A New Tree-Structure-Specified Multisignature Scheme for a Document Circulation System

Masaki Inamura\textsuperscript{1,2}, Kei’ichi Iwamura\textsuperscript{1}, Ryu Watanabe\textsuperscript{2}, Makoto Nishikawa\textsuperscript{2}, Toshiaki Tanaka\textsuperscript{2}.

1. Tokyo University of Science
2. KDDI R&D Laboratories Inc.

July 18, 2011.
Background & Goal

The Need for a Multisignature Scheme.

  - For Verification of Browsing a Document.
- An Agreement System for a Draft Proposal in Business.
  - For Certification as an agreement.

The Need for a Hierarchy-Specified Multisignature Scheme.

  - For Distinction between One Group and Others.
- An Agreement System for a Draft Proposal in Business.
  - For Difference in Rights & Duties of Employment Position.

We need a new scheme verifying not only who signs but also which division each of signers belongs to.
3

Strong Point of Our Scheme

1. Our scheme can run with parallel processing, so signing/verification time is very short.
2. Data-Size of multisignature is a constant or below, because of calculation on a elliptic curve.

* 📄: Signing Target (Doc. for Circulating). 🔒: Signature.

(c) Tokyo University of Science / KDDI R&D Labs. Inc. / KDDI Corp. 2011
Our Proposal (1) -Preparation-

Order-Specified Multisignature *Mention of Only Signing & Aggregation.

\( m = \text{Plain Text}, \ h = H(m), \ x_k = \text{Private Key}, \ v_k = \text{Public Key.} \)

\[
\begin{align*}
\text{Order } o_{\text{last}} & : & x_{o_{\text{last}}}, v_{o_{\text{last}}} &= x_{o_{\text{last}}}g & \sigma_{o_{\text{last}}} &= x_{o_{\text{last}}}h & S_{o_{\text{last}}} &= S_{o_{\text{last}}^{-1}} + (x_{o_{\text{last}}^{-1}}) \sigma_{o_{\text{last}}^{-1}} + \sigma_{o_{\text{last}}}
\end{align*}
\]

\[
\begin{align*}
\text{Order } o & : & x_o, v_o &= x_og & \sigma_o &= x_oh & S_o &= S_{o^{-1}} + (x_o^{-1}) \sigma_{o^{-1}} + \sigma_o
\end{align*}
\]

\[
\begin{align*}
\text{Order } 2 & : & x_2, v_2 &= x_2g & \sigma_2 &= x_2h & S_2 &= x_2 \sigma_1 + \sigma_2 & V_2 &= x_2v_1 + v_2
\end{align*}
\]

\[
\begin{align*}
\text{Order } 1 & : & x_1, v_1 &= x_1g & \sigma_1 &= x_1h & S_1 &= \sigma_1 & V_1 &= v_1
\end{align*}
\]
Our Proposal (2) -Proposition-

Browsing Verification System (Binary Tree-Structure, 3 Layers)

* Mention of Only Signing & Aggregation.

\[(m=\text{Doc. for Circulating}, \ h=H(m), \ x_{k,l} \text{ or } x_t=\text{Private Key}, \ v_{k,l} \text{ or } v_t=\text{Public Key.})\]

The Top Manager
(1 user)

\[\sigma_t = x_t h \]
\[S_t = S_{i,1} + S_{i,2} + (x_t - 1) \sigma_{i,1} + (x_t - 1) \sigma_{i,2} + \sigma_t \]
\[V_t = V_{i,1} + V_{i,2} + (x_t - 1)v_{i,1} + (x_t - 1)v_{i,2} + v_t \]

The Middle Manager
(2 users)

\[\sigma_{i,1} = x_{i,1} h \]
\[S_{i,1} = S_{b,1} + S_{b,2} + (x_{i,1} - 1) \sigma_{b,1} + (x_{i,1} - 1) \sigma_{b,2} + \sigma_{i,1} \]
\[V_{i,1} = V_{b,1} + V_{b,2} + (x_{i,1} - 1)v_{b,1} + (x_{i,1} - 1)v_{b,2} + v_{i,1} \]

The Laborer
(2 Pairs [4 users])

\[\sigma_{b,1} = x_{b,1} h \]
\[S_{b,1} = \sigma_{b,1} \]
\[V_{b,1} = v_{b,1} \]
\[\sigma_{b,2} \]
\[S_{b,2} \]
\[V_{b,2} \]
\[x_{b,2}, v_{b,2} \]
\[\sigma_{b,3} \]
\[S_{b,3} \]
\[V_{b,3} \]
\[x_{b,3}, v_{b,3} \]
\[\sigma_{b,4} \]
\[S_{b,4} \]
\[V_{b,4} \]
\[x_{b,4}, v_{b,4} \]

(c) Tokyo University of Science / KDDI R&D Labs. Inc. / KDDI Corp. 2011
Security Analysis

- **Order-Specified Legitimacy**
  - $V_{\text{order}}$ (or $V_t$)
    Forgeability of $V_{\text{order}}$ (or $V_t$) is equal to Solution of the CDH Problem in complexity.

- **Unforgeability**
  - $S_{\text{order}}$ (or $S_t$)
    Forgeability of $S_{\text{order}}$ (or $S_t$) is equal to Forgeability of the BLS signature in complexity.

- **Collusion-secure**
  - $x_o$ (or $x_{k,l} / x_t$)
    Each private key is an independent prime number, so $x_o1x_o2 \neq x_o3x_o4$.

- **Impossibility of Deceptive Participation**
  - $x_o$ (or $x_{k,l} / x_t$)
    The optional private key without himself/herself cannot be obtained because of the discrete logarithm problem.

⇒ Our scheme is secure with the random oracle.
Performance Evaluation

- Supposing Structure of Signers in the Simulation
  - Binary Tree with 9 Layers (511 Signers).

- Computer Spec.
  - CPU: Intel Core i7 870.
  - Memory Size: 3.24GB.
  - OS: Microsoft Windows7 Ultimate (32bits).

- Security Functions
  - Pairing Function: Tate Pairing.
  - Hash Function: \textit{MapToGroup} with SHA-256.

<table>
<thead>
<tr>
<th>Signing at each signer.</th>
<th>Verification by verifier.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Max 0.125[sec]</td>
<td>3.838[sec]</td>
</tr>
</tbody>
</table>
## Comparison with Existing Schemes

<table>
<thead>
<tr>
<th></th>
<th>Existing Order-Specified Multisignature Scheme. (e.g. T03)</th>
<th>Our Tree-Structure-Specified Multisignature Scheme.</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Data-Size</strong></td>
<td>$u_i s$ (Proportion to $u_l$.)</td>
<td>$2s$ (The Constant $S_\alpha$ &amp; $V_\alpha$.)</td>
</tr>
<tr>
<td><strong>Processing</strong></td>
<td><strong>Sign</strong> Sequential.</td>
<td><strong>Parallel.</strong></td>
</tr>
<tr>
<td></td>
<td><strong>Verify</strong> Sequential.</td>
<td><strong>Parallel.</strong></td>
</tr>
</tbody>
</table>

* Symbols

- $s$: The Data-Size of Single Signature.
- $u_l$: The Number of Signers at the Leaf.

Our scheme is more efficient than other existing schemes.
App. 1

The BLS Signature Scheme (1)

A Digital Signature Scheme Based on the GDH Group.

The GDH Group in Elliptic Curves.

$\mathbb{G}$: A Group on Elliptic Curves, $g$: An Element in $\mathbb{G}$, $e$: Pairing.

The CDH Problem:

For $a, b \in \mathbb{Z}_p^*$, given $g, ag, bg$, compute $abg$.

The CDH problem is hard to solve.

The DDH Problem:

For $a, b, c \in \mathbb{Z}_p^*$, given $g, ag, bg, cg$, decide $c = ab$.

The DDH problem is easy to solve
with comparison of $e(g, cg)$ to $e(ag, bg)$.

Then it is defined that $\mathbb{G}$ is the GDH group.
The BLS Signature Scheme.

$G$ : A GDH Group, $g \in G$.

$D$ : An Algorithm to solve the DDH Problem, $(a,b,c,d) \rightarrow \text{True or False } (a,b,c,d \in G)$.

$H$ : A One-way Function onto $G$, $\{0,1\}^* \rightarrow G$.

Key Generation: Private Key: $x \in \mathbb{Z}_p^*$, Public Key: $v := xg$.

Signing: $m \in \{0,1\}^* \Rightarrow h := H(m) \Rightarrow$ Signature: $\sigma := xh$.

Verification: $h' := H(m) \Rightarrow D(g, v, h', \sigma)$.

The BLS signature scheme is realizable with pairing.

(2001, D.Boneh et al.)
App. 3  The BLS Signature Scheme (3)

The Multisignature Scheme.

$G$ : A GDH Group, $g \in G$.

$D$ : An Algorithm to solve the DDH Problem,
(a,b,c,d) → True or False (a,b,c,d ∈ $G$).

$H$ : A One-way Function onto $G$, $\{0,1\}^* \rightarrow G$.

Key Generation: Private Key: $x_i \in \mathbb{Z}_p$, Public Key: $v_i := x_i g$.

Signing: $m \in \{0,1\}^* \Rightarrow h := H(m) \Rightarrow$ Signature: $\sigma := x_i h$.

Aggregation: $\sigma := \sum_{i=1}^n \sigma_i$.

Verification: $h' := H(m) \Rightarrow v := \sum_{i=1}^n v_i \Rightarrow D(g, v, h', \sigma)$.

A proposed scheme aggregates signatures by means of calculation on a elliptic curve.

A multisignature is as small as a single user’s signature.

Multisignature = $\sigma \times \alpha \times (\text{Signer}_1) + \beta \times (\text{Signer}_2) + \gamma \times (\text{Signer}_3)$

Aggregating on an elliptic curve.

Expression of a Multisignature by the Coordinates.
A signature specified order is introduced.

(1) Single Signature.
\[ \sigma_{↓↓1} = \text{file} \times \text{secret key} \]

(2) Relation between Two Members.
\[ \sigma_{↑-↓1} = \sigma_{↑-↓1} \times \sigma_{↓} \]

(3) Expansion for Plural.
\[ \sigma_{↑-↓1,↓2} = \sigma_{↑-↓1} \times \sigma_{↑-↓2} \times \sigma_{↓} \]

Any tree structures can be explained with continual the above method.