An Internet Based Anonymous Electronic Cash System

Israt Jahan¹, Mohammad Zahidur Rahman², K M Akkas Ali³, Israt Jerin⁴

¹²Department of Computer Science and Engineering, Jahangirnagar University, Dhaka, Bangladesh
³Institute of Information Technology, Jahangirnagar University, Dhaka, Bangladesh
⁴Britannia 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 spends 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 U₁, and shops Sᵢ and Sᵢ, 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 (U₁) needs e-cash, then Bank issues e-cash from user’s account in his (user’s) tamper-resistant device T, over an authenticated channel.
- When user (Uᵢ) wants to spend this e-cash, it is validated by bank (B) by blind signature issuing protocol.
- Uᵢ spends an e-cash by participating in a payment protocol with a shop Sᵢ over an anonymous channel, and
- Sᵢ 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

![Diagram](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 $x \in \mathbb{Z}_q$. 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 $(h', a') \in G_q \ast G_q$.
- 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 $(h', a')$ of the user is a triple $(Z', c', r')$ such that $c' = H(h', a')^x \mod q$.
- The secret key that corresponds to the public key $(h', a')$ 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 $\alpha_i = g_1^{\beta_2} g_2^{\alpha_2}$.

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 $x_{i2} \in \mathbb{Z}_q$, and stores it. $C_i$ sends $h_2 = g_1^{x_{i2}}$, to B, together with an appropriate verifiable description of the identity of $U_i$. It then generates independently at random a secret key $x_{i1} \in \mathbb{Z}_q$ for $U_i$. B lists this number $(h_2)$ 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 balance, that keeps track of the amount of money that is held by $U_i$.
- B makes $h_1 = g_1^{x_{i1}}$, known to $U_i$; this is the public key of $T_i$. B then computes $h_i = h_i h_1$ (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:

- **Ti** generates independently at random a number \( w_i \in \mathbb{Z}_q \), and sends \( a_i g_1^{w_i} \) to **C**. **Ti** stores \( w_i \) for later use in the payment protocol.

- **Ci** generates independently at random a vector \( (\alpha_1, \alpha_2, \alpha_3, \alpha_4) \in \mathbb{Z}_q^4 \), such that \( \alpha \neq 0 \mod q \). It then computes \( h_i' = (h g_2)^{a_1}, a_1' = a_1 g_1^{a_2} g_2^{a_3} g_3^{a_5} \), \( z_i' = z_i^{a_1}, \text{temp}_1 = h_i^{a_2}, \text{temp}_2 = (z_i')^{a_4} (h g_2)^{a_5} \).

- **C** stores \( h_i', a_1' \) and \( (a_1, a_2, a_3) \) and temp\(_1\), temp\(_2\), \( a_4 \) and \( a_5 \) for the later use in the payment protocol.

3.2.3 The pre-processing of blind signature issuing protocol

Payment of an amount requires **B** 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** verifies that \( w_i \) is still in memory, and that balance exceeds amount (**Ti** can read this value from spec). If this is the case, it computes \( d = H(h_i', a_1', \text{spec}) \) and \( r_i = dx_i + w_i \mod q \). It then decreases balance by amount, erases \( w_i \) from memory, and sends \( r_i \) to **C**\(_i\).

- **Ci** computes \( d = H(h_i', a_1', \text{spec}) \), and verifies that \( g_1^{r_i} h_i^{a_1} = a_0 \). If this is the case, **Ci** computes \( r_i' = a_0 (r_i + dx_i) + a_2 \mod q \), \( r_2 = dx_i + a_3 \mod q \). The pre-processing of payment protocol appears as follows:

| Tamper-resistant Device (**T**\(_i\)) | Bank (**B**): balance 
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Verify ( v ) ( \leftarrow ) ( v \leftarrow f_i(k, \text{seq, amount}) )</td>
<td>seq ( \leftarrow ) seq + 1</td>
</tr>
<tr>
<td>( v = f_i(k, \text{seq, amount}) ) then, seq ( \leftarrow ) seq + 1</td>
<td>balance ( \leftarrow ) balance + amount</td>
</tr>
</tbody>
</table>

Table 1: The withdrawal 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** and **B**. The blind signature issuing appears as follows:

**Computer**(**C**\(_i\)) **Bank**(**B**)

\[ w \in \mathbb{Z}_q \]
\[ a \leftarrow g_0^w \]
\[ b \leftarrow (h g_2)^w \]
\[ c' \leftarrow H(h_i', a_1', a \text{temp}_1, b^a \text{temp}_2) \]
\[ c \leftarrow c + a_4 \mod q \]
\[ \leftarrow (r) \rightarrow \]
\[ r \leftarrow cx + w \mod q \]

Table 2: The blind signature issuing protocol
If appropriate legal actions can be taken. The number (r1′, r2′) mod 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′, a′).

### Table 3: The preprocessing of payment protocol

### Table 4: The payment protocol

3.2.6 The payment protocol
The actual payment is done by means of the following on-line payment protocol between Cj and Sj.

- Cj sends (h′j, a′j, (z′i, c′j, x′), (r1′, r2′)) to Sj.
- Sj computes d = H(h′j, a′j, spec) and accepts the transferred information if and only if h′j ≠ 1, c′ = H(h′j, a′j, z′i), g1i^h′j mod q, and g1i^r1′g2i^r2′(h′j) = a′j.
- The payment protocol appears as follows:

### 3.2.7 The deposit Protocol
At a suitable time, preferably when network traffic is low, Sj sends the payment transcript, consisting of (h′j, a′j), (z′i, c′j, x′), (r1′, r2′) and spec, to B. B verifies that spec has been formed correctly by Sj. If this is the case, it searches its so-called deposit database to find out if it has stored (h′j, a′j) before.

There are two possible situations:

1. (h′j, a′j) is not in the deposit database. B then computes d = H(h′j, a′j, spec), and verifies the payment transcript by verifying that h′j ≠ 1, c′ = H(h′j, a′j, z′i), g1i^h′j mod q, and g1i^r1′g2i^r2′(h′j) = a′j. If these verifications hold, B stores (h′j, a′j, (z′i, c′j, x′)) and (r1′, r2′) in the deposit database, and credits the account of Sj by amount.

2. (h′j, a′j) 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 Sj 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′j, a′j) must have been double-spent with overwhelming probability. Since, B now has at its disposal a pair (r1′, r2′) from the new transcript and a pair, say (r1′′, r2′′), from the already deposited information, it can compute (r1′ - r1′′)/(r2′ - r2′′) mod q. B then searches its account database for joint public key g1i^r1′g2i^r2′(h′j) mod 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′j, a′j).
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