

基于 SM2 的代理保护代理签名的设计与实现

郭青霄¹, 张大伟¹, 常亮², 刘晓东³, 宋靖文¹

(1. 北京交通大学计算机与信息技术学院, 北京 100044;
2. 桂林电子科技大学广西可信软件重点实验室, 广西 桂林 541004;
3. 山东大学网络信息安全研究所, 山东 济南 250000)

摘要: 基于国产密码算法 SM2 数字签名标准, 设计了一种代理保护代理签名方案。通过对代理授权机制的设计, 实现了对代理签名者的保护; 通过对 SM2 数字签名算法的改进, 实现了代理者代表原始签名者的代理签名; 安全分析表明, 设计的算法具有可验证性、可识别性、可区分性、不可否认性和强不可伪造性。最后, 利用 C 语言及 Polarssl 库对该方案进行了实现, 并与 SM2 数字签名进行了性能比较。分析表明, 该算法在性能上具有较好的可用性。

关键词: 椭圆曲线; 代理签名; 代理保护代理签名; SM2 数字签名

中图分类号: TP391.1

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2017.00192

Design and implementation of proxy-protected proxy signature based on SM2

GUO Qing-xiao¹, ZHANG Da-wei¹, CHANG Liang², LIU Xiao-dong³, SONG Jing-wen¹

(1. School of Computer and Information Technology, Beijing Jiaotong University, Beijing 100044, China;
2. Guangxi Key Laboratory of Trusted Software, Guilin University of Electronic Technology, Guilin 541004, China;
3. Institute of Network Information Security, Shandong University, Jinan 250000, China)

Abstract: Based on the SM2 digital signature standard, a proxy-protected proxy signature scheme was designed. Through the design of the agent authorization mechanism, the protection of the proxy signer was realized. The proxy signature was realized by the proxy signer who represents the original signer through the improvement of the SM2 digital signature algorithm. The security analysis shows that the designed algorithm is verifiable, identifiable, distinguishable, non-repudiation, and strong unforgeability. Finally, the program was implemented using the C language and the Polarssl library, and the performance was compared with the SM2 digital signature. The analysis shows that the algorithm has good usability in performance.

Key words: elliptic curve, proxy signature, proxy-protected proxy signature, SM2 digital signature

1 引言

1985 年, N.Koblitz 和 V.Miller 各自独立地提出了椭圆曲线公钥密码 (ECC, elliptic curve

cryptography)^[1]。该密码体制的安全性主要基于椭圆曲线离散对数问题 (ECDLP, elliptic curve discrete logarithm problem) 的困难性。与传统离散对数问题相比, ECDLP 的求解难度要大得多, 目

收稿日期: 2017-07-06; 修回日期: 2017-08-27。通信作者: 张大伟, dwzhang@bjtu.edu.cn

基金项目: 国家自然科学基金资助项目 (No.61672091); 广西可信软件重点实验室研究课题基金资助项目 (No.KX201531)

Foundation Items: The National Natural Foundation of China (No.61672091), Key Laboratory of Trusted Software (No.KX201531)

前解决方法的计算复杂度都是指数级别的。文献[2]指出在相同安全强度下,椭圆曲线密码较其他公钥密码所需的密码规模要小很多。ECC的优势使其成为最具发展潜力和应用前景的公钥密码算法。在此背景下,我国开始组织和研究自主知识产权的ECC,于2004年研制完成了SM2算法,并于2016年8月成为中国国家密码标准(标准号为GB/T 32918-2016)^[3,4]。目前研究表明,256 bit的SM2算法与2 048 bit的RSA算法具有相同的安全等级。

1996年,Mambo、Usuda和Okamoto等^[5]提出了代理签名的概念。现有的代理签名方案大多是基于离散对数问题与大数因子分解问题的。2003年,白国强等^[6]将代理签名思想应用于椭圆曲线密码,提出了基于椭圆曲线的代理签名方案,但该方案并没有完全满足对代理签名者的保护等性质;2005年,张宁等^[7]对白国强等^[6]提出的方案进行研究分析,在代理签名的安全性方面做了改进,提出了代理保护代理签名方案;2008年,Chang等^[8]提出了基于ECDSA的代理保护代理签名方案。2010年,胡兰兰等^[9]提出了另一种改进的安全代理签名方案。

但截至目前,代理保护代理签名方案主要是在ECDSA的基础上进行研究的。本文将代理签名思想与国密算法SM2相结合,产生了一种新的代理签名方案,即基于SM2的代理保护代理签名方案,同时分析了这一方案的安全性及复杂性。该方案不仅满足代理签名的一般性质,还满足了文献[10]所提出的强不可伪造的性质,实现了对代理签名者的保护。

2 代理签名

2.1 代理签名简介

1976年,Diffie和Hellman^[11,12]利用公钥密码学的思想首次提出了数字签名概念。但在实际应用中,为满足签名权的安全传递等需求,需在数字签名的基础上进行扩展,从而产生了代理签名。文献[5]指出代理签名是指原始签名者可以授权代理者代表其行使签名权,验证者既能验证该签名的有效性,也能验证该签名是原始签名者所认可的^[13]。

代理签名的过程至少包含4部分,即代理授权、授权验证及代理密钥的生成、代理签名的生成、代理签名的验证。其流程如图1所示。

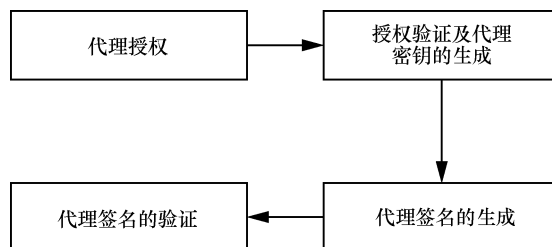


图1 代理签名的基本流程

Mambo等将代理签名分成3种:完全代理签名、部分代理签名和具有证书的代理签名^[10]。这里仅对部分代理签名进行介绍。

部分代理签名(partial delegation):原始签名者利用自己的私钥 s 计算出一个新的代理私钥 σ (为了安全考虑,要求不能由 σ 推导出 s),并通过安全通道将 σ 交给代理签名者,代理签名者可利用 σ 生成代理签名^[14]。该方法有以下2种类型的实现方案。

代理非保护代理签名(proxy-unprotected proxy signature):除了原始签名者外,只有被授权的代理签名者能够产生代理签名,其他第三方不能产生代理签名。

代理保护代理签名(proxy-protected proxy signature):除了被授权的代理签名者能够产生代理签名之外,包括原始签名者在内的其他任何人都不能产生代理签名。

2.2 代理签名特点

根据代理签名的定义,文献[10]指出代理签名体制应具备以下性质。

- 1) 可验证性:验证者能够验证代理签名是有效的。
- 2) 可区分性:代理签名和原始签名能够被其他人所区分。
- 3) 不可伪造性:除了原始签名者及其授权的代理签名者之外,其他人不能产生有效的代理签名。
- 4) 可识别性:代理签名者的身份能够被原始签名者从有效的代理签名中识别出来。
- 5) 不可否认性:代理签名者不能在原始签名

者面前否认自己所产生的签名。

文献[14,15]对上述的某些性质给出了更强的定义, 以确保对原始签名者和代理签名者是公平的, 举例如下。

6) 强不可伪造性: 原始签名者以及任何没有被授权的人均不能创建有效的代理签名。

实际应用中的代理签名方案并不一定要满足以上所有性质, 对于不同的应用场景可以提出不同的代理签名方案。

3 SM2 数字签名

SM2 算法的主要内容包括 3 部分: 数字签名算法、密钥协商协议和公钥加密算法。SM2 的数字签名部分适用于商用密码中的数字签名和验证。在 GM/T 0003 中规定了 SM2 椭圆曲线公钥密码算法的数字签名算法。

3.1 SM2 参数说明

(d_A, P_A) : 原始签名者 Alice 的公私钥对, 其中 $P_A = d_A G$ 。

M : 待签名的消息。

M' : 待验证的消息。

e : 密码杂凑函数作用于消息 M 的输出值。

e' : 密码杂凑函数作用于消息 M' 的输出值。

G : 椭圆曲线的一个基点, 其阶为素数 n 。

$H()$: 消息摘要函数。

$\text{mod } n$: 模 n 运算。

(r, s) : 发送的签名。

(r', s') : 收到的签名。

ID_A : 作为用户 A 具有长度为 entlen_A 比特的可辨别标识, $ENTL_A$ 是由整数 entlen_A 转换而成的 2 个字节。

$Z_A = H_{256}(ENTL_A \| ID_A \| a \| b \| x_G \| y_G \| x_A \| y_A)$ 。

$\overline{M} = Z_A \| M$, $\overline{M}' = Z_A \| M'$ 。

3.2 数字签名的生成算法

设待签名的消息为 M , 为了获得消息 M 的数字签名 (r, s) , 作为签名者的用户 A 可以实现以下运算步骤。

A1: 置 $\overline{M} = Z_A \| M$ 。

A2: 计算 $e = H(\overline{M})$ 。

A3: 用随机数发生器产生随机数 $k \in [1, n-1]$ 。

A4: 计算椭圆曲线点 $(x_1, y_1) = [k]G$ 。

A5: 计算 $r = (e + x_1) \text{mod } n$, 若 $r = 0$ 或 $r+k=n$, 则返回 A3。

A6: 计算 $s = ((1 + d_A)^{-1}(k - rd_A)) \text{mod } n$, 若 $s = 0$, 则返回 A3。

A7: 消息 M 的签名为 (r, s) 。

3.3 数字签名的验证算法

为了检验收到的消息 M' 及其数字签名 (r', s') , 作为验证者的用户 B 应实现以下运算步骤。

B1: 检验 $r' \in [1, n-1]$ 是否成立, 若不成立, 则验证不通过。

B2: 检验 $s' \in [1, n-1]$ 是否成立, 若不成立, 则验证不通过。

B3: 置 $\overline{M}' = Z_A \| M'$ 。

B4: 计算 $e' = H(\overline{M}')$ 。

B5: 计算 $t = (r' + s') \text{mod } n$, 若 $t = 0$, 则验证不通过。

B6: 计算椭圆曲线点 $(x_1', y_1') = [s']G + [t]P_A$ 。

B7: 计算 $R = (e' + x_1') \text{mod } n$, 检验 $R = r'$ 是否成立, 若成立, 则验证通过; 否则验证不通过。

文献[16]分析指出, SM2 数字签名使用了用户的参数及其系统参数, 提高了系统的安全性和认证性; 并且加入了较多的校验功能, 提高了签名验证系统的数据完整性和系统可靠性。

4 基于 SM2 的代理保护代理签名

本节将上述 SM2 数字签名体制与代理签名思想相结合, 设计了一种基于 SM2 的代理保护代理签名算法。在该算法中, 代理签名人 Bob 将代表原始签名人 Alice 签名, 验证者 Carol 对产生的代理签名进行验证。

4.1 参数说明

本节使用的部分参数与 3.1 节中的参数一致, 另外:

(d_B, P_B) : 代理签名者 Bob 的公私钥对, 其中 $P_B = d_B G$;

s_A : 原始签名者对代理签名者的委托密钥;

d_B : 代理签名密钥。

4.2 代理签名的流程

根据图 1 代理签名的基本流程, 算法主要进行了 4 部分, 其中 A、B、C 分别代表 Alice、Bob、Carol 的操作。

4.2.1 代理授权

在代理授权过程, 原始签名者 Alice 与代理签名者 Bob 要进行信息交互, Alice 根据交互信息生成授权信息, 具体过程如下。

B1: 用随机数生成器产生随机数 k_b , 其中 $k_b \in [1, n-1]$ 。

B2: 计算椭圆曲线上的点 $G_b = [k_b]G$ 。

B3: Bob 将 G_b 发送给 Alice。

其次, Alice 会进行下面的运算步骤。

A1: 用随机数生成器生成随机数 k_a , 其中 $k_a \in [1, n-1]$ 。

A2: 计算椭圆曲线上的点 $G_a = [k_a]G$ 。

A3: 计算椭圆曲线上的另一个点 $G_{ab} = (x_1, y_1) = [k_a]G_b$ 。

A4: 计算 $r_{ab} = x_1 \bmod n$, 若 $r_{ab} = 0$, 则返回步骤 A1。

A5: 计算 $s_A = k_a^{-1} r_{ab} d_A \bmod n$, 若 $s_A = 0$, 则返回步骤 A1。

A6: Alice 将 (G_a, G_{ab}, s_A) 作为代理签名的授权信息发送给 Bob。

其中, Alice 应向权威机构申请将公开的 G_a 及 G_{ab} 与 Bob 的身份信息进行绑定生成证书。这时, G_a 和 G_{ab} 的公开有两个作用: 一是对外界宣布代理签名者的身份; 二是提供授权验证和代理签名验证过程所需数据。但 s_A 需要 Alice 秘密发送给 Bob, 用于 Bob 计算代理签名密钥。

4.2.2 授权验证及代理密钥生成

当 Bob 收到授权信息后, 他首先要对授权信息进行验证, 确定授权信息确实来自原始签名者 Alice, 具体过程如下。

B4: 计算椭圆曲线上另一点

$$G_{ab} = (x_2, y_2) = [k_b]G_a。$$

B5: 计算 $r_{ab}' = x_2 \bmod n$, 当且仅当 $s_A G_a = r_{ab}' P_A$ 时接受委托。

B6: 一旦 Bob 接受了委托, 将计算 $d_B = s_A k_b^{-1} \bmod n$ 作为代理密钥进行签名。

4.2.3 代理签名的生成

当 Bob 需要代替 Alice 执行签名操作时, 使用代理密钥 d_B 进行签名。签名步骤如下。

B7~B9 同 3.2 节中的 A1~A3。

B10: 计算 $(x_3, y_3) = kG_{ab}$ 。

B11: 令 $r = (e + x_3) \bmod n$, 若 $r = 0$ 或

$r + k = n$, 则返回步骤 B9。

B12: 计算签名 $s = ((1 + d_B)^{-1}(k - rd_B)) \bmod n$, 若 $s = 0$ 则返回步骤 B9。

B13: 此时消息 M 的代理签名是 (G_{ab}, r, s) 。

此处值得注意的是, 在 B12 中计算签名利用的是代理签名密钥 d_B , 因为此密钥只有代理签名者知道, 所以其他所有人 (包括原始签名者) 均不能生成有效的代理签名, 即实现了对代理签名者的保护。

4.2.4 代理签名的验证

为了检验收到的消息 M' 及其数字签名 (G_{ab}, r', s') , 作为验证者 Carol 应实现以下运算步骤。

C1~C5 同 3.3 节中的 B1~B5。

C6: 计算椭圆曲线点 $X = (x_3', y_3') = [s']G_{ab} + [t]r_{ab}'P_A$, 计算 $R = (e' + x_3') \bmod n$, 当且仅当 $R = r'$ 时接受签名; 否则签名不能通过验证, “拒绝接受签名”。

4.3 代理签名方案的正确性和可验证性

1) 代理委托授权验证的正确性

定理 1 如果委托证书 (G_a, G_{ab}, s_A) 是有效的, 那么 $s_A G_a = r_{ab}' P_A$ 就是成立的。

证明 首先, 证明 $r_{ab}' = r_{ab}$ 。

由 $(x_2, y_2) = [k_b]G_a = k_b k_a G = k_a G_b = (x_1, y_1)$, 可得 $r_{ab}' = x_2 \bmod n = x_1 \bmod n = r_{ab}$ 。

又因为

$$s_A = k_a^{-1} r_{ab} d_A \bmod n = k_a^{-1} r_{ab}' d_A \bmod n$$

方程两边同时乘以 G , 则

$$s_A k_a G = r_{ab}' d_A G \Rightarrow s_A G_a = r_{ab}' P_A$$

由此可以证明 Bob 对 Alice 的代理委托授权信息 (G_a, G_{ab}, s_A) 是可以验证通过的。

2) 代理签名的可验证性

定理 2 该方案具有可验证性, 即验证者可以验证代理签名的有效性。

证明 由 $s = ((1 + d_B)^{-1}(k - rd_B)) \bmod n$ 得出

$$\begin{aligned} k &= ((1 + d_B)s' + r'd_B) \bmod n \\ &= (s' + d_B(s' + r')) \bmod n \\ &= (s' + d_B t) \bmod n \end{aligned}$$

方程两边同时乘以 G_{ab} , 则有

$$kG_{ab} = (s' + d_B t)G_{ab}$$

$$\begin{aligned}
&= s'G_{ab} + d_B t G_{ab} \\
&= s'G_{ab} + s_A k_b^{-1} t k_a k_b G \\
&= s'G_{ab} + s_A t k_a G \\
&= s'G_{ab} + k_a^{-1} r_{ab} d_A t k_a G \\
&= s'G_{ab} + r_{ab} t d_A G \\
&= s'G_{ab} + r_{ab} t P_A
\end{aligned}$$

上述过程证明了该算法满足代理签名可验证的性质，并且验证者可以直接使用原始签名者的公钥进行验证。

4.4 安全性分析

本节讨论几种针对拟议方案的安全性的可能攻击，该方案的安全性是基于以下两点：1) 散列函数单向性；2) ECDLP 的困难性。

攻击场景 1 原始签名者伪造有效的代理签名。

由 $s = ((1 + d_B)^{-1}(k - r d_B)) \bmod n$ 可知，原始签名者必须伪造出适当的 k 以及有效的代理密钥 d_B ，即满足以下 2 种情形之一才能伪造有效代理签名。1) 原始签名者需要知道代理签名者在代理授权阶段生成的随机数 k_b 的值，以求得 $d_B = s_A k_b^{-1} \bmod n$ 。然而随机数 k_b 本身是非公开的，而根据 ECDLP 可知，即使已知 $G_{ab} = [k_b]G_a$ ， k_b 的值也很难确定。2) 通过随机选取 k 和 d_B 得到正确的代理签名，这几乎是不可能的。

此处分析表明，原始签名者不能伪造代理签名，即满足强不可伪造的性质。

攻击情景 2 恶意代理签名者假冒原始签名者创建委托证书 (G_a, G_{ab}, s_A) 。

假设恶意代理签名者可以假冒原始签名者创建委托证书 (G_a, G_{ab}, s_A) ，那么他必须获得正确的 s_A ，才能通过代理验证阶段。由于 $s_A = k_a^{-1} r_{ab} d_A \bmod n$ ，这就意味着恶意代理签名者可以导出 $d_A = k_a s_A r_{ab}^{-1} \bmod n$ ，即攻击者可以导出原始签名者的秘密密钥 d_A 。这就与椭圆曲线密码系统的标准矛盾。所以攻击情景 2 不成立。

另外，综合攻击情景 1 的分析，代理签名者利用盲因子 k_b 对自己进行了保护，因此本文所提出的方案具有强不可伪造性以及代理签名者具有保护的功能。

攻击场景 3 恶意的代理签名者在接受原始签名者的委托证书 (G_a, G_{ab}, s_A) 后伪造委托证书 (G_a, G_{ab}', s_A') 。

假设恶意的代理其签名者通过计算 $G_{ab}' = cG_{ab} = (x', y')$ （其中， c 为任意随机数）， $\overline{r_{ab}} = x' \bmod n$ ， $s_A' = s_A r_{ab}^{-1} \overline{r_{ab}}$ 生成 (G_a, G_{ab}', s_A') 。虽然 (G_a, G_{ab}', s_A') 能够通过 4.2.2 节中 B5 阶段的验证公式，但在 4.2.1 节中已表明将 (G_a, G_{ab}', s_A') 发送给代理签名者时，将 G_a, G_{ab}' 与代理签名者身份信息绑定生成了证书，该证书也包含了原始签名者的信息。所以即使代理签名者成功伪造了 (G_a, G_{ab}', s_A') ，也无法向权威机构申请有效的委托证书。

攻击场景 4 攻击者假冒代理签名者，在代理授权过程与原始签名者交互产生代理密钥以进行中间人攻击。

借助于权威机构对原始签名者和代理签名者的公钥生成的数字证书可以有效地防止该中间人攻击。

4.5 方案所满足的性质

1) 可验证性

在 4.3 节中的定理 2 证明了该性质。

2) 可区分性

验证者可以利用原始签名的验证公式以及代理签名的验证公式对原始签名和代理签名进行区分。

3) 强不可伪造性

通过对 4.4 节攻击情景 1 的分析，可以确定该方案具有强不可伪造性。

4) 可识别性

由于在代理授权阶段 Alice 应向权威机构申请将公开的 G_a 及 G_{ab} 与 Bob 的身份信息进行绑定生成证书。所以原始签名者可以识别代理签名者的身份。

5) 不可否认性

由 4.2.1 节可得，由于进行了身份绑定所以代理签名者不能在原始签名者面前否认自己的签名。

5 实验分析

为了测试所提出的方案的正确性，本文使用 Microsoft Visual Studio 2010 工具以及 C/C++ 编程语言，在 64 位 Windows 操作系统上采用 Polarssl 中的密码算法，实现了基于 SM2 的代理保护代理签名算法。

5.1 实验参数

SM2 使用的素数域 256 位椭圆曲线的曲线参数如下。

```
p=FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF 00000000 FFFFFFFF FFFFFFFF
a=FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
FFFFFFFF 00000000 FFFFFFFF FFFFFFFC
b=28E9FA9E 9D9F5E34 4D5A9E4B CF6509A7
F39789F5 15AB8F92 DDBCBD41 4D940E93
n=FFFFFFFF FFFFFFFF FFFFFFFF FFFFFFFF
7203DF6B 21C052B 53BBF409 39D54123
xG=32C4AE2C 1F198119 5F990446 6A39C994
8FE30BBF F2660BE1 715A4589 334C74C7
yG=BC3736A2 F4F6779C 59BDCEE3 6B692153
```

D0A9877C C62A4740 02DF32E5 2139F0A0

其中, G 是椭圆曲线上的一个基点。

实验中对基于 SM2 代理保护代理签名的参数的输出如图 2 所示。

代理授权阶段的输出如图 3 所示。

授权验证及代理密钥生成阶段的输出如图 4 所示。

代理签名阶段的输出如图 5 所示。

代理签名验证阶段的输出如图 6 所示。

5.2 方案的复杂性分析

表 1 是对标准 SM2 数字签名与 SM2 代理保护代理签名的计算复杂度的分析(其中, T_M 表示点乘次数; T_{MM} 表示模乘次数; T_{INV} 表示模逆次数)。

```
*****基于SM2代理保护代理签名的参数*****
原始签名者的公钥-PA.X= 9A13A6727BBA1A78846634D8CC7434FBCCDB94A93E12AB3727321677479B2425
原始签名者的公钥-PA.Y= E4CD0BB8B5B1D27B34A69A59A2E59451322420932008508E88A2570EA3268047
原始签名者的私钥-dA= 9FCF861965027F235D41DD3700FE4D7FE2F73417DFA5819FEB5CD5C6F56FB113
```

图2 基于 SM2 代理保护代理签名的参数

```
*****原始签名者进行代理授权*****
代理签名者生成随机数Kb= C1A65FF676965A313877BA3BCE49DF063B13C256D1C21604C9177672D0FCC693
代理签名者计算得到的Gb.X= 32C4AE2C1F1981195F9904466A39C9943FE30BBFF2660BE1715A4589334C74C7
代理签名者计算得到的Gb.Y= BC3736A2F4F6779C59BDCEE36B692153D0A9877CC62A474002DF32E52139F0A0
原始签名者生成随机数Ka= F2682192FBFBFBOD147C4783C7A6EE9DF59EC9A6EBB7322C935E3421A5AA0D66
原始签名者计算得到的Ga.X= F198BCBD9F5745C802FCB7489AA7077039765735C95875C4D2043101EF941D7E
原始签名者计算得到的Ga.Y= 30EC63EE8DCE290FE698D172D55127EE4C96B949368C98CDC15D7CB0F5E064AC
原始签名者计算得到的Gab.X= 09156415692E7A116D4899E98CBF4D780AFA1F88E66A09DE0B4F63E17C818C3
原始签名者计算得到的Gab.Y= C59CDD4C4A61711C6A4728DF775C53DFE3FC385AB5A76B16149B862C39A4C8CA
原始签名者计算得到的Sa= 3626181AC837084E97FE7B2D759E247A33E2EDC50BB6FF32CB1AF0A1D7C1B721
原始签名者将Ga, Gab发给代理签名者
原始签名者将Sa通过安全通道发给代理签名者
```

图3 代理授权阶段

```
*****代理签名者进行授权验证及代理密钥的生成*****
代理签名者计算出来的Sa*Ga的X坐标= 5482481E3B6BA7199DBD7C85C3E0FE1671EA58FD7DBD793C32F25D61083A13B0
代理签名者计算出来的Sa*Ga的Y坐标= D5D7D119B8802382CECC35C783887FF12FD60CA18A679EE561546619ADEACF76
代理签名者计算出来的rab'*Pa的X坐标= 5482481E3B6BA7199DBD7C85C3E0FE1671EA58FD7DBD793C32F25D61083A13B0
代理签名者计算出来的rab'*Pa的Y坐标= D5D7D119B8802382CECC35C783887FF12FD60CA18A679EE561546619ADEACF76
Sa*Ga=rab'*Pa代理签名者授权验证成功!
代理签名者的代理签名密钥dB= 97E6C84D36DED1A1C408308F0FAB909CD6884EC07747FEF9011B232060FF6910
```

图4 授权验证及代理密钥生成阶段

```
*****代理签名者进行代理签名*****
代理签名者进行签名后得到(r, s)中的r= C730B9014D3878E58800BD4ED0FD47393A86D169C7B64F8F424C82EE21FD3433
代理签名者进行签名后得到(r, s)中的s= 186B62ABC09C4EC7D8F6FC35DE6C3E4FA73235E526BD61E3740912EBC815A04
```

图5 代理签名阶段

```

*****验证者对代理签名进行验证*****
验证者已知的信息：原始签名者的ID, 原始签名者的公钥, 消息m, 消息的签名 (r', s'), 以及公开信息Ga, Gb
代理签名验签时接受到的(r', s')的r' = C730B9014D3878E58800BD4ED0FD47393A86D169C7B64F8F424C82EE21FD3433
代理签名验签时接受到的(r', s')中的s' = 186B62ABC09C4BC7D8F6FC35DB6C3E4FA73235E526BD61E3740912EBC815A04
代理签名验签时计算的[s']Gb + [rab][t]Pa的X坐标= B47A017EC4DE827DAA2C2EACFA017DF940FCC4F793FB11A2F780EFF4712DC801
代理签名验签时计算的[s']Gb + [rab][t]Pa的Y坐标= 4C2C92E8C2E284FFB6A1228A8222E78E0B10D83712E37353BC6E73F58A71D0
代理签名验签时计算出的R= C730B9014D3878E58800BD4ED0FD47393A86D169C7B64F8F424C82EE21FD3433
R=r' 接受签名

```

图6 代理签名验证阶段

表1 该方案与标准 SM2 数字签名计算复杂度比较

方案	代理授权	代理验证及密钥生成	签名	验签
SM2 数字签名	—	—	$T_M + 2T_{MM} + T_{INV}$	$2T_M + 2T_{MM}$
基于 SM2 的代理保护代理签名	$3T_M + 3T_{MM} + T_{INV}$	$3T_M + 3T_{MM} + T_{INV}$	$T_M + 2T_{MM} + T_{INV}$	$2T_M + 3T_{MM}$

实验中对标准 SM2 数字签名与 SM2 代理保护代理签名的签名验签速度进行了多次测试, 计算了两者的平均时间如表 2 所示。

表2 该方案与标准 SM2 数字签名签名验签速度比较

方案	签名速度 (毫秒/次)	验签速度 (毫秒/次)
SM2 数字签名	35.2	139.7
基于 SM2 的代理保护代理签名	107.5	306.3

可以看出, 代理保护代理签名的签名时间大约是标准签名的 3 倍, 这是因为在 4.2.3 节中代理签名生成阶段的 B10 中用到的基点 $G_{ab} = k_a k_b G$ 比标准签名中的基点 G 大很多。表 2 中还显示该方案验签也比标准 SM2 验签慢, 这是由于该方案验签时计算的 $(x_3', y_3') = [s']G_{ab} + [t]r_{ab}P_a$ 比标准 SM2 验签时计算的 $(x_1', y_1') = [s']G + [t]P_a$ 多了一次标量乘法。以上实验数据表明, 基于 SM2 的代理保护代理签名方案是可以通过编程实现的, 但签名验签速度还有待进一步提高。

6 结束语

本文将代理签名思想应用于 SM2 数字签名, 提出了基于 SM2 的代理保护代理签名方案, 同时对该方案进行了安全性和复杂性的分析。分析表明, 该方案不仅满足代理签名的安全性, 还实现了对代理签名者的保护。实验结果表明, 本方案的签名验签速度稍慢于 SM2 数字签名, 但仍具有较好的可用性。

参考文献:

- [1] NEAL K. Elliptic curve cryptosystems[J]. Mathematics of Computation, 1987, 48(177): 203-209.
- [2] 杜洁璇. 椭圆曲线密码系统的研究与实现[D]. 北京: 北京邮电大学, 2010.
- [3] 李湛. 一种改进的椭圆曲线密码实现算法[J]. 电子科技, 2004(7): 31-33.
- [4] LI Z. An improved algorithm for Elliptic curve cryptography [J]. Electronic Science and Technology, 2004 (7): 31-33.
- [5] 汪朝晖, 张振峰. SM2 椭圆曲线公钥密码算法综述[J]. 信息安全研究, 2016, 2(11): 972-982.
- [6] WANG ZH, ZHANG Z F. Overview of SM2 Elliptic curve public key cryptography [J]. Information Security Research, 2016, 2 (11): 972-982.
- [7] MAMBO M, USUDA K, OKAMOTO E. Proxy signatures for delegating signing operation[C]. ACM Conference on Computer and Communications Security, 1996: 48-57.
- [8] 白国强, 黄淳, 陈弘毅, 等. 基于椭圆曲线的代理数字签名[J]. 电子学报, 2003, 31(11): 1659-1663.
- [9] BAI G Q, HUANG Z, CHEN H Y, et al. Proxy digital signature based on elliptic curve [J]. Acta Electronic Journal. 2003, 31 (11): 1659-1663.
- [10] 张宁, 傅晓彤, 肖国镇. 对基于椭圆曲线的代理签名的研究与改进[J]. 西安电子科技大学学报, 2005, 32(2): 280-283.
- [11] ZHANG N, FU X T, XIAO G Z. Study and improvement of agent signature based on Elliptic Curve[J]. Journal of Xidian University, 2005, 32 (2): 280-283.
- [12] CHANG M H, CHEN I T, CHEN M T. Design of Proxy Signature in ECDSA[C]//Eighth International Conference on Intelligent Systems Design and Applications, 2008. 2008: 17-22.
- [13] 胡兰兰, 郑康锋, 杨义先, 等. 一种改进的椭圆曲线安全代理签名方案[J]. 计算机应用研究, 2010, 27(2).

HU L L, ZHENG K F, YANG Y X, et al. An improved elliptic curve security proxy signature scheme [J]. Journal of Computer Applications, 2010, 27 (2).

- [10] 李继国, 曹珍富, 李建中, 等. 代理签名的现状与进展[J]. 通信学报, 2003(10): 114-124.

LI J G, CAO Z F, LI J Z, et al. Status and progress of proxy signature [J]. Journal of Communications, 2003 (10): 114-124.

- [11] DIFFIE W, HELLMAN M. Multiuser cryptographic techniques[J]. Proceedings of AFIPS 1976 NCC, 1976: 109-112.

- [12] DIFFIE W. The first ten years of public-key Cryptography[J]. IEEE, 1992: 135-175.

- [13] 周亮. 代理签名及扩展的关键问题研究[D]. 北京:北京邮电大学, 2008.

ZHOU L. Proxy signature and expansion of the key issues[D]. Beijing: Beijing University of Posts and Telecommunications, 2008.

- [14] LEE B, KIM H, KIM K. Strong proxy signature and its applications[J]. SCIS, 2001.

- [15] VARADHARAJAN V, ALLEN P, BLACK S. An analysis of the proxy problem in distributed systems[C]// 1991 IEEE Computer Society Symposium on Research in Security and Privacy 1991: 255-275.

- [16] 张焕国. SM2 数字签名算法[EB/OL]. 2015.<https://wenku.baidu.com/view/9dbf8a3d04a1b0717ed5dd12.html>.

ZHANG H G. SM2 digital signature algorithm[EB/OL]. <https://wenku.baidu.com/view/9dbf8a3d04a1b0717ed5dd12.html>.

作者简介:



郭青霄 (1992-), 女, 山东聊城人, 北京交通大学硕士生, 主要研究方向为支付安全、密码学货币。



张大伟 (1974-), 男, 辽宁朝阳人, 博士, 北京交通大学副教授, 主要研究方向为智能卡安全、支付安全和区块链技术。



常亮 (1980-), 男, 贵州赫章人, 博士, 桂林电子科技大学教授, 主要研究方向为知识表示与推理、可信计算。



刘晓东 (1975-), 男, 山东潍坊人, 博士, 山东大学讲师, 主要研究方向为信息安全及密码学。



宋靖文 (1995-), 女, 河北衡水人, 北京交通大学硕士生, 主要研究方向为支付安全、密码货币。