Outline

- Micropayment systems
  - Make small purchase over the Internet
- Two simple micropayment schemes
  - PayWord
  - MicroMint

PayWord and MicroMint

- Main goal
  - Minimize the number of public key operations
  - Use hash operations instead whenever possible
    - Hash functions are
      - 100 times faster than RSA signature verification
      - 10,000 times faster than RSA signature generation
PayWord

• Overview
  – Credit based scheme
  – Based on chains of paywords (hash values)
  – Broker gives a certificate to user to allow him/her to make paywords
  – User authenticates a complete chain to the vendor with a single public-key signature
  – User successively reveals each payword in the chain to make micropayment
  – Vendor gets money through broker.

PayWord (Cont’d)

• User-Broker relationship
  – User U establishes an account with broker B
    • Credit card number, expiration date, etc.
  – Broker B gives user U a certificate
    • Expiration date
    • Credit limit per vendor
    • Contact information of broker B
    • …
    – The certificate:
      • B will redeem authentic paywords produced by U turned in before the given expiration date.
      • Essentially allows U to produce paywords.

PayWord (Cont’d)

• User-Vendor relationships
  – Randomly choose \( w_0 \) and compute the paywords
  – User U sends Vendor V her commitment
    \[ M = \{ V, C_U, w_0, D, I_M \}_{SK_U} \]
    – Commitment is vendor-specific and user-specific

\( h \) : one-way hash function

\[
\begin{align*}
W_0 &\leftarrow W_i \\
W_j &\leftarrow W_i \\
W_k &\leftarrow W_i \\
W_{l} &\leftarrow W_i \\
\end{align*}
\]
PayWord (Cont’d)

• Payment
  – A payment $P$ from $U$ to $V$
  – $P = (w_i, i)$
  – $U$ spends her paywords in order
  – Variable-size payment
    • Example: $U$ has just paid $(w_3, 3)$. What should $U$ send to $V$ if she wants to pay 3 more cents?
    • $(____, _____)$

PayWord (Cont’d)

• Vendor-Broker relationship
  – For each User $U$, Vendor $V$ needs to send Broker $B$
    • The commitment $M$
    • The last payment $P = (w_l, l)$ received from $U$
  – Broker verifies $M$ and each payment $P = (w_l, l)$
  – Questions:
    • What’s the cost of verifying $P = (w_l, l)$?
    • What property(ies) of the hash function is used in PayWord?

MicroMint

• Overview
  – No public key operations
  – For unrelated low-value payments
  – Broker produces MicroMint coins
    • A coin is a bit string whose validity can be checked by anyone
  – Users purchase the coins
  – Users give the coins to vendors as payments
  – Vendors return coins to broker in turn for payments by other means.
MicroMint (Cont’d)

- Coins
  - Each coin is represented by a $k$-way collision that has distinct $x_i$’s.
  - The number of $x$-values that must be examined before one expects to see the first $k$-way collision is approximately
    \[ 2^{n(k-1)/k}, \]
    where $n$ is the number of bits in $y$.

- Minting coins
  - Equivalent to throwing balls into $2^n$ bins
    - Randomly select $x$, and compute $y=h(x)$.
    - Throw approximately $k*2^n$ balls
      - Roughly $1/2$ of the bins have at least $k$ balls.

- Question: If there are more than $k$ $x$’s in the same bin, can we make more than one coin out of it?
  - __________________

- Balance computational and storage requirements
  - Good coins: a coin is good only when the high-order $t$ bits are equal to a given value.
  - Reduce the storage requirements
  - Slow down the generation process
    - Tosses $k*2^n$ balls, but get $(1/2)*2^{n-t}$ coins.
MicroMint (Cont’d)

- Selling coins
  - Broker B remembers what coins User U gets
- Making payments
  - Vendor V can verify each coin
- Redemption
  - Vendor returns the coins to the broker
  - Broker checks coins and pays the vendor
    - Only pay for coins that have not been previously returned.

MicroMint (Cont’d)

- Double spending
  - Broker can detect doubly-spent coin
  - Broker can identify from which vendors he received such coins
  - Broker can link the doubly-spent coins with each user