4 Electronic Payment Systems

Literature:

Donal O'Mahony, Michael Peirce, Hitesh Tewari: Electronic Payment Systems for E-Commerce, 2nd ed., Artech House 2001

A Brief History of Cash Money

- Direct exchange of goods
  - Problematic since "double coincidence of wants" is required
- Commodity payment
  - Exchange with goods of well-known value (e.g. corn, salt, gold)
  - Leading to gold and silver coins
- Commodity standard
  - Tokens (e.g. paper notes) which are backed by deposits of the issuer
- *Fiat* money
  - Assuming a highly stable economy and government
  - Tokens no longer (or not fully) backed by deposits
  - Trust in the issuer replaces deposits
- Cash is used for 80% of all financial transactions
  - Cash is not free of transaction costs!
  - Replacement of coins/notes paid out of taxes

Forms of Payment

- Cash
- Cheques
  - Using “clearing house” between banks
- Giro, direct credit transfer *(Überweisung),* direct debit *(Lastschrift)*
  - Requires “clearing house”, today fully automated (“Automated Clearing House ACH”)
- Wire transfer
- Payment cards (cost usually borne by the merchant):
  - Credit card
    - » Associated with credit promise from bank
  - Charge card
    - » Requires full settlement of bill each month
  - Debit card
    - » Card used to initiate an immediate direct debit

Customer Preferences in Non-Cash Payment

- According to the *Bank for International Settlements*, www.bis.org
- Figures for 1998

Country	Cheques	Credit Transfer	Payment Cards	Direct Debit
USA	70 %	3.7 %	24.3 %	2.0 %
Netherlands	1.9 %	45 %	24.5 %	28.5 %
UK	28 %	19.3 %	33.1 %	19.4 %
Germany	4.8 %	50.6 %	5.1 %	39.5 %
Turkey	6.9 %	2.6 %	83.9 %	--

Customer and Merchant Preferences in German E-Commerce

- Study by Fittkau & Maaß (according to *Computerzeitung* 44, Oct 2004):
  - Used payment methods in German online commerce
  - Multiple assignment of preferences was possible
- Used payment methods:
  - 75 % traditional billing (payment after delivery, usually by credit transfer)
  - 42 % direct debit (Lastschrift)
  - 37 % credit card
  - 8 % electronic payment methods
- Electronic payment is still not well known and therefore rarely used
  - Payments for small amounts (less than € 2) are problematic

4 Electronic Payment Systems

Literature:

Donal O'Mahony, Michael Peirce, Hitesh Tewari: Electronic Payment Systems for E-Commerce, 2nd ed., Artech House 2001

Credit Card MOTO Transactions

- MOTO = Mail Order/Telephone Order
- Transactions without physical co-location of buyer and merchant
- Special rules:
  - Additional information
    - » Address
    - » Card security code
  - Often: Matching of delivery address and credit card billing address
- Extremely popular form of online payment
  - Data transfer secured by SSL, i.e. hybrid symmetric/asymmetric cryptosystem
- Disadvantages:
  - Many possibilities for fraud
  - Anonymity of customer not possible
  - High transaction cost – difficult for small amounts

SET

- SET = Secure Electronic Transactions
  - Standard by Visa and MasterCard 1996
  - Today almost without significance (after attempt to revive it in 1999)
  - But still a model for a thorough way to deal with the problem
- Scope restricted to authorization of credit card payments
  - No actual funds transfer
- Focus on trust model and authorization
  - Using public/private key cryptosystem
- Complex (three volumes specification)
  - But safe against all major risks
- Special PKI: All participants have to obtain (X.509) certificates
  - “Brand Certification Authority” (MasterCard/Visa)
  - Geopolitical Authority (optional)
  - Cardholder/Merchant/Payment CA

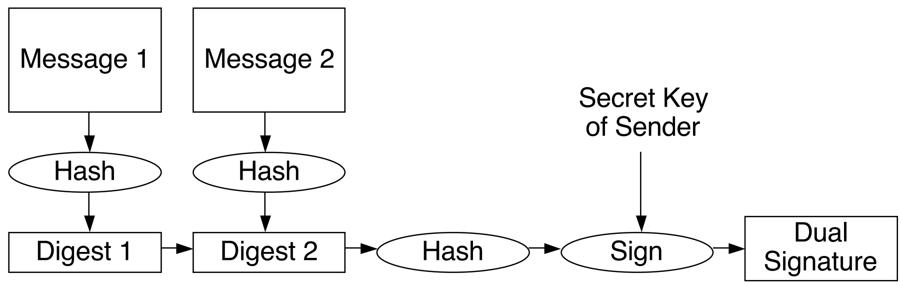
SET Initialization

- Initialization (PInitReq):
  - Cardholder to Merchant
  - Contains: Brand of card, list of certificates, “challenge” (to ensure freshness)
- Initialization Response (PInitRes):
  - Merchant to Cardholder
  - Contains: Transaction ID, response to challenge, certificates, “merchant challenge”

- Roles:
  - Cardholder (Buyer)
  - Merchant (Seller)
  - “Acquirer” (essentially credit card organization)
    - » Operating a “payment gateway”

Dual Signatures

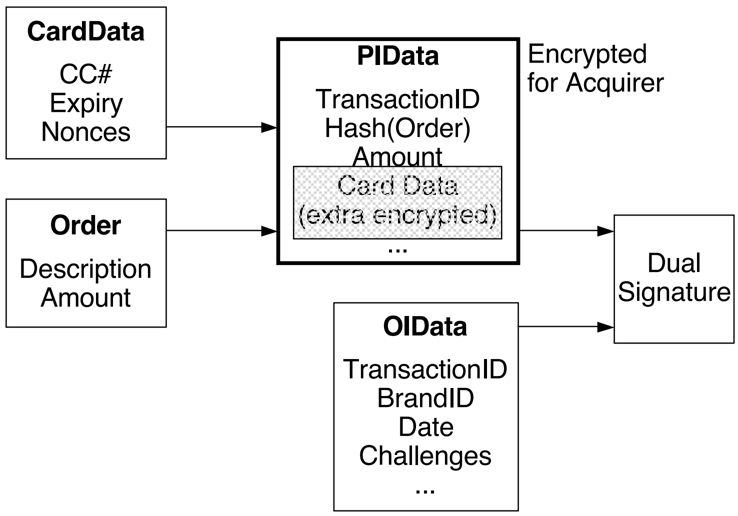
- General concept:
- Alice wants to send Message 1 to Bob and Message 2 to Carol, and she wants to assure Bob and Carol that the respective other message exists
  - To Bob she sends Message 1 and Digest 2
  - To Carol she sends Message 2 and Digest 1



SET Purchase

- Purchase Order (PReq):
  - Cardholder to Merchant
  - Order Information (OI):
    - » Identifies order description at the merchant
    - » Contains response to merchant challenge
    - » Includes random information ("nonce") for protection against dictionary attacks
  - Payment instructions (PI):
    - » Card data, purchase amount, hash of order, transaction ID
    - » Payment instructions are *encrypted* with acquirer's public key (merchant cannot read it)
    - » "Extra strong" encryption by using RSA (and not DES, for instance)
  - Dual signature for OI going to Merchant and PI going to Acquirer

SET Purchase Request Data



SET Authorization

- Authorization Request (AuthReq)
  - Merchant to Acquirer
  - Encrypted with Acquirer's public key
  - Signed with Merchant's secret key
- Contains: TransactionID, amount, Hash(Order), Hash(OIData), PIData, merchant details, cardholder billing address
  - Hash(Order) contained twice
    - » from merchant directly
    - » as part of PIData (encrypted, e.g. just forwarded from cardholder)
  - Can be used to verify that cardholder and merchant have agreed on order details
- Authorization Response (AuthRes)
  - Acquirer to Merchant
  - Contains: TransactionID, authorization code, amount, data, capture token (to be used for actual funds transfer)

4 Electronic Payment Systems

4.1 Traditional Payment Systems

4.2 Credit-Card Based Payment Standards

4.3 Electronic Cash and Micropayments

4.4 Practice of E-Payment

Literature:

Donal O'Mahony, Michael Peirce, Hitesh Tewari: Electronic Payment Systems for E-Commerce, 2nd ed., Artech House 2001

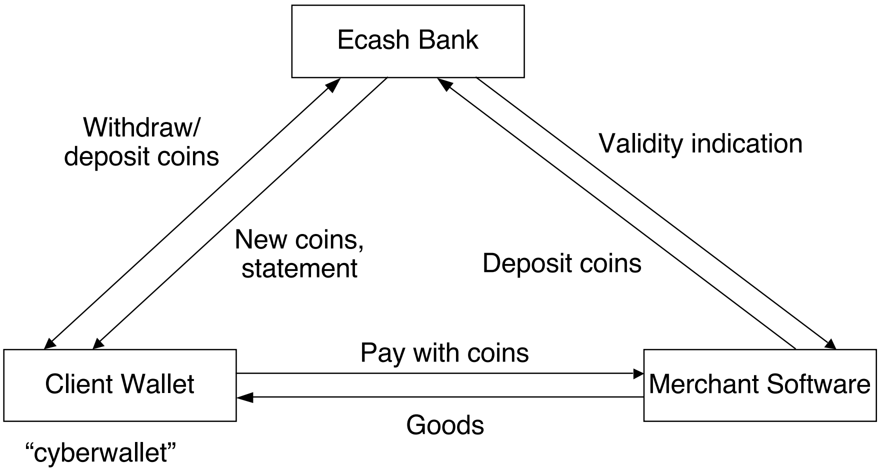
Electronic Cash

- Many attempts have been made to transfer the advantages of cash money to digital transactions:
  - Acceptability independent of transaction amount
  - Guaranteed payment – no risk of later cancellation
  - No transaction charges
    - » no authorization, no respective communications traffic
  - Anonymity
- There does not exist an electronic system which captures all of the above attributes!
  - But there are interesting approximations...

DigiCash / Ecash

- DigiCash (David Chaum)
  - Dutch/U.S. company, 1992
- Ecash
  - Electronic equivalent of cash, developed by DigiCash
  - Fully anonymous using cryptographic techniques
- History:
  - 1995: Mark Twain Bank, Missouri, started issuing real Ecash dollar coins
  - 1998: DigiCash bankruptcy
  - Relaunch as "eCash Technologies"
  - 2002: eCash Technologies taken over by InfoSpace
    - » Mainly to acquire valuable patents
- Ecash still an interesting model for electronic cash

Ecash Model



Minting Electronic Coins

- Each coin has a serial number
  - Serial number is generated by a client's "cyberwallet" software
  - Randomly chosen, large enough to avoid frequent duplicates (e.g. 100 bits)
- Coins, respectively their serial numbers, are signed by the bank
  - Bank does not know the serial number through "blinding" (see next slide)
  - Bank is not able to trace which coins are given to which person
- Bank uses different keys for different coin values
  - E.g. 5-cent, 10-cent, 50-cent signatures
- Contents of an electronic coin:
  - Serial number SN
  - Key version (can be used to obtain value, currency, expiry date)
  - Signature: F(SN), encrypted with one of the bank's secret keys
    - » Where F computes a hash code of SN and adds some redundant information – to avoid forging of coins

Blinding

- General concept:
- Alice wants Bob to sign a message without Bob seeing the content.
- Analogy: Envelope with message and a sheet of carbon paper
  - Signature on the outside of the envelope goes through to the contained message
- Procedure:
  - Blinding achieved by multiplication with random value *(blinding factor)*
  - Alice sends multiplied (blinded) message $B(M)$ to Bob
  - Bob signs blinded message: $Sign_{Bob}(B(M))$
  - Signature function and blinding (multiplication) are *commutative:*
    - » $Sign_X(B(M)) = B(Sign_X(M))$
  - Alice de-blinds message (by division with blinding factor)
  - The resulting message is $Sign_{Bob}(M)$, indistinguishable from a message directly signed by Bob

Avoiding Forged Coins

- Assuming the function F was omitted
  - Coin contains serial number *SN* in plaintext
  - Signature is just $SK_{\$1}(SN)$
- Forging a coin:
  - Choose a large random number *R*
  - Encrypt *R* with bank's $1 public key: $S = PK_{\$1}(R)$
  - Construct coins which contain *S* as serial number and *R* as signature
  - Now the coin can be verified (not distinguishable from real coin):

$$SK_{\$1}(S) = SK_{\$1}(PK_{\$1}(R)) = R$$

  - Therefore introduction of function F in coin definition

Avoiding Double Spending

- E-Coins are just pieces of data which can be copied
  - How to avoid that the same coin is spent several times?
- Ecash solution:
  - Central database of *spent coins*
  - Merchants must have an online connection with the Ecash bank
  - Before accepting a coin: check whether it has been spent already
- Problem:
  - Database of spent coins can become a performance bottleneck
  - Offline trade with coins is impossible

An Ecash Purchase

- Client has Ecash coins stored in his cyberwallet
- Merchant receives an order from the client
- Merchant sends a *payment request* to the client's cyberwallet
  - Amount, timestamp, order description, ...
- User is asked whether he/she wants to pay
- Coins for the (exact) amount are taken from wallet
  - There is no change with Ecash
  - Otherwise the merchant could record the serial numbers of his coins given to the client and try to identify the client
- Coins are encrypted with bank's public key when sent to merchant
  - Merchant just forwards them but cannot read anything
- To prove the payment:
  - Client generates a secret and includes (a hash of) it into the payment info.

The Perfect Crime

Bruce Schneier:

- An anonymous kidnapper takes a hostage.
- Kidnapper prepares a large number of blinded coins and sends them to the bank as a ransom demand.
- Bank signs the coins to save the hostage.
- Kidnapper demands that the signed coins are published, e.g. in newspaper or television. Pickup cannot be traced. Nobody else can unblind the coins but the kidnapper.
- Kidnapper saves the blinded coins to his computer, unblinds them, and has a fortune in anonymous digital cash
- Hopefully, kidnapper releases the hostage...

Off-Line Coins

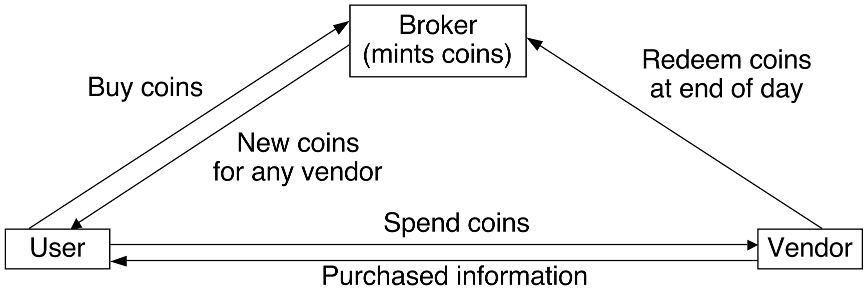
- Chaum/Pedersen 1992, Stefan Brands 1993:
  - Coins may consist of several parts
  - To use a coin in a payment transaction, one part of the coin must be revealed. Payer is not identified.
  - If the coin is used a second time, a second part of the coin is revealed – and the payer is identified.
  - This way, it is possible to trace double spendings after the fact, and to identify the origin of the double-spent coins.
- Algorithmic idea:
  - Identity *I* of user is encrypted with one-time random number *P*
    - » Is part of coin
  - Special *challenge-response* system: Merchant asks client for answer on a random challenge and stores the results
  - As soon as the merchant has *two* results for different challenges, he can calculate the information required to decrypt the identity of the payer

Macropayments and Micropayments

- Systems described above were designed for “macropayments”
  - Minimum granularity 1 cent (penny, etc)
- Prices for services often quoted in smaller quantities
  - See petrol prices...
  - Hundredth or thousandth of cent
- Micropayment:
  - Payment technology suitable for very small amounts
- Problem:
  - Transaction overhead from macropayment systems larger than value
- Advantage:
  - Losing an electronic micro-coin is not a serious damage
- Light-weight, fast, scalable protocols
- Historic pioneer: **Millicent** project (1995)
  - Digital Equipment Corporation (now part of Compaq, part of HP)
  - Key innovations: *Brokers* intermediating between vendors and *scrip* (digital cash valid only for a specific vendor)

MicroMint

- Developed by Ron Rivest and Adi Shamir (1996) (similar: *PayWord*)
- Idea:
  - Signing of e-coins by bank is computationally too expensive
  - Make it computationally difficult for everybody else but a broker to mint valid coins
  - Make it quick and efficient for everybody to verify a coin
  - No check for double spending

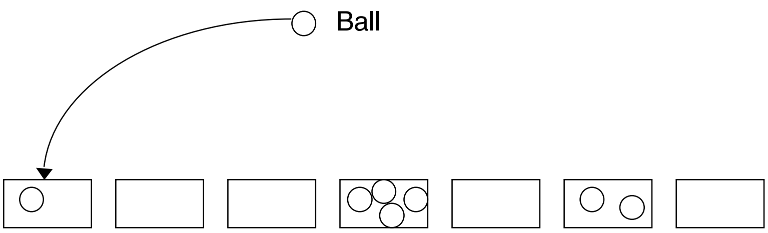


k-Way Hash Collisions

- MicroMint coin is a *k-way hash collision function*
- One-way hash function:
  $H(x) = y$
- Hash function collision:
  $H(x_1) = H(x_2) = y$
  – It is computationally hard to generate two values that map to the same value
- *k*-way hash function collision:
  – *k* different input values map to the same output value
- MicroMint coin (4-way hash collision):
  $C = [x_1, x_2, x_3, x_4]$ such that the hash function gives the same value for all x_i
- Verifying a MicroMint coin:
  – Just check the hash function value for the four given values

Minting MicroMint Coins

- Length of *x* and *y* values restricted to a fixed number of bits
  – Assuming y values are *n* bits long
- Analogy: Throwing balls at 2^n bins
  – "Balls" generated at random
  – "Bins" represent *y* values
- Successfully minted coin:
  – 4 balls in one bin
- Difficult to mint first coin, further coins much quicker



Preventing Forgery with MicroMint

- Special hardware:
  - Broker can gain speed advantage over attackers
- Short coin validity period:
  - Coins do not live more than a month
- Early minting:
  - Coins are minted a month or more before distribution – speed advantage
- Coin validity criterion:
  - May be changed every month, e.g. the used hash function
- Different bins:
  - Broker may remember the unused bins for the month and use them to detect forged coins
- ...

4 Electronic Payment Systems

Payment Service Providers

- Nowadays, many users apparently have learned to trust encrypted transmission over the Internet
  – Problem: Confidential data (e.g. credit card number, bank account) still known to merchant
- Solutions:
  – Build up high-trust merchant brands (e.g. Amazon)
  – Use independent third parties as *payment service provider*
    » Examples: FirstGate, PayPal
- Payment service provider:
  – Establishes account with user, keeps confidential data away from merchant
  – Provides easy tools for merchants to integrate payment functions into Wen shops
  – Accumulates small payments to monthly bills

Banner Advertising

- Advertising is often used as a form of payment on the Web
- Information services on the Web can be financed by advertising income
- Typical billing schemes for advertisers:
  – *Page impression:* Banner is put one time in front of a Web user
  – CPM: Cost per thousand (Roman 1.000 sign) page impressions
    » Typical cost range € 0.50 up to € 40 (figures from 2001)
  – CPC: Cost per click
    » Typical cost range € 0.10 up to € 1 (figures from 2001)

Mobile Network Based Payment Systems

- Example PayBox (www.paybox.net)
  - Registration with Payment Service Provider (paybox) – Customer obtains PIN
  - Payment request in E-Commerce or M-Commerce applications
  - Payment Service Provider calls back on mobile phone
  - Customer confirms payment by entering PIN
  - Confirmation by email/SMS
  - Mobile phone bill is *not* used for money transfer
- Add-on services:
  - Online credit transfer
  - User-to-user credit transfer via mobile phone
- Paybox company in Germany: Business closed 2003
  - Some success in Austria
  - www.payboxsolutions.com

Payment through Phone Bill

- Example T-Pay (Deutsche Telekom)
  - Billing data of phone bills are kept up to date
  - No additional bill for customer
  - Suitable for small amounts