High-Speed Public-Key based Electronic Cash Using Contactless IC Cards

NTT Information Sharing Platform Laboratories
Technology Overview

- The world’s first implementation of public-key based e-cash technology on dual contact/contactless interface IC cards.
- High-speed payment processing, less than 0.4 seconds.
Why are “High-Speed” and “Contactless” needed?

- Contactless allows use just by putting cards near terminals without removing it from your purse.
- High-speed contactless has wide application area:
  - Kiosks, Convenience stores,
  - Public transports (railways, buses), etc.
- Dual interface allows the use of legacy contact IC card terminals.
Why Public-Key Based E-Cash?

- Public-key cryptography is the only solution for valuable versatile e-cash.
  - e.g. Nation-wide general purpose e-cash cards.
- Symmetric cryptography can support only less-valuable local e-cash.
  - e.g. Domestic prepaid cards in a park, or railway’s stored fare cards, etc.
Why “Contactless” not available until now?

- RSA-like public-key cryptography requires co-processor on IC cards, but:
  - Co-processor consists of complex hardware, and consumes too much power;
  - Difficult to implement on contactless IC cards.
Why was E-Cash Slow?

- Slow digital sign generation on IC card
  - Public-key cryptography is much slower than symmetric schemes.

- Slow data communication
  - Contact IC card: 9,600bps
  - Exchange large amount of data (e.g. digital signature)

- Inefficient IC card command architecture
  - Lots of interaction between terminal and IC card occurs during a transaction. (e.g. transaction management)
Approach

- Elliptic Curve Cryptography (ECDSA*) to reduce computational complexity
- Precalculation – shorten processing time of digital signature generation
- Faster data communication
  - Contactless IC card: 106Kbps (≈10 times faster)
  - Data volume of ECDSA is shorter than that of RSA
- Efficient IC card command architecture
  - Reduce interaction between terminal and IC card

*ECDSA: Elliptic Curve Digital Signature Algorithm
### RSA vs. ECDSA

<table>
<thead>
<tr>
<th>Feature</th>
<th>RSA</th>
<th>ECDSA</th>
</tr>
</thead>
<tbody>
<tr>
<td>Key length</td>
<td>1024 bits</td>
<td>160 bits</td>
</tr>
<tr>
<td>Signature data length</td>
<td>128 bytes</td>
<td>42 bytes</td>
</tr>
<tr>
<td>Basic operations</td>
<td>Power of 1024 bit-long integers</td>
<td>XOR and arithmetics of 160 bit-long integers</td>
</tr>
<tr>
<td>Computational complexity</td>
<td>High</td>
<td>Low</td>
</tr>
<tr>
<td>Cryptographic strength</td>
<td>Almost equivalent</td>
<td></td>
</tr>
</tbody>
</table>
Technique of ‘Precalculation’

- In ECDSA, some parts can be calculated independently from the subject to be signed.
- Precalculation processes finished in advance.
  - During loading e-cash into IC card, or maintenance.
- Shorten signature generation time.

Precalculation

Store enough pairs of pre-calculated data into IC card. (e.g. during loading e-cash)

Signature generation

Combine subject with one of unused pairs of data.
Latest Result

- Payment processing time: **385ms** (total)

**PC Terminal**
- 15ms
  - Terminal authentication
  - ECDSA verification
  - Transaction management

**Communication**
- 180ms
  - Communication interface
    - To IC card: 106Kbps
    - To Terminal: 115Kbps

**IC Card**
- 190ms
  - Terminal authentication
  - ECDSA signature
  - Transaction logging

- Contactless communication

- **CPU**: PenIII 800MHz
- **RAM**: 256MB
- **OS**: Windows98

- **Communication interface**
  - To IC card: 106Kbps
  - To Terminal: 115Kbps

- **Infineon SLE66CL160S**
  - CPU: 16bit, 3.39MHz with EC-accelerator
  - RAM: 1.2KB
  - EEPROM: 16KB

- **Payment processing time**: 385ms (total)
Applications

Home PCs

AtMs

POS terminals

Internet banking/shopping

Banking

Payment

Fare payment, commuter pass

Mobile banking/shopping

Public transport

Contactless

Contact
Demonstration