椭圆曲线的可追踪门限签名

景运革

(运城学院 公共计算机教学部,山西 运城 044000)

摘 要：提出新的(t,n)门限数字签名方案,即基于椭圆曲线可追踪的门限数字签名方案,构造以椭圆曲线为基础和核心的门限数字签名,以实现用更短的密钥达到和 RSA 同等安全强度的门限签名系统。同时,实现在事后发生争执或需要追究责任时,可以通过仲裁机制追查出参与签名成员的身份。

关键词：椭圆曲线密码体制;数字签名;门限签名;可追踪

中图分类号：TP309 **文献标识码**：A **文章编号**：1674-7720(2010)21-0058-03

Elliptic curve traceable threshold signature

JING Yun Ge

(Public Computer Education, Yuncheng University, Yuncheng 044000, China)

Abstract: This paper proposes a new (t, n) threshold digital signature scheme based on elliptic curve traceable threshold signature scheme. In theory, the structure to the elliptic curve-based and core threshold digital signature scheme to achieve with a shorter and equal security to RSA key strength threshold signature system. Meanwhile, the realization of a dispute after the fact or in need of accountability, they can participate by signing an arbitration mechanism to trace the identity of the members.

Key words: elliptic curve cryptography; digital signature; threshold signature; traceable

自 1991 年 DESMEDT F 首次提出门限签名方案以来,门限签名引起了密码学界的广泛关注和研究,并且提出了各种各样的(t,n)门限群签名方案[1],也对这些方案提出了很多攻击方法和改进措施。与普通的数字签名相比,由于门限群签名需要多方参与,其安全性和健壮性有了很大的提高;与群签名相比,门限签名具有易操作性和方便性[2]。

椭圆曲线密码体制 ECC (Elliptic Curve Cryptography)[3-5] 的安全性是基于椭圆曲线上离散对数问题 ECDLP(Elliptic Curve Discrete Logarithm Problem)的。与其他公钥密码相比,椭圆曲线具有每比特数据最高的安全强度,这样的好处是计算参数更小、密钥更短、运算速度更快、签名也更加短小。

参考文献[1]证明了(t,n)门限群签名方案不能抵抗合谋攻击和伪造攻击,也不具备可追踪性。本文针对这些问题对上述方案进行了改进,提出了一种基于椭圆曲线的可追踪门限数字签名方案。该方案以椭圆曲线为基础,采用二次签名等方式,可有效地避免参考文献[1]所暴露的缺陷和不足。

1 椭圆曲线的可追踪门限签名方案

该方案根据分工不同,有三种角色,即签名者、签名组合者和秘密处理者。

签名者 p_i 进行门限签名操作。用集合 $T=\{p_1、p_2、p_3\cdots p_n\}$ 表示由 n 个签名者组成的签名者群体。该方案主要由参数选择、子密钥产生过程、签名过程、签名验证和事后追踪等 5 个部分组成。

签名组合者 C,收集单个签名者的操作结果,然后将收集的数据进行验证并组合。C 同时是群签名消息的唯一发布者,C 自选一个合适的签名体制 (称为群签名消息发布签名系统)来对群签名消息再次签名。这样通过将群签名消息的发布权进行控制以提高系统安全。C 保留其私钥 D_c,公开公钥 E_c。

秘密处理者 D,即可信任中心,主要是在系统初始化阶段处理、协调系统参数设置和子密钥分配等操作。一旦系统初始化成功则退出签名系统。

1.1 参数选择

设 $P、q$ 为大素数,$P|q-1$,F_q 是元素阶为 q 的有限

域，g 是 F_q 中阶为 P 的元素。E 是定义在 F_q 上的安全椭圆曲线，并保证其离散对数问题是难解的。$H()$ 是单向 Hash 函数。m 为消息内容。可信任中心 D 选取基点 P，使 $P\in E(F_q)$，其阶数 u 为大素数。d 为签名者群体 $T=\{p_1、p_2、p_3\cdots p_t\}$ 的私钥。T 的公开密钥为[5] $Q=dp\in E(F_q)$。

设 d_1 是签名者 p_i 的私钥，则 p_i 公钥为 $Q_i=d_ip$，$1\leqslant i\leqslant n$。因此，T 的公开系统参数是 $(E,P,Q,q,p,g,H())$。

1.2 子密钥 d_i 产生过程

可信任中心 D 将 T 的私有密钥 d 分成 n 等份，然后分发给 T 中的每一个成员 p_i，使得人数 $\geqslant t$ 的任意成员组合均可代表该群体签名，而任何成员人数 $<t$ 的子集不能签名。具体分发过程采用 Shamir 密码共享体制[6]，它基于 Lagrange 插值公式来实现的。

首先，秘密处理者 D 选择一个 $d\in F_q$ 作为签名者群体的私钥。其次，D 从 F_q 中选取 n 个不同的非零元 u_i $(1\leqslant i\leqslant n)$，$u_i$ 为签名者 p_i 的身份标记。然后，D 把 u_i 传给签名组织者 C 和 $p_i(1\leqslant i\leqslant n)$。为了实现 T 中的所有签名者共享私有密钥 d，D 在 F_q 上随机选取次数为 $t-1$ 的多项式 $f(x)$[7]。$f(x)=f_0+f_1x+f_2x^2+\cdots f_{t-1}x^{t-1}(\mathrm{mod}q)$，其中 $f(0)=f_0=d$，$f_1f_2\cdots f_{t-1}$ 为随机数。最后 D 计算多项式 $d_i=f(u_i)(\mathrm{mod}q)$，并将 d_i 秘密地传送给 p_i，d_i 就是签名者 p_i 的子密钥 $(1\leqslant i\leqslant n)$。

任何 t 个签名者可通过 Lagrange 内插法重构多项式，为不失一般性，设 t 个成员为 $p_1、p_2、p_3\cdots p_t$。

$$f(x)\sum_{i=1}^{t}\prod_{h\neq i,h=1}^{t}\frac{x-u_h}{u_i-u_h}f(u_i)(\mathrm{mod}q)=\sum_{i=1}^{t}\left(\prod_{h\neq i,h=1}^{t}\frac{x-u_h}{u_i-u_h}\right)d_{ui}(\mathrm{mod}q)$$

因此，密钥 d 可由 $d=f(0)$ 来恢复，即：

$$d=\sum_{i=1}^{t}b_id_{ui}，\text{其中 } b_1、b_2、b_3\cdots b_t \text{ 为 } b_i=\prod_{h\neq i,h=1}^{t}\frac{u_h}{u_h-u_i}$$

当 $t-1$ 个签名者想要恢复密钥 d 时，可以得到含有 t 个未知数的 $(t-1)$ 个线性方程，因此不能恢复 d。从而也就不能代表群体进行签名。

1.3 签名过程

在本方案中，签名的具体实施步骤如下：

(1) 签名组织者 C 从集合 T 中选择 t 个签名者要求其对文件 m 进行签名。若签名者同意签名，则回复肯定信息；否则，C 从剩余的签名者中重新选择一个以代替拒绝签名的成员，直到 C 收到 t 个肯定回复。为不失一般性，设同意签名者为 $p_1、p_2、p_3\cdots p_t$，每个签名者使用各自的子密钥为 $d_1、d_2、d_3\cdots d_t$。C 构建函数 $g(x)\prod_{i=1}^{t}(x-u_i)$ 计算 b_i，并发送给 p_i。

(2) 各签名者随机产生一个数 k_i，计算 $w_i=k_ip$ 和 $Q_i=d_ip$，并把 w_i 和 Q_i 发送给签名组织者 C。

(3) 签名组织者 C 计算点 $(x,y)=\sum_{i=1}^{t}w_i=kp$。设 $k=\sum k_i$，C 计算 $r=g^{x'}(\mathrm{mod}q)$，其中 x 为点 KP 的横坐标。再将 r 发给各签名者。

(4) 各签名者计算 $s_i=k_i+d_ib_iH(r|m)\mathrm{mod}\ q$，并 s_i 把发送给 C。

(5) C 计算并验证各签名者 p_i 的 w_i 是否满足 $w_i=s_ip-H(r|m)\mathrm{mod}\ b_iQ_i$，如果不满足则拒绝该签名者签名并结束本次签名过程。

(6) C 计算 $s=\sum_{i=1}^{t}s_i+g'(0)\mathrm{mod}\ q$，将 $(s,r,g(x))$ 用 C 的私有密钥 D_C 即群消息发布密钥进行签名，签名后分别为 $(s',r',g'(x))$。最后 C 将 $(s,r,g(x),s',r',g'(x),m)$ 输出作为签名者集合 T 对消息 m 的签名。

1.4 签名验证

接收者接收到消息后，需对签名进行验证。具体验证过程如下：

(1) 接收者首先用签名组织者 C 的公钥 E_c 处理，判定 $(s',r',g'(x))$ 是否是 $(s,r,g(x))$ 的合法签名。如果不是，则认为本次签名过程出现异常，同时放弃后续验证。

(2) 接收者计算 $g'(0)$。

(3) 接收者计算点 $(x',y')=(s-g'(0))P-H(r|m)Q$。

(4) 求出 r''，判断 r'' 是否等于 r，若是，则接受该签名，否则拒绝。具体计算和判断如下：

在本方案中，$s=\sum_{i=1}^{t}s_i+g'(0)\mathrm{mod}\ q$，$s_i=k_i+d_ib_iH\ (r|m)\mathrm{mod}\ q$，所以有：

$$s=\sum_{i=1}^{t}s_i+g'(0)\mathrm{mod}\ q=\sum_{i=1}^{t}(k_i+d_ib_iH(r|m)\mathrm{mod}\ q)=k+dH(r|m)+g'(0)\mathrm{mod}\ q$$

$$(x',y')=(s-g'(0))P-H(r|m)Q=(s-g'(0))P-H(r|m)dp=(s-g'(0))P-dH(r|m)p=kp=(x,y)$$

再计算 $r''=g^{x'}(\mathrm{mod}q)$。

又因为 $r=g^{x'}(\mathrm{mod}q)$，所以，如果签名和验证都正确，则应该有 $r''=g^{x'}(\mathrm{mod}q)=g^{x}(\mathrm{mod}q)$，从而就有 $r=r''$。因此通过验证 r 和 r'' 是否相等就可以说明该签名是否有效。

1.5 事后追踪

在事后发生争执或需要追究责任时，签名接收者出示所接收到得的 $(s,r,g(x),s',r',g'(x),m)$。仲裁者首先利用 E_c 验证 $(s',r',g'(x))$ 是否是 $(s,r,g(x))$ 的签名。如果是，则可以认定 $(s,r,g(x),s',r',g'(x),m)$ 确实是签名组织者 C 发送的消息。然后利用将签名者的身份标志 μ_i 代入 $g(x)$ 中，凡是满足 $g(x)=0$ 的均表示曾参与了该次签名过程。因此，可以在事后确定有哪些签名成员参与了某次签名。

2 安全性分析

2.1 抗群内成员伪造签名攻击

方案从两个方面可以抵抗 T 集合内成员的这种欺骗、伪造签名。首先在方案的签名过程中，每个签名者 p_i 必须用所拥有的 d_i 来构造 s_i。当 $i\neq j$ 时，$d_i\neq d_j$，$s_i\neq s_j$。又由于 d_i 是签名者 p_i 的私钥，别人不得而知，因此其他人

不能冒充 p_i 进行签名。另一方面，在签名过程中，利用步骤(5)中签名组织者 C 对各签名者的 w_i 和 $s_ip-H(r|m)b_iQ_i$，进行判断是否相等，以判决是否存在伪造签名情况。如果不相等，则说明本次签名出现异常，C 就终止本次签名。从而可有效地保证各签名者身份的真实性。如果每个签名者都是诚实且正常，则：

$$s_ip-H(r|m)b_iQ_i=s_ip-H(r|m)d_ib_ip=(s_ip-H(r|m)d_ib_i)p=k_ip=w_i$$

否则该等式不成立。因此 C 可利用该等式对各签名者身份的真实性进行判断。

2.2 抗群内成员合谋攻击

由于本方案采用了二次签名的方式，同时签名者的选取是由 C 来完成的，因此可以有效地抵抗 T 内成员的合谋攻击。在整个签名过程中 C 都必须参与其中。签名者的选取必须由 C 来完成，b_i 和 g'(0) 等数据计算要 C 来完成，S 的合成也要由 C 来完成。即使假设在这些过程中合谋者绕过 C 来进行，不要 C 的参与，T 内成员的合谋也是不可能成功的。这是因为群签名消息的发布权限完全是局限在 C 的身上，没有 C 的参与，T 内的成员是无法产生有效的群签名消息的。因此没有最后一步 C 的签名，接收者可以立即发现该签名消息异常，从而拒绝该签名。

2.3 抗 C 伪造签名攻击

由于在本方案中，C 只是对签名过程进行组织以及对签名消息进行发布，他除了掌握签名发布密钥外并不掌握系统其他的任何密钥，即 C 不能获取到签名者的私有密钥 d_i，也就是不能伪造 T 内成员进行签名。同时也不能通过构建多项式获取群的私钥 d。这是因为，在 Lagrange 公式中要恢复 d 必须要有 t 个签名者的私钥 d_i 和身份标志 u_i。而 C 只能获得签名者的身份标志，但是不能获得签名者的私钥。同时，由于在本方案中对群签名消息发布的权限进行了绑定和限制，整个群消息的发布只能是 C 来完成。因此，如果 C 和 T 内的 t 个成员进行合谋对某个文件 m′进行非法签名，在事后追查时，则首先就可以确定 C 是参与了 m′的签名。这是因为只有 C 知道他自己的私钥 D_c，别人不得而知，其他人也就不能对群签名消息进行发布。因此，只要接收者公布其接收到对 m' 的签名消息，仲裁者就可以通过验证确认 C 是否参与了合谋。一旦发现 C 参与了合谋就必须对其进行严厉的制裁，使其承担所有责任。因此，通过绑定群签名消息的发布权限在 C 身上，采用二次签名的方式从实际上阻止了合谋攻击。

2.4 抗群外人员选择消息攻击

选择消息攻击是指攻击者通过分析签名群体对多个消息的签名，从而构造出对另一消息(设为 N)的有效签名[7]。

攻击者可从签名者群体公开的参数和发送的数据得到以下信息：$E,P,Q,q,g(x),H(),S,r$。如果群体外攻击者要对消息 N 构造出一个有效的签名，则必须通过对以前的签名进行分析 (假设以对消息 m 的签名进行分析为例)，以构造一个 S_0、e_0 和 $g_0(x)$。攻击者将面临以下难题：

(1)要构建 $r_0=g^x \bmod q$，x 是点 KP 的横坐标，虽然点 P 公开，但 K 是 t 个签名者随机选择的参数 k_i，而 k_i 是并不公开的。若试图从截获的 $w_i=k_ip$ 中求出 k_i，可以得知这是 ECDLP。因此要构建出 r_0 是非常困难的。

(2)要构建 s_0，也就是构建 $s_0=\sum s_i+g'(0)=\sum(k_i+d_ib_iH(r|N))+g'(0)=k+dH(r|N)+g'(0)(\bmod q)$ 则必须要计算出 $k+dH(r|N)+g'(0)(\bmod q)$ 即计算出 K 和 d。通过前面的分析可知，计算 K 是一个 ECDLP 问题。对于 d 它是签名者群体的私钥并不公开，所以是无从得到的。但攻击者在这里可以试图通过对消息 m 的签名数据来逐个分析出 d_i 然后进行组合。假设，攻击者截获 $s_i=k_i+d_ib_iH(r|m)+(\bmod q)$，由于 k_i 未知，当 q 是一个足够大素数时，要从中求出 d_i 在计算上是不可能的。若试图通过截获的 $Q_i=d_ip$ 中计算出来 d_i，这同样是 ECDLP 问题。

本文提出了一个基于椭圆曲线的可追踪 (t,n) 门限数字签名方案，并对该方案进行了安全性分析。在本方案中，除了秘密共享阶段需要保密通信外，在签名和验证过程中都不需要进行保密通信，这可以保证方案方便使用。基于椭圆曲线密码体制保证了方案的安全性。通过将群签名消息的权限绑定在 C 身上，采用二次签名的方式，不仅可以抵抗外部成员的攻击，也可以有效地抵抗 T 集合内部成员的合谋攻击，同时还可以有效地防止签名组织者 C 的攻击。

参考文献

[1] 黄梅娟，张建中.一种安全的门限群签名方案[J].计算机应用研究，2006(6)：116-117.

[2] 王化群，张力军，赵君喜.基于椭圆曲线的无可信中心(t，n)门限群签名[J].信号处理，2006，2(22)：189-192.

[3] 张方国，王常杰，王育民. 基于椭圆曲线的数字签名与盲签名[J]. 通信学报，2001，18(2)：186-192.

[4] 龙芳. 数字签名算法研究[J].信息安全与通信保密，2007(5)：143-145.

[5] 秦志光，张险峰，周世杰，等.基于 ECC 的门限数字签名方案及其安全性 [J]. 电子科技大学学报，2005，34(1)：109-112.

[6] 王书文.秘密分享密码体制进展[J].西北民族大学学报，2003，24(2)：53-68.

[7] 杨义先，钮心忻.应用密码学[J].北京：北京邮电大学出版社，2005.

(收稿日期：2010-06-11)

作者简介：

景运革，男，1971 年生，硕士，工程师，主要研究方向：网络信息安全。