information α as the QR Code visualization graph and transmits it to the merchant.

After receiving the encrypted order α, the merchant uses code recognizer to verify the graph and decrypts the order α with its decryption key to obtain the order information. That is, $Dec_M(\alpha) = (\sigma, A, ID_i, ID_M, ID_B, MN, MP, r_2, y_2)$. The merchant adds the transaction information TI to the order and forms a quotation then signs with his signing key, that is, $s = Sig_M(\sigma, A, ID_i, ID_M, ID_B, MN, MP, r_2, y_2, TI)$. Then the merchant sends the signature s to the bank for verifying. Fig. 3 shows the process of the payment phase as below.

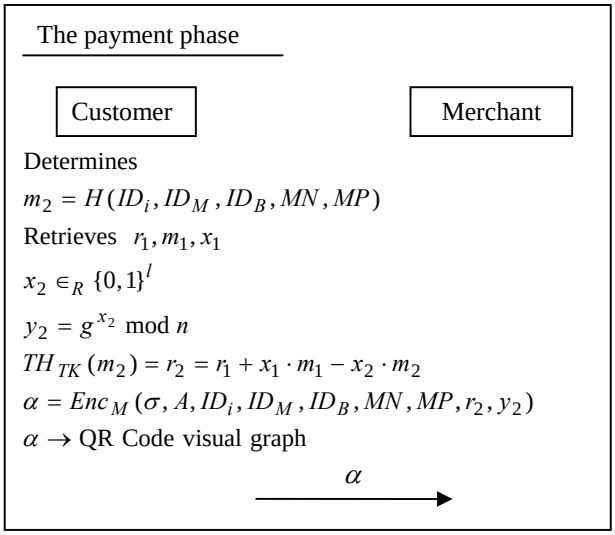


Fig. 3: The payment phase

(3) The deposit phase

After receiving the payment information, the bank uses the merchant's public key to decrypt the message and obtain the payment information (σ, A, ID_i, ID_M, ID_B, MN, MP, r_2, y_2, TI). The bank then verifies the validity of the quotation signed by the merchant; that is, check $Ver_M(s) \stackrel{?}{=}$ "True" and the customer's account balance for the purchase. The bank also verifies the legality of the customer's e-payment certificate, double-spending problem and order information. The verification process is shown as follows:

1. Verifies the merchant's quotation $Ver_M(s) \stackrel{?}{=}$ "True".
2. Combines the document $m_2 = H(ID_i, ID_M, ID_B, MN, MP)$.
3. Computes the trapdoor hash value $A = TH_{HK}(m_2, r_2, y_2) = g^{r_2} \cdot y_2^{m_2} \bmod n$.
4. Verifies the validity of e-payment certificate; that is $\sigma^e \stackrel{?}{=} H(ID_i, A) \bmod n$.

After verifying the above information, the bank accepts the payment information and temporarily freezes the *MP* amount in the customer's account. The bank transmits the verified transaction information to the merchant; the merchant accepts this order and delivers the merchandises to the customer. The customer transmits the transaction completion message to the bank when he/she received the merchandises. After receiving the completion message, the bank stores the transaction details (σ, ID_i, ID_M, ID_B, MN, MP, r_2, y_2, s, TI) in its database and deposits the amount *MP* to the merchant's account. Fig. 4 shows the process of the deposit phase.

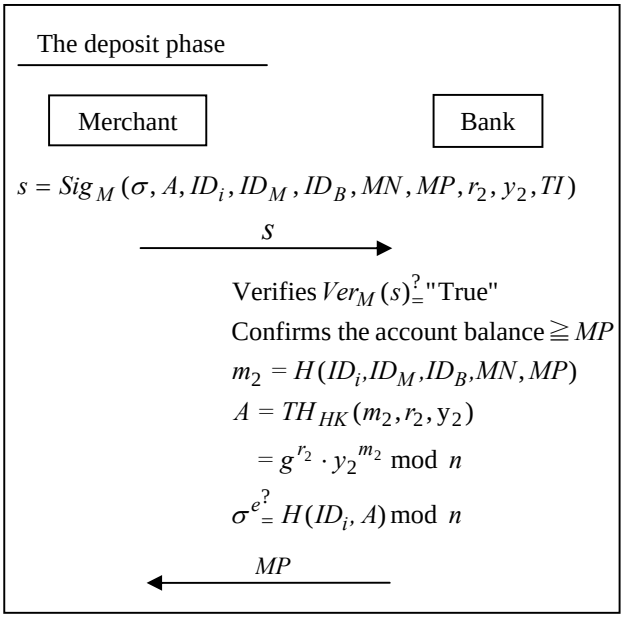


Fig. 4: The deposit phase

5. Security analysis

The mobile pre-paid system can achieve the following security requirements.

(1) Unforgeability

Every legal quotation has to be signed by the merchant with its signing key. If an attacker wants to forge the information of quotation, he/she need to obtain the merchant's signing key, but it is difficult to get. Moreover, only the legal customer possesses the trapdoor key. Without this trapdoor key, it would be difficult for an attacker to compute (m_2, r_2, y_2) such that the trapdoor hash value $A = TH_{HK}(m_2, r_2, y_2) = g^{r_2} \cdot y_2^{m_2} \bmod n$.

(2) Undeniability

For every transaction made between the customer and the merchant, the quotation has to be signed by the merchant's signing key. After the signature is completed, the merchant will transfer the quotation to the bank, the bank uses the public key published by the merchant to verify the quotation. The merchant cannot deny signing the information on the quotation. On the other hand, in the payment phase, the customer uses his/her own trapdoor key to compute $TH_{TK}(m_2)$ and obtain (m_2, r_2, y_2) and generate the trapdoor hash value A. Only the customer who knows the trapdoor key could perform the computation of $TH_{TK}(m_2)$. Therefore, the customer cannot deny the transaction information.

(3) Accuracy

The quotation generated by the merchant. Therefore, it can be provided to any party for verifying. If any party wants to check the accuracy of the quotation, he/she can use the public key published by the merchant to verify the signature *s* and the transaction information *TI*.

(4) Prevention of Double-Spending

The payment information (σ, A, ID_i, ID_M, ID_B, MN, MP, r_2, y_2, TI) is generated after the transaction between the customer and the merchant. The bank stores the payment information to its database. When the bank receives a new payment request from the customer, the bank will check its database to ensure that no other payment information exists in the database. If the same payment information is found, the request for the new payment will be rejected. It can prevent any possibility of double-spending problem from the customer's account efficiently.

6. Discussion

In this section, we would discuss the performance and the advantages of the mobile pre-paid system. The comparison table is shown as Table 1.

More flexible for E-cash

For the previous researcher's e-cash systems [1-6], the customer had to apply for the fixed amount of e-cash from the bank for payment. When the transaction amount would be more than the applied amount, it needs to apply for it again. Hence, in this proposed system, every transaction is paid directly to the merchant through the bank. The bank deposits the amount directly from the customer's account, so the customer's amount consumption is based on his/her account balance. Therefore, there are no problems of idle balance and cannot afford it.

Reducing the online computation and more efficiently

Chang and Lai [5] proposed the protocol that requires a huge number of modulo and hash operations to complete the purchase and deposit phases. The details are shown as Table 1. Comparing with both protocols, the computational cost doesn't greater than Chang and Lai's protocol too much. Nevertheless, customers must to apply for new e-cash when he/she used it. It needs more waiting time and computational cost, which is inconvenient for the customer. There are no above problems in our protocol, because the deduction is based on customer's account. Applications of such protocols with mobile devices have a number of shortcomings including limited battery power and low computational capabilities. The protocols developed in this paper including the trapdoor hash functions to reduce the limitations of online computation cost by using only integer multiplication, making it effective for applications with mobile devices.

Table 1: Computational cost evaluation

Computational cost		Withdraw	Payment	Deposit
Chang and Lai's scheme	Customer	$1T_{exp}$ $+2T_{mul}$ $+1T_{div}$ $+1T_H$	$1T_{exp}$ $+2T_{mul}$ $+1T_{div}$ $+1T_H$	0
	Merchant	0	$2T_{exp}+2T_H$	0
	Bank	$1T_{exp}$	$3T_{exp}+2T_H$	$2T_{exp}$ $+2T_H$
Our protocol	Customer	$4T_{exp}$ $+1T_{mul}$	$1T_{exp}$ $+2T_{imul}$ $+2T_{add}$ $+1T_{asym}$ $+1T_H$	0
	Merchant	0	$1T_{exp}$ $+1T_{asym}$	0
	Bank	$1T_{exp}$ $+1T_H$	0	$4T_{exp}$ $+1T_{mul}$ $+1T_H$

T_{exp}: The exponential operation.
T_{asym}: The asymmetric encryption/decryption operation.
T_{mul}: The modulo multiplication operation.
T_{imul}: The integer multiplication operation.
T_{div}: The modulo division operation.
T_{add}: The integer addition operation.
T_H: The one-way hash function.

7. Conclusions

In this paper, we proposed a novel e-cash payment protocol using trapdoor hash function on smart mobile devices. Customers can easily use the smart mobile

devices to apply for the e-cash and purchase the merchandises conveniently. The customer doesn't withdraw e-cash from the bank anymore. This system deducts the amount of the transaction from the customers' accounts directly and resolves the idle remainder of the previous e-cash systems. By using trapdoor hash function, this protocol can reduce the computational cost and achieve the security requirements of e-commerce systems. Our protocol have more computational cost than Chang and Lai's scheme in the withdraw phase. However, if the e-cash doesn't be fitted with the price of the merchandises that needs to apply for the e-cash again, it needs more computational cost. In the future, we'll figure out the more efficient protocol not only for the customer but also for the bank.

References

[1] D. Chaum, "Blind signature for untraceable payments", In: *Proceedings of advances in Cryptology*, Springer-Verlag, New York, pp.199-203, 1983.

[2] 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, 2004.

[3] Y. Y. Chen, J. K. Jan, and C. L. Chen, "A novel proxy deposit protocol for e-cash systems", *Applied Mathematics and Computation*, Vol. 163, No. 2, pp. 869-877, 2005.

[4] C. L. Chen and M. H. Liu, "A traceable E-cash transfer system against blackmail via subliminal channel", *Electronic Commerce Research and Applications*, Vol. 8, No. 6, pp. 327-333, 2009.

[5] C. C. Chang and Y. P. Lai, "A flexible Date-attachment Scheme on E-cash", *Computers & Security,* Vol. 22, No. 2, pp.160-166, 2003.

[6] W. S. Juang, "D-cash: A flexible pre-paid e-cash scheme for date-attachment", *Electronic Commerce Research and Applications*, Vol. 6, No. 1, pp. 74-80, 2007.

[7] C. C. Chang, S. C. Chang, and J. S. Lee, "An on-line electronic check system with mutual authentication", *Computers & Electrical Engineering*, Vol. 35, No. 5, pp. 757-763, 2009.

[8] W. K. Chen, "Efficient on-line electronic checks", *Applied Mathematics and Computation*, Vol. 162, No. 3, pp. 1259-1263, 2005.

[9] J. E. Hsien, C. C. Hsueh, and C. Y. Chen, "An electronic traveler's check system", *Conference on Theory and Practice for Electronic Commerce*, pp. 164–169, 2001.

[10] H. T. Liaw, J. F. Lin, and W. C. Wu, "A new electronic traveler's check scheme based on one-way hash function", *Electronic Commerce Research and Applications*, Vol. 6, No. 4, pp. 499-508, 2007.

[11] H. Krawczyk and T. Rabin, "Chameleon signatures", *Symposium on Network and Distributed Systems Security* (NDSS'00), pp.143-154, 2000.

[12] A. Shamir and Y. Tauman, "Improved online / offline signature schemes", *Advances in Cryptology-CRYPTO'01*, LNCS 2139, pp. 355-367, 2001.

[13] F. Y. Yang, S. H. Chiu, and C. M. Liao, "Trapdoor Hash Functions with Efficient Online Computations", *The Proceedings of Multimedia and Networking Systems Conference 2006* (MNSC 2006)**,** 2006.

[14] F. Y. Yang, "Efficient trapdoor hash function for digital signatures", *Chaoyang Journal*, Vol. 12, pp. 351- 357, 2007.

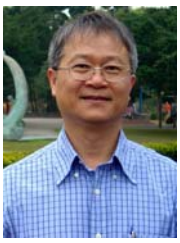
[15] F. Y. Yang, "Improvement on a trapdoor hash function", *International Journal of Network Security*, Vol. 9, No. 1, July, pp. 17-21, 2009.

[16] T. Okamoto, M. Tada, and A. Miyaji, "Efficient 'on the fly' signature schemes based on integer factoring", *Proceedings of the 2nd International Conference on Cryptology in India, INDOCRYPT'01*, LNCS 2247, pp. 275-286, 2001.

[17] G. Poupard and J. Stern, "On the fly signatures based on factoring", *Proceedings of the 6th ACM Conference on computer and communications security* (CCS), pp. 48-57, 1999.

[18] D. Pointcheval, "The composite discrete logarithm and secure authentication", *Public-Key Cryptography* 2000 (PKC), LNCS 1751, pp. 113-128, 2000.

Jian-Sen Wang received the B.S. degree in Computer Science and Information Engineering from Chaoyang University of Technology in 2009. Now he is a student of Dual Degree Program between the University of Aizu, Japan and Chaoyang University of Technology, Taiwan. Research interests include Information Security, E-Business System, Security of E-Business, and E-Commerce.



Fuw-Yi Yang received the B.Sc. degree and M.Sc. degree in the electronic engineering from National Taiwan University of Science and Technology, Taiwan, and the Ph.D. degree in the Department of Applied Mathematics, National Chung Hsing University, Taiwan. He is currently an associate professor with the Department of Computer Science and Information Engineering in Chaoyang University of Technology. He is a member of the Chinese Cryptology and Information Security Association (CCISA). His research interests include computer cryptography, network security, and information security.

Incheon Paik received the M.E. and Ph.D. degrees in Electronics Engineering from Korea University in 1987 and 1992, respectively. During 1993-2000, he worked as an associate professor in Soonchunhyang university, Korea. From 1996 to 1998, he was a visiting researcher of State Key Laboratory, Beihang University, Beijing, China. Also he leaded

the Electronic Commerce S/W Research Center (now, ITRC) funded by the Ministry of Information & Communication. Now he is an associate professor in the University of Aizu, Japan. He has organized several international conferences and served as a referee for several international journals such as IEICE, JAAMAS, IEEE TSC, etc. Research interests include Semantic Web, Web Services, Web Service Composition, Web Data Mining, Business Process Management and Integration, e-Business System, Software Component, Security for e-Business, and Agents on Semantic Web. He is a member of ACM, IEEE, and IPSJ.