

Compact E-Cash

Jan Camenisch, Susan Hohenberger, and
Anna Lysyanskaya

EuroCrypt 2005

Summary of New Scheme

- Off-line (no need to contact bank during spending)
- Anonymous
- Can identify or trace (in second variation) double spender
 - Can trace without trusted third party
 - First time tracing possible while maintaining anonymity
- Extremely small wallets
 - 2^k coins in k space
 - All algorithms $O(k)$ rather than $O(2^k)$

Background: The Basic E-Cash Idea

- Three parties
 - Bank
 - User
 - Merchant
- Setup
 - User opens non-anonymous account with Bank using real money
- Withdraw
 - User gets digital “coins” from bank, deducting from account
- Spend
 - User gives coins to Merchant

Background: The Basic E-Cash Idea

- Deposit
 - Merchant gives coins to Bank, crediting their own account
- Bank shouldn't be able to tell whose coins were redeemed!

Protocol Framework

- A scheme defines the following algorithms:
 - BKeygen
 - UKeygen
 - Withdraw
 - Spend
 - Deposit
 - Identify
 - VerifyGuilt
 - Trace*
 - VerifyOwnership*
- * = optional extension

BKeygen and UKeygen

- Just take basic parameters (group generator, etc.)
- Produce keys for Bank or User

Withdraw

- Protocol between User and Bank
- User ends up with wallet W containing some coins
- Bank debits user's account

Spend

- Protocol between User and Merchant
- User gives ZKP of valid coin
- Merchant ends up with serial number S of “spent” coin, along with prove of validity V
- User updates their wallet to remember not to spend that coin again

Deposit

- Protocol between Merchant and Bank
- Merchant sends S and V to Bank
- Bank verifies and checks whether it has received S before
 - If no, credits Merchant's account
 - If yes, Bank is able to identify User (next protocol) using V and the previous V'

Identify

- Algorithm which identifies a double spender
- Takes serial number S and two proofs of validity V and V'
- Produces public key of User and proof of guilt P

VerifyGuilt

- Algorithm to allow others (e.g., a court) to publicly confirm the guilt a double spender
- Takes S , V , V' , P
- Determines whether P is valid

Building Blocks: Discrete Log Based ZKP

- Basic ZKP for variety of discrete log things
- Knowledge of a discrete log
 - Modulo a prime
 - Modulo a composite
- Proof that a commitment opens to product of two other committed values
- Etc.
- Disjunctions and conjunctions of above

Building Blocks: Special Pseudorandom Function

- Dodis and Yamploskiy
- A pseudorandom function with special properties
 - Allows a ZKP that its output was produced from certain set of inputs (without revealing actual input)

Building Blocks: CL Signatures

- Camenisch and Lysyanskaya
- Uses Pedersen commitments
 - Just a particular number theoretic commitment function
- A signature scheme
 - Allow *efficient* ZKP of knowledge of a signature on the opening of a Pedersen commitment
 - ZKP for such signatures on openings of commitments already known, but not efficient (based on representation of everything as big circuit)

Scheme

- Here is a basic sketch of how this works
(on whiteboard)

Open Problems

- Coins of arbitrary denominations
- Tracing without funny / suspect assumptions (i.e., XDH) and while keeping highly efficient wallets