An Electronic Cash Scheme with Crime-Resistance

Chun-Ta Li[1], Yan-Chi Lai[2], Min-Shiang Hwang[3] and Yen-Ping Chu[4]
[1]*Department of Computer Science, National Chung Hsing University*
phd9307@cs.nchu.edu.tw
[2]*Department of Information Management, Chaoyang University of Technology*
[3]*Department of Management Information Systems, National Chung Hsing University*
mshwang@nchu.edu.tw
[4]*Department of Computer Science and Information Engineering, Tunghai University*
ypchu@thu.edu.tw

ABSTRACT

One of the potential problems with electronic cash schemes is the perfect crimes. No one can trace back to a particular owner after the message-signature pair is published. Therefore, a malicious adversary may make use of this unconditional anonymity to blackmail or kidnap someone to get clean and untraceable money. In this paper, we propose an electronic cash scheme satisfying all the security requirements of electronic cash systems. Also, we believe that our e-cash scheme can resist the perfect crimes.

Keywords: blind signature, electronic cash, Internet, perfect crime, privacy

1: Introduction

In 1983, D. Chaum proposed the first electronic cash (e-cash) system [2, 3]. In Chaum's e-cash system, there are three participants, namely, the consumer, bank, and merchant. Further, there are three phases in Chaum's e-cash protocol, including withdrawal, payment and deposit phase. First, the consumer withdraws the e-cash from the bank, and the bank checks the bank account of the consumer to see whether the consumer has a good balance or not. If so, the consumer acquires the e-cash from the bank over the network. Then, the consumer can spend the e-cash at any merchant. The merchant deposits the e-cash in the bank and the bank verifies the e-cash and checks whether or not it has been spent previously. Finally, the bank deposits the corresponding amount of money in the merchant's bank account.

However, von Solms and Naccache [11] showed that Chaum's protocol is vulnerable to the perfect crime attack in 1992. Later, Kugler and Vogt [9] proposed an on-line payment scheme to prevent blackmailing in 2001. However, Han et al [6] pointed out that the Kugler-Vogt scheme cannot solve the problem of impersonation and proposed a practical scheme to defeating blackmailing. Unfortunately, Han et al scheme still cannot solve the problem of blackmailing [4]. The perfect crimes result from a very powerful characteristic of the blind signatures: untraceability. No one can trace back to a particular owner after the message-signature pair has been published. Therefore, a malicious adversary may take advantage of this unconditional anonymity and commit such crimes as blackmail or kidnapping because ill-gotten money cannot be traced. Kidnappers and blackmailers can commit perfect crimes by using anonymous communication channels and anonymous e-cash. Furthermore, a good electronic cash scheme should satisfy the following requirements, including unforgeability, untraceability, verifiability, double-spending-resistance, and perfect-crimes resistance [1, 5, 7, 8, 10].

The rest of this paper is organized as follows. In Section 2, we briefly describe the Chaum's scheme [2, 3], essential requirements of general electronic cash and demonstrate the perfect crime attack [11] on the Chaum's scheme. In Section 3, we propose an improved scheme that is satisfying all of requirements mentioned above and further analyze it to see if all of the requirements are satisfied in Section 4. Finally, Section 5 concludes this paper.

2: Brief Review and Weakness of Chaum's E-Cash Scheme

In this section, we will briefly review the Chaum's scheme and show the perfect crime attack on the Chaum's scheme in Section 2.1 and 2.2 respectively. Besides, we are going to describe the essential requirements of electronic cash in Section 2.3.

2.1: The Chaum's E-Cash Scheme

Let m, r and $B(m, r)$ are represent as a serial number, a blind factor and a blinding function, respectively. Next, assume that notations of bank, consumer and merchant are B, C and M, respectively. Suppose that the notations of public key and private key are *PKi* and *SKi*, where *i* is represent as a participant. In withdrawal phase, the consumer computes $M' = B(m, r)$ and sends $E_{PKB}(M')$ to bank, where $E_K(.)$ is asymmetric encryption with key K. Next, the bank computes $D_{SKB}(E_{PKB}(M'))$ to decrypt out M' and the bank checks the account of the consumer to see whether there is enough saving to transfer to e-cash, where $D_K(.)$ is asymmetric decryption with key K. If it is enough, the bank signs the blinded message M' by

computing $S' = E_{SKB}(M')$. Then, the bank sends E_{PKC} (S') to the consumer and the consumer decrypts out S' with his/her private key. Finally, the consumer employs the unblinding function with the corresponding blind factor r to obtain the signature on m by computing $S = B^{-1}(S', r)$. Here, (m, S) denotes the e-cash which was published by the bank.

In payment phase, the consumer pays the e-cash (m, S) for any merchandize on any web site over the Internet. First, he/she must encrypt e-cash with the merchant's public key *PKM* for preventing any intruder from intercepting the e-cash. Then, the merchant uses its private key *SKM* to obtain the e-cash from the consumer by computing $D_{SKM}(E_{PKM}(m, S))$. Finally, it uses the bank's public key *PKB* to verify whether or not the e-cash was authorized by the bank. If the e-cash (m, S) is valid, the following equation $m = E_{PKB}(S)$ holds.

In deposit phase, the merchant sends $E_{PKB}(m, S)$ to the bank. Then, the bank uses its private key *SKB* to decrypt out the e-cash (m, S) and checks whether the e-cash was published by itself or not by verifying $m \stackrel{?}{=} E_{PKB}(S)$. If it holds, the bank adds the funds to merchant's account and informs merchant of acceptance. Then the merchant gives the merchandize to the consumer.

2.2: Perfect Crime Attack on the Chaum's E-Cash Scheme

First, we assume the criminal's name is Sinise. He perpetrates the perfect crime by taking the steps below.

Step 1: Sinise opens a bank account and kidnaps a baby.

Step 2: Sinise chooses a set of m_i (m_1, m_2, …, m_n) and a set of r_i (r_1, r_2, …, r_n).

Step 3: Sinise employs the blinding function with the blind factors r_i to get the blinded message M_i'(M_1', M_2', …, M_n') by computing $M_i' = B(m_i, r_i)$. And then he threatens the victim that he will kill the baby if the victim does not send all M_i' to the bank for requesting the e-cash.

Step 4: The victim sends those blinded messages M_i' to the bank according to the instructions.

Step 5: The bank uses its private key *SKB* to sign the received request M_i' by computing $S_i' = E_{SKB}(M_i')$. And then the bank sends the blind signature S_i' to the victim.

Step 6: After receiving the blind signature S_i', the victim publishes the sets S_i on a newspaper.

Step 7: Sinise can obtain the blind signature S_i' by buying the newspaper and unblinds it with the blind factor r_i by computing $S_i = B^{-1}(S_i', r_i)$.

Step 8: Sinise frees the baby. Sinise can now freely spend all this ransom money (m_i, S_i) without worrying about the danger of ever being identified.

2.3: Essential Requirements for Electronic Cash Schemes

In this subsection, we will briefly describe the essential requirements of general electronic cash. These requirements are shown as follows.

• **Unforgeability**: Any one except authorized organizations such as banks cannot issue valid electronic cash.

• **Untraceability**: The relationship between some e-cash and a consumer is untraceable for the bank, except for the case of authorized revocation (ex. double spending tracing).

• **Verifiability**: Every participant can make sure that the e-cash, either in withdrawal, payment or deposit, is published by the authorized organization through the verification procedure.

• **Double-Spending-resistance**: In general, each piece of e-cash can only be spent once. If any malicious party wants to pay the same e-cash a second time for some merchandise, the bank must be able to recognize that. And the bank should be able to identify the malicious party by means of authorized revocation.

• **Double-Depositing-resistance**: In contrast to Double-Spending-resistance,
Double-Depositing-resistance is defined that each piece of e-cash can only be deposit once. If any malicious merchant wants to deposit the same e-cash a second time in the bank, the bank must be able to recognize that. And the consumer could be able to claim his legality in the transaction.

• **Perfect-Crimes-resistance**: In the previous sections, we have discussed how a perfect crime can be conducted if Chaums's scheme is deployed. An ideal electronic cash scheme should withstand a variety of crimes. However, the crimes are always changing forms and diversifying, and therefore we may not be able to list all the possible crimes and guarantee that an e-cash scheme can avoid all crimes. The e-cash scheme should operate with the aids of various financial and network security mechanisms to keep perfect crimes from working.

3: The Proposed Electronic Cash Scheme

In this section, we will gradually introduce our electronic cash scheme. We list some notations in Table 1 and the whole structure of our proposed electronic cash scheme is illustrated in Figure 1. Our electronic cash scheme consists of four phases and with four participants carrying them out: consumer, a trusted third party, a bank and a merchant. In our scheme, we shall use public cryptosystems to make sure that the necessary information gets transmitted over a public channel, and blind signature techniques will be adopted to protect the private information of the participants. The four participants are responsible for the following tasks.

• **Consumer (C)**: People who can purchase merchandise over the Internet by spending the e-cash.

• **Trusted Third Party (TA)**: In our scheme, we add a trusted third party in our e-cash scheme that it is responsible for generating the necessary information for consumer to use the e-cash (ex. serial number) and making correct judgments in case of disputes (ex. double spending).

• **Bank (B)**: Bank is responsible for issuing the e-cash to

the consumer and depositing the e-cash in the merchant's account when the consumer pays e-cash for merchandise.

• **Merchant (M)**: The merchant supplies a variety of merchandise that can be purchased by the consumer over the Internet. And the merchant must verify whether the e-cash is issued by the bank or not when obtaining the e-cash from the consumer. In order to transfer the e-cash to her/his own account, the merchant must deposit the e-cash in the bank. The details of four phases are described as follows.

ID_i	an unique identification of the user *i*
A_i	a bank account of the user *i*
$h(.)$	a public one-way hashing function
sn	an unique serial number
n_i	a set of unique e-cash number
PKi/*SKi*	a public key of the user *i* and its corresponding private key
$E_K(m)$	an encryption function to encrypt message *m* with the key *K*
$D_K(m)$	a decryption function to decrypt message *m* with the key *K*
$B(m,r)$	a blinding function for message *m* with a blind factor *r*
$B^{-1}(m,r)$	a unblinding function for message m with a blind factor *r*
R	a transaction receipt signed by both the consumer and the merchant

Table 1: Notations used through our proposed e-cash scheme

3.1: Registration Phase

Step 1: In order to request authorized e-cash from the bank, the consumer must use his/her private key to generate a certificate by computing $S_C = E_{SKC}(ID_C)$, and then the consumer sends $E_{PKTA}(ID_C, S_C)$ to the TA. When the TA receives the request from the consumer, it decrypts out (ID_C, S_C) and verifies the certificate by computing ID_C ?= $D_{PKC}(S_C)$. If it holds, the TA generates a unique serial number *sn* and records (sn, ID_C) in its database.

Step 2: In order to prove that the serial number *sn* is generated by the TA, it uses its private key *SKTA* to sign *sn* by computing $S_{TA} = E_{SKTA}(sn)$. Then, TA sends $X = E_{PKC}(S_{TA}, sn)$ and $Y = E_{PKB}(S_{TA}, sn)$ to the consumer and the bank, respectively.

Step 3: After receiving *X* from the TA, the consumer decrypts out (S_{TA}, sn) and verifies *sn* by computing sn ?= $D_{PKTA}(S_{TA})$. Finally, if the result is true, the consumer uses this *sn* to apply for the e-cash. Similarly, the bank executes the same verification procedure on *Y* and then records the serial number *sn* in its database to detect possible double spending later.

3.2: Withdrawal Phase

Step 1: After the registration phase, the consumer shall acquire the serial number *sn* from TA, and then he/she can withdraw e-cash from the bank now. Firstly, he/she uses the private key to sign the bank account information by computing $S_A = E_{SKC}(A_C)$. Next, before the consumer submits the serial number *sn* and e-cash number n_i to the bank, he/she employs a random number r_1 as the blind factor to blind the *sn* and n_1(assumed that i=1) by computing $M' = B(h(sn, n_1), r_1)$. Here, n_i is a set of unique e-cash number that allowing the consumer to use his/her unique serial number to request some different e-cashes and the duplications of n_i is not allowed in our scheme. Finally, the consumer sends $W = E_{PKB}(A_C, S_A, M')$ to the bank for requesting the legitimate e-cash.

Step 2: After receiving the request *W* from the consumer, the bank uses its private key to decrypt out (A_C, S_A, M'). Then, the bank verifies the signature S_A with the consumer's public key *PKC* by checking A_C ?= $E_{PKC}(S_A)$. If it holds, the bank withdraws the e-cash from the consumer's account. Then, the bank uses its private key *SKB* to sign the blinded message *M'* by computing $S' = E_{SKB}(M')$. and sends *S'* back to the consumer by computing $W_1 = E_{PKC}(S')$.

Step 3: Upon receiving W_1, the consumer decrypts out *S'* and employs the unblinding function $S = B^{-1}(S', r_1)$ to obtain the e-cash (sn, n_1, S) from the bank. Note that (sn, n_1) was signed by the bank, but the bank did not get any information about the serial number *sn* when it was signed.

3.3: Payment Phase

Step 1: In this phase, the consumer can pay the e-cash (sn, n_1, S) to the merchant to buy the merchandise. In order to acquire the transaction receipt from the merchant, the consumer generates a random number receipt and signs it with his/her private key *SKC* by computing $S_r = E_{SKC}(receipt)$. Then, the consumer employs the blinding function with another blind factor r_2 and computes $S_r' = B(S_r, r_2)$. And for security, the consumer uses the merchant's public key *PKM* to encrypt the e-cash (sn, n_1, S) by computing $C = E_{PKM}(sn, n_1, S, S_r')$.

Step 2: After receiving the encrypted e-cash *C* from the consumer, the merchant decrypts out (sn, n_1, S, S_r') with its private key. Finally, the merchant checks whether the e-cash was really published by the bank or not by verifying $h(sn, n_1)$?= $D_{PKB}(S)$. If it holds, the merchant grants the transaction and gives the merchandise and the receipt to the consumer. Here, the receipt $Sr'' = E_{SKM}(S_r')$ is generated by signing the blinded message S_r' with the private key *SKM*.

Step 3: Once the merchandise and the blinded receipt *Sr"* are received, the consumer uses the unblinding function with the corresponding blind factor r_2 such as $R = B^{-1}(Sr'', r_2) = E_{SKM}(S_r) = E_{SKM}(E_{SKC}(receipt))$. In the end of the transaction, the consumer obtains the merchandise and the corresponding receipt *R* signed by both the consumer and the merchant. In order to prevent the merchant from double depositing the e-cash in the

bank, the consumer must keep the receipt well.

3.4: Deposit Phase

Step 1: To deposit the e-cash (sn, n_1, S) in merchant's bank account, the merchant sends $E_{PKB}(sn, n_1, S)$ to the bank.
Step 2: After receiving the encrypted e-cash (sn, n_1, S), the bank decrypts it by computing (sn, n_1, S) = $D_{SKB}(E_{PKB}(sn, n_1, S))$. Then, the bank checks whether the e-cash was published by itself or not by verifying $h(sn, n_1)$?= $D_{PKB}(S)$. If it holds, the bank adds the amount to the merchant's account and announces the merchant that transaction is successful. Finally, the bank marks the (sn, n_1) non-fresh.

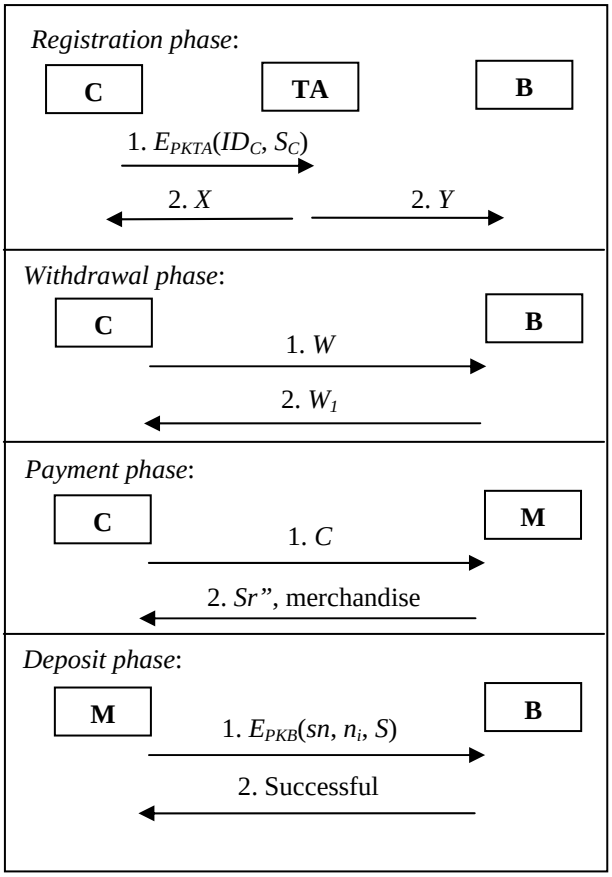

Figure 1: The whole structure of our proposed e-cash scheme

4: Analysis

In this section, we shall analyze our e-cash scheme to see if all of the requirements we brought up earlier are satisfied.
Unforgeability: In our e-cash scheme, only the authorized organization, namely bank, can issue valid e-cash. Each penny of the lawful e-cash is signed by the bank and the signing key is known only to the bank.
Untraceability: Blind signature techniques are what we employ in our scheme to satisfy the untraceability requirement. The bank cannot get any knowledge when it signs the blinded message M' in the withdrawal phase. In the deposit phase, when the merchant deposits the e-cash (sn, n_i, S) in the bank, the bank cannot get any knowledge as to the relationship between the e-cash and the specific consumer. Therefore, we believe that our e-cash scheme can meet this requirement.
Verifiability: Our e-cash scheme is based on public key cryptosystems, hence, the consumer can verify whether the serial number sn was generated by the TA by computing $h(sn)$?= $D_{PKTA}(S_{TA})$ in registration phase. Besides, the consumer can ensure that the e-cash (sn, n_i, S) was issued by the bank by computing $h(sn, n_i)$?= $D_{PKB}(S)$ in the withdrawal phase. Finally, in the payment and deposit phases, the e-cash (sn, n_i, S) can also be verified by using the bank's public key PKB.
Double-Spending-resistance: In our e-cash scheme, each sum of e-cash is recorded in the bank's database and the generated serial number is recorded as well in the TA's database. When the bank receives e-cash from any merchant, it checks whether the e-cash status is fresh or not. If the bank detects that the e-cash is not fresh which means the occurrence of double spending, it requests the TA to reveal the identity of the consumer, namely the ID number ID_C. Therefore, the bank can then know the malicious user.
Double-Depositing-resistance: In payment phase, the consumer can obtain a transaction receipt R which the receipt was signed by the consumer and the merchant. If the merchant framed the consumer for double spending, the consumer can make use of the receipt as the evidence to argue that it's not true. Besides, the receipt R prevents the merchant to double depositing the e-cash in the bank.
Perfect-Crimes-resistance: In Chaum's e-cash scheme, the serial number m is chosen by the consumer. According to von Solms and Naccache's claim, the criminal can freely spend the money without any danger of ever being identified. We believe that the crux of the problem is the serial number m. In our opinion, the serial number should be generated by a trusted third party like the ministry of finance. In our design, if the criminal wants to acquire the money from the victim, since he/she cannot create the serial number by her-/himself, the criminal can only threaten the victim to request the serial number from the TA. Assume the victim applies for the sn in compliance with the instruction. When the criminal obtains the e-cash (sn, n_i, S) and the hostage is set free, the victim can announce that (sn, n_i, S) is ill-gotten money. If the criminal spent the ill-gotten money to buy anything from a merchant, the merchant can report this to the police and helps the police to arrest the criminal. Hence, we believe that the criminal cannot freely spend the money without any danger if the serial number of the e-cash is generated by a trusted third party.

5: Conclusions

It is pointed out that the major problem of the e-cash systems is the perfect crimes. In this paper, we proposed an improved electronic cash scheme. In the proposed

e-cash scheme, we adopt the conventional database to ensure that the e-cash will not be double spent by the consumer. Besides, we suggest that the serial number of each sum of e-cash should be published by a trusted third party like the ministry of finance to discourage the criminal from misusing the power of anonymity.

ACKNOWLEDGEMENT

This work is supported in part by National Science Council and Taiwan Information Security Center at NCTU.

REFERENCES

[1] N. Alexandris, M. Burmester, V. Chrissikopoulos and Y. Desmedt, "Secure linking of customers, merchants and banks in electronic commerce," *Future Generation Computer Systems*, 16(4):393–401, 2000.

[2] D. Chaum, "Blind signatures for untraceable payments," In *Advances in Cryptology(CRYPTO'82)*, pages 199–203, 1982.

[3] D. Chaum, "Blind signatures system," In *Advances in Cryptology(CRYPTO'83)*, pages 153–156, 1983.

[4] X. Chen, F. Zhzng and Y. Wang, "A new approach to prevent blackmailing in e-cash," *(http://eprint.iacr.org/2003/055.pdf)*, 2003.

[5] Y. Frankel, Y. Tsiounis and M. Yung, "Fair off-line e-cash made easy," In *ASICRYPT*: *Advances in Cryptology – International Conference on the Theory and Application of Cryptology*, Springer-Verlag, 1998.

[6] D. G. Han, Y. Y. Park, Y. H. Park, S. Lee, D. H. Lee and H. J. Yang, "A practical approach defeating blackmailing," In *Proceedings of Information Security and Privacy(ACISP2002)*, Springer-Verlag, pages 464-481, 2002.

[7] Min-Shiang Hwang, Iuon-Chung Lin and Li-Hua Li, "A simple micropayment scheme," *Journal of Systems and Software*, 55(3):221–229, 2001.

[8] Min-Shiang Hwang, Eric Jui-Lin Lu, and Iuon-Chung Lin, "Adding timestamps to the secure electronic auction protocol," *Data & Knowledge Engineering*, 40(2):155–162, 2002.

[9] D. Kugler and H. Vogt, "Marking: a privacy protecting approach against blackmailing," In *Proceedings of the 4th International Workshop on Practice and Theory in Public Key Cryptography: Public Key Cryptography(PKC 2001)*, Springer-Verlag, pages 137–152, 2001.

[10] Shingo Miyazaki and Kouichi Sakurai, "A practical off-line digital money system with partially blind signatures based on the discrete logarithm problem," *IEICE Transactions on Fundamentals*, E83-A(1):106–108, 2000.

[11] S. von Solms and D. Naccache, "On blind signature and perfect crime," *Computers & Security*, 11:581–583, 1992.