

基于离散对数上的多级多代理签名方案

杨迎辉,任俊峰

(河南理工大学 数学与信息科学学院,河南 焦作 454000)

摘要:通过对代理多重数字签名、多重代理数字签名和多级代理数字签名方案的深入研究,结合三者的思想,提出了一个基于离散对数上的多级多代理数字签名方案,前三者均可以看作该签名的特殊形式,即若 $A_{ij}(1 \leq j \leq n_i)$ 是 $A_{i-1,j}$ 信任的第 i 级代理签名人,当 $i = 1, n_0 > 1, n_1 = 1$ 时该签名就是代理多重签名;当 $i = 1, n_0 = 1, n_1 > 1$ 时该签名就是多重代理签名;当 $i > 1, n_0 = 1, \dots, n_i = 1$ 是该签名就是多级代理签名。本方案不仅具有前三者签名类型的优点,而且更具有—般性,应用范围也更加广泛。

关键词:多级多代理;离散对数;数字签名

中图分类号:TP309.3

文献标识码:A

文章编号:1673-629X(2010)12-0181-04

Multi-Level Multi-Proxy Signature Scheme Based on Discrete Logarithms

YANG Ying-hui, REN Jun-feng

(School of Mathematics and Information Science, Henan Polytechnic University, Jiaozuo 454000, China)

Abstract: Researched on the proxy multi-signature, multi-proxy signature and multi-level proxy signature schemes, and according to the former thoughts, proposed a multi-level multi-proxy signature scheme based on discrete logarithms, the formers can be considered a special form of the proposed scheme, in other words, proxy signer $A_{ij}(1 \leq j \leq n_i)$ is $A_{i-1,j}$'s trusted i -level proxy signer, if $i = 1, n_0 > 1, n_1 = 1$, the proposed scheme is a proxy multi-signature scheme; if $i = 1, n_0 = 1, n_1 > 1$, it is a multi-proxy signature scheme; if $i > 1, n_0 = 1, \dots, n_i = 1$, it is a multi-level proxy signature scheme, therefore, it has the advantages of the formers and is more extensive in the application.

Key words: multi-level multi-proxy; discrete logarithms; digital signature

0 引言

代理签名是一种特殊的数字签名,它是将原始签名人的签名权委托给代理签名人,让代理签名人代表原始签名人去行使签名权。一个代理签名方案须满足以下基本性质:可区别性;不可伪造性;可验证性;可识别性;不可否认性。

自1996年 Mambo、Usuda 和 Okamoto^[1]首次提出了代理签名的概念以来,人们对它进行了广泛的研究^[2~12],近来,伊丽江^[2]等人与祁名、Harn^[3]分别提出了一个新的代理签名方案:代理多重签名。王晓明、符方伟^[4]分别指出他们的方案是不安全的,并给出了相应的改进。2006年金永明^[5]等人利用双线性对提出了基于超椭圆曲线的多级代理签名方案。文中结合代

理多重签名和多级代理签名的思想,提出了一个基于离散对数上的多级多代理签名方案,该方案更具有—般性。多重代理签名、代理多重签名、多级代理签名可以看作是该方案的特殊形式。本方案在现实中的应用范围也更加广泛。例如, n 个不同的单位要联合发布一个文件,该文件必须有这些单位共同签名才生效。做法可以有两种:

(1) 这些单位共同在该文件上签字;

(2) 这些单位可以委托一个他们都信任的代理人,代表他们在该文件上签字。

对于第一种情况,可以用多重签名方案^[6]解决,对于第二种情况,可以用代理多重签名方案^[4]解决,但是若这些部门想委托多个代理人,或代理人有事想再次委托给别人来进行代理签名,对于这些情况,目前还没有现成的方案来实现。

文中针对以上问题,提出了一类新的代理签名方案可以有效的解决这类问题,称之为多级多代理签名

收稿日期:2010-05-04;修回日期:2010-08-09

基金项目:河南理工大学青年基金资助项目(Q2010-58)

作者简介:杨迎辉(1982-),男,河南洛阳人,助教,从事密码学与信息安全研究。

(multi-level multi-proxy signature)。多级多代理签名一般过程如下:

设 $A_{0j}(1 \leq j \leq n_0)$ 是 n_0 个原始签名人, $A_{1j}(1 \leq j \leq n_1)$ 是 A_{0j} 信任的代理签名人, 代理签名人可以再次委托签名权, $A_{ij}(1 \leq j \leq n_i)$ 是 A_{i-1j} 信任的代理签名人, 即代理签名权经过 i 级委托后, 由代理签名人 $A_{ij}(1 \leq j \leq n_i)$ 对文件进行签名, 最后由验证者 B 对代理签名进行验证。当 $i = 1, n_0 > 1, n_1 = 1$ 时该签名就是代理多重签名; 当 $i = 1, n_0 = 1, n_1 > 1$ 时该签名就是多重代理签名; 当 $i = 1, n_0 = 1, \dots, n_i = 1$ 时该签名就是多级代理签名。因此, 多重代理签名、代理多重签名、多级代理签名都可以看作是该方案的特殊形式。

1 代理多重签名方案

本节简单介绍伊丽江^[2]等人提出的代理多重签名方案。

令 A_1, \dots, A_n 是 n 个原始签名人, B 是原始签名人 A_i 都信任的一个代理签名人, m 是要签名的文件。每个原始签名人 A_i 有一个公钥为 v_i 和一个密钥 s_i , 使得 $s_i \in Z_{p-1}^*, v_i = g^{s_i} \bmod p$, 其中 $1 \leq i \leq n$ 。

1.1 Mambo 型代理多重签名方案

(1) 代理密钥的生成阶段。

子代理密钥的生成。 A_i 随机选择 $k_i \in Z_{p-1}^*$, 计算出 $K_i = g^{k_i} \bmod p$ 和 $\sigma_i = s_i + k_i K_i \bmod p - 1$ 。

子代理密钥的发送。 A_i 将 (σ_i, K_i) 作为子密钥通过安全的信息通道发送给 B 。

子代理密钥的验证。代理签名人 B 验证等式 $g^{\sigma_i} = v_i K_i^{k_i} \bmod p$ 是否成立, 若等式成立, (σ_i, K_i) 就是一个有效的子代理密钥, 否则若等式不成立, B 就拒绝接受这个子代理密钥, 让原始签名人 A_i 重新发送一个有效的子代理密钥, 要么就终止协议(即拒绝代理 A_i 进行签名)。

代理密钥的生成。若代理签名人 B 确认所有的子代理密钥 (σ_i, K_i) 都是有效的, 其中 $1 \leq i \leq n$, 那么他

计算 $\sigma = \sum_{i=1}^n \sigma_i$, 把 σ 作为代理多重签名的代理密钥。

(2) 代理签名的生成阶段。当代理签名人 B 代表所有的原始签名人 A_1, \dots, A_n 对文件 m 进行签名时, 他用代理密钥 σ 作为普通签名的密钥来执行一般的签名运算, 生成的代理多重签名是 $(m, \text{Sign}_\sigma(m), K_1, \dots, K_n)$, 其中 $\text{Sign}_\sigma(m)$ 是用一般的数字签名算法在代理密钥 σ 下生成的关于消息 m 的代理多重数字签名。

(3) 代理签名的验证阶段。验证人先计算出 $v' = v_1 \cdots v_n K_1^{K_1} \cdots K_n^{K_n} \bmod p$, 然后把 v' 作为代理公钥, 再对代理多重签名 $\text{Sign}_\sigma(m)$ 进行普通签名验证。

1.2 Kim 型代理多重签名方案

(1) 代理密钥的生成阶段。

子代理密钥的生成。 A_i 随机选择 $k_i \in Z_{p-1}^*$, 计算出 $K_i = g^{k_i} \bmod p$ 和 $e_i = h(m_w, K_i)$, 然后 A_i 再计算出 $\sigma_i = e_i s_i + k_i \bmod p - 1$ 。

子代理密钥的发送。 A_i 将 (σ_i, K_i, m_w) 作为子密钥通过安全的信息通道发送给 B 。

子代理密钥的验证。代理签名人 B 验证等式 $e_i = h(m_w, K_i)$ 和 $g^{\sigma_i} = v_i K_i^{K_i} \bmod p$ 是否成立, 若等式成立, (σ_i, K_i, m_w) 就是一个有效的子代理密钥, 否则若等式不成立, B 就拒绝接受这个子代理密钥, 让原始签名人 A_i 重新发送一个有效的子代理密钥, 要么就终止协议(即拒绝代理 A_i 进行签名)。

代理密钥的生成。若代理签名人 B 确认所有的子代理密钥 (σ_i, K_i, m_w) 都是有效的, 其中 $1 \leq i \leq n$, 那么他计算 $\sigma = \sum_{i=1}^n \sigma_i$, 把 σ 作为代理多重签名的代理密钥。

(2) 代理签名的生成阶段。当代理签名人 B 代表所有的原始签名人 A_1, \dots, A_n 对文件 m 进行签名时, 他用代理密钥 σ 作为普通签名的密钥来执行一般的签名运算, 生成的代理多重签名是 $(m, \text{Sign}_\sigma(m), K_1, \dots, K_n, m_w)$, 其中 $\text{Sign}_\sigma(m)$ 是用一般的数字签名算法在代理密钥 σ 下生成的关于消息 m 的代理多重数字签名。

(3) 代理签名的验证阶段。验证人先计算 $e_i = h(m_w, K_i)$, $(1 \leq i \leq n)$ 和 $v' = v_1^{e_1} \cdots v_n^{e_n} K_1 \cdots K_n \bmod p$, 然后把 v' 作为代理公钥, 再对代理多重签名 $\text{Sign}_\sigma(m)$ 进行普通签名验证。

2 文中提出的多级多代理签名方案

2.1 安全参数的选取阶段

SDC(share distribution center)是所有人都信任的共享分配中心, SDC 选取 p, q 两个大素数, 且 $q \mid (p-1)$, g 是 $\text{GF}(p)$ 的本原元, 阶数为 q , h, H 是安全的 Hash 函数, 设 $A_{0j}(1 \leq j \leq n_0)$ 是原始签名人, n_0 是原始签名人的数量, $A_{ij}(1 \leq j \leq n_i)$ 是第 i 级代理签名人, n_i 是第 i 级代理签名人的数量, B 是验证人, w_{i-1} 是第 $i-1$ 级签名人与第 i 级代理签名人间的授权证书, 它不仅包含第 $i-1$ 级签名人与 i 级代理人的身份、签名范围, 还包括第 $i-1$ 级对第 i 级代理人的再

授权范围等事项的说明。 $y_{ij} = g^{x_{ij}} \bmod p$, 其中 $x_{ij} \in Z_q^*$, y_{ij} 分别是 A_{ij} 的私钥和公钥, $y_{iG} = \prod_{j=1}^{n_i} y_{ij} \bmod p$, 为了避免类似祁名^[3]方案中的伪造攻击, SDC 对所有参与签名和验证人的公钥进行验证, 并在公开文件夹中公布 $(w_0: y_{0j}, y_{0G}, \dots, w_i: y_{ij}, y_{iG})$ 。注: SDC 的公开文件夹中的内容只能由 SDC 改写, 其他任何人都无法改写。

2.2 代理签名的授权阶段

● 第 1 级授权: 即原始签名人对第 1 级代理签名人的授权。

(1) 原始签名人 A_{0j} 随机选取 $k_{0j} \in Z_q^*$, 计算 $r_{0j} = g^{k_{0j}} \bmod p$, 广播 r_{0j} 给其他原始签名人 A_{0i} , DC (designated clerk) 和 SDC (SDC 对 r_{0j} 验证, 并计算出 $r_0 = \prod_{j=1}^{n_0} r_{0j} \bmod p$, 放入公开文件夹中)。每个原始签名人计算, $r_0 = \prod_{j=1}^{n_0} r_{0j} \bmod p$, $s_{0j} = n_0 r_0 k_{0j} + n_0 H(w_0) x_{0j} \bmod q$, A_{0j} 发送 s_{0j} 给 DC。

(2) DC 收到 s_{0j} 后, 计算 $s_0 = \sum_{j=1}^{n_0} s_{0j} \bmod q$, 从公开文件夹中取得公钥, 通过

$$g^{s_0} = r_0^{n_0 r_0} \cdot y_{0G}^{n_0 H(w_0)} \bmod p \quad (1)$$

验证 s_0 的有效性。若不成立, 通过 $g^{s_{0j}} = r_{0j}^{n_0 r_0} \cdot y_{0j}^{n_0 H(w_0)} \bmod p$, 验证每个 s_{0j} 的有效性, 其中 $1 \leq j \leq n_0$ 。若某个 s_{0j} 不成立, DC 通知 A_{0j} 重新生成有效的 s_{0j} 。DC 将 (r_0, s_0, w_0) 发送给代理签名人 A_{1j} 。

(3) A_{1j} 收到 (r_0, s_0, w_0) 后, 每个 A_{1j} 通过 $g^{s_0} = r_0^{n_0 r_0} \cdot y_{0G}^{n_0 H(w_0)} \bmod p$, 验证 s_0 的有效性。若有效, 计算 $s_{0j} = x_{1j} + s_0 \bmod q$, s_{0j} 是 A_{1j} 的代理私钥, 其中 $1 \leq j \leq n_1$ 。否则终止协议。

● 第 i 级授权: 即第 $i-1$ 级代理签名人对第 i 级代理签名人授权。

(1) A_{i-1j} 选取随机数 $k_{i-1j} \in Z_q^*$, 计算 $r_{i-1j} = g^{k_{i-1j}} \bmod p$, 并广播 r_{i-1j} 给其他代理签名人 A_{i-1i} , DC 和 SDC (SDC 对 r_{i-1j} 验证, 放入公开文件夹中并计算出 r_{i-1})。每个代理人计算 $r_{i-1} = \prod_{j=0}^{n_{i-1}} r_{i-1j} \bmod p$, $s_{i-1j} = n_{i-1} r_{i-1} k_{i-1j} + n_{i-1} H(w_{i-1}) x_{i-1j} + n_{i-2}^{-1} s_{i-2} \bmod q$, A_{i-1j} 发送 s_{i-1j} 给 DC。

(2) DC 收到 s_{i-1j} 后, 计算 $s_{i-1} = \sum_{j=1}^{n_{i-1}} s_{i-1j} \bmod q$, 从公开文件夹中取得公钥, 由

$$g^{s_{i-1}} = \left(\prod_{j=0}^{i-1} r_{jG}^{H(w_j)} \right)^{n_{i-1}} \bmod p \quad (2)$$

验证 s_{i-1} 的有效性, 若无效, 通过 $g^{s_{i-1j}} = r_{i-1j}^{n_{i-1} r_{i-1}}$, $y_{i-1j}^{n_{i-1} H(w_{i-1})} \prod_{k=0}^{i-2} (r_k^{r_k} \cdot y_{kG}^{H(w_k)}) \bmod p$ 验证每个 s_{i-1j} 的有效性, 其中 $1 \leq j \leq n_{i-1}$ 。若某个 s_{i-1j} 不成立, DC 通知 A_{i-1j} 重新生成有效的 s_{i-1j} 。DC 将 $(r_{i-1}, s_{i-1}, w_{i-1})$ 发送给代理签名人 A_{ij} 。

(3) A_{ij} 收到 $(r_{i-1}, s_{i-1}, w_{i-1})$ 后, 每个 A_{ij} 通过 (1) 式验证 s_{i-1} 的有效性。若有效, 计算 $s_{i-1j} = x_{ij} + s_{i-1} \bmod q$, s_{i-1j} 是 A_{ij} 的代理私钥。否则终止协议。

2.3 代理签名的产生阶段

设共进行了 i 级代理后开始对消息 m 进行签名, 即由代理人 A_{ij} 对消息 m 签名。

(1) A_{ij} 取随机数 $t_{ij} \in Z_q^*$, 计算 $u_{ij} = g^{t_{ij}} \bmod p$, 并广播 u_{ij} 给其他代理签名人 A_{ii} 和 DC。每个代理人计算 $u_i = \prod_{j=1}^{n_i} u_{ij} \bmod p$, 和 $v_{ij} = u_i t_{ij} + s_{i-1j} n_{i-1}^{-1} h(m, u_i) \bmod q$, A_{ij} 发送 (u_i, v_{ij}, m) 给 DC 作为消息 m 的子代理签名。

(2) DC 收到 (u_i, v_{ij}, m) 后, 通过 $g^{v_{ij}} = u_i^{u_i} \cdot y_{ij}^{n_{i-1} h(m, u_i)} \cdot \left(\prod_{j=0}^{i-1} r_j^{r_j} y_{jG}^{H(w_j)} \right)^{h(m, u_i)} \bmod p$ 验证 v_{ij} 的有效性, 若某个 v_{ij} 不成立, DC 通知 A_{ij} 重新生成有效的 v_{ij} 。

(3) DC 计算 $v_i = \sum_{j=1}^{n_i} v_{ij} \bmod q$, 把 (u_i, v_i, m) 作为的数字签名发送给 B 。

2.4 代理签名的验证阶段

验证者 B 收到 (u_i, v_i, m) 后, 从公开文件夹中取出必要的数据, 通过

$$g^{v_i} = u_i^{u_i} \cdot y_{iG}^{n_{i-1} h(m, u_i)} \cdot \left[\prod_{j=0}^{i-1} (r_j^{r_j} y_{jG}^{H(w_j)}) \right]^{n_i h(m, u_i)} \bmod p \quad (3)$$

来验证签名的有效性。

当原始签名人或某一级代理人要收回其代理人的代理签名权时, 只需要通知 SDC 删除公开文件中关于他的信息, 其代理人就不能再行使代理权, 若其代理人继续使用原来的信息进行代理签名, 那么在验证过程中, 验证人不能从公开文件夹取得信息, 因此不能通过验证, 这样就防止了代理权的滥用。

3 提出方案的正确性及安全性分析

3.1 方案的正确性分析

定理 1. DC (designated clerk) 及代理人 A_{1j} 都可以通过 (1) 式验证 s_0 的有效性。

证明:

$$\begin{aligned}
 g^{s_0} &= g^{\sum_{j=1}^{n_0} s_{0j} \bmod q} = \prod_{j=1}^{n_0} g^{n_0 r_0 k_{0j} + n_0 H(w_0) r_{0j} \bmod p} \\
 &= \prod_{j=1}^{n_0} (r_0^{n_0 r_0} \cdot y_{0j}^{n_0 H(w_0)}) \bmod p \\
 &= r_0^{n_0 r_0} \cdot y_{0G}^{n_0 H(w_0)} \bmod p
 \end{aligned}$$

定理 2 DC(designated clerk) 及代理人 A_{ij} 都可以
通过(2)式验证 s_{i-1} 的有效性。

证明:

$$\begin{aligned}
 g^{s_{i-1}} &= g^{\sum_{j=1}^{n_{i-1}} s_{i-1j} \bmod q} \\
 &= \prod_{j=1}^{n_{i-1}} g^{n_{i-1} r_{i-1} k_{i-1j} + n_{i-1} H(w_{i-1}) x_{i-1j} + n_{i-2}^{-1} s_{i-2} \bmod p} \\
 &= \prod_{j=1}^{n_{i-1}} (r_{i-1j}^{n_{i-1} r_{i-1}} \cdot y_{i-1j}^{n_{i-1} H(w_{i-1})} \cdot g^{n_{i-2}^{-1} s_{i-2}}) \bmod p \\
 &= r_{i-1}^{n_{i-1} r_{i-1}} \cdot y_{i-1G}^{n_{i-1} H(w_{i-1})} \cdot \\
 &\quad \left(\prod_{j=0}^{i-2} r_j^{r_j} y_{jG}^{H(w_j)} \right)^{n_{i-1}} \bmod p \\
 &= \left(\prod_{j=0}^{i-1} r_j^{r_j} y_{jG}^{H(w_j)} \right)^{n_{i-1}} \bmod p
 \end{aligned}$$

定理 3 验证人 B 可以通过(3)式验证签名的有效性。

证明:

$$\begin{aligned}
 g^{v_i} &= g^{\sum_{j=1}^{n_i} v_{ij} \bmod q} = \prod_{j=1}^{n_i} g^{u_i r_{ij} + s_{i-1j} n_{i-1}^{-1} h(m, u_i)} \bmod p \\
 &= \prod_{j=1}^{n_i} u_i^{r_{ij}} \cdot g^{(x_{ij} + s_{i-1j}) n_{i-1}^{-1} h(m, u_i)} \bmod p \\
 &= u_i^{n_i} \cdot \prod_{j=1}^{n_i} [y_{ij}^{n_{i-1}^{-1} h(m, u_i)} \cdot \\
 &\quad \left(\prod_{j=0}^{i-1} r_j^{r_j} y_{jG}^{H(w_j)} \right)^{h(m, u_i)}] \bmod p \\
 &= y_{iG}^{n_{i-1}^{-1} h(m, u_i)} \cdot \left[\prod_{j=0}^{i-1} (r_j^{r_j} y_{jG}^{H(w_j)}) \right]^{n_i h(m, u_i)} \bmod p
 \end{aligned}$$

3.2 方案的安全性分析

(1) 根据公钥 y_{ij} , 任何人都不能计算出 x_{ij} , 因为这是解离散对数难题。

(2) 任何人(包括授权人)不能替代签名人 A_{ij} 进行代理签名。因为要想替代 A_{ij} 签名, 必须有代理私钥 s_{i-1j} , 而任何人不能伪造出 A_{ij} 的代理私钥 s_{i-1j} 。因为 $s_{i-1j} = x_{ij} + s_{i-1} \bmod q$, 要想得到代理私钥 s_{i-1j} , 必须求出代理人的私钥, 伪造者是无法求出的。

(3) 不可否认性。如果在签名的验证阶段, 代理签名 (u_i, v_i, m) 通过验证, 在验证方程中有授权人及代理人签名的公钥, w_i 也有授权人及签名人的信息, 也

就是说, 授权人不能否认对代理人的授权, 签名人不能否认都文件的签名。

4 结束语

结合代理多重签名和多级代理签名的特点, 提出了一个基于离散对数上的多级多代理签名方案, 它不仅具有二者的优点, 而且代理多重签名, 多重代理签名以及多级代理签名都可以看作是它的一种特殊形式, 在应用方面也更具有一般性。

参考文献:

- [1] Mambo M, Usuda K, Okamoto E. Delegation of the power to sign messages[J]. IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences, 1996, E79-A(9):1338-1354.
- [2] 伊丽江, 白国强, 肖国镇. 代理多重签名[J]. 计算机研究与发展, 2001, 38(2):204-206.
- [3] 祈明, Harn L. 基于离散对数的若干新型代理签名方案[J]. 电子学报, 2000, 28(11):114-115.
- [4] 王晓明, 符方伟. 一种代理多重数字签名方案的安全分析[J]. 通信学报, 2002, 23(4):98-102.
- [5] 金永明, 徐秋亮, 陈泽雄. 椭圆曲线上的多级代理签名方案[J]. 计算机工程与应用, 2006(20):98-102.
- [6] Ohta K, Okamoto T. A digital multi-signature scheme based on the Fiat-Shamir scheme[C]//Advances in Cryptology-ASIACRYPT'91. [s.l.]:Springer-Verlag, 1991:139-148.
- [7] Kim S, Park S, Won D. Proxy signature, revisited[C]//ICICS'97, Lecture Notes in Computer Science. Berlin: Springer, 1997:223-232.
- [8] Tseng S F, Yang C Y, Hwang M S. A nonrepudiable threshold multi-signature multi-proxy multi-signature scheme with shared verification[J]. Future Generation Computer Systems, 2004, 20(5):887-893.
- [9] Hsu C L, Tsai K Y, Tsai P L. Cryptanalysis and improvement of nonrepudiable threshold multi-proxy multi-signature scheme with shared verification[J]. Information Sciences, 2007, 177(2):543-549.
- [10] 杨迎辉, 孙艳蕊, 袁喜凤. 改进的门限多代理多重共享验证签名方案[J]. 计算机工程, 2008, 34(23):170-172.
- [11] 蔡庆华, 陈文莉. 基于双线性对的代理签名[J]. 计算机技术与发展, 2006, 16(9):230-232.
- [12] 蔡庆华. 基于双线性对的代理盲签名[J]. 计算机技术与发展, 2006, 16(11):166-167.

中国计算机学会会刊、中国科技核心期刊
《计算机技术与发展》欢迎投稿, 欢迎订阅!