**American Journal of Engineering Research (AJER)**
e-ISSN: 2320-0847 p-ISSN : 2320-0936
Volume-4, Issue-4, pp-148-152
www.ajer.org
Research Paper Open Access

# An Internet Based Anonymous Electronic Cash System

Israt Jahan[1], Mohammad Zahidur Rahman[2], K M Akkas Ali[3], Israt Jerin[4]
[1,2]*Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh*
[3]*Institute of Information Technology, Jahangirnagar University, Dhaka, Bangladesh*
[4]*Britannia University, Paduar Bazar, Bissaw Road, Comilla, Bangladesh*

***ABSTRACT:*** *There is an increase activity in research to improve the current electronic payment system which is parallel with the progress of internet. Electronic cash system is a cryptographic payment system which offers anonymity during withdrawal and purchase. Electronic cash displays serial numbers which can be recorded to allow further tracing. Contrary to their physical counterparts, e-cash have an inherent limitation; they are easy to copy and reuse (double-spending). An observer is a tamper-resistant device, issued by the Internet bank, which is incorporated with the Internet user's computer that prevents double-spending physically, i.e., the user has no access to her e-cash and therefore he cannot copy them. In this paper, we shall present an anonymous electronic cash scheme on the internet which incorporates tamper-resistant device with user-module.*

***KEYWORDS-*** *E-cash, Double-spending, Tamper-resistant device, Blind signature, Internet banking.*

## I. INTRODUCTION

Electronic commerce is one of the most important applications for the internet. The prerequisite for establishing an electronic marketplace is a secure payment. Several electronic protocols have been proposed to implement different kinds of payment: credit card payments, micropayments, and digital e-cash. Cryptographically, the most challenging task is the design of digital e-cash for every payment system mentioned above we have the requirement that the payment token has to be unforgeable. In 1982, D. Chaum [7] presented the notion of blind signatures that offer the possibility to design electronic e-cash. The bank signs a set of data chosen by the user which guarantees both the unforgeability of the e-cash and their anonymity, since the bank does not get any information about data it signed. But blind signatures solve only half of the problem: since digital data can be copied, a user can spend a valid e-cash several times (double-spending) if the deposit of e-cash is not done on-line [3]. To validate each e-cash on-line means that the vendor has to contact the bank in every purchase. From the efficiency's point of view this is undesirable. Therefore, we restrict our attention to off-line systems, i.e., the vendor has to check the validity of e-cash without contacting the bank. An e-cash is constructed in a way that allows its owner to spend it anonymously once, but reveals his identification if he spent it twice [5]. From a theoretic point of view this solution is quite elegant. But in practice it is unsatisfactory. A way to prevent the user physically from copying her coins is to store essential parts of a coin in a tamper-resistant device called the observer [7].

## II. AN E-CASH MODEL WITH TAMPER-RESISTANT DEVICE

An internet based anonymous off-line electronic e-cash scheme [1, 8 and 9] with tamper –resistant device consists of three collections of probabilistic, polynomially- bounded parties [2], a bank B, users $U_i$, and shops $S_j$, and four main procedures: withdrawal, blind signature issuing, payment and deposit (Figure 1). Users and shops maintain separate account with the Internet Bank [10].

- When user ($U_i$) needs e-cash, then Bank issues e-cash from user's account in his (user's) tamper-resistant device $T_i$ over an authenticated channel.
- When user ($U_i$) wants to spend this e-cash, it is validated by bank (B) by blind signature issuing protocol.
- $U_i$ spends an e-cash by participating in a payment protocol with a shop $S_j$ over an anonymous channel, and
- $S_j$ performs a deposit protocol with the bank B, to deposit the user's e-cash into his account.

(1) Withdrawal protocol
(2) Blind signature issuing protocol
(3) Payment protocol
(4) Deposit protocol

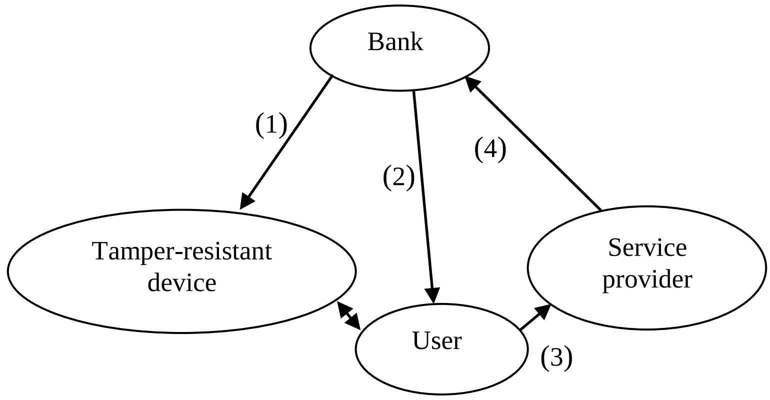


**Figure 1: Model of e-cash with tamper-resistant device**

## III. AN INTERNET BASED ANONYMOUS E-CASH SYSTEM

We shall now represent an anonymous off-line e-cash transaction system on the Internet.

### 3.1 The Bank's setup protocol

- All arithmetic is performed in a group $G_q$ of prime order q chosen by bank (B). The bank generates independently at random four numbers $g_0$, $g_1$, $g_2$, $h \in G_q$ and a number $x \in Z_q$. The bank also determines a collision-free hash function H(.) such as to make the Schnorr signature scheme secure [4]. A public key that is issued by the bank to the user is a pair ( $h'_i$, $a'_i$) $\in G_q * G_q$.
- The number x is the secret key of the bank, and the corresponding public key is the tuple ($g_0$, $g_1$, $g_2$,h, $G_q$, H(.)). A certificate of the bank on the public key ( $h'_i$, $a'_i$) of the user is a triple ($z'_i$, $c'$, $r'$) such that $c'$=H($h'_i$, $a'_i$, $z'_i$, $g_o^{r'}h^{-c'}$, $(h_i')^{r'}$ $(z_i')^{-c'}$).
- The secret key that corresponds to the public key ($h'_i$, $a'_i$) of the user is a pair (($\beta_1$, $\alpha_1$), ($\beta_2$, $\alpha_2$)), such that $h_i'=g_1^{\beta 1}$ $g_2^{\alpha 1}$ and $\alpha_i'= g_1^{\beta 2}$ $g_2^{\alpha 3}$.

### 3.2 The actions

The Internet bank will be denoted by B, the user by $U_i$, and the service provider by $S_j$. The computer of $U_i$ is denoted by $C_i$, and his tamper-resistant device by $T_i$.

#### 3.2.1 Account establishment protocol

$U_i$ installs on his computer, a software program for performing the protocols. When $U_i$ opens an account with B, the following procedure takes place.

- $C_i$ generates independently at random a secret key $x_{i2} \in Z_q$, and stores it. $C_i$ sends $h_{i2} = g_1^{xi2}$, to B, together with an appropriate verifiable description of the identity of $U_i$. It then generates independently at random a secret key $x_{i1} \in Z_q$ for $U_i$. B lists this number ($h_{i2}$) in its so-called account database, together with at least a balance variable that keeps track of the amount of money that $U_i$ has in its account with B, and the description of $U_i$'s identity.
- B then issues to $U_i$ a tamper-resistant device $T_i$ which has stored in non-volatile memory at least the following items: the numbers $x_{i1}$ and $g_1$, and a description of $G_q$; code to perform its role in the protocols; and a counter variable, from now on denoted by *balce,* that keeps track of the amount of money that is held by $U_i$.
- B makes $h_{i1}=g_1^{xi1}$, known to $U_i$; this is the public key of $T_i$. B then computes $h_i=h_{i1}h_{i2}$ (the joint public key of $T_i$ and $U_i$ and stores $h_i$ in his account database along with its other information on $U_i$). The bank B does not know the joint secret key, ($x_{i1}$+$x_{i2}$) mod q, of $T_i$ and $U_i$.
- Finally, B computes $(h_i g_2)^x$, which will henceforth be denoted by $z_i$ known to $U_i$.

### 3.2.2 Withdrawal protocol

The withdrawal of electronic cash appears as follows:

$T_i$ is assumed to have in common with B a secret key k. This secret key, and a sequence number, seq, (which has been set to some initial value, such as zero), have been stored by B before issuing $T_i$ to $U_i$. In addition, the description of a one-way function $f_1(.)$ has been stored by B in $T_i$. B decreases the balance, *balce'*, of $U_i$ by amount. It then increases seq by one, and transfers v $\leftarrow f_1$(k, seq, amount) to $T_i$ by sending it to $C_i$. $T_i$ receives v from $C_i$. It then computes $f_1$(k, seq, amount), and compares it for equality with v. If equality holds, it increases seq by one, and balance by amount.

The withdrawal protocol appears as follows:

| Tamper-resistant Device ($T_i$) | | Bank (B) |
|---|---|---|
| | | *balce'*$\leftarrow$*balce'*- amount |
| Verify | $\leftarrow$( v)----- | v $\leftarrow f_1$(k, seq, amount) |
| v = $f_1$(k, seq, amount) | | seq$\leftarrow$ seq+1 |
| then, seq$\leftarrow$ seq+1 | | |
| *balce*$\leftarrow$*balce* + amount | | |

**Table 1: The withdrawal protocol**

### 3.2.3 The Pre-processing of blind signature issuing protocol

Payment of an amount requires $U_i$ to provide the service provider with a signature on the amount (and additional data). To prepare for the withdrawal of a blind signature on e-cash, $T_i$ and $C_i$ perform the following off-line processing.

- Ti generates independently at random a number $w_i \in_R Z_q$, and sends $a_i g_1^{wi}$ to $C_i$. $T_i$ stores $w_i$ for later use in the payment protocol.
- $C_i$ generates independently at random a vector $(\alpha_1,\alpha_2, \alpha_3,\alpha_4, \alpha_5) \in Z_q^5$, such that $\alpha \neq 0$ mod q. It then computes $h_i' \leftarrow (h_i g_2)^{\alpha 1}$, $a_i' = a_i^{\alpha 1} g_1^{\alpha 2} g_2^{\alpha 3}$ , $z_i' \leftarrow z_i^{\alpha 1}$, $temp_1$ $h^{\alpha 4} g_0^{\alpha 5}$, $temp_2 \leftarrow (z_i')^{\alpha 4}(h_i g_2)^{\alpha 1 \alpha 5}$.
- $C_i$ stores ($h_i'$, $a_i'$) and ($\alpha_1$, $\alpha_2$, $\alpha_3$) and $temp_1$, $temp_2$, $\alpha_4$ and $\alpha_5$ for the later use in the payment protocol.

### 3.2.4 The blind signature issuing protocol

The issuing of blind signature [6] is done by means of the following on-line certificate issuing protocol between $C_i$ and B. The blind signature issuing appears as follows:

| Compouter($C_i$) | | Bank(B) |
|---|---|---|
| | | $w \in Z_q$ |
| | | $a \leftarrow g_0^w$ |
| | $\leftarrow$(a,b)-------------- | $b \leftarrow (h_i g_2)^w$ |
| $c' \leftarrow H(h_i', a_i', a\ temp_1, b^{\alpha 1} temp_2)$ | | |
| $c \leftarrow c' + \alpha_4$ mod q | ---------(c) $\rightarrow$ | |
| | $\leftarrow$(r )------------ | r $\leftarrow$cx + w mod q |

**Table 2: The blind signature issuing protocol**

### 3.2.5 The pre-processing of payment protocol

To pay to $S_j$ an amount, $T_i$ and $C_i$ perform the following pre-processing.

- $C_i$ determines the specification, denoted by *spec*, of the payment. This number is a concatenation, in a standardized format, of that is to be transferred, the time and date of transaction, and an identification number that is uniquely associated with $S_j$. Additional data fields may be included in variable spec. $C_i$ then sends ($h_i'$, $a_i'$) and spec to $T_i$.
- $T_i$ verifies that $w_i$ is still in memory, and thatbalance exceeds amount ($T_i$ can read this value from spec). If this is the case, it computes $d=H(h_i',a_i',spec)$ and $r_1=dx_{i1} + w_i$ mod q. It then decreases balance by amount, erases $w_i$ from memory, and sends $r_i$ to $C_i$.
- $C_i$ computes $d=H(h_i', a_i',spec)$, and verifies that $g_1^{r1} h_{i1}^{-d}=a_i$. If this is the case, $C_i$ computes $r_1'=\alpha_1(r_1+dx_{i2})+ \alpha_2$ mod q, $r_2 \leftarrow d\alpha_1+\alpha_3$ mod q. The pre-processing of payment protocol appears as follows:

**User computer($C_i$)** **Tamper-resistant device($T_i$)**

-------- $(h_i', a_i') \rightarrow$ $d=H(h_i', a_i', spec)$

$balce \rightarrow balce$ - amount

$r_1=dx_{i1}+w_i$

$\leftarrow (r_1)$--------- erases $w_i$

$d=H(h_i', a_i', spec)$

verify

$g_1^{r1}h_{i1}^{-d} = a_i$

$r_1' \leftarrow \alpha_1(r_1+dx_{i2})+ \alpha_2 \bmod q$

$r_2 \leftarrow d\alpha 1+\alpha 3 \bmod q$

**Table 3: The preprocessing of payment protocol**

### 3.2.6 The payment protocol

The actual payment is done by means of the following on-line payment protocol between $C_i$ and $S_j$.

- $C_i$ sends $(h_i', a_i'),(z_i',c',r'),(r_1',r_2)$to $S_j$.
- $S_j$ computers d in the same way as did $C_i$ and $T_i$ and accepts the transferred information if and only if $h_i'\neq 1$ ,$c'=H(h'_i, a'_i, z'_i, g_o^{r'}h^{-c'}, (h_i')^{r'} (z_i')^{-c'})$ and $g_1^{r1'}g_2^{r2}(h_i')^{-d}=a_i'$
- The payment protocol appears as follows:

**Computer($C_i$)** **Service Provider($S_j$)**

-- $(h_i', a_i'),(z_i',c',r'),(r_1',r_2) \rightarrow$

Check

$d=H(h_i', a_i', spec)$

$c'=H(h'_i, a'_i, z'_i, g_o^{r'}h^{-c'}, (h_i')^{r'} (z_i')^{-c'})$

$g_1^{r1'}g_2^{r2}(h_i')^{-d} = a_i'$

**Table 4: The payment protocol**

### 3.2.7 The deposit Protocol

At a suitable time, preferably when network traffic is low, $S_j$ sends the payment transcript, consisting of $(h_i',a_i)$, $(z'_i, c', r')$, $(r_1',r_2)$ and spec, to B.

B verifies that spec has been formed correctly by $S_j$. If this is the case, it searches its so-called deposit database to find out if it has stored $(h_i',a_i)$ before.

There are two possible situations:

1. $(h'_i,a'_i)$ is not in the deposit database. B then computes $d=H(h_i', a_i', spec)$, and verifies the payment transcript by verifying that $h_i'\neq 1$, $c'=H(h'_i, a'_i, z'_i,g_o^{r'}h^{-c'}, (h_i')^{r'} (z_i')^{-c'})$ and $g_1^{r1'}g_2^{r2}(h_i')^{-d}=a_i'$. If these verifications hold, B stores $(h_i', a_i'),(z_i',c',r')$ and $(r_1',r_2)$ in the deposit database, and credits the account of $S_j$ by amount.
2. $(h_i',a_i)$ is already in the deposit database. In that case a fraud has occurred. If *spec* of the already stored information is identical to that of the new payment transcript, then $S_j$ is trying to deposit the same transcript twice.

Otherwise, B verifies the transcript as described insituation 1. If the verification holds (the payment transcript is valid), then the certified public key $(h'_i,a_i)$ must have been double-spent with overwhelming probability. Since, B now has at its disposal a pair $(r_1',r_2)$ from the new transcript and a pair, say $(r_1'',r_2')$, from the already deposited information, it can compute $(r_1'- r_1'')/( r_2- r_2') \bmod q$. B then searches its account database for joint public key $g_1^{(r1'- r1'')/( r2- r2')}$. Since, the identity of the corresponding account holder is known to B, appropriate legal actions can be taken. The number $(r_1'- r_1'')/( r_2- r_2') \bmod q$ serves as the proof of B that the traced user has compromised his tamper-resistant device and has double-spent the certified public key $(h'_i,a_i)$ .

## IV. DISCUSSIONS

In the e-cash scheme with tamper-resistant device, the user's secret is shared between the user and his observer. The combined secret is a modular sum of the two shares, so one share of the secret reveals no information about the combined secret. Co-operation of the user and the tamper-resistant device is necessary in order to create a valid response to a challenge during a payment transaction. It prevents the tamper resistant device from leaking any information about the user.

## V. CONCLUSIONS

We presented electronic cash system which provides a physical defense against double-spending detection. To guarantee the prevention of double-spending, the bank has to be sure that the tamper-resistant device cannot be tampered with by the users. The use of a tamper-resistant device is a kind of first line of defense. If the user cannot manipulate the device, the tamper-resistant device can prevent double-spending. If the user succeeds in tampering the observer, the double-spending detection identifies the user afterwards.

## REFERENCES

[1] Ogiela, M.R., & Sulkowski, P. Improved Cryptographic protocol for digital coin exchange. *15th International Symposium on Soft Computing and Intelligent Systems (SCIS)*, 2014, 1148-1151.

[2] Miers, I., Garman, C., Green, M., & Rubin, A.D. Zerocoin: Anonymous Distributed E-Cash from Bitcoin. *IEEE Symposium on Security and Privacy (SP)*, 2013, 397-411.

[3] Jiangxiao Zhang., Zhoujun Li., Hua Guo., & Chang Xu. Efficient Divisible E-Cash in the Standard Model. *IEEE International Conference on Green Computing and Communications (GreenCom)*, 2013, 2123-2128.

[4] Yun-kyung Lee., Seung-Wan Han., Sok-joon Lee., Byung-ho Chung., & Deok Gyu Lee. Anonymous Authentication System Using Group Signature. *International Conference on Complex, Intelligent and Software Intensive Systems*, 2009, 1235-1239.

[5] Israt Jahan., Mohammad Zahidur Rahman., & Md. Golam Moazzem. Review of Anonymous Electronic Payment System. *Journal of Electronics and Computer Science*, 2003, 25-39.

[6] Tsiounis, Y.S. Efficient Electronic Cash: New Notions and Techniques. *Northern University, Massachusetts*. PhD Thesis, 1997.

[7] Chaum, D., & Pedersen, T. Wallet Databases with Observers. *In proceedings of CRYPTO*, 1993, 89-105.

[8] Brands, S. Untraceable Off-line Cash in Wallets With Observers. *In proceedings of CRYPTO*, 1993, 303-318.

[9] Schnorr, C.P. Efficient Signature Generation by Smart Cards. *Journal of Cryptology*, 1991, 4(3), 161-174.

[10] Chaum, D. Blind Signatures for Untraceable Payments. *Advances in cryptology-Proceedings of Crypto'82, Lecture Notes in Computer Science, Springer-Verlag*, 1982, 199-203.