

a simple protocol for digital cash by elliptic curves

Shuo Shen & Samuel S. Wagstaff, Jr.

Introduction

Digital cash is an important application of modern cryptology. Traditional cash spread germs, and people can steal it from the owner. Checks and credit card have an audit trail; you can't hide to whom you gave money. Digital cash can avoid all these defects. There have been a lot of excellent protocols designed for digital cash, which allows secure and untraceable transactions. In this poster, we try to create a simple and efficient protocol of digital cash by use the powerful mathematics theory of elliptic curve.

Protocol

(0) Definitions for the parameters used in the protocol:

E : an elliptic curve over a field K
 n : a integer not divisible by the characteristic of K
 $E[n]$: the set of n -torsion points in E
 g : a point in $E[n]$
 μ_n : the group of n th roots of unity in the algebraic closure of K
 $e_n : E[n] \times E[n] \longrightarrow \mu_n$: the Weil pairing defined on $E[n]$
 x, x', y, m : integers in K

(1) Parties in the protocol:

There are three different parties in this protocol: a bank, client and merchant. Their secret keys are as following:

Bank secret key: x
Client secret key: y
Merchant secret key: m

And bank has yg and mg in his database as the identify information for the client and merchant, respectively.

(2) Create the digital cash:

- [a] The client go to bank, identifies himself to the bank and tells bank the value of the coin he want to withdraw from his bank. (say, \$100)
- [b] The bank chooses a random number l of 20 digits and also a random number x' . Give the client the coin (a group of numbers): $\{l, 100, g_1=(x+x')yg, g_2=(x+x')g\}$
- [c] The bank put $\{l, 100, x'\}$ in his data base.

(3) Transaction procedure when client use the digital cash to merchant

- [d] Merchant opens the signature and make sure that this coin hasn't been spent at his store before, then he calculates $A=e_n(g_1, mg)$, $B=e_n(g_2, mg)$, and gives A to the client.
- [e] Client calculates $y'=y^{-1}$, $C=A^{y'}$ and gives C to merchant.
- [f] Merchant calculates whether $B=C$, if it does, the digital cash is valid. And merchant creates a time stamp T with the $A^{y'}$ coin and lets client sign it, get $S_c(T)$.
 $(C=A^{y'}=e_n(g_1, mg)^{y'}=e_n((x+x')yg, mg)^{y'}=e_n((x+x')yg, g)^{y'}=e_n((x+x')yg, g)=B)$

(4) Merchant takes the digital cash to bank to cash the digital cash

- [g] Merchant brings: $\{l, 100, g_2, S_c(T), mg\}$ to the bank
- [h] Bank opens the coin and checks its database. If there is no l in the database, bank give merchant \$100 and store $\{the\ signed\ coin, S_c(T), mg\}$ in his database.
- [i] If l is already in the data base and mg is same, then the merchant is cheating, catch the merchant.
- [j] If l is already in the database and mg is different then the client is cheating. Then bank have to show the merchant the name of the merchant which already cash the coin. After merchant confirm it, he give g_1 to bank. Bank calculate $(x+x')^{-1}g_1=(x+x')^{-1}(x+x')yg=yg$ to reveal the client's identity and catch the client.

Possible attack and solution:

- (1) The coin is lost or stolen
The people can't use the coin at all since the client's secret key y is needed during [e] part of the protocol.
And the client also can't pretend to lose the coin and use it later, since only he himself can use the coin.
- (2) Double spending
The client can't double spend the coin at the same merchant, since the merchant have the coin information when the coin is used first time and there is also time stamp when the coin is used. If the client double spend the coin at different merchant, he will be caught during the [j] part of the protocol
- (3) The Merchant double cashing the coin
Then the merchant will be caught during the [i] part of the protocol.
The merchant cannot create two time stamps by himself and say the innocent client want to cheat, because the time stamp has to be signed by the merchant himself.
The merchant also cannot cooperate with another merchant to cash the coin twice, because they don't have two time stamp signed by client.
- (4) Anonymity proof
As showed in the [j] part of the protocol, the bank can't get client's identity information unless the client is proved to use the coin twice, in this case, the client deserve being caught.
If and only if when the merchant and the bank cooperate, merchant gives bank g_1 , then the client's transaction activity is revealed. Fortunately this case is not likely to happen, if it does happen, even the traditional cash cannot make the transactions anonymous. So at least the digital cash is not worse than traditional cash in anonymity.

Conclusion

This simple protocol of digital cash provides more convenience than traditional cash and provides enough anonymity no less than the traditional cash.

Because the elliptic curve theorem is used in the protocol, the size of of the secret keys and other random generalized integers can be smaller since the discrete logarithm problem in the elliptic curve group E is much harder than the discrete logarithm problem in a field K .

Besides, there are only two rounds of calculations during the transactions, it is simpler than most other digital cash protocols.