Digital Cash

Guilin Wang

The School of Computer Science, University of Birmingham

1 What Is Digital Cash?

Money is important for us since we can use it to purchase the desired goods and services in a convenient and anonymous way. However, carrying paper cash may be a little annoying sometimes, since the paper cash could be stolen by thieves and become a media for bacteria. So, using the existing cryptographic tools and security measures, can we replace paper cash with digital cash? In this lecture, we will discuss this topic.

Digital cash refers to an format of electronic data that mimic the functionalities of paper cash, such as anonymity and transferability. Basically, digital cash is supposed to be stored, transferred, paid, and verified electronically via use of different digital devices (computers, mobile phones etc.) and computer networks, like the Internet.

As you know, some forms of money are already in digital formats.

- Using credit or debit cards to buy a variety of goods and services.
- Checking your bank account balance via electronic banking service.
- Transferring money between different accounts via e-banking or Electronic Funds Transfer (EFT).
- Using smart cards to take public transportation vehicles or buy drinks from vending machines etc. For example, Chipknip in the Netherland, the Octopus cards in Hong Kong, and EZ-link in Singapore.
- ...

However, all these forms of electronic money are not digital cash, because they do not meet some essential requirements for digital cash, such as anonymity, unlinkability, and/or transferability. The concept of digital cash (or e-cash) was introduced by Chaum in 1982 [2]. As the use of paper cash, the model of digital cash usually has three parties, a customer, a merchant, and the bank. According to [4, 5], digital cash are ideally expected to satisfy the following properties:

(1) **Security**: The digital cash cannot be forged and/or reused by a user illegally.
(2) **Privacy (Untraceability)**: Nobody, including the bank, cannot reveal the relationship between the identities of customers and the digital cash they used to purchase some goods and services. This property actually includes two requirements, i.e., **anonymity** and **unlinkability**. The former means neither or both of the bank and merchants cannot figure out the customers' identities, while the latter requires that nobody can find any linkage between two transactions made by the same customer.
(3) **Transferability**: The digital cash can be transferred between different customers without the involvement of the bank before the e-cash is finally stored into one customer's account.
(4) **Off-Line Payment**: When a user purchases some goods or services from a merchant, the merchant can verify the validity of the e-cash without to make any on-line enquiry with the bank.
(5) **Divisibility**: Without the bank's help, a user should be able to subdivide a piece of e-cash in a given amount into smaller pieces of e-cash.

(6) **Independence**: The security of the e-cash does not rely on any physical locations and/or special devices so that the digital cash can be easily exchanged via computer networks like any confidential electronic information.

2 On-line Digital Cash

Now, we want to know how to realize digital cash. As the first solution, you may think about letting the bank sign some data to generate a digital cash with a given amount or denomination. Such a scheme is unforgeable and independent, but linable and not anonymous (the bank knows who withdraws a particular piece of e-cash). So, we need some advanced cryptographic tools.

2.1 Blind Signatures

To achieve the privacy, Chaum invented a new primitive called **blind signatures**. Intuitively, a blind signature scheme allows a signature requestor to get the signer's digital signature without revealing the signed message to the signer. In contrast to our paper world, this looks a little weird. However, as an interesting tool blind signatures are quite useful in some applications where anonymity is crucial, such as e-voting and digital cash (We shall see the latter soon.)[1]. More specifically, the basic idea of blind signatures is something like that:

- To get Bob's signature on a message m, Alice first blinds m by computing $m' = Blind(m, r)$, where r is a random blinding factor and $Blind(\cdot, \cdot)$ is the blinding function.
- Then, the blinded message m' (instead of the original message m) is sent to Bob.
- After that, using his private signing key Bob signs m' by calculating $s' = Sign(m', sk)$, and returns s' to Alice.
- Finally, using the secret blinding factor r Alice unblinds s' to get Bob's signature s for message m. That is, such a scheme is designed so that

$$s = Unblind(Sign(Blind(m, r), sk), r) \equiv Sign(m, sk). \tag{1}$$

Blind signatures can be implemented for many signature schemes, including RSA, Schnorr, and DSA etc. As an example, we take a look for RSA, since this may be the simplest blind signature. To get Bob's RSA signature on message m without revealing m itself, Alice first sends Bob the blinded message $m' = mr^e \bmod N$ by picking a random number r as the blinding factor. Here, we suppose (e, d) are the Bob's public and private exponents, and N is the corresponding RSA modulus. Then, Bob returns his signature s' on m', i.e., $s' = (m')^d \bmod N$. Finally, Alice gets Bob's signature s on message m by unblinding s' as follows:

$$s = s'r^{-1} \bmod N. \tag{2}$$

This scheme works well since s is just Bob's normal RSA signature on m. The reason is that we have

$$s = s'r^{-1} \equiv (m')^d r^{-1} \equiv (mr^e)^d r^{-1} \equiv m^d r r^{-1} \equiv m^d \bmod N. \tag{3}$$

Think about why the above scheme satisfies the anonymity and unlinability.

[1] Zero-knowledge (ZK) proof is also a little strange when we first learn it. However, ZK proof is really an amazing and very powerful primitive in cryptography.

2.2 On-Line Digital Cash

Using blind signatures, we can realize a on-line digital cash scheme as illustrated in Figure 1 and further explained as follows. The whole scheme consists of three steps:

- **Withdraw Cash**: A customer Alice prepares a message m and gets its signature s from the Bank via a blind signature protocol. At the same time, the bank deducts the corresponding amount of money from Alice's account.
- **Payment**: When Alice wants to purchase some goods or service from an merchant Bob, she sends the e-cash (m, s) to Bob. Then, Bob can verify if s is the bank's valid signature for message m. However, this is not enough to guarantee the validity of the e-cash. Bob needs to make an on-line interaction with the bank to assure that (m, s) has never been spent by anybody (we assume the bank maintain a database to store all spent e-cash). Once this procedure is finished, Bob and Alice can finish their transaction.
- **Payment**: After the e-cash (m, s) has been checked by the bank, the corresponding amount of money is also credit into Bob's account.

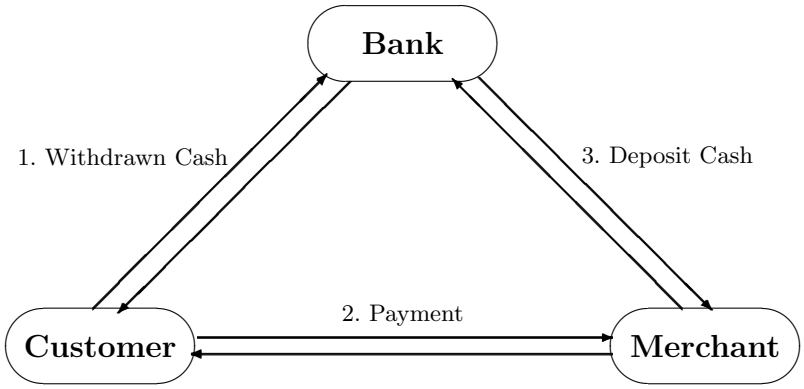

Figure 1. . The Framework of On-Line Digital Cash

The above solution meets the requirements of security, privacy, and independence, but does not satisfy other properties. In particular, the following two issues are quite important:

1. **Double Spending**: To prevent a customer from using the same piece of e-cash double or multiple times, on-line verification with the bank is necessary. However, this leads to lacking of both off-line payment and transferability.
2. **Denomination**: How to determine the denomination of a piece of e-cash? Any of the following three methods can be employed. (a) The bank uses different public keys to sign different pieces of digital cash with different denominations. (b) The customer sends the bank n items of blinded data such that all these messages specify the same amount of money. The bank randomly asks the customer to open $(n-1)$ items for checking, and signs the rest one if everything is ok. (c) A special version of blind signature, called *partially blind signatures* [1], allows the finally signed message m to explicitly include some fixed and previously known information, like amount of money, the bank's name, and the expiration date etc.

3 Off-Line Digital Cash

Can we get rid of the on-line verification with the bank in the above digital cash scheme? Without additional assumptions, this is impossible. The reason is that if Bob can accept a

digital coin, Charlie will also accept a copy of the same digital coin. So, to design a off-line digital cash system we have to introduce a trusted party, which could be a tamper-resistant device (TRD). Intuitively, you can imagine it as a smart card in a real life world. The basic idea is that once you spend or transfer a digital coin to another customer, your identity information will be partially revealed. However, if you do not double-spend this coin, your true identity will never be revealed really. In contrast, if any user does double-spend this coin, his/her identity will be recovered so that the bank could punish that user.

The model of off-line digital cash is similar to that of off-line scheme. The main difference is that when a user Alice withdraws a digital coin from the bank, her identity Alice is securely esrowed and stored together with the cash file using the idea of *secret splitting*. For example, we can use 100 pairs $A_i = (LA_i, RA_i)$ to store Alice's identity such that

$$\mathsf{Alice} = LA_i \otimes RA_i, \text{ for any } i = 1, 2, \cdots, 100. \tag{4}$$

Here, LA_i and RA_i are securely committed by Alice's TRD and each LR_i is random number. When Alice wants to transfer it to another user Bob, she needs to reveal either LA_i ro RA_i (for each i) to Bob according to Bob's random challenge. If Alice can do this properly, Bob accepts her coin. At the same time, Bob's identity will also be esrowed automatically by another 100 pairs $B_i = (LB_i, RB_i)$ such that $\mathsf{Bob} = LB_i \otimes RB_i$. To finish their transaction, a specially designed machine (like a card reader) may be needed.

The point is that once any user double-spends a particular digital coin, his/her identity will be revealed by the bank finally. So, in the essence the off-line digital cash scheme does not prevent double-spending immediately but can thwart the double-spending effecitively. It is not difficult to see that this scheme satisfies all desired properties except divisibility. However, TRDs and the related system may be costly and inconvenient, and then cause negative influence on the wide acceptance of the scheme. In [4], Okamoto and Ohta proposed a digital cash scheme with divisibility. In their scheme, the data size for a payment is about 20 kilobytes, and each transaction can be completed in several seconds.

4 Other Issues

Digital cash is an interesting concept in the security community. However, it has not taken off until nowadays. In fact, DigiCash, the pioneering company in digital cash, was declared bankruptcy in 1998. This company was founded by David Chaum in 1988, but failed to attract enough investment. There are many reasons for the unpopularity of digital cash. Here are some of them:

- The anonymity may benefit to illegal activities, such as money laundering and perfect crime.
- The use of trusted parties in off-line digital cash may be not practical in real life.
- The paper cash and credit/debit system is already quite good though it may have this or that weakness.
- ...

On the other hand, some forms of electronic money are quite successful, though they are not digital cash systems. The best example may be eBay [7], an online auction and shopping web system. In contrast to digital cash, eBay dos not meet strong privacy such as anonymity and unlinkability but provides a more secure transaction way than the credit card system. Even in 2005, eBay already had 157 million users in 34 countries and the annual profit was more than one billion US dollars.

5 Summary

In this handout, we briefly reviewed the concept of blind signatures and then discussed how to design on-line and off-line digital cash systems. We also listed a number of desired properties for such a system. Two most important issues about digital cash are anonymity and double spending.

References

1. Masayuki Abe and Tatsuaki Okamoto: Provably Secure Partially Blind Signatures. In: *Proc. of Advances in Cryptology - CRYPTO' 00*, pp. 271-286, LNCS 1880, Springer-Verlag, 2000.
2. David Chaum. Blind Signatures for Untraceable Payments. In: *Proc. of Advances in Cryptology - CRYPTO' 82*, pp. 199-203, Plenum Press, 1983.
3. David Chaum, Amos Fiat, and Moni Naor. Untraceable Electronic Cash. In: *Proc. of Advances in Cryptology - CRYPTO' 88*, pp. 319-327, LNCS 403, Springer-Verlag, 1988.
4. Tatsuaki Okamoto and Kazuo Ohta. Universal Electronic Cash. In: *Proc. of Advances in Cryptology - CRYPTO' 91*, pp. 324-337, LNCS 576, Springer-Verlag, 1991.
5. Bruce Schneier. *Applied Cryptography, Protocols, Algorithms, and Source Code in C*, Section 6.1: Digital Cash. John Wiley & Sons, Inc, 1993.
6. Digital Cash and Net Commerce. http://www2.pro-ns.net/~crypto/toc12.html.
7. Ebay. http://en.wikipedia.org/wiki/EBay
8. Electronic Money. http://en.wikipedia.org/wiki/Electronic_money.
9. EZ-Link. http://en.wikipedia.org/wiki/EZ-Link