Electronic Payment

Prof. Bart Preneel
COSIC – KU Leuven - Belgium
Firstname.Lastname(at)esat.kuleuven.be
http://homes.esat.kuleuven.be/~preneel
December 2014

Goals

• Understand how physical payment systems can be replaced by electronic payment systems
• Understand principles behind prepaid (e.g. Proton), debit (e.g. Maestro), credit (e.g. EMV)
• Understand electronic coins and micropayments

Overview

• Traditional payment
• Principle of electronic cash
• Electronic purse
• Credit card transactions
• Micropayments
• Electronic cash: on-line
• Electronic cash: off-line

Traditional payment

• Cash
• Instruction:
  – check
  – credit card
  – debit card

Cash

• bearer instrument
• off-line payments
• low and medium value
• privacy, coins not traceable
• widely accepted

• bank: risk of forgery, cost of transport
• user: theft and loss, change, physical presence
• government: money laundering

€/$ Counterfeiting

2014
> 15 billion notes in circulation
fraudulent: 670,000 or 1 in 22,000
+/- € 800 billion genuine in 2011
new 5/10 € bill in May '13/Sep '14

1995: $15.5 million (1% digitally produced)
2005: $61 million (45% digitally produced)
Fraudulent: 1 to 2 in 10000
$1000 billion genuine in 2013

UK pound: 1 in 4170 counterfeit!
Electronic Payment

Common features $/EURO

pattern detected by scanners and copiers

Card fraud rates (Europe) 2000
Source: Lafferty Publications

<table>
<thead>
<tr>
<th>Card Type</th>
<th>Fraud Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Belgian Debit</td>
<td>0.02%</td>
</tr>
<tr>
<td>CB (France)</td>
<td>0.04%</td>
</tr>
<tr>
<td>Maestro (Europay)</td>
<td>0.06%</td>
</tr>
<tr>
<td>UK Debit</td>
<td>0.14%</td>
</tr>
<tr>
<td>Visa EU Credit</td>
<td>0.04%</td>
</tr>
<tr>
<td>Visa USA Credit</td>
<td>0.06%</td>
</tr>
<tr>
<td>Europay Credit</td>
<td>0.10%</td>
</tr>
<tr>
<td>Canada Credit</td>
<td>0.15%</td>
</tr>
<tr>
<td>UK Credit</td>
<td>0.16%</td>
</tr>
<tr>
<td>Cartes Bancaires Abroad</td>
<td>0.47%</td>
</tr>
</tbody>
</table>

US Credit Card Fraud

Source: Celent Communications via Lafferty Publications

Payment by Instruction

- Convenient
- Reduced risk
- Identify users: manual signatures, magstripe cards, smart cards
- Traceable
- Verification expensive:
  - credit/debit card: on-line, tamper resistant modules
  - check: off-line, delay, processing cost

Electronic Cash
Electronic Cash

- Convenient, no physical presence
- Reduced risk
- Cost effective for low value
- Untraceable and unlinkable
- More expensive than traceable systems, new technology
- Verification inexpensive:
  - on-line: no tamper resistant modules
  - off-line: reduced risk, doublespending

Payment
= authenticated transfer of value

- (data origin) authentication
  - symmetric: MAC
  - asymmetric: digital signature
- transfer of value: replay!
  - Prevent replay: tamper resistance, challenge response
  - Detect replay: nonce, timestamp
- risk management

Pay first: Electronic Purse Scheme

- Pay first: electronic purse (2)
  - Customer: smart card with
    - counter: value
    - MAC key
    - RSA certificate
  - Merchant: terminal
    - off-line
    - loads value from card
    - contains smart card
  - Issuer/Acquirer:
    - database for reconciliation of all transactions

Pay first: electronic purse (3)

- Load value: on-line to issuer
- Payment
  - off-line
  - check for blacklist
  - keys from both terminal and customer card
- Deposit: on-line (weekly)

- anonymity: issuer identifies user based on account number
- traceable and linkable
- relies on tamper resistance

Pay first: electronic purse (4)

- 1 layer: Banksys

Proton card: $ID^d \mod n$
With $d$ private key of Banksys
Pay first: electronic purse (5)

- National schemes: Proton, Clip, Mondex
- CEPS: Common European Purse Specification
  - standards exist, not deployed
  - relies in part on public-key cryptography
- Limitations: no card to card payment. Why?

Pay later: EMV

- Magstripe with PIN (on-line) or manual signature (off-line)
- Smart card with DES and RSA certificate
  - off-line PIN verification
  - on-line card verification above threshold (risk management)
- Smart card with RSA (dynamic)
  - needs PKI (card scheme-issuer-card)
  - off-line verification in terminal
  - on-line for high risk

Static Data Authentication based on a digital signature - initialisation

EMV: dynamic data authentication

- Three layers:
  - EPI
  - Issuers
  - Cards

Certificate for dynamic data authentication of a credit card

Dynamic Data Authentication based on a digital signature - initialisation
Credit cards today

- Magstripe/hologram/embossing (US) (no chip)
- SDA: Static Data Authentication (UK)
  - 3-DES based MAC + static RSA signature
  - vulnerable to cloning
- DDA: Dynamic Data Authentication
  - 3-DES based MAC
  - Dynamic RSA signature of random string for entity authentication
- CDA: Combined Data Authentication
  - 3-DES based MAC
  - RSA signature on random string and on payment details
  - more secure; still the issue of mafia fraud

Micropayments

- Only 1 expensive payment
  - authorisation/commitment using digital signature
- Sub-payments are cheap
  - off-line computation of hash value
- Sub-payment and deposit very small
  - hash value (100 bits)
- Lampot chain idea:

```
\[ x_0, f(x_0), f(f(x_0)), \ldots, f(x_{t-1}), x_t \]
```

Micropayments (2)

```
Customer

Generate random \( x_0 \)
Compute \( x_t = f(x_0) \)

\( x_0 \rightarrow \text{SIG}(x_0) \rightarrow \text{Authorize} t \) payments
```

```
\( x_{t-i} \rightarrow f^{-i}(x_0) \rightarrow \text{Check} f^{-i}(x_{t-i}) = x_t \)
```

x: 96..128 bit string, \( f \) one-way function

Micropayments: Micromint

- idea of Rivest and Shamir
- collision resistant hash function \( h \)
  - finding collisions is hard………………
  - unless you perform a massively parallel pre-computation
- coin = collision pair
  - (\( x, x' \)) with \( x \neq x' \) and \( h(x) = h(x') \)
- easy to check validity
- update function \( h \) on a regular basis

Micropayments: Micromint

- \( n \)-bit hash function
  - If you evaluate the hash function in \( r \) points, you expect \( r^2/2^{n+1} \) collisions if \( r << 2^n \)
  - Cost of finding 1 collision (\( r=1 \)): \( 2^{n+1} \) steps
  - Cost per collision: \( r/(r^2/2^{n+1}) = 2^{n+1}/r \)
- Example: \( n = 120, r = 2^{72} \)
  - cost of finding a single collision: \( 2^{65.5} \) steps
  - With \( r = 2^{72} \) expect \( 2^{23} \) collisions; cost for each collision is only \( 2^{23} - 2^{49} \) steps
  - So making a coin is much cheaper for the government (large scale, precomputation) than for an attacker

Micropayments: Bitcoin (2009)

- Designed by Satoshi Nakamoto
- Distributed generation and verification
- Transactions
  - irreversible
  - inexpensive
  - over anonymous peer-to-peer network
  - broadcasted within seconds and verified within 10 to 60 minutes by inclusion in hash chain
  - double spending prevention using a central database (chain mechanism)
- Pseudonymous (believed by many to be anonymous)… but
Micropayments: Bitcoin (2009)

- Bitcoins
  - can be mined by anyone but economies of scale
  - finding nonce such that
    SHA-2(SHA-2 (previous hash || transaction data || nonce))
    has a required number (d) of leading zeroes
  - proof of work: hard to compute but easy to check
  - if more solutions are found, d is increased
  - currently massive hardware investment
- hard limit of about 21 million
- divisible to 8 decimal places yielding a total of approx. $21 \times 10^{14}$ units
- system assumes that longest chain is correct chain (majority of computational power can create new "true" chain)

Micropayments: Bitcoin (ctd)

- Bitcoins
  - transferred from one public key to another using a digital signature computed with the private key of the payer
  - one user can have of course many key pairs
  - not anonymous: public keys can be clustered and many can be linked to identities
- Incidents
  - June 2012: massive devaluation
  - June 2012: Mt. Gox hacked - largest Bitcoin exchange (which trades Bitcoins for real world dollars and vice versa)
  - September 2012: Bitfloor hacked - $250,000 USD in Bitcoins inappropriately transferred to a single account
  - August 2013: bug in Random Number Generator in Java on Android results in theft of Bitcoins
  - February 2014: Mt. Gox temporarily closed

Electronic Cash: On-line

- Coin C is RSA signature:
  - $C = x^d \mod n$
  - with $x =$ encoded version of 160-bit string $x$
  - verify signature using $(e,n)$ (note $e.d = 1 \mod \lambda(n)$)
  - detect double-spending on-line
  - denominations: different values of $e$
    - $e_1=3$: 1 cent; $e_2=5$: 2 cents; $e_3=7$: 4 cents,...
  - No anonymity!

Electronic coins (1) + anonymity

- Payment message:
  - $E_{Pub\_shop}(ID_{shop} \ || \ ID_{trans} \ || \ C_1 \ || \ C_2 \ || \ldots \ || C_l)$
  - $E_{Pub\_shop}$ prevents stealing of spent coins
  - $ID_{trans}$ random transaction identifier
- Payer is untraceable
- Coins of payer are unlinkable
- Payee is NOT anonymous: allows for some audit
Electronic Payment

Electronic Cash: Off-line

- 1 Coins in envelope
- 2 Signed coins in envelope
- 3 Coins
- 4 Goods/Services

Customer

Bank

Merchant

Electronic Cash: Off-line (2)
Doublespending!!!

- 3 Coins
- 4 Goods
- 5 Deposit

Customer

Bank

Merchant 1

Merchant 2

Prevent doublespending through tamper resistance

Doublespending detection “after the fact”

Electronic Cash: Offline (3)

- Doublespending detection after the fact requires more sophisticated blinding protocols (restrictive blinding, Brands93)
  - One payment allows user stay anonymous, but identity leaks after 2 payments

Extensions

- revokable cash
- divisable coins
- fault and loss tolerance
- anonymous fingerprinting
- unlinkable credentials:
  - one can show that one is 18 years old, without revealing one’s identity

More information and some links

- www.visa.com: Travelmoney
- www.mastercard.com
- D. Chaum, S. Brands, Minting electronic cash, IEEE Spectrum, February 1997 (introductory article)
- P. Wayner, Digital cash: Commerce on the net, Morgan Kaufmann, 1997
- D. O’Mahony, M. Peirce, H. Tewari, Electronic payment systems, Artech House, 1997