

文章编号:1001-9081(2006)07-1631-03

## 对一种门限代理签名方案的密码分析及改进

王天银<sup>1</sup>, 蔡晓秋<sup>1,2</sup>, 张建中<sup>2</sup>

(1. 洛阳师范学院 数学科学学院, 河南 洛阳 471022;

2. 陕西师范大学 数学与信息科学学院, 陕西 西安 710062)

(yinyang790720@yahoo.com.cn)

**摘 要:**对一种基于双线性对的新型门限代理签名方案进行了密码分析,发现该门限代理签名方案不能抵抗伪造攻击和公钥替换攻击。对该方案进行了改进,改进后的门限代理签名方案克服了原门限代理签名方案的安全隐患,并且保留了原门限代理签名方案的优点。

**关键词:**双线性对; 门限代理签名; 伪造攻击; 公钥替换攻击

**中图分类号:** TP309 **文献标识码:** A

## Cryptanalysis and improvement of a threshold proxy signature scheme

WANG Tian-yin<sup>1</sup>, CAI Xiao-qiu<sup>1,2</sup>, ZHANG Jian-zhong<sup>2</sup>

(1. Mathematics and Information Science College, Luoyang Normal University, Luoyang Henan 471022, China;

2. College of Mathematics and Information Science, Shaanxi Normal University, Xi'an Shaanxi 710062, China)

**Abstract:** Through the cryptanalysis of a threshold proxy signature scheme based on bilinear pairings, it was found that the scheme couldn't resist forgery attack and public-key substitute attack. An improvement to mend the security leaks was proposed, which overcame the disadvantages and retained the merits of the original threshold proxy signature scheme.

**Key words:** bilinear pairings; threshold proxy signature; forgery attack; public-key substitute attack

## 0 引言

1996 年, Mambo, Usuda 和 Okamoto 首先提出了代理签名的概念<sup>[1,2]</sup>, 随着代理签名的发展, 门限代理签名受到了广泛关注,  $(t, n)$  门限代理签名是代理签名的一种变形, 它是由门限体制和代理签名体制结合而产生的一种新的签名体制。门限代理签名体制的代理签名密钥由  $n$  个代理签名人分享保管, 只有  $t$  个或更多的代理签名人代表原始签名人运用各自的代理签名密钥碎片才能产生对消息  $m$  的签名, 少于  $t$  个则不能。

文献[3]和文献[4]分别独立地提出了门限代理签名体制, 文献[5]指出文献[3]和文献[4]的门限代理签名方案是不安全的, 并给出了一个改进方案。文献[6]进一步指出文献[5]的方案不能抵抗公钥替换攻击, 并给出了一个更安全的不可否认门限代理签名方案。然而上述门限代理签名方案都需要原始签名人和代理签名人广播和传送许多数据, 并且要进行大量的指数运算, 因而通信量和计算量非常大。文献[7]给出了一个基于双线性对的新型门限代理签名方案(Q-C-X 方案), 具有较小的通信量和计算量。但本文通过对 Q-C-X 方案的密码分析发现该方案不能抵抗伪造攻击和公钥替换攻击, 并对 Q-C-X 方案进行了改进。改进后的门限代理签名方案克服了 Q-C-X 方案的安全隐患, 并且保持了 Q-C-X 方案通信量小、计算复杂性低等优点。

## 1 Q-C-X 方案

### 1.1 初始化过程

设  $G_1$  和  $G_2$  分别为阶是素数  $q$  的加群和乘群,  $P$  为 CDH 群

$G_1$  的生成元, 且离散对数问题在群  $G_1$  和  $G_2$  中是计算上不可行的,  $e: G_1 \times G_1 \rightarrow G_2$  为一个安全的双线性对。  $H_1(\cdot): \{0, 1\}^* \times G_1 \rightarrow Z_q^*$ ,  $H_2(\cdot): \{0, 1\}^* \rightarrow G_1 \setminus \{1\}$  为两个强无碰撞安全单向 Hash 函数。原始签名人  $p_0$  随机选择  $x_0 \in Z_q^*$  作为私钥  $sk_0$ , 计算  $Y_0 = x_0 P$  作为公钥  $pk_0$ , 并让 CA 认证。代理签名人  $p_1, p_2, \dots, p_n$  随机选择  $x_i \in Z_q^*$  作为各自的私钥  $sk_i$ , 计算  $Y_i = x_i P$  作为公钥  $pk_i$ , 并让 CA 认证。授权书  $m_\omega$  详细描述了代理的诸多事宜。

### 1.2 代理密钥生成协议

1) 原始签名人  $p_0$  随机选取  $r \in Z_q^*$ , 计算  $U = rP$ ,  $h = H_1(m_\omega, U)$ ,  $Q = H_2(m_\omega)$ ,  $V = (r + hx_0)Q$ ,  $s = n^{-1}(hr + x_0)$ 。然后把授权书  $m_\omega$ 、对  $m_\omega$  的签名  $\sigma = (U, V)$  和  $s$  发送给每个代理签名人。

2) 每个代理签名人验证:  $e(P, V) \stackrel{?}{=} e(U + hY_0, H_2(m_\omega))$ ;  $nsP \stackrel{?}{=} hU + Y_0$ 。当且仅当上面两式都成立才接受  $(\sigma, s)$ 。若  $(\sigma, s)$  有效, 代理签名人  $p_i$  随机选择  $k_i \in Z_q^*$ , 计算并广播  $k_i P$ , 同时计算出  $s_i = s + x_i + k_i$  作为自己的代理秘密。

3) 代理签名人  $p_i$  随机选取一个多项式  $f_i(z) = s_i + a_{i,1}z + a_{i,2}z^2 + \dots + a_{i,t-1}z^{t-1}$ , 然后  $p_i$  计算并广播  $a_{i,j}P$  ( $j = 1, 2, \dots, t-1$ ), 最后将  $f_i(j)$  秘密发送给  $p_j$  ( $j = 1, 2, \dots, n; j \neq i$ )。

4) 代理签名人  $p_i$  从代理签名人  $p_j$  那里收到  $f_j(i)$  ( $j = 1, 2, \dots, n; j \neq i$ ) 后, 验证  $f_j(i)P \stackrel{?}{=} \sum_{k=0}^{t-1} i^k a_{j,k}P$ , 若成立,  $p_i$  计算  $x_i' = \sum_{j=1}^n f_j(i)$ , 并广播  $Y_i' = x_i'P$ 。通过计算  $\sum_{j=1}^n \sum_{k=0}^{t-1} i^k a_{j,k}P$  其

收稿日期: 2006-01-23; 修订日期: 2006-03-20

基金项目: 国家自然科学基金资助项目(10271069); 河南省自然科学基金资助项目(0511010300)

作者简介: 王天银(1979-), 男, 河南南阳人, 硕士, 主要研究方向: 密码学; 蔡晓秋(1980-), 女, 河南许昌人, 硕士研究生, 主要研究方向: 密码学; 张建中(1960-), 男, 陕西周至人, 教授, 博士, 主要研究方向: 密码学、信息安全。

他人也能得到  $Y_i' = x_i'P$ 。

### 1.3 代理签名生成及验证

设  $m$  为待签名的消息,不失一般性,在这里假定  $p_1, p_2, \dots, p_t$  为  $t$  个代表原始签名人对消息  $m$  进行签名的代理签名人。签名过程如下:

1) 每个代理签名人  $p_i (i = 1, 2, \dots, t)$  计算:  $\omega_i =$

$$\prod_{j \in \{1, 2, \dots, t\}, j \neq i} \frac{j}{j-i}, \sigma_i = x_i' \omega_i H_2(m);$$

2)  $t$  个代理签名人收集完所有的部分签名后  $\sigma_i$ , 对  $\sigma_i$  用下式——验证其有效性:

$$e(P, \sigma_i) \stackrel{?}{=} e(\omega_i Y_i', H_2(m))$$

若每个部分签名  $\sigma_i$  都能通过验证,则它们合作产生  $\sigma' =$

$$\sum_{i=1}^t \sigma_i \text{ 和 } K = \sum_{m=1}^n k_m P, \text{ 完整有效的门限代理签名为 } \langle m, U, m_\omega, \sigma', K \rangle。$$

接收者验证  $e(P, \sigma') \stackrel{?}{=} e(H_1(m_\omega, U)U + Y_0 + K + \sum_{i=1}^n Y_i, H_2(m))$ , 若成立,则签名有效,否则予以拒绝。

## 2 对 Q-C-X 方案的密码分析

对 Q-C-X 方案,给出了两种攻击,从而说明该方案是不安全的。

### 2.1 伪造攻击

任何人可以伪造对任意消息  $m'$  的有效门限代理签名  $\langle m', U', m_\omega, \sigma', K' \rangle$ 。

伪造攻击过程如下:

攻击者首先任选  $U' \in G_1, l \in Z_q^*$ , 计算  $K' = lP - (\sum_{i=1}^n Y_i + Y_0 + H_1(m_\omega, U')U')$ ,  $\sigma' = lH_2(m')$ , 然后把  $\langle m', U', m_\omega, \sigma', K' \rangle$  作为对消息  $m'$  的代理签名传给接收者。

由于  $e(H_1(m_\omega, U')U' + Y_0 + K' + \sum_{i=1}^n Y_i, H_2(m')) = e(lP, H_2(m')) = e(P, lH_2(m')) = e(P, \sigma')$ , 所以攻击者伪造的门限代理签名  $\langle m', U', m_\omega, \sigma', K' \rangle$  可以通过验证。

### 2.2 公钥替换攻击

原始签名人  $p_0$  或任何代理签名人  $p_i (1 \leq i \leq n)$  都可以通过公钥替换攻击伪造对任意消息  $m'$  的有效门限代理签名  $\langle m', U', m_\omega, \sigma', K' \rangle$ 。

不失一般性,假定  $p_1$  欲伪造代理签名。攻击如下:

代理签名人  $p_1$  首先任选  $K' \in G_1, U' \in G_1, l \in Z_q^*$ , 计算:

$$Y_1' = lP - (\sum_{j=2}^n Y_j + Y_0 + K' + H_1(m_\omega, U')U'), \sigma' = lH_2(m').$$

接着要求 CA 把他的公钥改为  $Y_1'$ , 然后把  $\langle m', U', m_\omega, \sigma', K' \rangle$  作为对消息  $m'$  的门限代理签名传给接收者。

易知:  $e(P, \sigma') = e(H_1(m_\omega, U')U' + Y_0 + K' + \sum_{i=1}^n Y_i, H_2(m'))$ 。

所以代理签名人  $p_1$  伪造的签名  $\langle m', U', m_\omega, \sigma', K' \rangle$  可以通过验证。

## 3 对 Q-C-X 方案的改进

### 3.1 初始化过程

参数设置同原方案。不同之处在于密钥认证中心 CA 要

求每个要求认证其公钥  $pk$  的用户能够同时提供知道该公钥  $pk$  对应的私钥  $sk$  的零知识证明。具体步骤如下: CA 随机选择  $r \in Z_q^*$ , 计算  $R = rP$ , 然后把  $R$  发送给相应用户, 用户计算  $B = skR$ , 然后发送给 CA, CA 验证  $rp_k \stackrel{?}{=} B$ , 若成立, 则给予认证, 否则拒绝认证。

### 3.2 代理密钥生成协议

1) 原始签名人  $p_0$  随机选取  $k_0 \in Z_q^*$ , 计算并广播  $U_0 = k_0P$ , 代理签名人  $p_i (1 \leq i \leq n)$  随机选择  $k_i \in Z_q^*$ , 计算并广播  $U_i = k_iP$ 。

2)  $p_0$  计算:  $U = \sum_{i=0}^n U_i, h = H_1(m_\omega, U), \delta = x_0 H_2(m_\omega, U), s = n^{-1}(hk_0 + x_0)$ , 然后把  $(m_\omega, \delta, s)$  发送给每个代理签名人  $p_i$ 。

3) 代理签名人  $p_i$  收到  $(m_\omega, \delta, s)$  后, 计算  $U = \sum_{i=0}^n U_i,$

$h = H_1(m_\omega, U)$ 。然后验证:  $e(P, \delta) \stackrel{?}{=} e(Y_0, H_2(m_\omega)), nsP \stackrel{?}{=} hU_0 + Y_0$ 。当且仅当上面两个等式都成立时接受  $(m_\omega, \delta, s)$ 。若  $(m_\omega, \delta, s)$  有效, 计算  $s_i = s + x_i + hk_i$  作为自己的代理秘密。接着随机选取系数在  $Z_q$  中, 次数为  $t-1$  的多项式  $f_i(z) = s_i + a_{i,1}z + a_{i,2}z^2 + \dots + a_{i,t-1}z^{t-1}$ 。然后计算并广播  $a_{i,j}P (j = 1, 2, \dots, t-1)$ , 并且将  $f_i(j)$  秘密发送给  $p_j (j = 1, 2, \dots, n; j \neq i)$ 。

4)  $p_i$  从  $p_j$  那里收到  $f_j(i) (j = 1, 2, \dots, n; j \neq i)$  后, 验证  $f_j(i)P \stackrel{?}{=} \sum_{k=0}^{t-1} i^k a_{j,k}P$ , 若此式不成立, 则  $f_j(i)$  无效, 要求  $p_j$

重发; 否则  $p_i$  计算出自己的代理密钥  $x_i' = \sum_{k=1}^n f_k(i)$ , 并广播

$$Y_i' = x_i'P。$$

通过计算  $\sum_{j=1}^n \sum_{k=0}^{t-1} i^k a_{j,k}P$ , 其他人也能得到  $Y_i' = x_i'P$ 。若令  $f(z) = \sum_{i=1}^n f_i(z)$ , 则易知  $x_i' = f(i), Y_i' = f(i)P$ 。

### 3.3 代理签名生成协议

设  $m$  为待签名的消息, 不失一般性, 在这里也假定  $p_1, p_2, \dots, p_t$  为  $t$  个代表原始签名人  $p_0$  对消息  $m$  进行签名的代理签名人。签名过程如下:

1) 每个代理签名人  $p_i (i = 1, 2, \dots, t)$  计算:  $\omega_i =$

$$\prod_{j \in \{1, 2, \dots, t\}, j \neq i} \frac{j}{j-i}, \sigma_i = x_i' \omega_i H_2(m)。$$

2)  $t$  个代理签名人收集完所有的部分签名  $\sigma_i (i = 1, 2, \dots, t)$  后, 对  $\sigma_i$  用下面的等式——验证其有效性:

$$e(P, \sigma_i) \stackrel{?}{=} e(\omega_i Y_i', H_2(m))$$

若上式不成立, 则知  $p_i$  没有给出正确的部分签名, 或  $p_i$  是一个欺诈者, 可以找其他代理签名人重做 1)。若每个部分签名  $\sigma_i$  都能通过验证, 则它们合作产生  $\sigma = \sum_{i=1}^t \sigma_i$ , 完整有效的门限代理签名为  $\langle m, U, m_\omega, \sigma \rangle$ 。

### 3.4 代理签名的验证

接收者验证  $e(P, \sigma) \stackrel{?}{=} e(H_2(m), H_1(m_\omega, U)U + Y_0 + K' + \sum_{i=1}^n Y_i)$ , 若成立, 签名有效。

签名的正确性可由如下方程给出:

$$e(P, \sigma) = e(P, \sum_{i=1}^t \sigma_i) = e(P, \sum_{i=1}^t x_i' \omega_i H_2(m))$$

$$\begin{aligned}
&= e\left(H_2(m), \sum_{i=1}^n f_i(0)P\right) \\
&= e\left(H_2(m), \sum_{i=1}^n (s + H_1(m_\omega, U)k_i + x_i)P\right) \\
&= e\left(H_2(m), (H_1(m_\omega, U)U + Y_0 + \sum_{i=1}^n Y_i)\right)
\end{aligned}$$

## 4 对改进方案的密码分析

### 4.1 安全性

下面分析改进方案克服了原方案的安全隐患,并且满足门限代理签名的其他安全要求。

**定理 1** 改进方案可以抵抗本文提出的伪造攻击和公钥替换攻击。

**证明:** 改进方案的完整有效门限代理签名为  $\langle m, U, m_\omega, \sigma \rangle$ , 删除了原门限代理签名方案签名  $\langle m, U, m_\omega, \sigma' \rangle$ ,  $K >$  中的多余参数  $K$ , 因此避免了针对参数  $K$  的伪造攻击。

改进方案也可抵抗公钥替换攻击, 因为若攻击者(不妨设为代理签名人  $p_1$ ) 欲通过公钥替换攻击的方法伪造对消息  $m'$  的有效门限代理签名,  $p_1$  必须首先任选  $U' \in G_1, l \in Z_q^*$ , 计算:  $Y_1' = lP - (\sum_{j=2}^n Y_j + Y_0 + H_1(m_\omega, U')U')$ ,  $\sigma' = lH_2(m')$ 。

接着必须要求 CA 把他的公钥改为  $Y_1'$ , 但是在改进方案中 CA 要求  $p_1$  必须提供知道公钥  $Y_1'$  对应的私钥  $x_1'$  的零知识证明。由于在群  $G_1$  中, 离散对数问题在计算上是不可行的, 所以  $p_1$  不能求出公钥  $Y_1'$  对应的私钥  $x_1'$ , 也就不能提供知道公钥  $Y_1'$  对应的私钥  $x_1'$  的零知识证明, 因此不能伪造对消息  $m'$  的有效门限代理签名  $\langle m', U', m_\omega, \sigma' \rangle$ 。所以原始签名人  $p_0$  或任何代理签名人  $p_i (1 \leq i \leq n)$  都不能通过公钥替换攻击的方法伪造对任意消息  $m'$  的有效门限代理签名  $\langle m', U', m_\omega, \sigma' \rangle$ 。证毕。

**强不可伪造性** 攻击者伪造的对消息  $m'$  的签名  $\langle m', U', m_\omega, \sigma' \rangle$  要想通过接收者验证, 必须使  $\langle m', U', m_\omega, \sigma' \rangle$  满足验证方程:

$$e(P, \sigma') = e\left(H_2(m'), H_1(m_\omega, U')U' + Y_0 + \sum_{i=1}^n Y_i\right)$$

由于在群  $G_1$  中, 离散对数问题在计算上是不可行的,  $H_1(\cdot), H_2(\cdot)$  为两个强无碰撞安全单向 Hash 函数, 并且  $e$  为一个安全的双线性对, 因此已知  $\langle U', m_\omega, \sigma' \rangle$  之中的任何两个, 去求第三个在计算上都是不可行的, 因此改进方案满足强不可伪造性。

同原方案一样, 改进方案使用了授权书  $m_\omega$ , 并且授权书

$m_\omega$ 、原始签名人和代理签名人的公钥都要在代理签名的验证过程中出现, 因此满足门限代理签名的可验证性、可区分性等其他安全性要求。

### 4.2 性能

表 1 列举了改进方案和原方案在每个阶段所需计算量的比较。令  $A_m, M_m, I_m, A, S$  和  $P$  表示模加运算、模乘运算、求逆运算、点加、数乘运算和双线性对运算。

表 1 计算复杂性比较

运 算	代理密钥生成阶段		代理签名生成阶段		签名验证阶段	
	原方案	改进方案	原方案	改进方案	原方案	改进方案
$A_m$	$2 + n(1 + nt)$	$1 + n(1 + nt)$	0	0	0	0
$M_m$	1	1	0	0	0	0
$I_m$	1	1	0	0	0	0
$A$	$2 + n(t - 1)$	$2 + n(t + n + 2)$	$t + n - 2$	$t - 1$	$n + 2$	$n + 1$
$S$	$2 + n(2t + 5)$	$n(t + n + 1)$	$2t + n$	$2t$	1	1
$P$	$2n$	$2n$	$2t$	$2t$	2	2

从表 1 容易看出, 改进方案在代理密钥生成阶段的计算量大于原方案, 但在代理签名生成阶段和验证阶段计算量比原方案低, 总的来说改进方案计算复杂性略高于原方案。但改进方案所需的通信量比原方案稍低。

### 参考文献:

- [1] MAMBO M, USUDA K, OKAMOTO E. Proxy signatures for delegating signing operation[A]. Proceedings of the 3rd ACM Conference on Computer and Communications Security[C]. New Dehi, India: ACM Press, 1996. 48 - 57.
- [2] MAMBO M, USUDA K, OKAMOTO E. Proxy signatures: delegation of the power to sign messages[J]. IEICE Transactions Fundamentals, 1996, E79A(9): 1338 - 1354.
- [3] ZHANG K. Threshold proxy signature schemes[A]. Proceedings of the First International Workshop on Information Security[C]. Berlin: Springer-Verlag, 1997. 191 - 197.
- [4] KIM S, PARK S, WON D. Proxy signatures, revisited[A]. Proceedings of International Conference on Information and Communications Security[C]. Berlin: Springer-Verlag, 1997. LNCS1334. 223 - 232.
- [5] SUM HM, LEE NY, HWANG T. Threshold proxy signatures[J]. IEEE Proceedings of Computers & Digital Techniques, 1999, 146 (5): 259 - 263.
- [6] 李继国, 曹珍富. 一个改进的门限代理签名方案[J]. 计算机研究与发展, 2002, 39(11): 1513 - 1518.
- [7] 钱海峰, 曹珍富, 薛庆水. 基于双线性对的新型门限代理签名方案[J]. 中国科学(E), 2004, 34 (6): 711 - 720.

(上接第 1630 页)

- [9] LISTON T. Welcome to my tarpit - the tactical and strategic use of labrea [EB/OL]. <http://www.hackbus ters.net/LaBrea/LaBrea.txt>, 2001.
- [10] PROVOS N. A virtual honeypot framework [R/OL]. [http://www.citi.umich.edu/techrepts/reports/re ports/citi-tr-03-1.pdf](http://www.citi.umich.edu/techrepts/reports/citi-tr-03-1.pdf), 2003.
- [11] KREIBICH C. Honeycomb automated ids signature creation using honeypots [EB/OL]. <http://www.cl.cam.ac.uk/.cpk25/honeycomb/>, 2003.
- [12] The HoneyNet Project [EB/OL]. <http://www.honeynet.org>, 2005.
- [13] MEADOWS C. A formal framework and evaluation method for network denial of service[A]. Proceedings of the 1999 IEEE Computer Security Foundations Workshop[C]. IEEE Computer Society Press, 1998.
- [14] FREILING FC, HOLZ T, WICHESKI G. Botnet tracking: exploring a root-cause methodology to prevent distributed denial-of-service attacks [Z]. Aachener Informatik Berichte, RWTH Aachen, 2005.
- [15] 诸葛建伟. Botnet 简介[R/OL]. <http://www.icst.pku.edu.cn/honeynetweb/honeynetcn/TechnicalReports.htm>, 2005 - 05.