Digital Cash

Neeraj Venugopal
011541907
Computer Engineering Department,
College of Engineering
San Jose State University, San Jose, CA 95112
neeraj.venugopal@sjsu.edu

Harika Reddy Billuri
012462164
Computer Engineering Department,
College of Engineering
San Jose State University, San Jose, CA 95112
harikareddy.billuri@sjsu.edu

Yashaswi Doddaveerappa
012468066
Computer Engineering Department,
College of Engineering
San Jose State University, San Jose, CA 95112
yashaswi.doddaveerappa@sjsu.edu

Namratha Venkatesh Murthy
012475021
Electrical Engineering Department,
College of Engineering
San Jose State University, San Jose, CA 95112
Namratha.venkateshmurthy@sjsu.edu

ABSTRACT

The need of a payment system which enables the electronic transactions is growing with the use of Internet. Present day electronic payment systems have a major problem, they cannot handle the security. People like to use paper cash because it is easy to carry around, they can make a payment with the received cash and they don't need to ask a third party like a bank to perform their payments. Paper cash can, however, be stolen or lost and no one compensates for the lost or stolen money.

Checks, debit cards and credit cards reduce risk of lost cash for people but allow people to invade your privacy. Your financial transactions are always being monitored. People need a way to protect their anonymity to protect their privacy. Considering the increase of electronic services such as Internet, the need for more efficient electronic payments has become an essential fact.

In this report, the type of electronic cash focused on is Digital Cash. This report describes the concept of digital cash, a way to implement electronic payments in an environment of mutual trust between the bank and the system users. Digital cash offers a solution to the problems of paper cash and today's debit cards, credit cards and check; it is secure and protects people's privacy by protecting the user anonymity and makes payment untraceable. This is achieved by using secret splitting, blind signature and bit commitment protocols.

KEYWORDS

Digital Cash, Secret Splitting, Bit Commitment, Blind Signature, Digital Signature

1 INTRODUCTION

Digital cash (also known as electronic cash, e-cash, D-cash) refers to a system in which a person can securely pay for goods or services electronically without necessarily involving a bank to mediate the transaction. It is an anonymous token-based electronic system. In simpler words, a customer/user who has value can transfer that value to a merchant and the merchant can accumulate the value and redeem it with the issuer of that money. It is the issuer that loads the value to the customer in the first place.

The customer can use digital cash to pay over the Internet. Digital cash is a payment system which enables a secure off-line transaction. Digital cash can be used as electronic money since it keeps its user's anonymity, enables off-line transactions, is portable and at the same time offers the ability of electronic transactions.

Unlike real cash, digital cash does not have physical existence (paper cash). But digital cash has value that makes possible digital cash exchangeable for paper cash. Digital cash is a transfer protocol for some underlying form of money such as real cash.

2 PROPERTIES

2.1 Security

The transaction protocol must ensure that high-level security is maintained through sophisticated encryption techniques. For instance, Alice can pass digital cash to Bob such that either of them, or others, cannot alter or reproduce the electronic token.

2.2 Anonymity

Anonymity assures the privacy of a transaction on multiple levels. It can maintain the anonymity of the person. The transaction carried out is not traceable. Both Alice and Bob have the option to remain anonymous in relation to the payment. Furthermore, at the second level, they have the option to remain completely invisible to the mere existence of a payment on their behalf.

2.3 Portability

The security and use of the digital cash is not dependent on any physical location. The cash can be transferred through computer networks and off the computer network into other storage devices. Alice and Bob can walk away with their digital cash and transport it for use within alternative delivery systems, including non-computer network delivery channels. Digital wealth should not be restricted to a unique, proprietary computer network.

2.4 Two-way payments

The digital cash can be transferred to other users. Essentially, peer-to-peer payments are possible without either party required to attain registered merchant status in contrast with today's card-based systems. Alice, Bob, Carol, and David share an elaborate dinner together at a trendy restaurant and Alice pays the bill in full. Bob, Carol, and David each should then be able to transfer one-fourth of the total amount in digital cash to Alice.

2.5 Off-line capability

The protocol between the two exchanging parties is executed off-line, meaning that neither of the parties is required to be host-connected to process. Availability must be unrestricted. Alice can freely pass value to Bob at any time of day without requiring third-party authentication.

2.6 Divisibility

A digital cash token for a given amount can be subdivided into pieces of cash in smaller amounts. The cash must be fungible so that reasonable portions of change can be made. Alice and Bob can approach a provider or exchange house and request digital cash breakdowns into the smallest possible units. The smaller the breakdowns, the better it is to enable high quantities of small-value transactions.

2.7 Infinite duration

The digital cash does not expire. It maintains value until lost or destroyed provided that the issuer has not debased the unit to nothing or gone out of business. Alice can store a token somewhere safe for ten or twenty years and then retrieve it for use.

2.8 Wide acceptability

The digital cash is well-known and accepted in a large commercial zone. Primarily a brand issue which implies recognition of and trust in the issuer. With several digital cash providers displaying wide acceptability, Alice should be able to use her preferred unit in more than just a restricted local setting.

2.9 User-friendliness

The digital cash should be simple to use from the perspective of both the sending and the receiving. Simplicity leads to mass use which leads to wide acceptability. Alice and Bob does not require an advanced degree in cryptography as the protocol machinations should be transparent to the immediate user.

2.10 Unit-of-value or monetary freedom

Another important need is that the digital cash is denominated in market-determined, non-political monetary units. Alice and Bob can issue non-political digital cash denominated in any defined unit which competes with governmental-unit digital cash.

3 STRUCTURE

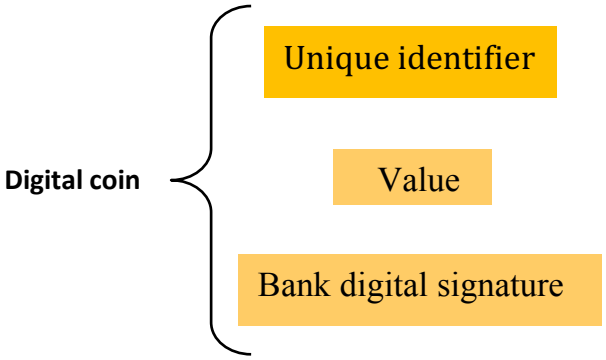
Digital cash transactions include:

- A financial institution (The bank).
- A payer or customer (Alice).
- A payee or a merchant (Bob).

Figure 1: Digital Cash Transaction

Digital coin consists of the following elements:

- Serial number - a unique number that identifies the coin.
- Denomination - the actual value of the coin.
- Digital Signature - the signature provided by the bank



4 OPERATION

The idea of Digital Cash Transaction is shown from Figure 2.

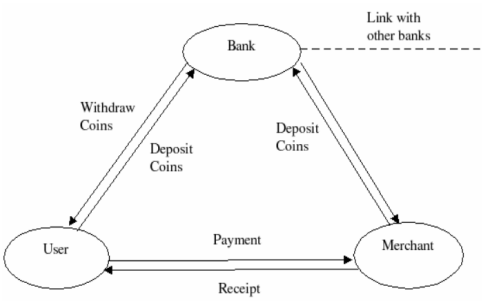


Figure 2: Block Diagram for steps involved in Digital Transaction

Steps involved in digital cash transaction:

- Customer withdraw digital cash from bank and stores in a digital wallet.
- Customer orders the product he require via internet at merchant's site.
- Merchant asks for payment.
- Customer pays bill, sending digital cash to the merchant.
- Merchant sends digital cash to the bank for validation.
- Bank validates digital cash, then merchant deposit digital cash.
- Finally, merchant sends product and receipt to the customer.

There are several protocols used to perform digital cash transaction. In this project, protocol 4 (Section 4.1) is used.

4.1 Protocol 4

If it turns out that the person who bought the money order tried to cheat the merchant, the bank would want to know who that person was. To do that requires moving away from a physical analogy and into the world of cryptography.

The technique of secret splitting can be used to hide Alice's name in the digital money order.

(1) Alice prepares *n* anonymous money orders for a given amount. Each of the money orders contains a different random uniqueness string, *X*, one long enough to make the chance of two being identical negligible. On each money order, there are also *n* pairs of identity bit strings, *I*1, *I*2, ..., *I*n. (Yes, that's *n* different pairs on *each* check.) Each of these pairs is generated as follows: Alice creates a string that gives her name, address, and any other piece of identifying information that the bank wants to see. Then, she splits it into two pieces using the secret splitting protocol (see Section 4.2). Then, she commits to each piece using a bit-commitment protocol (see Section 4.3).

For example, *I*37 consists of two parts: *I*37L and *I*37R. Each part is a bit-committed packet that Alice can be asked to open and whose proper opening can be instantly verified.

Any pair (e.g., *I*37L and *I*37R, but not *I*37L and *I*38R), reveals Alice's identity. Each of the money orders looks like this:

Amount

Uniqueness String: X Identity Strings: I1 = (I1L, I1R)

I2 = (I2L, I2R) ....

In = (InL, InR)

(2) Alice blinds all *n* money orders, using a blind signature protocol (see Section 4.4). She gives them all to the bank.

(3) The bank asks Alice to unblind *n* - 1 of the money orders at random and confirms that they are all well-formed. The bank checks the amount, the uniqueness string, and asks Alice to reveal all of the identity strings.

(4) If the bank is satisfied that Alice did not make any attempts to cheat, it signs the one remaining blinded money order. The bank hands the blinded money order back to Alice and deducts the amount from her account.

(5) Alice unblinds the money order and spends it with a merchant.

(6) The merchant verifies the bank's signature to make sure the money order is legitimate.

(7) The merchant asks Alice to randomly reveal either the left half or the right half of each identity string on the money order. In effect, the merchant gives Alice a random *n*- bit selector string, *b*1, *b*2, ..., *b*n. Alice opens either the left or right half of *I*i, depending on whether *b*i is a 0 or a 1.

(8) Alice complies.

(9) The merchant takes the money order to the bank.

(10) The bank verifies the signature and checks its database to make sure a money order with the same uniqueness string has not been previously deposited. If it hasn't, the bank credits the amount to the merchant's account. The bank records the uniqueness string and all of the identity information in a database.

(11) If the uniqueness string is in the database, the bank refuses to accept the money order. Then, it compares the identity string on the money order with the one stored in the database. If it is the same, the bank knows that the merchant copied the money order. If it is different, the bank knows that the person who bought the money order photocopied it. Since the second merchant who accepted the money order handed Alice a different selector string than did the first merchant, the bank finds a bit position where one merchant had Alice open the left half and the other merchant had Alice open the right half. The bank XORs the two halves together to reveal Alice's identity**.**

4.2 Secret Splitting Protocol

Secret Splitting enables you to split a secret into different shares and give these shares in the custody of multiple persons without disclosing the secret itself.

Secret splitting is especially useful in situations where you feel uncomfortable in sharing a secret with others and you have doubts about the reliability of some of them. You don't want one to misuse the secret behind the other's back. For example, you invented a new cheese sauce that is tasteless than your competitors, it is important for you to keep it as a secret. This calls for secret splitting. There are ways to take a message and divide it up into pieces.

Each piece by itself means nothing but when put together and the message appears. If the message is the recipe and each employee have a piece, then only together can they make the sauce. If any employee resigns with his single piece of the recipe, his information is useless by itself.

Here's a protocol in which Bob can split a message between Tom and Jerry:

a) Bob generates a random-bit string, R, the same length as the message, M.

b) Bob XORs M with R to generate S.

$M \oplus R = S$

c) Bob gives R to Tom and S to Jerry.

To reconstruct the message, Tom and Jerry have only one step to do:

d) Tom and Jerry XOR their pieces together to reconstruct the message.

$R \oplus S = M$

4.3 Bit Commitment Protocol

The bit commitment protocol was developed to prevent people from changing answers. For instance, you want to prove that you know which party will win the next election. You can write your answer in a file, encrypt the file and give it to your friend. When the election is over,

you would give the key to your friend and he would decrypt the file and know if you were lying or not.

The problem is that you could cheat by having two different keys and such that each of them would give a different result. Then depending on what the election's result will be you can choose the write key and proving that you knew the result.

Bit commitment protocols are designed to prevent this deception from taking place.

There are three different kinds of Bit commitment protocols:

a) Bit commitment using symmetric cryptography.

b) Bit commitment using one-way function.

c) Bit commitment using Pseudo-Random-sequence Generators**.**

4.4 Blind Signature Protocol

One of the simplest blind signature schemes is based on RSA signing**.** A traditional RSA signature is computed by raising the message m to the secret exponent d modulo the public modulus. The blind version uses a random value r, such that r is relatively prime to N (i.e. gcd(r,N) = 1). R is raised to the public exponent e modulo N, and the resulting value r^e mod N is used as a blinding factor. The author of the message computes the product of the message and blinding factor, i.e.

$$m' \equiv mr^e \pmod{N}$$

and sends the resulting value m' to the signing authority. Because r is a random value and the mapping r->r^e mod N is a permutation it follows that r^e mod N is random too. This implies that m' does not leak any information about m. The signing authority then calculates the blinded signature s' as:

$$s' \equiv (m')^d \pmod{N}.$$

s' is sent back to the author of the message, who can then remove the blinding factor to reveal s, the valid RSA signature of m:

$$s \equiv s' \cdot r^{-1} \pmod{N}$$

This works because RSA keys satisfy the equation $r^{ed} \equiv r \bmod N$ and thus

$$s \equiv s' \cdot r^{-1} \equiv (m')^d r^{-1} \equiv m^d r^{ed} r^{-1} \equiv m^d r r^{-1} \equiv m^d \pmod{N},$$

hence *s* is indeed the signature of *m*.

In practice, the property that signing one blinded message produces at most one valid signed messages is usually desired. This means one vote per signed ballot in elections, for example. This property does not hold for the simple scheme described above: the original message and the unblinded signature is valid, but so is the blinded message and the blind signature, and possibly other combinations given a clever attacker. A solution to this is to blind sign a cryptographic hash of the message, not the message itself**.**

5 RESULTS

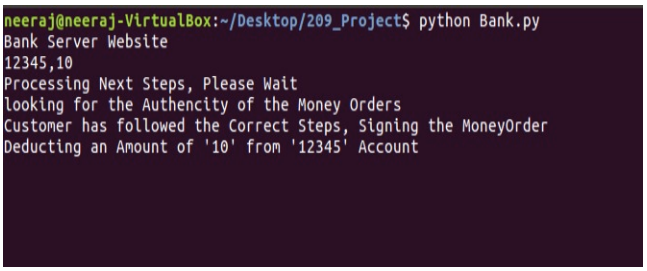


Figure 3: Bank Interface

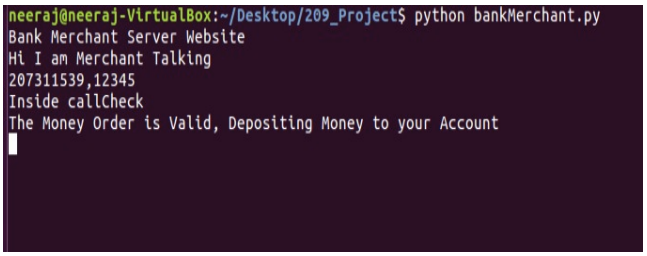


Figure 4: Bank Merchant Interface

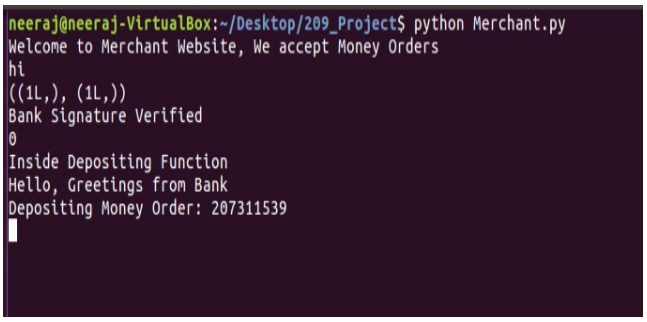


Figure 5: Merchant Interface

```
neeraj@neeraj-VirtualBox:~/Desktop/209_Project$ python Customer.py
Welcome to the Customer Portal.
Please Enter the Customer ID:
12345
Please Enter the Amount you want to be present in Money Order:
10
'Alice' your balance requirements have been met.
Enter the number of money orders you want to process
2
Secret Splitting, Bit Commitment and the MoneyOrder Files have been Performed.
Blind the Money Orders

All the Money Orders Have Been Blinded, Contacting Bank for Next Steps.
Unblind n - 1 Money Orders and reveal the identity Strings
12345
Unblinded n - 1 Money Orders, Verifying with the bank
Signing the Last Money Order, for the customer to use
Signed the Money Order, You can use them
Your Money Order is Ready to use, do let us know whether you want to transfer
1
Preparing for Merchant Connection
Revealing all the Left Identity Strings
Revealing all my left Strings
('74425649654f5163696269694a4d435361',)
('4654707863736164674d645a525a67684a',)
End
neeraj@neeraj-VirtualBox:~/Desktop/209_Project$
```

Figure 6: Customer Interface

```
neeraj@neeraj-VirtualBox:~/Desktop/209_Project$ cat 207311539.txt
------------------------------------------------
Money Order Number: 207311539
------------------------------------------------
Money Order Amount: 10
------------------------------------------------
Unique String of Customer: 3132333435416c696365416c6963654044696769436173682e636f6d
------------------------------------------------
1 Iteration:
Left String Value:  74425649654f5163696269694a4d435361
Right String Value: 45726dd65cae3ccc52aa51380cc3d8a691
------------------------------------------------
2 Iteration:
Left String Value:  4654707863736164674d645a525a67684a
Right String Value: 7764a0b110d99daa48dc003ae151257d4c
------------------------------------------------
```

Figure 7: Money Order Format

```
neeraj@neeraj-VirtualBox:~/Desktop/209_Project$ mysql
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 233
Server version: 5.7.22-0ubuntu0.16.04.1 (Ubuntu)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql> use DigitalCash;
Reading table information for completion of table and column names
You can turn off this feature to get a quicker startup with -A

Database changed
mysql> select * from CustomerDetails;
+----------+-----------+---------------+---------------------+
| Cust_Id  | Cust_Name | AccountBalance | Address             |
+----------+-----------+---------------+---------------------+
|    12345 | Alice     |          1987 | Alice@DigiCash.com  |
|    96206 | bob       |          1234 | bob@DigiCash.com    |
| 11541907 | Neeraj    |           500 | Neeraj@DigiCash.com |
+----------+-----------+---------------+---------------------+
3 rows in set (0.00 sec)

mysql>
```

Figure 8: Amount deducted from the Account

6 DISADVATAGES

Though there are several benefits of digital cash, there are also some potential drawbacks of this system. With electronic cash, criminals would be able to perform illegal transactions anonymously and untraceably. Having to perform transactions with paper cash in person reduces the anonymity of a transaction. Moving money electronically, in small denominations to avoid suspicion would be easy compared to the paper money alternative. There is also a pressing issue regarding the technology involved in digital cash. Power failures, loss of records, and undependable software often cause a major setback in promoting the technology.

7 CONCLUSIONS

Digital cash promises to be a revolutionary method for conducting business. It will allow transactions to occur between parties on opposite sides of the globe with the same ease as going to the corner gas station to buy gas. However, this new technology will present us new challenges. Electronic cash also gives us the ability to trade in non-governmental units of currency. Electronic cash transactions have the potential to become as popular as credit card transactions. Banks or other financial institutions will then, essentially, be minting these electronic coins which will be backed by the financial stability of a corporation rather than a government. This will certainly have an impact on world economies.

ACKNOWLEDGMENTS

The work described in this paper was made possible and achievable by the contribution of Gokay Saldamli.

REFERENCES

[1] Applied Cryptography by Bruce Schneier
[2] http://positivemoney.org/publications/digital-cash/
[3] http://innovbfa.viabloga.com/files/M_Farsi__Digital_cash__1997.pdf.
[4] http://users.telenet.be/d.rijmenants/en/secretsplitting.htm.
[5] https://www.youtube.com/watch?v=6A136yz4gpc
[6] http://en.ecommercewiki.info/payment/electronic_options/digital_cash
[7] https://en.wikipedia.org/wiki/Blind_signature