# Multiple-Use Transferable E-Cash

Pratik Sarkar
Bengal Engineering and Science University
West Bengal,India

## ABSTRACT

Ecash is a concept of electronic cash which would allow users to carry money in form of digital coins. Transaction can be done both offline and online in absence of a third party/financial institution. This paper proposes an offline model which supports multiple usage of transferable ecoin. The protocol is based on RSA, digital signature and a two-step encryption process. In this two step encryption, the user account details are encrypted in the coin using unique numbers in each step. The first encryption takes place during the successful receipt of the coin, where a receive end number is used for encryption,which is unique for every receipt. The second step of encryption takes place during successful spending of the coin,where a spending end receive number is used for encryption, which is unique for every spenfing of the coin. These two unique numbers comprise the major part of encryption in this model, prevents double spending and preserves user anonymity.

## Keywords:

E-Cash, E-Commerce, Security.

## 1. INTRODUCTION

Electronic cash is one of the developing applications of cryptology because of its widespread use. The different modes of electronic cash systems include the e-commerce service providers, net banking facilities and the card based systems, such as credit and debit cards, and other offline schemes. It has become an alternative mode- to physical paper money, making payments. Electronic cash systems can be classified into online and offline mode based on the requirement for connectivity with a central server. In both the systems the user opens an account in the bank. They vary in their mode of payment.

In an online system, the transaction between two users is carried out in the presence of a bank. The first user provides his account details and amount of payment to the bank for verification. The bank verifies the credentials of the user and transfers the amount of money from the first users account to the second users account. In this system, a central authority (i.e. bank) is required, to verify the users credentials, and a network connection with the bank is required. If the connection cannot be made with the bank then the users cannot perform the transaction. This compromises the anonymity of the user and is dependent on the availability of a network connection with the bank. In an offline system, the transaction between two users is carried out in the absence of a bank. The user deposits a certain amount of money in the bank and receives digital coins of equivalent denominations. The user carries digital coins and performs monetary transactions by exchanging those coins following certain protocols, without the presence of any central authority. This offline electronic cash is called ecash. The user details are not revealed to the bank during the transactions and it remains secured unless the user performs any forging or copying the coin.

The concept of electronic cash was first introduced by David Chaum [8]. The main idea behind electronic cash is to provide an alternate paper-less monetary system. There has been a lot of work in the field of offline ecash [7, 9, 10, 11, 12, 13, 14, 15, 16, 17]. The security of paper cash depends on the difficulty in producing the coins and bills. The ecash cannot rely on physical features for its security. It depends on the cryptographic techniques and protocols for its security. An ideal offline ecash system should have the following properties:

a. Offline transaction The users can make payments in an offline mode, i.e. they do not need to be connected to bank during the payment.

b. User Anonymity (Untraceability) The bank should not be able to compute the transaction details or the user details from the coin. The privacy of the user details should be protected.

c. Security The cryptographic techniques should protect the coin from being forged or spent more than once.

d. Transferability The users should be able to transfer the coin.

e. Independence from physical conditions The ecash should be able to transmit through networks or as digital files without any dependence on physical conditions.

The offline ecash deals with a series of problems in implementing these four features. The user may copy a coin and spend it more than once. This is called multiple spending. The offline system faces a serious problem in providing against multiple spending. Another problem is the forging of illegal ecash. Various cryptographic techniques are applied to prevent it and provide transferability of the ecash files.

Many ecash systems have been proposed. Some of the popular digital currency companies are Digicash, Faircash and Bitcoin. Digicash[2] was the first of its kind. It was founded by David Chaum in 1990. Faircash is a similar digital currency company[1]. It is a multi-purpose, multi-hop, pre-paid and anonymous electronic payment system. It makes use of a secured chip (CASTOR), which is completely under user's control. The chip uses a new identification technology called Electronic Identity Mutation (EIM) to fabricate uniquely electronic identities. But it has a major flaw[3] in its multiple-spending detection. In case of multiple spending the faircash system requires the cooperation of the users of the ecoin and access the transaction log of the ecoin. This compromises the anonymity of the user and the bank is able to identify all the users,

of the ecoin, in order to find out the actual multi-spender. Bitcoin was first introduced by Satoshi Nakamoto[4]. It is based on a peer-to-peer network where payments are made online in the absence of a central authority. The network timestamps transaction records by hashing them into a ongoing chain of hash-based proof-of-work. The longest chain serves as the proof of transactions occurred and it helps to detect a multiple spender in case of illegal spending. But the nodes must constantly update themselves with the longest chain in order to keep a track of all the records of transactions that took place while they were offline. The security of this system heavily relies on the largest chain and assumes that most of its nodes are honest. If majority of the nodes in the longest chain are dishonest then the scheme would fail to identify any multiple spender. The transactions are made public in the longest chain through hashing and the scheme also requires online transactions. This compromises the user privacy and relies on network availability. There has been further proposed extensions for Bitcoin, like Zerocoin[6].
This paper proposes a concept similar to Bitcoin. It is an offline model of electronic cash which allows the user to transfer the coins. It contains different coins of different denominations to provide for payments of different values. The security of this scheme depends on RSA signatures and a two step encryption process. The transaction details of the user are stored in the coin in two steps of encryption. The encryptions are done by introducing two new features - spending end number and receive end number. The receive end number is used to encrypt the user details when he successfully receives the coin. The spending end number is used to encrypt the user details when he successfully spends the coin. On successful receipt of the coin, the user encrypts his account details in the coin using the receive end number. This is the first step of encryption where the user account details are contained. When the user spends the coin, he encrypts a key to the previous encryption using his spending end number. This encryption contains the key to the first encryption. The second encryption can be broken by the bank only in case of double spending. Using this concept of encryption, the coin supports multiple-transferability, i.e. it can be used multiple times by different users before being deposited in the bank. It also tackles the problem of double spending and counterfeiting by storing the users transaction history in an encrypted form in the coin. Anonymity is preserved as the bank can only decrypt the transaction details of the multiple-spender without gaining any access to the other user details in the coin.

## 2. USER ACCOUNT CREATION

Every user creates an account in the bank. During account creation every $i^{th}$ user gets a unique identity/account number $I_i$ . The Bank and user generate a pair of RSA[18] keys for every user's account. The bank keeps the key - ( $d_i$ , $n_i$ ) as private and makes the - ( $e_i$ , $n_i$ ) key public for the user to encrypt his identity in the coin during transaction. The bank uses its private key to decrypt and obtain the user identity from the user history in case of double spending.

## 3. DESCRIPTION OF COIN

In this model every coin has its own identity number. This identity number is used to differentiate between every coin. Each coin has a particular face value which represents its denomination. Each $k^{th}$ coin has a unique identity number $A_k$ and a face value $F_k$. Bank generates a pair of RSA keys as its signature keys. It provides digital signature [5] on the coin using its private key ( $d'_u$ , $n'_u$ ) . The public key ( $e'_u$ , $n'_u$ ) is kept in the coin to verify the bank's signature and hence the validity of the coin.This part of the coin is only viewable. Once the bank generates the coin, this part of the coin becomes immutable. Users can only verify the coin using the public key and signatures. Another part of the coin is the user history part. It contains the transaction details of the user in an encrypted form.
[ $A_k$ , $(A_k)^{d'_u}$ , $F_k$ , $(A_k \ XOR \ F_k)^{d'_u}$ , ( $e'_u$ , $n'_u$ ) , {User History} ]

## 4. VERIFICATION OF COIN

During transaction the coin is verified against counterfeiting and denomination change. The digital signatures, provided by the bank on the coin's identity number and the face value, are checked using the public key of the coin.

$((A_k)^{d'_u})^{e'_u} \bmod n'_u = A_k$ [From RSA] (1)

ii. $((A_k \ XOR \ F_k)^{d'_u})^{e'_u} \bmod n'_u = (A_k \ XOR \ F_k)$ [From RSA] (2)

iii. $(A_k \ XOR \ F_k)$ XOR $A_k = F_k$ [Verification of denomination] (3)

### 4.1 Counterfeiting

If $A_k$ is generated by the counterfeiter, it cannot receive the digital signature $(A_k)^{d'_u}$ as he does not know the bank private key of the coin and he cannot generate it from the public key of the other coins. The counterfeit coin gets discarded during the verification step (1) and it cannot be generated, provided the bank's private key for the coin is not compromised.

### 4.2 Changing denominations

The counterfeiter can try to change the denomination of the coin by changing $F_k$. But the scheme prevents such denomination change by using $(A_k \ XOR \ F_k)^{d'_u}$ as the mode of verification for $F_k$ value. Let the changed denomination value be $F_k$' value. The $F_k$' gets discarded as it cannot pass the verification steps(2 and 3). So the denomination of the coin cannot be changed.

## 5. USER HISTORY

In this model the transaction details are stored in the coin. In bitcoin the user trasaction details are transmitted as a chain of digitally-signed transactions [4] in a peer-to-peer network for verification of chain of ownerships. Here, a similar idea has been proposed, where the user transactions details are encrypted into the coin using a two step encryption. The encryption is done using two numbers. These two numbers are - spending end number and receive end number. The user account details are encrypted into the coin using these two numbers for future verification of double spending of the coin. Once encrypted and stored in the coin, the user details cannot be modified. It can be just viewed by the bank, for verification purposes.

$\{ S_{ij} , (S_{ij} \ XOR \ I_i)^{e_i} , (S_{ij} \ XOR \ n_i).b_{ij} , (n_i \ XOR \ b_{ij}) \}$

The user details are encrypted into the coin in a 2 step encryption process. The first step of encryption takes place when the user successfully receives the coin. Every $i^{th}$ user has a receive end

number $S_{ij}$ , which is unique for $j^{th}$ receipt of coin. The $S_{ij}$ value is different for every $j^{th}$ receipt of coin by $i^{th}$ user. He uses this number to encrypt his details into the user history part of the coin, when he successfully receives the coin. Once added into the user history part of the coin, the

The second step of encryption takes place when the user successfully spends the coin. Every $i^{th}$ user has a spending end number $b_{ij}$ which is unique for $j^{th}$ spending of coin. The $b_{ij}$ value is different for every $j^{th}$ spending of coin by $i^{th}$ user. He uses this number along with the receive end number to encrypt his details into the user history part of the coin, when he successfully spends the coin. The existing user history cannot be modified by the users. Only new transaction details can be added in it.

## 6. WITHDRAWAL PROTOCOL

The bank generates the coin - ($A_k$ , $F_k$) . $A_k$ is the coin identity number and $F_k$ is the denomination of the coin. It uses the private key ( $d'_u$ , $n'_u$ ) of the RSA pair to provide digital signature on the coin's contents and stores the public key ( $e'_u$ , $n'_u$ ) in the coin. The bank signs as follows:

sign($A_k$) = $(A_k)^{d'_u}$

sign($F_k$) = $(A_k \ XOR \ F_k)^{d'_u}$

The bank sends the coin to the user as :

$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) )$

The user verifies the bank's signatures. He generates a random number, the receive end number, $S_{ij}$ on successful verification of coin's identity and receives the coin. This $S_{ij}$ is unique for every receipt. Next time when the user receives a coin he will use a different $S_{ij}$. The user encrypts his identity $I_i$ into the coin using the $S_{ij}$ value and his public RSA account key ( $e_u$ , $n_u$ ) as follows:

Encryption of $I_i$ = ( $S_{ij}$ XOR $I_i$)$^{e_i}$ mod $n_i$
The contents of the coin with the user after withdrawal:

$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) , \{ S_{ij}, ( S_{ij}$ XOR $I_i)^{e_i}$ mod $n_i\} )$

## 7. SPENDING PROTOCOL

The user spends the coin to a merchant. The merchant verifies the bank's signatures in the coin using the coin's public keys. After successful verification, the user generates a random number, the spending end number - $b_{ij}$ which is used to encrypt the user account details into the coin. The user computes ( $b_{ij}$ *XOR* $n_i$ ) and ($s_{ij}$ *XOR* $n_i$) .$b_{ij}$ and incorporates them in the coin and the transaction takes place. This incorporation denotes the successful spending of the coin by the user. ( $b_{ij}$ *XOR* $n_i$ ) is used for calculation of $n_i$ in case of double spending. ( $s_{ij}$ *XOR* $n_i$) .$b_{ij}$ is used for storing the account key of user in encrypted form in the coin which can be decrypted only in case of double spending. The contents of the coin, passed on to the merchant:
$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) , \{ S_{ij}, ( S_{ij}$ XOR $I_i)^{e_i}$ mod $n_i$ , ( $b_{ij}$ *XOR* $n_i$ ) , ( $s_{ij}$ *XOR* $n_i$) .$b_{ij}$ $\} )$

The merchant completes the spending protocol by generating his receive end number and encrypting his identity $I_m$ ( here *i=m* ) into the coin, just as the user does at the end of the withdrawal protocol.

## 8. AFTER *M* - TRANSACTIONS

The coin can undergo any number of transactions before being deposited in the bank. Let us assume the coin underwent *m* - transactions before being deposited in the bank. The coin underwent *m* receipts and *m*-1 spending protocols. The contents of the coin after *m* -transactions :
$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) ,$
$\{ S_{1j}, ( S_{1j}$ XOR $I_1)^{e_1}$ mod $n_1$ , ( $b_{1j}$ *XOR* $n_1$ ) , ( $S_{1j}$ *XOR* $n_1$) .$b_{1j}$ $\} ,$
$\{ S_{2j}, ( S_{2j}$ XOR $I_2)^{e_2}$ mod $n_2$ , ( $b_{2j}$ *XOR* $n_2$ ) , ( $S_{2j}$ *XOR* $n_2$) .$b_{2j}$ $\} , \ldots$
$\{ S_{mj}, ( S_{mj}$ XOR $I_m)^{e_m}$ mod $n_m$ $\} )$
[ i=1,2,...m]

## 9. DEPOSIT PROTOCOL

The $m^{th}$-user deposits the coin in the bank. He incorporates his details in the coin using the spending end number $b_{mj}$ as follows :
$\{$ ( $b_{mj}$ *XOR* $n_m$ ) , ( $S_{mj}$ *XOR* $n_m$) .$b_{mj}$ $\}$
The bank receives the coin and verifies its signature in the coin. It checks for double spending by using the $A_k$ value of the coin. The bank checks in its record whether any other coin with the same identity number has been deposited or not. If a coin with same identity number has been deposited earlier, then double spending has occurred and the coin is sent for double spending verification protocol. If double spending has not been detected then the coin is deposited in the bank and the user history and coin identity number is stored by the bank for double spending verifications against future deposits of coins.

## 10. DOUBLE SPENDING

The bank checks in its record for any deposited coin with the same $A_k$ value as this. If it does not exist then double spending has not occured. If another coin with the same $A_k$ value has been deposited earlier then double spending has occurred. If double spending has occurred then there must be atleast one user who has received the coin, copied it and spent it twice. That user will have same *S* value in both the coins but different *b* values in the coins. Both the coins will have same coin identity number, denomination and signatures. Only the user history details will vary.
Let the first coin be:

$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) ,$
$\{ S, ( S$ XOR $I)^{e}$ mod $n$ , ( $b_1$ *XOR* $n$ ) , ( $S$ *XOR* $n$) .$b_1$ $\}$ (4)

Let the second coin be:

$(A_k , F_k , (A_k)^{d'_u}, (A_k \ XOR \ F_k)^{d'_u}, ( e'_u , n'_u ) ,$
$\{ S, ( S$ XOR $I)^{e}$ mod $n$ , ( $b_2$ *XOR* $n$ ) , ( $S$ *XOR* $n$) .$b_2$ $\}$ (5)

NOTE: The user history details are simplified into *S*, *I*, *e*, *b* and *n* values for ease of calculation. Both the *S* values for the coins will be same as the coin is received once by the user and spent twice or more. Only the *b* values will differ. The bank needs to obtain

*I* using *S* and *n* from (4) and (5). It computes ( *S XOR n* ) as follows:

( ( $S$ *XOR* $n$) .$b_1$ XOR ( $S$ *XOR* $n$) .$b_2$ ) / ( ( $b_1$ *XOR* $n$ ) XOR ( $b_2$ *XOR* $n$ ) )

= ( ( $S$ *XOR* $n$) ( $b_1$ XOR $b_2$ ) )/ ( $b_1$ XOR $b_2$ )

= $S$ *XOR* $n$ (6)

The bank uses the ( $S$ XOR $n$ ) value to find *n* as follows from(4) and (6) :

( $S$ *XOR* $n$ ) *XOR S*

= *n* (7)

For the corresponding *n* value bank searches for the private key *d* from the set of ( $d_i$ , $n_i$ ) values and find the corresponding *d* value for the double-spending user. It decrypts and obtains the *I* from (4) as follows:

(( $S$ XOR $I$)$^e$ )$^d$ mod $n$

= $S$ *XOR* $I$ (7)

Using (4) and (7), he decrypts *I* using *S* and ( $S$ *XOR* $I$) values as:

( $S$ *XOR* $I$ ) *XOR S*

= *I*
Tha bank obtains the identity/account number *I* of the double spender.

## 11. ANONYMITY OF USER

The bank cannot obtain the identity number of any user unless that user has double spent a coin. The bank cannot decrypt any user details from 1 record and it cannot identify more than 1 record of any user from the user history details as user identity number is encrypted in the coin using unique(for every transaction) receive and spending end numbers. The bank cannot obtain details from one record.

( $S$ *XOR* $n$).$b$ XOR ( $S$ *XOR* $n$).$b$

= ( $S$ *XOR* $n$) (*b XOR b*)

= 0
The anonymity of user is compromised only when there are two or more coins with same *S* value for a user with different $b_j$ values, i.e case of double spending. Hence user anonymity is preserved.

### Acknowledgment

I would like to thank Abhishek Chanda and Sourav Sengupta for their valuable suggestions.

## 12. CONCLUSION

This paper proposes a system for electronic transactions which supports multiple transferability of the ecoin. It started with using digital signatures and RSA but it introduced a two-step encryption process to tackle the problem of double spending and preserve user anonymity. This encryption is difficult to break except in case of double spending. The unique nature of both the spending end and receive end numbers prevent the anonymity of the user from being compromised. The user must provide unique numbers for encryption, for each transaction to prevent leakage of information to the bank on deposit of coin. This encryption scheme increases the size of the coin after every transaction. There has to be further work on this model, to control the size of the coin. This will also reduce the number of records the bank has to keep, for double spending verification, after successful deposit of ecoins. There is another problem in this model. The spending and receive end numbers have to be made unique from within the protocols, without depending on the user for providing unique numbers. It can be done by relating some user specific details with the coin's identity number. User specific detail is anything that is unique to every user. As each coin has its own unique identity number, so relating any user's specific detail with it, will give a unique number for every transaction.

## 13. REFERENCES

[1] Kreft, H , Adi,W "fairCASH - A Digital Cash Candidate for the proposed GCC Gulf Dinar.

[2] DigiCash , http://en.wikipedia.org/wiki/DigiCash.

[3] Yen Choon Ching, Heinz Kreft , Faircash: Concepts and Framework.

[4] Satoshi Nakamoto, "Bitcoin: A Peer-to-Peer Electronic Cash System".

[5] R.L. Rivest, A. Shamir, and L. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems".

[6] Ian Miers, Christina Garman, Matthew Green, Aviel D. Rubin, Zerocoin : Anonymous Distributed E-Cash from Bitcoin.

[7] Brands, S., Untraceable Off-line Cash in Wallet with Observers, Proceedings of Crypto 93, pp.302-318 (1994).

[8] Chaum, D., Fiat, A., and Naor, M., Untraceable Electronic Cash, Proceedings of Crypto 88, pp.319-327 (1990).

[9] Damingo, S. and Di Crescenzo, G., Methodology for Digital Money based on General Cryptographic Tools, Proceedings of Eurocrypt 94.

[10] De Santis, A. and Persiano, G., Communication Efficient Zero-Knowledge Proofs of Knowledge (with Applications to Electronic Cash), Proceedings of STACS 92.

[11] Eng, T. and Okamoto, T. Single-Term Divisible Coins, Proceedings of Eurocrypt 94.

[12] Ferguson, N., Single Term Off-line coin", Proceedings of Eurocrypt 93, pp.318-328 (1994).

[13] Franklin, M. and Yung, M., Secure and Efficient Off-Line Digital Money, Proceedings of ICALP 93, pp. 449-460 (1993).

[14] Hayes, B., Anonymous One-Time Signatures and Flexible Untraceable Electronic Cash , Proceedings of AuscrvDt 90.

[15] Okamoto, T., and Ohta, K., Disposable Zero-Knowledge Authentication and Their Applications to Untraceable Electronic Cash, Proceedings of Crypto 89.

[16] Okamoto, T., and Ohta, K., Universal Electronic Cash, Proceedings of Crypto 91.

[17] Pailles, J.C., New Protocols for Electronic Money, Proceedings of Auscrypt 92.

[18] Rivest; Ronald L. (Belmont, MA), Shamir; Adi (Cambridge, MA), Adleman; Leonard M. (Arlington, MA), December 14, 1977, U.S. Patent 4,405,829.