基于证书的多代理多签名

周才学 谭旭杰

(九江学院信息科学与技术学院 九江 332005)

摘 要 给出了基于证书的多代理多签名的形式化定义和安全模型,并提出一个不使用双线性对的具体方案。在随机预言机模型中,在椭圆曲线离散对数是困难问题的假设下,对方案进行了安全性证明。效率分析表明,方案具有较高的效率。

关键词 基于证书的多代理多签名,随机预言机模型,公钥替换攻击,双线性对

中图法分类号 TP309 **文件标识码** A

Certificate-based Multi-proxy Multi-signature Scheme

ZHOU Cai-xue TAN Xu-jie

(School of Information Science and Technology, University of Jiujiang, Jiujiang 332005, China)

Abstract This paper gave out a formal definition and security notions of certificate-based multi-proxy multi-signature schemes, and proposed a concrete scheme without using bilinear pairings. It was proved to be existentially unforgeable in random oracle model under elliptic curve discrete logarithm assumption. Performance analysis shows the scheme is of high efficiency.

Keywords Certificate-based multi-proxy multi-signature, Random oracle model, Public key replacement attack, Bilinear pairings

1 引言

早期的公钥密码体制主要是基于PKI(public key infrastructure)的,它的主要缺点是公钥管理需要较高的费用。后来,人们提出了基于身份的密码体制[1],它可以有效地降低公钥的管理费用,但是存在密钥托管问题。于是,人们又提出了无证书密码体制[2],它既降低了公钥的管理费用又不存在密钥托管问题,展示了巨大的优越性。与此同时,人们还提出了基于证书的密码体制[3],它具有与无证书密码体制相同的优越性,同时又可克服无证书密码体制需要安全信道来传送部分私钥的问题,是一个很有前途的公钥密码体制。

代理签名[4]是某个原始签名人在自己不方便亲自签名的情况下,可以委托某个信任的人来代替他进行签名,该概念一经提出就受到了广泛的关注。2001年,Hwang等人[5]把代理签名的概念扩展到多代理多签名的情形,即一群原始签名人把他们的签名权委托给一群代理签名人,授权时必须所有原始签名人合作才行,生成代理签名时也必须所有代理签名人合作才行。

把基于证书的密码体制与代理签名的概念相结合的工作首先由Kang等人[6]给出。随后许多基于证书的各种代理签名方案被陆续提出[7-9]。但目前还未见有基于证书的多代理多签名的报道,所以本文提出一个基于证书的多代理多签名方案。

在代理签名的安全模型方面,Boldyreva等人[10]首先进行了尝试,他们的安全模型全面考虑了攻击者的攻击能力。随后,Wang等人[11]指出Boldyreva等人的安全模型存在两个缺陷并进行了改进。

在基于证书的签名的安全模型方面,Kang等人[6]首先进行了尝试。随后,Li等人[12]指出,在基于证书的密码体制中也存在公钥替换攻击。2008年,Wu等人[13]总结以前的基于证书的签名的安全模型,提出了3种安全模型,分别是一般攻击者模型、强攻击者模型和超级攻击者模型。此后,又陆续有基于证书的签名方案被提出,但都是基于已有的安全模型。

在基于证书的代理签名的安全模型方面,李继国等人[14]和陈江山[15]各提出一个安全模型,但他们的安全模型没有考虑攻击者伪造标准签名的问题。

基于以上分析,本文主要参考Wang等人[11]的代理签名的安全模型和Wu等人[13]的基于证书的一般攻击者模型,提出基于证书的多代理多签名的安全模型。然后,在随机预言机模型中对本文方案进行了安全性证明。最后,给出了方案的效率分析,结果表明方案是高效的。

2 一些基本概念

2.1 椭圆曲线离散对数问题

设E是定义在有限域F_q上的椭圆曲线,G是椭圆曲线上的加法群。给定$P, aP \in G, a \in Z_q$,要求计算a。

2.2 基于证书的多代理多签名的形式化定义

一个基于证书的多代理多签名方案由以下8个算法组成:

(1)初始化(Setup):输入安全参数k,算法输出系统主私

周才学(1966—),男,硕士,副教授,主要研究方向为密码学与网络信息安全。

钥 s、主公钥 P_{pub} 和系统公开参数 $Params$。

(2)用户密钥产生(UserKeyGen):输入系统公开参数 $Params$ 和用户身份 ID,算法输出用户的私钥和公钥对(x_{ID},y_{ID})。

(3)证书产生(CertGen):输入系统公开参数 $Params$、主私钥 s、用户身份 ID、用户公钥 y_{ID} 和时段参数 j,算法输出用户该时段的证书 $Cert_{ID,j}$。

(4)标准签名算法(StandardSign):输入系统公开参数 $Params$、用户密钥对(x_{ID},y_{ID})、用户证书 $Cert_{ID,j}$、消息 m,算法输出一个签名 σ。

(5)标准签名验证算法(StandardVerify):输入系统公开参数 $Params$、消息 m、签名 σ、用户的公钥 y_{ID},算法输出 1 或 0。如果输出是 1 表示签名正确,否则表示签名不正确。

(6)代理授权产生(DelegationGen):输入系统公开参数 $Params$、所有原始签名人的密钥对(x_{o_i},y_{o_i})($i=1,\cdots,n$)和证书 $Cert_{o_i,j}$($i=1,\cdots,n$)、授权信息 m_w(它由所有原始签名人共同产生,记录着对代理消息类型的限制、代理的有效期、原始签名者和代理签名者的身份和公钥等信息),算法用标准签名算法对授权信息 m_w 进行签名,最后输出代理授权 $Cert_{op}$ 并传输给所有代理签名者。每个代理签名人都可验证代理授权的正确性,如果不正确,则重新生成代理授权。

(7)多代理多签名算法(MProxyMSign):输入系统公开参数 $Params$、授权信息 m_w、所有代理签名人 ID_{p_i}($i=1,\cdots,n$)的密钥对(x_{p_i},y_{p_i})和证书 $Cert_{p_i,j}$、代理授权 $Cert_{op}$ 和消息 m,算法输出多代理多签名 σ_p。

(8)多代理多签名验证算法(MProxyMVerify):输入系统公开参数 $Params$、授权信息 m_w、消息 m、多代理多签名 σ_p、n 个原始签名者的公钥 y_{o_i}($i=1,\cdots,n$)、n 个代理签名人的公钥 y_{p_i}($i=1,\cdots,n$),算法输出 1 或 0。如果输出是 1 表示多代理多签名正确,否则表示不正确。

2.3 多代理多签名的安全模型

在基于证书的密码体制中,某个身份为 ID 的用户必须同时拥有证书 $Cert_{ID,j}$ 和私钥 x_{ID} 才能产生正确的签名。

在基于证书的密码体制中,存在两类攻击者[3]。第一类攻击者 A_I 不知道系统主私钥,模拟的是一般用户的攻击,他能替换任意用户的公钥;第二类攻击者 A_{II} 知道系统主私钥,可以产生任意用户的证书,但不知道用户私钥,他不能替换用户的公钥。

本文的安全模型主要是参考 Wu 等人[13]的基于证书的一般攻击者模型和 Wang 等人[11]的代理签名的安全模型提出的。

定义 1(基于证书的多代理多签名的安全性) 在适应性选择消息、适应性选择授权和适应性选择身份攻击下,如果不存在任何多项式有界的攻击者(A_I 或 A_{II})以不可忽略的概率赢得以下游戏,则称一个基于证书的多代理多签名方案是安全的。

1. 初始化:输入安全参数 k,挑战者 D 运行 Setup,输出系统主私钥 s、主公钥 P_{pub} 和系统公开参数 $Params$。如果是第一类攻击者 A_I,D 返回 P_{pub} 和 $Params$ 给 A_I;如果是第二类攻击者 A_{II},D 还要返回 s 给 A_{II}。

2. 阶段 1:攻击者适应性地进行多项式有界次的以下询问:

(a)用户生成询问:攻击者选择用户身份 ID,如果该 ID 的密钥对还未产生,D 运行用户密钥产生算法生成该用户的私/公钥对(x_{ID},y_{ID}),并返回 y_{ID} 给攻击者,再把(x_{ID},y_{ID})放入表 L_k;否则直接返回 y_{ID} 给攻击者。

(b)私钥询问:攻击者选择一个已经生成的用户身份 ID,D 查询表 L_k,返回 x_{ID} 给攻击者。如果用户的公钥被替换则返回 $\perp$。

(c)替换公钥询问(仅适合于攻击者 A_I):在任何时候,A_I 选择一个新的公钥 y'_{ID} 替换原来的公钥 y_{ID},并且攻击者不用提供知道相应私钥的证明。D 更新表 L_k 为(—,y'_{ID})。

(d)证书询问(仅适合于攻击者 A_I):攻击者选择用户 ID 和公钥 y_{ID},D 运行证书产生算法产生该用户在时段 j 的证书 $Cert_{ID,j}$,并返回给攻击者。如果用户的公钥被替换则不允许进行此项查询。因为 CA(Certificate Authority)不会为用户在相同时段内发行两份不一样的证书。

(e)标准签名询问:攻击者选择用户 ID 和消息 m,假设该用户的密钥和证书已经产生(如果未产生 D,就先调用用户密钥产生算法和证书产生算法),D 运行标准签名算法产生一个签名 σ,并返回给攻击者,D 记录(ID,m)到表 S_{qu} 中。如果用户的公钥被替换,则返回 $\perp$。

(f)代理授权询问:攻击者选择一个授权信息 m_w。分两种情况:

情况 1:攻击者作为原始签名者之一,D 作为代理签名者之一运行代理授权产生算法,最后产生代理授权 $Cert_{op}$。D 验证 $Cert_{op}$ 的正确性,如果正确,把(m_w,$Cert_{op}$)加入到表 $Warrp$ 中。

情况 2:攻击者作为代理签名者之一,D 作为原始签名者之一运行代理授权产生算法,最后产生代理授权 $Cert_{op}$,D 通过公开信道返回 $Cert_{op}$ 给攻击者。D 把 m_w 加入到表 $Warro$ 中。

(g)多代理多签名询问:攻击者选择一个授权信息 m_w、消息 m,其中 $m_w \in Warrp$,m 适合 m_w。D 从 $Warrp$ 中获得代理授权 $Cert_{op}$,从 m_w 中获得 n 个代理签名人的身份,然后运行多代理多签名算法产生一个多代理多签名 σ_p 给攻击者,D 把(m_w,m)加入到表 PS_{qu} 中。如果任何一个代理签名者的公钥被替换,则返回 $\perp$。

3. 伪造:假如以下任一事件发生,则攻击者获胜。

E1:攻击者输出一个用户 ID' 的伪造的标准签名(m',σ'),该签名能通过标准签名验证算法,且(ID',m')$\notin S_{qu}$。对于第一类攻击者 A_I,$y_{ID'}$ 可以是替换后的公钥,要求其没有对 ID' 的证书进行查询;对于第二类攻击者 A_{II},$y_{ID'}$ 是用户的原始公钥,要求其没有对 ID' 的私钥进行查询。

E2:攻击者冒充代理签名者之一 ID_{p_i} 输出一个伪造的多代理多签名(m',m_w',σ_p'),该签名能通过多代理多签名验证,且(m',m_w')$\notin PS_{qu}$。对于第一类攻击者 A_I,他可以替换任何一个原始签名者或代理签名者的公钥,但要求他没有对代理签名者 ID_{p_i} 的证书进行过查询;对于第二类攻击者 A_{II},他不可以替换任何一个原始签名者或代理签名者的公钥,且他没有对 ID_{p_i} 的私钥进行过查询。

E3:攻击者冒充原始签名者之一 ID_{o_i} 输出一个伪造的多代理多签名(m',m_w',σ_p'),该签名能通过多代理多签名验证,且 $m_w' \notin Warro$。对于第一类攻击者 A_I,他可以替换任何一

个原始签名者或代理签名者的公钥，但要求他没有对原始签名者 ID_{o_i} 的证书进行过查询；对于第二类攻击者 A_{II}，他不可以替换任何一个原始签名者或代理签名者的公钥，且要求他没有对 ID_{o_i} 的私钥进行过查询。

攻击者的优势定义为他在以上游戏中获胜的概率。

3 基于证书的多代理多签名方案

3.1 具体方案

Setup：输入安全参数 k，CA 产生两个大的素数 p 和 q，然后在有限域 $GF(p)$ 上确定一条安全的椭圆曲线 $E(F_p)$，设 P 为椭圆曲线循环群 G 中任意一阶为 q 的生成元。CA 随机选取私钥 $s\in Z_q^*$，计算公钥 $P_{pub}=sP$，选取 3 个安全 Hash 函数 H_1,H_2,H_3，它们均将 $\{0,1\}^*$ 映射到 Z_q^*，公开系统参数 $\{p,q,P,P_{pub},H_1,H_2,H_3\}$，保密私钥 s。

UserKeyGen：用户随机选取 $x\in Z_q^*$ 作为私钥，计算公钥 $y=xP$。n_1 个原始签名人的密钥对为 $(x_{o_i},y_{o_i})(i=1,2,\cdots,n_1)$，$n_2$ 个代理签名人的密钥对为 $(x_{p_i},y_{p_i})(i=1,2,\cdots,n_2)$。

CertGen：每个原始签名人发送他的公钥 y_{o_i} 和身份信息 ID_{o_i} 给 CA，CA 验证用户身份后，随机选择 $r_{o_i}\in Z_q^*$，计算 $R_{o_i}=r_{o_i}P$，$h_{1o_i}=H_1(P_{pub},ID_{o_i},y_{o_i},j,R_{o_i})$，$cert_{o_i}=r_{o_i}+s\cdot h_{1o_i}$，发送证书 $Cert_{o_i}$ 给每个原始签名人 ID_{o_i}，并广播 R_{o_i}，其中 j 表示时段参数；同样，每个代理签名人 ID_{p_i} 发送他的公钥 y_{p_i} 和身份信息 ID_{p_i} 给 CA，CA 验证用户身份后，随机选择 $r_{p_i}\in Z_q^*$，计算 $R_{p_i}=r_{p_i}P$，$h_{1p_i}=H_1(P_{pub},ID_{p_i},y_{p_i},j,R_{p_i})$，$cert_{p_i}=r_{p_i}+s\cdot h_{1p_i} \bmod q$，发送证书 $Cert_{p_i}$ 给每个代理签名人 ID_{p_i}，并广播 R_{p_i}。每个原始签名人 $ID_{o_i}(i=1,\cdots,n_1)$ 验证 $cert_{o_i}P=R_{o_i}+h_{1o_i}P_{pub}$ 等式来确定证书的有效性。同理，每个代理签名人 $ID_{p_i}(i=1,\cdots,n_2)$ 通过验证 $cert_{p_i}P=R_{p_i}+h_{1p_i}P_{pub}$ 等式来确定证书的有效性。

Sign：签名人身份为 ID，他随机选择 $r\in Z_q^*$，计算 $R=rP$，$h_{21}=H_2(m,R,R_{ID})$，$h_{22}=H_2(m,R,y_{ID})$，$d=r+Cert_{ID}\cdot h_{21}+x_{ID}\cdot h_{22} \bmod q$，签名 $\sigma=(m,R,d,R_{ID})$。

Verify：验证者收到签名 $\sigma=(m,R,d,R_{ID})$ 后，计算 $h_{1ID}=H_1(P_{pub},ID_{ID},y_{ID},j,R_{ID})$，$h_{21}=H_2(m,R,R_{ID})$，$h_{22}=H_2(m,R,y_{ID})$，然后验证等式 $dP=R+h_{21}\cdot(R_{ID}+h_{1ID}P_{pub})+h_{22}\cdot y_{ID}$ 是否成立。成立则验证者接受签名，否则拒绝。

DelegationGen：(1)原始签名人 $ID_{o_i}(i=1,\cdots,n_1)$ 随机选择 $r_i\in Z_q^*$，计算并广播 $R_i=r_iP$；(2) $ID_{o_i}(i=1,\cdots,n_1)$ 计算 $R_o=\sum_{i=1}^{n_1}R_{o_i}$，$R=\sum_{i=1}^{n_1}R_i$，$h_{21}=H_2(m_w,R,R_o)$，$h_{22o_i}=H_2(m_w,R,y_{o_i})$，$d_i=r_i+Cert_{o_i}\cdot h_{21}+x_{o_i}\cdot h_{22o_i} \bmod q$，其中 m_w 为授权信息，它记录着对代理消息类型的限制、代理的有效期、原始签名者和代理签名者的身份和公钥等信息，由所有原始签名者合作产生。n_1 个原始签名人确定某个人为合成者，合成者验证 $d_iP=R_i+h_{21}\cdot(R_{o_i}+h_{1o_i}P_{pub})+h_{22o_i}y_{o_i}(i=1,\cdots,n_1)$ 的正确性，如果都正确，则计算 $d=\sum_{i=1}^{n_1}d_i$，并通过公开信道传送 $(m_w,R,d,R_{o_1},R_{o_2},\cdots,R_{o_{n_1}})$ 给每个代理签名者 $ID_{p_i}(i=1,\cdots,n_2)$，每个代理签名者都可通过验证 $dP=R+h_{21}\cdot(R_o+\sum_{i=1}^{n_1}(h_{1o_i})P_{pub})+\sum_{i=1}^{n_1}(h_{22o_i}\cdot y_{o_i})$ 等式来确认代理授权的有效性。

MProxyMSign：

(1) $ID_{p_i}(i=1,\cdots,n_2)$ 随机选择 $u_i\in Z_q^*$，计算 $U_i=u_iP$；

(2) $ID_{p_i}(i=1,\cdots,n_2)$ 计算 $U=\sum_{i=1}^{n_2}U_i$，$R_o=\sum_{i=1}^{n_1}R_{o_i}$，$R_p=\sum_{i=1}^{n_2}R_{p_i}$，$H_o=\sum_{i=1}^{n_1}h_{1o_i} \bmod q$，$H_p=\sum_{i=1}^{n_2}h_{1p_i} \bmod q$，$h_{31}=H_3(m,m_w,U,R,H_o)$，$h_{32p_i}=H_3(m,m_w,U,y_{p_i},H_p)$，$h_{33}=H_3(m,m_w,U,R_p,H_p)$，$v_i=u_i+d\cdot h_{31}+x_{p_i}\cdot h_{32p_i}+Cert_{p_i}\cdot h_{33} \bmod q$。

(3)所有代理签名人指定一个合成者，合成者验证 $v_iP=U_i+h_{31}(R+h_{21}\cdot(R_o+H_oP_{pub})+\sum_{i=1}^{n_1}(h_{22o_i}\cdot y_{o_i}))+h_{32p_i}y_{p_i}+h_{33}(R_{p_i}+h_{1p_i}P_{pub})(i=1,\cdots,n_2)$ 的正确性，如果都正确，则计算 $v=\sum_{i=1}^{n_2}v_i \bmod q$，最后，多代理多签名为：$\sigma_p=(m,m_w,U,v,R_p,R,R_o,H_o,H_p)$。

MProxyMVerify：验证者收到 $\sigma_p=(m,m_w,U,v,R_p,R,R_o,H_o,H_p)$ 后，首先验证 m 是否适合 m_w，以及代理的有效期，原始签名人和代理签名人的身份和公钥是否适合 m_w，然后验证者计算 $h_{21}=H_2(m_w,R,R_o)$，$h_{22o_i}=H_2(m_w,R,y_{o_i})$，$h_{31}=H_3(m,m_w,U,R,H_o)$，$h_{32p_i}=H_3(m,m_w,U,y_{p_i},H_p)$，$h_{33}=H_3(m,m_w,U,R_p,H_p)$，验证以下等式是否成立：

$$vP=U+n_2h_{31}(R+h_{21}\cdot(R_o+H_oP_{pub})+\sum_{i=1}^{n_1}(h_{22o_i}\cdot y_{o_i}))+\sum_{i=1}^{n_2}(h_{32p_i}y_{p_i})+h_{33}(R_p+H_pP_{pub})$$

如果成立，验证者接受多代理多签名，否则拒绝。

说明：为了得到固定长度密文，本文让代理签名者计算 $H_o=\sum_{i=1}^{n_1}h_{1o_i} \bmod q$，$H_p=\sum_{i=1}^{n_2}h_{1p_i} \bmod q$，$R_o=\sum_{i=1}^{n_1}R_{o_i}$，$R_p=\sum_{i=1}^{n_2}R_{p_i}$，并传输 H_p、H_o、R_p 和 R_o，从而避免传输 $R_{o_1},R_{o_2},\cdots,R_{o_{n_1}}$ 和 $R_{p_1},R_{p_2},\cdots,R_{p_{n_2}}$。

3.2 安全性分析

定理 1(类型 I 攻击下的不可伪造性) 在随机预言机模型中，在适应性选择消息、适应性选择授权和适应性选择身份攻击下，假如存在一个攻击者 A_I 能在多项式时间内以 ε_1 的优势伪造一个标准签名、以 ε_2 的优势冒充代理签名者之一伪造一个多代理多签名，或者以 ε_3 的优势冒充原始签名者之一伪造一个多代理多签名，他最多进行 $q_{H_i}(i=1,2,3)$ 次 Hash 询问，则存在一个挑战者 D，能在多项式时间内用以下优势解决椭圆曲线离散对数问题。

$$Adv_D^{EC\text{-}DL}\geqslant\left(\frac{\varepsilon_1^3}{(q_{H_1}+q_{H_2})^6}-\frac{3}{2^k}\right)\varepsilon_1+\left(\frac{\varepsilon_2^3}{(q_{H_1}+q_{H_3})^6}-\frac{3}{2^k}\right)\varepsilon_2+\left(\frac{\varepsilon_3^3}{(q_{H_1}+q_{H_3})^6}-\frac{3}{2^k}\right)\varepsilon_3$$

证明：设挑战者 D 接收一个随机的椭圆曲线离散对数问题的实例 (P,xP)，$x\in Z_q^*$，x 的值未知，D 的目标是输出 x。D 以 A_I 为子程序充当定义 1 游戏中的挑战者。D 运行 Setup 设置系统参数，把系统公开参数 $\{p,q,P,P_{pub},H_1,H_2,H_3\}$ 发给 A_I，其中 $P_{pub}=xP$。C 维护 $L_1,L_2,L_3,Warrp,Warro,S_{qu},PS_{qu},L_k,L_c$ 9 张表，这些表的初始值都为空。A_I 能做如下询问，为简单起见，假设 A_I 的询问都是不同的，并在作其它询问之前已经作过用户生成询问。

H_1 询问：如果 $(P_{pub},ID_i,y_i,j,R_{ID_i},h_1)$ 在表 L_1 中，返回此 h_1；否则随机选取 $h_1\in Z_q^*$，返回此 h_1 并加入到表 L_1 中。

H_2 询问：如果(m,R,R_{ID},h_2)或(m,R,y_{ID},h_2)在表 L_2 中，返回此 h_2；否则随机选取 $h_2\in Z_q^*$，返回此 h_2 并加入到表 L_2 中。

H_3 询问：如果(m,m_w,U,R,H_o,h_3)或(m,m_w,U,R_p,H_p,h_3)或$(m,m_w,U,y_{ID_i},H_p,h_3)$在表 L_3 中，返回此 h_3；否则随机选取 $h_3\in Z_q^*$，返回此 h_3 并加入到表 L_3 中。

用户生成询问：A_I 选择一个新的用户身份 ID_i。D 随机选取 $x_{ID_i}\in Z_q^*$ 作为用户私钥，设置他的公钥为 $y_{ID_i}=x_{ID_i}\cdot P$，并把$(ID_i,x_{ID_i},y_{ID_i},\Theta=0)$加入到表 L_k 中，$\Theta=0$ 表示公钥未被替换，返回 y_{ID_i} 给 A_I。

私钥询问：A_I 选择一个用户身份 ID_i，D 查询表 L_k，如果 $\Theta=0$，返回 x_{ID_i} 给 A_I；如果 $\Theta=1$，返回 $\perp$ 给 A_I。

替换公钥询问：在任何时候，A_I 选择一个新的公钥 y'_{ID_i} 替换原来的公钥 y_{ID_i}。D 更新表 L_k 为$(ID_i,-,y'_{ID_i},\Theta=1)$。

证书询问：A_I 选择用户 ID_i 和公钥 y_{ID_i}，D 查询表 L_k，如果 $\Theta=1$，返回 $\perp$ 给 A_I；否则，D 随机选择 $Cert_{ID_i},h_{1ID_i}\in Z_q^*$，计算 $R_{ID_i}=Cert_{ID_i}P-h_{1ID_i}\cdot P_{pub}$，将$(P_{pub},ID_i,y_{ID_i},j,R_{ID_i},h_{1ID_i})$加入到表 L_1 中，如果 L_1 中已经有该条记录则重新选择 $Cert_{ID_i},h_{1ID_i}$，再重复上述计算过程，最后返回$(Cert_{ID_i},R_{ID_i})$给 A_I，把$(ID_i,y_{ID_i},Cert_{ID_i},R_{ID_i})$放入表 L_c。从 A_I 的观点来看，这是一个正确的证书。

标准签名询问：A_I 选择用户 ID_i 和消息 m，D 查询表 L_k，如果 $\Theta=1$，D 返回 $\perp$ 给 A_I；否则，D 分别从表 L_k 和 L_c 中获得 ID_i 的私钥和证书，D 可以按正常方式完成签名，返回签名 $\sigma=(m,R,d,R_{ID_i})$给 A_I，并记录(ID_i,m)到表 S_{qu} 中。

代理授权询问：攻击者选择一个授权信息 m_w，分两种情况。

情况 1：攻击者作为原始签名者之一 ID_{o_i}，D 作为代理签名者之一 ID_{p_i}，运行代理授权产生算法。最后产生代理授权 $Cert_{op}$，D 验证$(m_w,Cert_{op})$的正确性。如果正确，把$(m_w,Cert_{op})$加入到表 $Warrp$ 中。

情况 2：攻击者作为代理签名者之一 ID_{p_i}，D 作为原始签名者之一 ID_{o_i}，运行代理授权产生算法。D 查询表 L_k 中的 $ID_{o_i}(i=1,\cdots,n_1)$，如果任何一个 $\Theta=1$，D 返回 $\perp$ 给 A_I；否则，D 分别从表 L_k 和 L_c 中获得 ID_{o_i} 的私钥和证书，D 可以按正常方式完成代理授权产生算法，最后产生代理授权 $Cert_{op}$，D 通过公开信道返回 $Cert_{op}$ 给攻击者，并把 m_w 加入到表 $Warro$ 中。

多代理多签名询问：攻击者选择一个授权信息 m_w、消息 m。D 验证是否 $m_w\in Warrp$，m 适合 m_w。如果不符，D 返回 $\perp$ 给 A_I；否则，D 从 m_w 中获得 n_1 个原始签名人 $ID_{o_i}(i=1,\cdots,n_1)$和 n_2 个代理签名人 $ID_{p_i}(i=1,\cdots,n_2)$的身份，D 查询表 L_k 中的 $ID_{p_i}(i=1,\cdots,n_2)$，如果任何一项的 $\Theta=1$，D 返回 $\perp$ 给 A_I；否则，D 从表 $Warrp$ 中查询 m_w 获得代理授权 $Cert_{op}$，从表 L_k 和 L_c 中查询 $ID_{p_i}(i=1,\cdots,n_2)$获得私钥 $x_{p_i}(i=1,\cdots,n_2)$和证书 $Cert_{p_i}(i=1,\cdots,n_2)$，然后运行多代理多签名算法产生一个多代理多签名 σ_p 给攻击者，最后把(m_w,m)加入到表 PS_{qu} 中。

最终 A_I 停止询问并输出一个伪造。

E1：A_I 输出一个用户 ID'的伪造的标准签名(m',σ')，该签名能通过标准签名验证算法，且$(ID',m')\notin S_{qu}$，$Cert_{ID',j}\notin L_c$，y'_{ID}可以是替换后的公钥。设伪造的签名是 $\sigma'=(m',R',d',R_{ID'})$，则由多分叉引理[10]对 h_{21} 和 H_1 用相同的输入和不同的预言机实例，可以得到同一个消息 m'的 4 个伪造的签名 $\sigma^{(1)}=(m',R',d^{(1)},R_{ID'})$、$\sigma^{(2)}=(m',R',d^{(2)},R_{ID'})$、$\sigma^{(3)}=(m',R',d^{(3)},R_{ID'})$和 $\sigma^{(4)}=(m',R',d^{(4)},R_{ID'})$，设 $h_{1ID'}^{(1)}$和 $h_{1ID'}^{(2)}$是对应 H_1 的 2 个不同实例的不同 Hash 值，$h_{21}^{(1)}$、$h_{21}^{(2)}$、$h_{21}^{(3)}$和 $h_{21}^{(4)}$是对应 h_{21}的 4 个不同实例的不同 Hash 值。由于

$$d^{(1)}P=R'+h_{21}^{(1)}\cdot(R_{ID'}+h_{1ID'}^{(1)}\cdot P_{pub})+h'_{22}\cdot y_{ID'}$$

$$d^{(2)}P=R'+h_{21}^{(2)}\cdot(R_{ID'}+h_{1ID'}^{(1)}\cdot P_{pub})+h'_{22}\cdot y_{ID'}$$

$$d^{(3)}P=R'+h_{21}^{(3)}\cdot(R_{ID'}+h_{1ID'}^{(2)}\cdot P_{pub})+h'_{22}\cdot y_{ID'}$$

$$d^{(4)}P=R'+h_{21}^{(4)}\cdot(R_{ID'}+h_{1ID'}^{(2)}\cdot P_{pub})+h'_{22}\cdot y_{ID'}$$

由以上 4 个等式可得：$((d^{(1)}-d^{(2)})(h_{21}^{(1)}-h_{21}^{(2)})^{-1}-(d^{(3)}-d^{(4)})(h_{21}^{(3)}-h_{21}^{(4)})^{-1})\cdot(h_{1ID'}^{(1)}-h_{1ID'}^{(2)})^{-1}=x$，从而 D 解决了椭圆曲线离散对数问题。

E2：A_I 冒充代理签名者之一 ID_{p_i} 输出一个伪造的多代理多签名(m',m_w',σ_p')，该签名能通过多代理多签名验证。$y_{p_i}(i=1,\cdots,n_2)$和 $y_{o_i}(i=1,\cdots,n_1)$可以是替换后的公钥，且 $(m',m_w')\notin PS_{qu}$，$Cert_{ID_{p_i}}\notin L_c$。

E3：A_I 冒充原始签名者之一 ID_{o_i} 输出一个伪造的多代理多签名(m',m_w',σ_p')，该签名能通过多代理多签名验证。$y_{p_i}(i=1,\cdots,n_2)$和 $y_{o_i}(i=1,\cdots,n_1)$可以是替换后的公钥，且 $m_w'\notin Warro$，$Cert_{ID_{o_i}}\notin L_c$。

以上两种情况下，设伪造的多代理多签名是 $\sigma_p'=(m',m_w',U',v',R_o',R_p',R',H_o',H_p')$，则由多分叉引理[10]对 h_{33} 和某个 h_{1p_i} 用相同的输入和不同的预言机实例，可得 4 个伪造的多代理多签名 $\sigma_p^{(1)},\sigma_p^{(2)},\sigma_p^{(3)},\sigma_p^{(4)}$。设 $h_{1p_i}^{(1)}$和 $h_{1p_i}^{(2)}$是对应 h_{1p_i}的 2 个不同实例的不同 Hash 值，$h_{33}^{(1)}$、$h_{33}^{(2)}$、$h_{33}^{(3)}$、$h_{33}^{(4)}$是对应 h_{33}的 4 个不同实例的不同 Hash 值，由 $h_{1p_i}^{(1)}$和 $h_{1p_i}^{(2)}$可得 $H_p^{(1)}$ 和 $H_p^{(2)}$。于是 4 个伪造的多代理多签名为：

$$\sigma_p^{(1)}=(m',m_w',U',v^{(1)},R_o',R_p',R',H_o',H_p^{(1)})$$

$$\sigma_p^{(2)}=(m',m_w',U',v^{(2)},R_o',R_p',R',H_o',H_p^{(1)})$$

$$\sigma_p^{(3)}=(m',m_w',U',v^{(3)},R_o',R_p',R',H_o',H_p^{(2)})$$

$$\sigma_p^{(4)}=(m',m_w',U',v^{(4)},R_o',R_p',R',H_o',H_p^{(2)})$$

由于

$$v^{(1)}P=U'+n_2h'_{31}(R'+h'_{21}(R_o'+H_o'P_{pub})+\sum_{i=1}^{n_1}(h'_{22o_i}\cdot y'_{o_i}))+\sum_{i=1}^{n_2}(h'_{32p_i}y'_{p_i})+h_{33}^{(1)}(R_p'+H_p^{(1)}P_{pub})$$

$$v^{(2)}P=U'+n_2h'_{31}(R'+h'_{21}(R_o'+H_o'P_{pub})+\sum_{i=1}^{n_1}(h'_{22o_i}\cdot y'_{o_i}))+\sum_{i=1}^{n_2}(h'_{32p_i}y'_{p_i})+h_{33}^{(2)}(R_p'+H_p^{(1)}P_{pub})$$

$$v^{(3)}P=U'+n_2h'_{31}(R'+h'_{21}(R_o'+H_o'P_{pub})+\sum_{i=1}^{n_1}(h'_{22o_i}\cdot y'_{o_i}))+\sum_{i=1}^{n_2}(h'_{32p_i}y'_{p_i})+h_{33}^{(3)}(R_p'+H_p^{(2)}P_{pub})$$

$$v^{(4)}P=U'+n_2h'_{31}(R'+h'_{21}(R_o'+H_o'P_{pub})+\sum_{i=1}^{n_1}(h'_{22o_i}\cdot y'_{o_i}))+\sum_{i=1}^{n_2}(h'_{32p_i}y'_{p_i})+h_{33}^{(4)}(R_p'+H_p^{(2)}P_{pub})$$

由以上 4 个等式得：$((v^{(1)}-v^{(2)})(h_{33}^{(1)}-h_{33}^{(2)})^{-1}-(v^{(3)}-v^{(4)})(h_{33}^{(3)}-h_{33}^{(4)})^{-1})\cdot(H_p^{(1)}-H_p^{(2)})^{-1}=x$，从而 D 解决了椭圆曲线离散对数问题。

在 E1、E2 和 E3 3 种情况下，由多分叉引理的概率得：

$$Adv_D^{EC\text{-}DL} \geqslant \left(\frac{\varepsilon_1^3}{(q_{H_1}+q_{H_2})^6}-\frac{3}{2^k}\right)\varepsilon_1+\left(\frac{\varepsilon_2^3}{(q_{H_1}+q_{H_3})^6}-\frac{3}{2^k}\right)\varepsilon_2+\left(\frac{\varepsilon_3^3}{(q_{H_1}+q_{H_3})^6}-\frac{3}{2^k}\right)\varepsilon_3$$

定理 2(类型 II 攻击下的不可伪造性) 在随机预言机模型中，在适应性选择消息、适应性选择授权和适应性选择身份攻击下，假如存在一个攻击者 A_{II} 能在多项式时间内以 ε_1 的优势伪造一个标准签名，以 ε_2 的优势冒充代理签名者之一伪造一个多代理多签名，或者以 ε_3 的优势冒充某个原始签名者之一伪造一个多代理多签名，他最多进行 q_{H_i} $(i=1,2,3)$ 次 Hash 询问，q_c 次用户生成询问，q_k 次私钥询问，则存在一个挑战者 D，能在多项式时间内用以下优势解决椭圆曲线离散对数问题。

$$Adv_D^{EC\text{-}DL} \geqslant \frac{1}{q_c}\left(1-\frac{1}{q_c}\right)^{q_k}\left[\left(\frac{\varepsilon_1}{q_{H_2}}-\frac{1}{2^k}\right)\varepsilon_1+\left(\frac{\varepsilon_2}{q_{H_3}}-\frac{1}{2^k}\right)\varepsilon_2+\left(\frac{\varepsilon_3^3}{(q_{H_2}+q_{H_3})^6}-\frac{3}{2^k}\right)\varepsilon_3\right]$$

设一个随机的椭圆曲线离散对数问题的实例为 (P,xP)，$x\in Z_q^*$，x 的值未知，把挑战身份的公钥设为 xP，运用多分叉引理[10]和通用分叉引理[16]，定理 2 可类似得到证明。

3.3 效率分析

由于未见有基于证书的多代理多签名的报道，所以效率比较方面，本文主要与非基于证书的多代理多签名方案进行比较。2013 年，文献[17]提出一个基于身份的多代理多签名方案，方案也没有使用耗时的双线性对计算，通过与使用双线性对的文献[18,19]的比较，得出他们的方案效率较高，在采用了具体的实验数据后，他们方案的多代理多签名阶段计算量分别只有文献[18]的 13.61%，文献[19]的 14.21%；多代理多签名验证阶段分别只有文献[18]的4.6%，文献[19]的 4.59%。本文与文献[17]的比较结果列于表 1，其中 Tm 表示椭圆曲线上的点乘运算，n_1 表示原始签名人的总数，n_2 表示代理签名人的总数。从表 1 可以看出，文献[17]的密文长度随 n_1 和 n_2 线性增长，而本文为固定密文长度；计算量方面，两个方案的差别不大。所以，本文方案的计算效率与文献[17]差不多，而密文长度大大缩短，因此属于高效的方案。

表 1 相关方案的效率比较

方案	密文长度	多代理多签名	多代理多签名验证
文献[17]	$\|m\|+\|m_w\|+(n_1+n_2+2)\|G\|+2\|q\|$	$(2n_1+3n_2+1)Tm$	$(2n_1+2n_2+2)Tm$
本文方案	$\|m\|+\|m_w\|+4\|G\|+3\|q\|$	$(n_1+4n_2+3)Tm$	$(n_1+n_2+6)Tm$

结束语 多代理多签名能实现一群原始签名人授权一群代理签名人的情景。基于证书的密码体制既能简化公钥的管理，又能解决密钥托管问题。本文给出了基于证书的多代理多签名的形式化定义和安全概念，并提出一个具体方案。方案没有采用耗时的双线性对运算，与其它方案相比效率较高，并且是可证安全的。方案还具有在原始签名人和代理签名人之间不需要安全信道的优点，适用性强。

参 考 文 献

[1] Shamir A. Identity-based cryptosystems and signature schemes [C]// Proceeding of Crypto'84. LNCS 196, Berlin: Springer-Verlag, 1984: 47-53

[2] Al-Riyami S S, Paterson K G. Certificateless public key cryptography[C]// Proceeding of ASIACRYPT 2003. LNCS 2894, Berlin: Springer-Verlag, 2003: 452-473

[3] Gentry C. Certificate-based encryption and the certificate revocation problem[C]// Proc. of EuroCrypt. Berlin: Springer-Verlag, 2003: 272-293

[4] Mambo M, Usuda K, Okamoto E. Proxy signature: delegation of the power to sign messages[J]. IEICE Trans. Fundamentals, 1996, E79-A(9): 1338-1353

[5] Hwang S J, Chen C C. A New Multi-Proxy Multi-Signature Scheme[C]// 2001 National Computer Symposium: Information Security. Taiwan, 2001: 19-26

[6] Kang B G, Park J H, Hahn S G. A certificate-based signature scheme[C]// Proc. of CT-RSA. Berlin: Springer-Verlag, 2004: 99-111

[7] 陈辉焱，李巍，苏艳芳. 一种基于证书的代理环签名方案[J]. 计算机工程，2012，38(16)：149-152

[8] 韩春霞，王琳杰. 基于证书的多重代理多重盲签名方案研究[J]. 长江大学学报：自然科学版，2013，10(19)：61-64

[9] 翟正元，高德智，梁向前，等. 新的基于证书的代理盲签名方案[J]. 计算机工程与应用，2014，50(4)：57-62

[10] Boldyreva A, Palacio A, Warinschi B. Secure proxy signature schemes for delegation of signing rights[J]. Journal of cryptology, 2012, 25(1): 57-115

[11] Wang Qin, Cao Zhen-fu. Security arguments for partial delegation with warrant proxy signature schemes[EB/OL]. [2004-11-17]. http://erpint.iacr.org/2004/315.pdf

[12] Li Ji-guo, Huang Xin-yi, Mu Yi, et al. Certificate-based signature: security model and efficient construction[C]// Proc. of EuroPKI'2007. 2007: 110-125

[13] Wu Wei, Mu Yi, Susilo W, et al. Certificate-based signatures revisited[J]. Journal of universal computer science, 2009, 15(8): 1659-1684

[14] Li Ji-guo, Xu Li-zhong, Zhang Yi-chen. Provably secure certificate-based proxy signature schemes[J]. Journal of Computers, 2009, 4(6): 444-452

[15] 陈江山. 基于证书的代理签名和盲签名[D]. 漳州：漳州师范学院，2012

[16] Bellare M, Neven G. Multi-signatures in the plain public-key model and a general forking lemma[C]// Proc of CCS'06. Alexandria: ACM, 2006: 390-399

[17] Tiwari N, Padhye S, He D. Efficient ID-based multiproxy multisignature without bilinear maps in ROM[J]. Annals of telecommunications, 2013, 68(3-4): 231-237

[18] Li Xiang-xue, Chen Ke-fei. ID-based multi-proxy signature, proxy multi-signature and multi-proxy multi-signature schemes from bilinear pairings[J]. Appl Math Comput, 2005(169): 437-450

[19] Sahu R A, Padhye S. An ID-based multi-proxy multisignature scheme[C]// Proceeding of IEEE international conference on computer & communication technology (ICCCT-2010). 2010: 60-63