软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2012,23(9):2416−2429 [doi: 10.3724/SP.J.1001.2012.04246]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

标准模型下的代理签名:构造模型与证明安全性*

谷 科[1,2+], 贾维嘉[2,3], 王四春[1], 石良武[1]

[1](湖南商学院 信息学院,湖南 长沙 410205)

[2](中南大学 信息科学与工程学院,湖南 长沙 410083)

[3](香港城市大学 计算机科学系,香港)

Proxy Signature in the Standard Model: Constructing Security Model and Proving Security

GU Ke[1,2+], JIA Wei-Jia[2,3], WANG Si-Chun[1], SHI Liang-Wu[1]

[1](Information College, Hu'nan University of Commerce, Changsha 410205, China)

[2](School of Information Science and Engineering, Central South University, Changsha 410083, China)

[3](Department of Computer Science, City University of Hong Kong, Hong Kong, China)

+ Corresponding author: E-mail: gk4572@163.com

Gu K, Jia WJ, Wang SC, Shi LW. Proxy signature in the standard model: constructing security model and proving security. *Journal of Software*, 2012,23(9):2416−2429 (in Chinese). http://www.jos.org.cn/1000-9825/4246.htm

Abstract: Current proxy signature schemes are not proved for their security in the complete provable security model of proxy signature. In this paper, we show a complete provable security model for proxy signature based on Boldyreva's provable security model, and a new identity-based proxy signature scheme are proposed in the standard model, which is based on Paterson's scheme. In the complete provable security model for proxy signature, the new scheme is proved to have the existential identity-based proxy signature unforgerability under an adaptive chosen message attack, which has a security reduction to CDHP. Comparing with other proxy signature schemes based on public key cryptosystem in the standard model, the concept of the identity is introduced into the new scheme, and the new scheme is more secure.

Key words: proxy signature; provable security; identity; CDHP

摘 要: 目前已经提出的代理签名方案缺乏在完整的代理签名安全模型下证明方案的安全性.在 Boldyreva 等人提出的代理签名安全模型的基础上,对代理签名的可证安全模型进行详细的形式化定义,提出一种完整的代理签名可证安全模型.同时,为了展示该安全模型的有效性和可扩展性,对 Paterson 等人提出的标准模型下基于身份的签名方案进行扩展,提出在标准模型下基于身份的代理签名方案,并在可证安全模型下,证明新方案具有在自适应选择消息攻击下存在基于身份的代理签名不可伪造性,其安全性在标准模型下可归约于 CDH 问题假定.新方案与标准模型下基于公钥密码体制的代理签名方案相比,不仅增加了用户身份的概念,还具有更完备的安全性.

关键词: 代理签名;可证安全性;身份;CDH 问题

* 基金项目: 教育部人文社科项目(12YJAZH142); 香港城市大学研究项目基金(9681001, 6351006); 香港城市大学战略研究发展基金(7008110); 深港创新圈基金(ZYB200907080078A); 湖南省社科基金重点项目(09ZDB080); 湖南省软科学研究项目(2010ZK3029)

收稿时间: 2011-08-14; 定稿时间: 2012-04-20

中图法分类号: TP309 文献标识码: A

代理签名是由 Mambo 等人在 1996 年提出的签名方案[1].在代理签名方案中,原始签名者(delegator)授权自己的签名权限给一个或者多个代理者(proxy signer).代理签名作为一种普通签名的扩展,不仅需要具有普通签名的一般特性,如不可伪造性、不可否认性等,还需要具有原始签名与代理签名可区分性、抗原始签名者攻击等性质.由于代理签名方案比普通签名方案更为复杂,因此,代理签名的可证安全模型[2−8]也区别于普通签名的安全模型[9−12].

Boldyreva 等人于 2003 年提出了第一个代理签名的可证安全模型[2],该模型基于注册密钥模式(register key model),为学者证明代理签名方案的安全性提供了基准.但该安全模型没有考虑授权书(warrant)的因素,因此存在一定缺陷.由于授权书作为一种授权的传递信息,包含了代理签名的权限和签名规则,因此在代理签名方案中是十分重要的.其后,Malkin 等人提出了一种多层次的代理签名可证安全模型[3],虽然加入了授权书的概念,但该模型同样基于注册密钥模式,并且较为复杂,因此难以应用于方案的安全性证明.

2008 年,Schuldt 等人在文献[4]中提出了一种较完整的代理签名可证安全模型,该模型加入了授权书的概念,没有基于注册密钥模式,能够在代理密钥暴露的情况下用于证明方案的不可伪造性,为学者证明代理签名方案的安全性提供了较严格的标准.但该模型没有对敌手的能力进行具体的刻画,因此模型较为抽象.而文献[5−7]虽然给出了代理签名方案安全性的完整证明,但在这 3 篇文献中,代理签名安全模型定义的 3 种攻击情况缺乏逻辑上的联系,不能构成一个完备的整体;并且没有对代理签名方案存在的攻击情况进行深入分析,没有对敌手的能力进行进一步刻画.所以,定义的 3 种攻击情况是否是对代理签名方案中各种攻击情况的完备覆盖还需要进一步研究.

最近,Boldyreva 等人在文献[8]中对代理签名安全模型的相关工作[3−5]进行了总结分析,对目前已经提出的代理签名安全模型在如何刻画敌手能力的问题上提出了质疑;同时认为,在代理签名方案中,自我代理行为是一种安全性脆弱的情况.因此,Boldyreva 等人在文献[8]中提出了一种更为精确的代理签名安全模型.在该安全模型中,对代理签名方案分 4 种情况进行了安全分析.不过,该文献并没有对代理签名安全模型进行完全的形式化定义.因此,目前大部分代理签名方案[5−7,13−23]都缺乏在完整的代理签名安全模型下证明其安全性的过程,因此,这些方案的安全性仍然值得进一步研究.

所以,提出一种更安全的代理签名方案是十分有意义的,这也是本文的主要动机.

本文在文献[8]的基础上提出一种更详细的代理签名可证安全模型,该模型对代理签名方案在 4 种情况下的安全分析进行完整的形式化定义.同时,为了展示该安全模型的有效性与扩展性,本文对 Paterson 等人提出的标准模型下基于身份的签名方案[12]进行了扩展,提出在标准模型下基于身份的代理签名方案,并在可证安全模型下,证明新方案具有在自适应选择消息攻击下存在基于身份的代理签名不可伪造性,其安全性在标准模型下可归约于 CDH 问题假定.本文方案与其他文献提出的标准模型下的代理签名方案相比,具有更完备的可证安全性,同时,引入了用户身份的概念使得方案具有更好的实用性.虽然本文方案增加了用户身份的计算,但仅增加了较少的计算开销.因此,本文主要工作如下:

(1) 在文献[8]的基础上对代理签名可证安全模型进行完整的形式化定义,提出了一种更详细的代理签名可证安全模型;
(2) 在文献[12]的基础上提出了一个新的基于身份的代理签名方案;
(3) 为了说明安全模型的有效性与可扩展性,对新方案的安全性在代理签名可证安全模型下进行详细的分析证明.

本文第 1 节进行背景知识介绍.第 2 节提出代理签名模型.第 3 节提出代理签名可证安全模型.第 4 节提出新的代理签名方案.第 5 节对新方案进行分析证明.第 6 节是总结与推论.

1 相关知识

1.1 双线性变换

双线性变换:设 G_1 和 G_2 分别为 q 阶的循环群,g 为 G_1 的生成元,则有 e:$G_1\times G_1\to G_2$,并且 e 满足条件 $e(g^a,g^b)=e(g,g)^{a\cdot b}$ 以及 $e(g,g)\neq 1$.即,e 满足双映射性、非退化性和可计算性.

1.2 计算Diffie-Hellman问题及问题假定

定义 1(计算 Diffie-Hellman(CDH)问题). 设 G_1 为 q 阶的循环群,g 为 G_1 的生成元,对于$\forall(g,g^a,g^b)\in G_1$,其中,$a,b\in Z_q$,计算 $g^{a\cdot b}$.

定义 2. 如果不存在一个概率多项式的算法在时间 t 内以至少ε的概率解决 G_1 上 CDH 问题,我们则称 (t,ε)-CDH 问题假设在该群 G_1 上成立.

2 代理签名模型

与普通签名模型相比,代理签名模型增加了原始签名者对代理签名者进行签名授权的步骤;同时,在代理签名模型中,代理签名的私钥生成也依赖于原始签名者的授权信息.因此,代理签名模型比普通签名模型更为复杂.本文在文献[8]的基础上给出了由 6 种抽象算法组成代理签名模型,具体描述如下:

- Setup 算法:输入安全参数 1^k,输出系统全部系统参数;
- KeyGen 算法:系统生成公私钥对(pk_i,sk_i),$i\in\{delegator,proxy\}$,其中,pk_i 为公钥,sk_i 为私钥,因此有原始签名者(授权者)公私钥对(pk_{de},sk_{de})和代理签名者公私钥对(pk_{pr},sk_{pr});
- Delegate 算法:算法输入$(pk_{de},sk_{pr},sk_{de},w)$,其中,$w$ 为授权书,算法输出代理授权信息δ与 ProxyKeyGen 算法交互完成代理签名的授权过程;
- ProxyKeyGen 算法:ProxyKeyGen 算法与 Delegate 算法交互完成代理签名私钥的生成,算法输入$(\delta,pk_{de},pk_{pr},sk_{pr})$,算法输出$(pk_{de},pk_{pr},w,psk_{pd})$,其中,$psk_{pd}$ 为代理签名私钥;
- ProxySign 算法:代理者对消息 m 进行签名,算法输入$(pk_{de},pk_{pr},w,psk_{pd},m)$,然后输出代理签名$(pk_{de},pk_{pr},w,\sigma)$(签名中的公钥 pk_{de} 和 pk_{pr} 可以公开发布,不包括在签名中);
- ProxyVerify 算法:签名接收者对代理者对消息 m 的代理签名$(pk_{de},pk_{pr},w,\sigma)$进行验证,算法输入$(m,pk_{de},pk_{pr},w,\sigma)$,然后输出布尔值 *accept* 或者 *reject*.

3 代理签名可证安全模型

与普通签名方案相比,代理签名方案包含一个原始签名者对代理者的授权过程,因此在代理签名方案中,各个角色间的行为关系将更加复杂.

文献[4]给出了一个较为完整的代理签名可证安全模型,在该模型中,敌手与挑战者之间通过进行一个安全询问游戏(security query game)完成对代理签名方案的安全性证明.该模型最大化敌手的优势,敌手可以控制除用户 u^*之外的所有有用信息,并能够对签名 Oracle 和 Hash Oracle(如果存在)发出任何询问,最后伪造签名.但该模型只对代理签名方案中的 3 种情况进行了安全分析,而根据 Boldyreva 等人在文献[8]中对代理签名安全模型相关工作的总结与分析,代理签名的安全模型[8]需要对如下 4 种情况进行安全分析:

情况 1.使用用户 u^*的公钥验证普通签名,分析代理签名方案在无授权代理的情况下作为普通签名方案的安全性;

情况 2.用户 u^*进行自我代理的情况下,验证代理签名的安全性;

情况 3.分析敌手是否能够伪造一个被敌手控制的用户(bad user)授权给 u^*的代理签名.在这类情况中,我们定义敌手的攻击为原始签名者攻击;

情况 4.分析敌手是否能够在没有获得 u^*授权下,通过一个被敌手控制的用户伪造一个假装 u^*授权的代理

签名.

虽然 Boldyreva 等人在文献[8]中提出一种更精确的代理签名可证安全模型,但该文仅仅对代理签名可证安全模型进行了描述性的论述,缺少形式化的定义过程.因此,本文在文献[4,8]的基础上对代理签名可证安全模型中的 4 种情况进行完整的形式化定义.一般来说,一个签名方案的安全性是通过一个敌手与一个挑战者间的安全游戏来说明,但为了更清晰地描述本文的代理签名可证安全模型,本文通过构造算法 B 与敌手 F 进行交互,对代理签名方案分 4 种情况展开攻击实验(attack experiment).则对于敌手 F,方案至少存在一个用户 u^*不被任意敌手 F 控制(敌手 F 可以为一个或者多个),且最大化敌手 F 的优势;敌手 F 可以控制除用户 u^*之外的所有有用信息(包括用户 u^*的授权书和公钥),且能够共享 4 种情况下所获得的全部信息;并存在模拟器 S,在通过有限次代理签名私钥询问和签名询问后,能够伪造出包括消息和签名的合法签名视图.为了更清晰、简单地描述模型,本文形式化定义的代理签名可证安全模型建立在公钥密码体制和标准模型[11]上.

算法 B 的符号定义如下(为了简化模型便于理解,原始签名者和代理者之间的具体交互过程被省略):

① n 个用户的集合Ω,用户 $u_i\in\Omega,i\in\{1,2,\ldots,n\}$且 n 足够大;

② 被控制用户集合 $D,D\subset\Omega$,用户 $d_i\in D,i\in\{1,2,\ldots,|D|\}$,其中,$|D|$表示集合 D 的大小;

③ 不被控制用户集合 $U,U\subseteq\Omega$,用户 $u_i^*\in U$,$i\in\{1,2,\ldots,|U|\}$,其中$|U|$表示集合 U 的大小;

④ *cma* 表示选择消息查询阶段,*forge* 表示签名伪造阶段;

⑤ *req_proxykey*(*param*1,*param*2,*param*3,*param*4)表示代理签名私钥查询,负责完成原始签名者与代理者(在用户集合Ω中除原始签名者之外的某个用户)之间进行的授权交互过程并产生代理签名私钥.其中:*param*1,*param*2,*param*3 为输入参数,*param*1 表示原始签名者私钥,*param*2 表示授权书,*param*3 表示代理者公钥;*param*4 为输出参数,表示代理签名私钥;

⑥ $req_sig_{psk}(\cdot)$表示代理签名查询,其中,*psk* 表示代理签名私钥,点·表示查询的任意消息,$req_sig_{psk}(\cdot)$返回结果.在情况 1 中,*req_sig*(·)表示普通签名查询;

⑦ *versign*(*param*1,*param*2,*param*3,*param*4)表示验证签名,其中:*param*1 表示消息;*param*2 表示代理签名;*param*3 为一个元组,表示原始签名者和代理者的公钥集合;*param*4 表示授权书,*versign*(·)返回结果,其值为布尔值.在情况 1 中,*versign*(·)表示验证普通签名,*param*3 表示某个用户的公钥,*param*4 将默认为空值;

⑧ k 表示安全参数,F 表示敌手,n 表示用户数,T 表示攻击类型,其值取枚举型{*type*1,*type*2,*type*3,*type*4},*type*1 代表情况 1,*type*2 代表情况 2,*type*3 代表情况 3,*type*4 代表情况 4,q_e 表示代理签名私钥询问最大次数,q_s 表示签名询问最大次数,且有 $q_e\geqslant q_s$;

⑨ 运算符号:$\xleftarrow{R}$ 表示随机选择,$\xleftarrow{R}\{...\}$ 表示随机从集合中选择,$F\leftarrow$表示敌手 F 获得相应的结果,$\leftarrow F$ 表示敌手 F 输出相应的结果.

同时,在我们分情况描述算法 B 之前,设算法 B 已经完成如下步骤:

(a) 系统初始化:*Setup*(k,n);

(b) 获得某个用户 u_i^* 的公钥:$i\xleftarrow{R}\{1,2,...,|U|\},F\leftarrow\left(pk_{u_i^*|u_i^*\in U}\right)$;

(c) 结束查询阶段的标志置 0:*StopCma*←0;

(d) 记录查询次数的变量置 0:*Num*←0;

(e) 获得全部用户的授权书:$F\leftarrow(w_{u\in\Omega})$.

(1) 在情况 1 下,*T*=*type*1,则算法 B 描述如下:

Experiment $B(k,F,T,n)$

```
{while (StopCma==0 && Num≤q_s)                    //循环查询
   { StopCma ← F(cma, req_sig(·), pk_{u_i^*|u_i^*∈U});   //签名询问
     Num←Num+1;                                   //如果 Num 没有超过签名询问次数,则可继续查询
   }
```

$\left(m,\sigma^*,pk_{u_i^*|u_i^*\in U}\right)\leftarrow F(forge,*)$; //*表示对应的参数,这里表示 u_i^* 的私钥;伪造签名

if $\left(versign\left(m,\sigma^*,pk_{u_i^*|u_i^*\in U}\right)==1\ \&\&\ StopCma==0\right)$ return (*success*←1);

//如果验证签名的结果为 1 且对消息 *m* 在查询阶段没有被查询过,则敌手伪造签名成功

else return (*fail*←0);

}

(2) 在情况 2 下,*T*=*type*2,则算法 B 描述如下:

Experiment *B*(*k*,*F*,*T*,*n*)

{while (*StopCma*==0 && (*Num*≤q_e||*Num*≤q_s)) //循环查询

{ if (*Num*≤q_e) $F\leftarrow req\_proxykey\left(*,w_{u_i^*},pk_{u_i^*|u_i^*\in U},psk\right)$; //自我代理签名私钥查询

if (*Num*≤q_s) $StopCma\leftarrow F\left(cma,req\_sig_{psk}(\cdot),\left(pk_{u_i^*|u_i^*\in U},pk_{u_i^*|u_i^*\in U}\right)\right)$;

Num←*Num*+1;

}

$\left(m,\sigma^*,\left(pk_{u_i^*|u_i^*\in U},pk_{u_i^*|u_i^*\in U}\right)\right)\leftarrow F\left(forge,(*,*),w_{u_i^*}\right)$; //其中,*表示对应的参数,这里表示 u_i^* 的私钥

if $\left(versign\left(m,\sigma^*,\left(pk_{u_i^*|u_i^*\in U},pk_{u_i^*|u_i^*\in U}\right),w_{u_i^*}\right)==1\ \&\&\ StopCma==0\right)$ return (*success*←1);

else return (*fail*←0);

}

(3) 在情况 3 下,*T*=*type*3,则算法 B 描述如下:

Experiment *B*(*k*,*F*,*T*,*n*)

{while (*StopCma*==0 && (*Num*≤q_e||*Num*≤q_s))

{if (*Num*≤q_e)

{ $j\xleftarrow{R}\{1,2,...,|D|\}$; //敌手从被控制用户集合中获得某个用户的公私钥

$F\leftarrow\left(sk_{d_j|d_j\in D},pk_{d_j|d_j\in D}\right)$;

$F\leftarrow req\_proxykey\left(sk_{d_j|d_j\in D},w_{d_j},pk_{u_i^*|u_i^*\in U},psk\right)$; //代理签名私钥查询

}

if (*Num*≤q_s) $StopCma\leftarrow F\left(cma,req\_sig_{psk}(\cdot),\left(pk_{d_j|d_j\in D},pk_{u_i^*|u_i^*\in U}\right)\right)$;

Num←*Num*+1;

}

$\left(m,\sigma^*,\left(pk_{d_j|d_j\in D},pk_{u_i^*|u_i^*\in U}\right)\right)\leftarrow F\left(forge,\left(sk_{d_j|d_j\in D},*\right),w_{d_j}\right)$; //伪造签名

if $(versign\left(m,\sigma^*,\left(pk_{d_j|d_j\in D},pk_{u_i^*|u_i^*\in U}\right),w_{d_j}\right)==1\ \&\&\ StopCma==0)$ return (*success*←1);

else return (*fail*←0);

}

(4) 在情况 4 下,*T*=*type*4,则算法 B 描述如下:

Experiment *B*(*k*,*F*,*T*,*n*)

{while (*StopCma*==0 && (*Num*≤q_e||*Num*≤q_s))

{if (*Num*≤q_e)

{ $j\xleftarrow{R}\{1,2,...,|D|\}$; //敌手从被控制用户集合中获得某个用户的公私钥

$F \leftarrow \left(sk_{d_j|d_j\in D}, pk_{d_j|d_j\in D}\right)$;

$F \leftarrow req\_proxykey\left(*, w_{u_i^*}, pk_{d_j|d_j\in D}, psk\right)$;

}

if ($Num \leqslant q_s$) $StopCma \leftarrow F\left(cma, req\_sig_{psk}(\cdot), \left(pk_{u_i^*|u_i^*\in U}, pk_{d_j|d_j\in D}\right)\right)$;

$Num \leftarrow Num+1$;

}

$\left(m, \sigma^*, \left(pk_{u_i^*|u_i^*\in U}, pk_{d_j|d_j\in D}\right)\right) \leftarrow F\left(forge, \left(*, sk_{d_j|d_j\in D}\right), w_{u_i^*}\right)$; //伪造签名

if $\left(versign\left(m, \sigma^*, \left(pk_{u_i^*|u_i^*\in U}, pk_{d_j|d_j\in D}\right), w_{u_i^*}\right) == 1 \ \&\& \ StopCma == 0\right)$ return ($success \leftarrow 1$);

else return ($fail \leftarrow 0$);

}

本文通过算法 B 在 4 种情况下描述了代理签名可证安全模型.与算法 B 交互,敌手首先经过系统初始化阶段,然后进入循环查询阶段和伪造签名阶段.在循环查询阶段,敌手可以进行代理签名私钥询问和签名询问,如果敌手在查询阶段成功,则可以结束查询阶段,进入伪造签名阶段.最后,如果敌手伪造的对某消息的代理签名验证正确,并且该消息在查询阶段没有被询问过,则敌手伪造签名成功,算法 B 返回 1;否则敌手失败,算法 B 返回 0.其中,敌手 F 在攻击的不同阶段运行模拟器获得的视图与实际执行过程不可区分.

定义 3(代理签名的安全性). 令 PS=($Setup$,$KeyGen$,$Delegate$,$ProxyKeyGen$,$ProxySign$,$ProxyVerify$)是代理签名方案,k 为安全参数,T 为攻击类型,其值取枚举型{$type1$,$type2$,$type3$,$type4$},n 为用户数,F 表示敌手,$\Pr(B_{PS}(k,F,T,n)==1)$为算法 B 返回 1 的概率.那么,敌手攻破方案 PS 的优势可定义为

$$Adv_{PS,F}^{ps\text{-}uf}(k,T,n,q_e,q_s,t) = \max_{T\in\{type1,type2,type3,type4\}}\{\Pr(B_{PS}(k,F,T,n)==1)\},$$

其中,q_e 为代理签名私钥询问次数,q_s 为签名询问次数,t 为算法 B 的运行时间.因此,如果敌手攻破方案 PS 的优势可以被忽略,那么方案 PS 将被认为是安全的.

虽然本文的代理签名可证安全模型是建立在公钥密码体制和标准模型[11]上,但具有较强的可扩展性.因此,如果安全模型扩展到基于身份的密码体制,则根据文献[12]定义的安全模型,仅需对本文的安全模型作两点改变:(1) 对本文模型中的用户公私钥部分进行用户身份计算的替换;(2) 在情况 1 下增加对用户身份的私钥查询阶段.由于具体内容大体相同,本文不再详细叙述该扩展的安全模型,而通过对新方案的安全性证明展示安全模型的有效性与可扩展性.

4 新的代理签名方案

本文根据第 2 节描述的代理签名模型,对 Paterson 等人提出的标准模型下基于身份的签名方案[12]进行扩展,提出在标准模型下基于身份的代理签名方案.新方案与标准模型下基于公钥密码体制的代理签名方案相比,增加了用户身份的概念.新方案不仅继承了标准模型下基于身份的签名方案的安全性质,而且也能够满足代理签名方案的安全性质.方案 PS 由 6 种算法组成,具体描述如下:

- Setup 算法

PKG(用户私钥生成中心)系统选择两个 q 阶的循环群 G_1 和 G_2,g 为 G_1 的生成元,并存在一个映射 e:$G_1\times G_1\to G_2$.同时,存在 3 个无碰撞的 Hash 函数: $H:\{0,1\}^*\to\{0,1\}^{n_w}$, $H_u:\{0,1\}^*\to\{0,1\}^{n_u}$ 和 $H_m:\{0,1\}^*\to\{0,1\}^{n_m}$, H 用于把授权书的二进制串映射成固定长度为 n_w 的二进制串,H_u 和 H_m 分别用于把身份和消息的二进制串映射成固定长度为 n_u 和 n_m 的二进制串;

PKG 随机选择 $g_1,\theta\in G_1$;同时,随机选择 $u',m'\in G_1$ 以及两个向量 U=(u_i)和 M=(m_i),其中,$u_i\in G_1$,$m_i\in G_1$,U 和 M

的长度分别为 n_u 和 n_m;最后,PKG 输出公共参数 $params=(G_1,G_2,e,g,g_1,\theta,u',U,m',M)$,同时生成用户公共参数列表 UPK_List,用于发布注册用户生成的公共参数,初始值为空.

- KeyGen 算法

为不失一般性,我们设有原始签名者和代理者两个用户,设原始签名者的身份 u_{de} 为一个长度为 n_u 的二进制串,V_{de} 表示 u_{de} 中比特值为 1 的位置 i 的集合,即 $V_{de}\subseteq\{1,2,\ldots,n_u\}$;对于原始签名者,PKG 随机选择 $x_{de},r_{de}\in Z_q$,计算原始签名者的私钥为

$$sk_{de}=(sk_{de,1},sk_{de,2})=\left(g_1^{x_{de}}\cdot\left(u'\prod_{i\in V_{de}}u_i\right)^{r_{de}},g^{r_{de}}\right).$$

同时,生成原始签名者的用户公共参数 $pk_{de}=g^{x_{de}}$ 加入用户公共参数列表 UPK_List;然后,PKG 通过安全的方式发送私钥 sk_{de} 给原始签名者,原始签名者通过与 keyGen 算法同样的方式获得 V_{de} 集合,然后对收到的私钥 sk_{de} 进行验证 $e(sk_{de,1},g)\stackrel{?}{=}e(g_1,pk_{de})\cdot e\left(u'\prod_{i\in V_{de}}u_i,sk_{de,2}\right)$,如果相等,则接受;否则,可以要求 PKG 重新生成私钥 sk_{de}.

同样,对于代理者,PKG 随机选择 $x_{pr},r_{pr}\in Z_q$,生成代理者的私钥:

$$sk_{pr}=(sk_{pr,1},sk_{pr,2})=\left(g_1^{x_{pr}}\cdot\left(u'\prod_{i\in V_{pr}}u_i\right)^{r_{pr}},g^{r_{pr}}\right),$$

并生成代理者的用户公共参数 $pk_{pr}=g^{x_{pr}}$ 加入用户公共参数列表 UPK_List;然后,PKG 通过安全的方式发送私钥 sk_{pr} 给代理者,代理者对收到的私钥 sk_{pr} 进行验证,如果相等,则接受;否则,可以要求 PKG 重新生成私钥 sk_{pr}.

- Delegate 算法

原始签名者随机选择 $s\in Z_q$,计算 $\delta_1=sk_{de,1}\cdot(\theta^s)^{H(w)}=g_1^{x_{de}}\cdot\left(u'\prod_{i\in V_{de}}u_i\right)^{r_{de}}\cdot(\theta^s)^{H(w)},\delta_2=g^s$,其中,$w$ 为授权书,并令 $\delta_3=sk_{de,2}=g^{r_{de}},\delta_4=w$,于是生成代理签名授权信息$\delta=(\delta_1,\delta_2,\delta_3,\delta_4)$,然后发送$\delta$给代理签名者.

- ProxyKeyGen 算法

代理签名者收到δ后验证 $e(\delta_1,g)\stackrel{?}{=}e(g_1,pk_{de})\cdot e(\theta,\delta_2)^{H(w)}\cdot e\left(u'\prod_{i\in V_{de}}u_i,\delta_3\right)$,如果相等,则接受;否则,要求原始签名者重新授权.代理签名者根据授权信息δ生成代理签名私钥 $psk_{pd}=(psk_{pd,1},psk_{pd,2},psk_{pd,3},psk_{pd,4})$,其中,

$$psk_{pd,1}=sk_{pr,1}\cdot\delta_1=sk_{pr,1}\cdot sk_{de,1}\cdot(\theta^s)^{H(w)},psk_{pd,2}=sk_{pr,2},psk_{pd,3}=\delta_3=sk_{de,2},psk_{pd,4}=\delta_2=g^s.$$

- ProxySign 算法

代理签名者代表原始签名者对消息 m 进行签名,代理签名者通过与 keyGen 算法同样的方式对长度为 n_m 的消息 m 的二进制串进行处理获得 Q 集合,Q 表示 m 中比特值为 1 的位置 j 的集合,即 $Q\subseteq\{1,2,\ldots,n_m\}$;然后,代理签名者随机选择 $r_m\in Z_q$,计算:

$$\sigma_1=psk_{pd,1}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m}=sk_{pr,1}\cdot sk_{de,1}\cdot(\theta^s)^{H(w)}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m},\sigma_2=g^{r_m}.$$

同时,令$\sigma_3=psk_{pd,2},\sigma_4=psk_{pd,3},\sigma_5=psk_{pd,4}$,最后输出代理签名$(w,\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5))$并发送给签名接收者.

- ProxyVerify 算法

签名接收者对代理者对消息 m 的代理签名$(w,\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5))$进行验证:

$$e(\sigma_1,g)=e(g_1,pk_{pr})\cdot e(g_1,pk_{de})\cdot e\left(m'\prod_{j\in Q}m_j,\sigma_2\right)\cdot e\left(u'\prod_{i\in V_{pr}}u_i,\sigma_3\right)\cdot e\left(u'\prod_{i\in V_{de}}u_i,\sigma_4\right)\cdot e(\theta,\sigma_5)^{H(w)}.$$

如果相等,则输出布尔值 *accept*;否则,输出 *reject*.

5 方案分析

5.1 正确性分析

本节对新方案中使用的两个验证式进行推导分析.

(1) 代理签名者对从原始签名者处收到代理签名授权信息$\delta=(\delta_1,\delta_2,\delta_3,\delta_4)$进行推导验证,有

$$e(\delta_1,g)=e\left(g_1^{x_{de}}\cdot\left(u'\prod_{i\in V_{de}}u_i\right)^{r_{de}}\cdot(\theta^s)^{H(w)},g\right)$$

$$=e(g_1^{x_{de}},g)\cdot e\left(\left(u'\prod_{i\in V_{de}}u_i\right)^{r_{de}},g\right)\cdot e((\theta^s)^{H(w)},g)$$

$$=e(g_1,pk_{de})\cdot e\left(u'\prod_{i\in V_{de}}u_i,sk_{de,2}\right)\cdot e(\theta^s,g)^{H(w)}$$

$$=e(g_1,pk_{de})\cdot e\left(u'\prod_{i\in V_{de}}u_i,\delta_3\right)\cdot e(\theta,g^s)^{H(w)}$$

$$=e(g_1,pk_{de})\cdot e(\theta,\delta_2)^{H(w)}\cdot e\left(u'\prod_{i\in V_{de}}u_i,\delta_3\right).$$

(2) 签名接收者对代理者对消息 m 的代理签名$(w,\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5))$进行推导验证,有

$$e(\sigma_1,g)=e\left(sk_{pr,1}\cdot sk_{de,1}\cdot(\theta^s)^{H(w)}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m},g\right)$$

$$=e\left(g_1^{x_{pr}}\cdot\left(u'\prod_{i\in V_{pr}}u_i\right)^{r_{pr}}\cdot g_1^{x_{de}}\cdot\left(u'\prod_{i\in V_{de}}u_i\right)^{r_{de}}\cdot(\theta^s)^{H(w)}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m},g\right)$$

$$=e(g_1,pk_{pr})\cdot e\left(u'\prod_{i\in V_{pr}}u_i,sk_{pr,2}\right)\cdot e(g_1,pk_{de})\cdot e\left(u'\prod_{i\in V_{de}}u_i,sk_{de,2}\right)\cdot e(\theta^s,g)^{H(w)}\cdot e\left(m'\prod_{j\in Q}m_j,\sigma_2\right)$$

$$=e(g_1,pk_{pr})\cdot e(g_1,pk_{de})\cdot e\left(m'\prod_{j\in Q}m_j,\sigma_2\right)\cdot e\left(u'\prod_{i\in V_{pr}}u_i,\sigma_3\right)\cdot e\left(u'\prod_{i\in V_{de}}u_i,\sigma_4\right)\cdot e(\theta,\sigma_5)^{H(w)}.$$

5.2 效率分析

5.2.1 计算性能分析

新方案利用 PKG 系统生成用户私钥、系统参数以及用户公共参数,虽然 PKG 系统会带来私钥和参数托管等问题,但根据 Boneh 等人在亚密会议 2008 上提出的广义的基于身份加密方案(generalized identity-based encryption)[24],利用 PKG 系统构造的基于身份的加密方案被公认为是一种标准且安全的方案.因此,基于 PKG 系统构造的签名方案也将是一种标准的方案[12].不过在新方案中,随着注册用户数量的增加,不仅用户公共参数列表 UPK_List 长度增加,对 PKG 系统维护的工作量也将增加.因此,如何在 PKG 系统中减少用户公共参数列表 UPK_List,提出在标准模型下更高效的基于身份的代理签名方案,将是本文下一步的工作.同时,分析新方案,代理者在进行代理签名时需要计算 $\sigma_1=psk_{pd,1}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m}$ 和 $\sigma_2=g^{r_m}$,由于 $\sigma_2=g^{r_m}$ 可以预先计算,因此,代理者的平均在线计算量为 n_m+1 次群元素的乘法运算和一次指数运算;而对于验证方,从验证式

$$e(\sigma_1,g)=e(g_1,pk_{pr})\cdot e(g_1,pk_{de})\cdot e\left(m'\prod_{j\in Q}m_j,\sigma_2\right)\cdot e\left(u'\prod_{i\in V_{pr}}u_i,\sigma_3\right)\cdot e\left(u'\prod_{i\in V_{de}}u_i,\sigma_4\right)\cdot e(\theta,\sigma_5)^{H(w)}$$

分析可知,由于 $e(g_1, pk_{pr})\cdot e(g_1, pk_{de}), m'\prod_{j\in Q} m_j, u'\prod_{i\in V_{pr}} u_i$ 和 $u'\prod_{i\in V_{de}} u_i$ 都可以预先计算,因此,验证方的在线计算量为 5 次双线性对计算、1 次 hash 函数计算、1 次指数运算和 4 次乘法运算.

5.2.2 与其他方案的比较分析

目前,许多代理签名方案[7,13,14,16−23]都已经被提出,这些方案都是基于各种数学假设,如大素数分解问题[14,16]、离散对数问题[13,17,18]和 CDH 问题[7,19−23]等.相对于建立在随机预言机模型下的方案,建立在标准模型下的本文方案具有更好的可证安全性.同时,本文方案的安全性是建立在 CDH 问题的困难性上,与其他建立在大素数分解问题与离散对数问题的方案相比[13,14,16−18],虽然本文方案利用双线性对计算需要更大的计算消耗,但本文方案具有更短的系统参数和签名长度.另外,在安全参数相同的情况下,本文方案具有更强的安全性.同时,为了更好地比较分析本文方案与其他方案,我们详细地比较了本文方案与同样建立在 CDH 问题上的另外两种标准模型下的方案[7,20].由于本文方案加入了用户身份的概念,因此相对于另外两种没有基于身份的方案[7,20]来说,本文方案扩展了标准模型下代理签名方案的应用范围,且具有更完备的安全性.表 1~表 3 展示了 3 种方案的详细对比情况.表 1 展示了 3 种方案的公私钥长度对比情况.本文方案加入了用户身份的概念,因此虽然相对于其他两种方案具有较长的私钥,但本文方案不需要用户公钥的存在.表 2 展示了 3 种方案的代理授权和签名长度的对比情况.由于本文方案加入了用户身份的概念,因此具有相对较长的代理授权和签名长度.表 3 展示了 3 种方案的计算性能对比情况(3 种方案都不考虑预先计算的情况).比较于另外两种方案,本文方案在签名上需要最少的计算开耗;不过,由于本文方案在验证签名过程中需要对用户身份进行验证,因此增加了部分计算开耗,大概比另外两种方案多出两个双线性对的计算时间.

Table 1 Key length comparison of three schemes

表 1 3 种方案的公私钥长度对比

	私钥长度	公钥长度
方案[20]	$2\cdot\|Z_q\|$	$2\cdot\|G_1\|$
方案[7]	$2\cdot\|Z_q\|$	$2\cdot\|G_1\|$
本文方案	$2\cdot\|G_1\|$	无

说明:$|Z_q|$表示 Z_q 中的元素长度,$|G_1|$表示 G_1 中的元素长度.

Table 2 Delegation and signature length comparison of three schemes

表 2 3 种方案的代理授权和签名长度对比

	代理授权长度	签名长度
方案[20]	$2\cdot\|G_1\|+\|w\|$	$3\cdot\|G_1\|+\|w\|$
方案[7]	$2\cdot\|G_1\|+\|w\|$	$3\cdot\|G_1\|+\|w\|$
本文方案	$3\cdot\|G_1\|+\|w\|$	$5\cdot\|G_1\|+\|w\|$

说明:$|G_1|$表示 G_1 中的元素长度,$|w|$表示授权书的长度.

Table 3 Performance comparison of three schemes

表 3 3 种方案的计算性能对比

	签名	验证
方案[20]	$(2\cdot n_m+4)\cdot C_{mul1}+4\cdot C_{exp}$	$2\cdot n_m\cdot C_{mul1}+5\cdot C_{pairing}+3\cdot C_{mul2}$
方案[7]	$C_h+3\cdot C_{mul1}+4\cdot C_{exp}$	$C_h+(n_m+1)\cdot C_{mul1}+(n_m+1)\cdot C_{exp}+5\cdot C_{pairing}+3\cdot C_{mul2}$
本文方案	$(n_m+1)\cdot C_{mul1}+2\cdot C_{exp}$	$C_h+n_m\cdot C_{mul1}+2\cdot n_u\cdot C_{mul1}+C_{exp}+7\cdot C_{pairing}+5\cdot C_{mul2}$

说明:C_{mul1} 表示 G_1 上的一次乘法操作,C_{mul2} 表示 G_2 上的一次乘法操作,

C_{exp} 表示 G_1 上的一次指数操作, $C_{pairing}$ 表示一次双线性对计算,C_h 表示一次哈希函数计算.

5.3 安全性分析

新方案的安全性将在本文的代理签名安全模型下,通过敌手 F 启动算法 B 模拟挑战者与其进行交互来证明,证明新方案的安全性在标准模型下可归约于 CDH 问题假定.由于在本文的代理签名安全模型中,方案需要分 4 种情况来证明其安全性,因此本节在 4 种情况下给出方案的安全性定理.同时,在方案安全性的证明过程中,

为了更好地说明分析,本节将省略在安全模型中定义的细节,并且最大化敌手的优势.同时,由于敌手获得了所有用户的授权书,因此去除 Hash Oracle 的询问.

定理 1. 如果对于(t',ε')-CDH 问题假定成立,那么本文的代理签名方案的普通签名方案是(t,ε,q_e,q_s)-安全的,其中,

$$\varepsilon'=\frac{\varepsilon}{16\cdot(q_e+q_s)\cdot q_s\cdot(n_u+1)\cdot(n_m+1)},$$
$$t'=t+O((q_e\cdot n_u+q_s\cdot(n_u+n_m))\cdot C_{mul}+(q_e+q_s)\cdot C_{exp}).$$

q_e是签名私钥询问次数,q_s是签名询问次数,n_u是签名者身份的二进制串长度,n_m是消息 m 的二进制串长度,C_{mul}表示群元素的乘法运算时间,C_{exp}表示指数运算时间.

证明:本文方案是对 Paterson 等人提出的标准模型下基于身份的签名方案[12]的扩展,由于 Paterson 等人在文献[12]中已经证明该标准(普通)签名方案是(t,ε,q_e,q_s)-安全的,因此本文方案的普通签名方案同样也是(t,ε,q_e,q_s)-安全的,具体证明过程略. □

定理 2. 如果对于(t',ε')-CDH 问题假定成立,那么本文签名方案在情况 2 下是(t,ε,q_e,q_s)-安全的,其中,

$$\varepsilon'=\frac{\varepsilon}{16q(q_e+q_s)q_s(n_u+1)(n_m+1)},$$
$$t'=t+O(q_e\cdot[(n_u+4)\cdot C_{mul}+5.4\cdot C_{exp}+3\cdot C_{pairing}]+q_s\cdot[(n_m+n_u+1)\cdot C_{mul}+8.8\cdot C_{exp}]).$$

q 是群的阶,q_e是代理签名私钥询问次数,q_s是签名询问次数,n_u是身份 u 的二进制串长度,n_m是消息 m 的二进制串长度,C_{mul}表示群元素的乘法运算时间,C_{exp}表示指数运算时间,$C_{pairing}$表示双线性对运算时间.

证明:对于存在 n 个用户的代理签名方案 PS,假设存在一个(t,ε,q_e,q_s)的敌手 F 攻击方案,可以构造一个算法 B,在至多 t'时间内以至少ε'的概率解决 CDH 问题.本文的证明将基于文献[12]的证明思路,因此,对于给定的$(g,g^a,g^b)\in G_1$,其中,$a,b\in Z_q$,为了能够计算 $g^{a\cdot b}$,算法 B 模拟挑战者与敌手 F 进行交互,具体交互过程如下:

- 系统设置(setup)

令 $l_u=2(q_e+q_s)$,$l_m=2q_s$,随机选择 $k_u\in Z_{l_u}$,$k_m\in Z_{l_m}$,且有 $0\leqslant k_u\leqslant n_u$,$0\leqslant k_m\leqslant n_m$;对于给定的 q_e,q_s,n_u,n_m,假设有 $l_u(n_u+1)<q$,$l_m(n_m+1)<q$;随机选择 $x'\in Z_{l_u}$、长度为 n_u 的向量 $X=(x_i)$,其中,$x_i\in Z_{l_u}$;随机选择 $z'\in Z_{l_m}$、长度为 n_m 的向量 $Z=(z_j)$,其中,$z_j\in Z_{l_m}$;同时,随机选择 $y',w'\in Z_q$ 以及选择长度为 n_u 的向量 $Y=(y_i)$和长度为 n_m 的向量 $W=(w_j)$,其中,$y_i,w_j\in Z_q$.为了使符号更容易转换,定义如下 4 个函数:

$$F(u)=x'+\sum_{i\in V}x_i-l_u\cdot k_u,\ J(u)=y'+\sum_{i\in V}y_i,\ K(m)=z'+\sum_{j\in Q}z_j-l_m\cdot k_m,\ L(m)=w'+\sum_{j\in Q}w_j.$$

现在,算法 B 构造系统参数:$g_1=g^b$,随机选择$\ell\in Z_q$,设置:

$$\theta=g_1^{\ell}\cdot g,u'=g_1^{-l_u\cdot k_u+x'}\cdot g^{y'},u_i=g_1^{x_i}\cdot g^{y_i},1\leqslant i\leqslant n_u,m'=g_1^{-l_m\cdot k_m+z'}\cdot g^{w'},m_j=g_1^{z_j}\cdot g^{w_j},1\leqslant j\leqslant n_m,$$

则 $u'\prod_{i\in V}u_i=g_1^{F(u)}\cdot g^{J(u)}$,$m'\prod_{j\in Q}m_j=g_1^{K(m)}\cdot g^{L(m)}$,并输出用户公共参数列表 UPK_List.

在情况 2 下,敌手 F 获得全部需要的资源,如全部用户的授权书、相应用户的用户公共参数等,则敌手 F 进入询问阶段.

- 询问(queries)

算法 B 与敌手 F 进行交互,回答敌手 F 的如下询问,询问分为两个阶段:

(1) 代理签名私钥询问

在情况 2 下,进行代理签名私钥询问,模拟某个不被控制的用户 u^*进行自我授权的交互视图.由于算法 B 仅知道用户 u^*的公共参数 $pk_{u^*}=g^{x_{u^*}}$ 但不知道用户 u^*的私钥 sk_{u^*},则对于 $F(u^*)\neq 0 \bmod q$,随机选择 $r_{u^*}\in Z_q$,计算

$$sk_{u^*}=\left(sk_{u^*,1},sk_{u^*,2}\right)=\left((pk_{u^*})^{-\frac{J(u^*)}{F(u^*)}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{r_{u^*}},(pk_{u^*})^{-\frac{1}{F(u^*)}}\cdot g^{r_{u^*}}\right).$$

令 $r'_{u^*}=r_{u^*}-\frac{x_{u^*}}{F(u^*)}$,则 sk_{u^*} 是有效的用户 u^*的私钥(与文献[12]的证明同理).然后,随机选择 $s\in Z_q$,计算 $\delta_1=sk_{u^*,1}\cdot(\theta^s)^{H(w_{u^*})},\delta_2=g^s$,并令 $\delta_3=sk_{u^*,2},\delta_4=w_{u^*}$,其中,$w_{u^*}$ 为用户 u^*的授权书,于是生成代理签名授权信息 $\delta=(\delta_1,\delta_2,\delta_3,\delta_4)$,发送$\delta$给代理签名者 u^*(模拟完成自我授权);模拟代理签名者 u^*收到δ后对其进行验证,由于 $r'_{u^*}=r_{u^*}-\frac{x_{u^*}}{F(u^*)},sk_{u^*}$ 是有效的用户 u^*的私钥,因此授权验证能够通过;最后,根据授权信息δ计算 $psk_{u^*,1}=sk_{u^*,1}\cdot\delta_1=sk_{u^*,1}\cdot sk_{u^*,1}\cdot(\theta^s)^{H(w_{u^*})}$,并令 $psk_{u^*,2}=sk_{u^*,2},psk_{u^*,3}=sk_{u^*,2},psk_{u^*,4}=g^s$,于是生成用户 u^*的自我代理签名私钥 $psk_{u^*}=\left(psk_{u^*,1},psk_{u^*,2},psk_{u^*,3},psk_{u^*,4}\right)$.

那么,对敌手 F 来说,算法 B 所产生的关于某个用户身份 u^*的自我代理签名私钥与实际所产生的自我代理签名私钥不可区分.如果 $F(u^*)=0 \bmod q$,则算法 B 不能进行,模拟终止.

(2) 签名询问

进行一个关于不被控制的用户 u^*对消息 m 的自我代理签名询问.当 $K(m)\neq 0 \bmod q$ 时,算法随机选择 $r_{u^*},r_m,s\in Z_q$,计算

$$\sigma_1=(pk_{u^*})^{-\frac{2\cdot L(m)}{K(m)}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m}\cdot\theta^{s\cdot H(w_{u^*})},\sigma_2=(pk_{u^*})^{-\frac{2}{K(m)}}\cdot g^{r_m},\sigma_3=g^{r_{u^*}},\sigma_4=g^{r_{u^*}},\sigma_5=g^s.$$

最后,输出代理签名 $(w_{u^*},\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5))$ 并发送给敌手 F.

当 $r'_m=r_m-\frac{2\cdot x_{u^*}}{K(m)}$,代理签名 $(w_{u^*},\sigma=(\sigma_1,\sigma_2,\sigma_3,\sigma_4,\sigma_5))$ 为有效签名,这是因为

$$\begin{aligned}\sigma_1&=(pk_{u^*})^{-\frac{2\cdot L(m)}{K(m)}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m}\\&=(g^{2\cdot x_{u^*}})^{-\frac{L(m)}{K(m)}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m}\cdot\theta^{s\cdot H(w_{u^*})}\\&=g_1^{2\cdot x_{u^*}}\cdot(g_1^{K(m)}\cdot g^{L(m)})^{-\frac{2\cdot x_{u^*}}{K(m)}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}}\cdot(g_1^{K(m)}\cdot g^{L(m)})^{r_m}\cdot\theta^{s\cdot H(w_{u^*})}\\&=g_1^{2\cdot x_{u^*}}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r'_m}\cdot\theta^{s\cdot H(w_{u^*})},\\\sigma_2&=(pk_{u^*})^{-\frac{2}{K(m)}}\cdot g^{r_m}=g^{-\frac{2\cdot x_{u^*}}{K(m)}}\cdot g^{r_m}=g^{r_m-\frac{2\cdot x_{u^*}}{K(m)}}=g^{r'_m},\end{aligned}$$

则对敌手 F 来说,算法 B 所产生的用户 u^*对消息 m 的自我代理签名与真实的自我代理签名不可区分.如果 $K(m)=0 \bmod q$,则算法 B 不能进行,模拟终止.

- 伪造阶段(forgery)

如果算法 B 在询问阶段没有终止,那么敌手 F 至少可以成功地以ε的概率返回用户 u^*对消息 m^*的有效的自我代理签名 $(w_{u^*},\sigma^*=(\sigma_1^*,\sigma_2^*,\sigma_3^*,\sigma_4^*,\sigma_5^*))$,其中,

$$\sigma^*=(\sigma_1^*,\sigma_2^*,\sigma_3^*,\sigma_4^*,\sigma_5^*)=\left(g_1^{2\cdot x_{u^*}}\cdot\theta^{s^*\cdot H(w_{u^*})}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}^*}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m^*},g^{r_m^*},g^{r_{u^*}^*},g^{r_{u^*}^*},g^{s^*}\right).$$

因此,当 $F(u^*)\neq 0 \bmod q$,或者 $K(m^*)\neq 0 \bmod q$,或者 $\ell\cdot H(w_{u^*})\neq 0 \bmod q$ 时,算法 B 停止;

当 $F(u^*)=0 \bmod q$,且 $K(m^*)=0 \bmod q$,且 $\ell \cdot H(w_{u^*})=0 \bmod q$ 时,算法 B 可计算(设 $a=x_{u^*}$):

$$\left(\frac{\sigma_1^*}{\left(g^{r_{u^*}^*}\right)^{2\cdot J(u^*)}\cdot(g^{r_m^*})^{L(m^*)}\cdot(g^{s^*})^{H(w_{u^*})}}\right)^{\frac{1}{2}}=\left(\frac{g_1^{2\cdot x_{u^*}}\cdot\theta^{s^*\cdot H(w_{u^*})}\cdot\left(u'\prod_{i\in V_{u^*}}u_i\right)^{2\cdot r_{u^*}^*}\cdot\left(m'\prod_{j\in Q}m_j\right)^{r_m^*}}{\left(g^{r_{u^*}^*}\right)^{2\cdot J(u^*)}\cdot(g^{r_m^*})^{L(m^*)}\cdot(g^{s^*})^{H(w_{u^*})}}\right)^{\frac{1}{2}}$$

$$=\left(\frac{g_1^{2\cdot x_{u^*}}\cdot(g^{\ell}\cdot g)^{s^*\cdot H(w_{u^*})}\cdot(g_1^{F(u^*)}\cdot g^{J(u^*)})^{2\cdot r_{u^*}^*}\cdot(g_1^{K(m^*)}\cdot g^{L(m^*)})^{r_m^*}}{\left(g^{r_{u^*}^*}\right)^{2\cdot J(u^*)}\cdot(g^{r_m^*})^{L(m^*)}\cdot(g^{s^*})^{H(w_{u^*})}}\right)^{\frac{1}{2}}$$

$$=(g_1^{2\cdot x_{u^*}})^{\frac{1}{2}}$$

$$=g_1^{x_{u^*}}$$

$$=g^{a\cdot b}.$$

即,输出 CDH 问题.

现在我们分析一下算法 B 模拟成功的概率.因为需要完整的运行整个算法才能解决 CDH 问题,所以算法 B 在询问阶段、伪造阶段都不能终止.与文献[12]中的概率分析相同,我们可以得到如下算法 B 模拟成功的概率为

$$P(B\_Success)\geqslant\frac{1}{4(q_e+q_s)(n_u+1)}\cdot\frac{1}{4q_s(n_m+1)}\cdot\frac{1}{q}=\frac{1}{16q(q_e+q_s)q_s(n_m+1)(n_u+1)}.$$

所以有 $\varepsilon'=\dfrac{\varepsilon}{16q(q_e+q_s)q_s(n_u+1)(n_m+1)}$.

因此,如果算法 B 没有被终止,敌手 F 可以以ε的概率伪造一个有效的用户 u^*的自我代理签名,并且算法 B 可以通过解决 CDH 问题计算出 $g^{a\cdot b}$.在算法 B 的整个运行过程中,如果不计整数加法与乘法运算时间以及 hash 函数的计算时间,且在代理签名私钥询问中对授权代理信息进行验证时部分计算可以预先进行,则算法 B 完成整个运算过程的时间大约需要

$$t'=t+O(q_e\cdot[(n_u+4)\cdot C_{mul}+5.4\cdot C_{exp}+3\cdot C_{pairing}]+q_s\cdot[(n_m+n_u+1)\cdot C_{mul}+8.8\cdot C_{exp}]).$$

因此,算法 B 能够在时间 t'内以ε'的概率解决 G_1 上 CDH 问题,但这与(t',ε')-CDH 问题假定矛盾.因此,本文的签名方案在情况 2 下是(t,ε,q_e,q_s)-安全的. □

定理 3. 如果对于(t',ε')-CDH 问题假定成立,那么本文签名方案在情况 3 和情况 4 下都是(t,ε,q_e,q_s)-安全的,其中,

$$\varepsilon'=\frac{\varepsilon}{16q(q_e+q_s)q_s(n_u+1)(n_m+1)},$$

$$t'=t+O(q_e\cdot[(n_u+4)\cdot C_{mul}+5.4\cdot C_{exp}+3\cdot C_{pairing}]+q_s\cdot[(n_m+n_u+4)\cdot C_{mul}+7.4\cdot C_{exp}]).$$

q 是群的阶,q_e 是代理签名私钥询问次数,q_s 是签名询问次数,n_u 是身份 u 的二进制串长度,n_m 是消息 m 的二进制串长度,C_{mul} 表示群元素的乘法运算时间,C_{exp} 表示指数运算时间,$C_{pairing}$ 表示双线性对运算时间.

证明:与定理 2 证明同理,在情况 3 和情况 4 下,敌手已经控制一方(原始签名者或者代理签名者)的私钥,因此在代理签名私钥询问阶段,模拟算法仅需要先模拟出未被控制一方的私钥再生成代理授权信息.由于其证明过程大部分与定理 2 相似,因此这里忽略. □

6 结束语

本文在总结分析文献[2–8]的基础上对代理签名可证安全模型进行了详细的形式化定义,提出了一种完整

的安全模型.同时,为了展示安全模型的有效性与可扩展性,给出了一个在标准模型下基于身份的代理签名方案的可证安全实例.本文给出的代理签名可证安全模型最大化地定义了敌手的优势,并分 4 种情况对代理签名方案的安全性进行了定义.同时,本文在文献[12]的基础上提出一种在标准模型下基于身份的代理签名方案,并在可证安全模型下证明了新方案具有在自适应选择消息攻击下存在基于身份的代理签名不可伪造性,其安全性可归约于 CDH 问题假定.本文提出的代理签名安全模型是完备实用的,同时,提出的代理签名方案与标准模型下的其他代理签名方案相比具有更完整的可证安全性.不过,新方案的效率有待进一步提高,这也是本文下一步研究的重点:如何进一步改进方案,提高方案的效率.

References:

[1] Mambo M, Usuda K, Okamoto E. Proxy signature for delegating signing operation. In: Proc. of the 3rd ACM Conf. on Computer and Communications Security. New York: ACM Press, 1996. 48−57. [doi: 10.1145/238168.238185]

[2] Boldyreva A, Palacio A, Warinschi B. Secure proxy signature schemes for delegation of signing rights. http://eprint.iacr.org/2003/096.pdf [doi: 10.1007/s00145-010-9082-x]

[3] Malkin T, Obana S, Yung M. The hierarchy of key evolving signatures and a characterization of proxy signatures. In: Cachin C, Camenish J, eds. Proc. of the Advances in Cryptology-EUROCRYPT 2004. LNCS 3027, Berlin: Springer-Verlag, 2004. 306−322. [doi: 10.1007/978-3-540-24676-3_19]

[4] Jacob C. N. Schuldt, Kanta Matsuura, Kenneth G. Paterson. Proxy signatures secure against proxy key exposure. In: Cramer R, ed. Proc. of the Public Key Cryptography-PKC 2008. LNCS 4939, Berlin: Springer-Verlag, 2008. 141−161. [doi: 10.1007/978-3-540-78440-1_9]

[5] Xu F, Cui J, Huang H. A provably-secure proxy signature scheme from bilinear pairings. Acta Electronica Sinica, 2009, 37(3):439−443 (in Chinese with English abstract).

[6] Ming Y, Wang YM. Designated verifier proxy signature scheme without random oracles. Journal of Electronics & Information Technology, 2008,30(3):668−671 (in Chinese with English abstract).

[7] Sun Y, Xu CX, Yu Y, Mu Y. Strongly unforgeable proxy signature scheme secure in the standard model. Journal of Systems and Software, 2011,84(9):1471−1479. [doi: 10.1016/j.jss.2011.02.041]

[8] Boldyreva A, Palacio A, Warinschi B. Secure proxy signature schemes for delegation of signing rights. Journal of Cryptology, 2012, 25(1):57−115. [doi: 10.1007/s00145-010-9082-x]

[9] Boneh D, Boyen X. Short signatures without random oracles. In: Cachin C, Camenish J, eds. Proc. of the Advances in Cryptology-EUROCRYPT 2004. LNCS 3027, Berlin: Springer-Verlag, 2004. 56−73. [doi: 10.1007/978-3-540-24676-3_4]

[10] Cha J, Cheon J. An identity-based signature from gap diffie-Hellman groups. In: Desmedt YG, ed. Proc. of the Public Key Cryptography-PKC 2003. LNCS 2567, Berlin: Springer-Verlag, 2003. 18−30. [doi: 10.1007/3-540-36288-6_2]

[11] Waters B. Efficient identity-based encryption without random oracles. In: Cramer R, ed. Proc. of the Advances in Cryptology-EUROCRYPT 2005. LNCS 3494, Berlin: Springer-Verlag, 2005. 114−127. [doi: 10.1007/b136415]

[12] Paterson KG, Schuldt JCN. Efficient identity-based signatures secure in the standard model. In: Batten L, Safavi-Naini R, eds. Proc. of the ACISP 2006. LNCS 4058, Berlin: Springer-Verlag, 2006. 207−222. [doi: 10.1007/11780656_18]

[13] Cui SJ, Wen FT. Improvement of a forward-secure proxy signature scheme. In: Proc. of the Computer Engineering and Technology 2010 (ICCET 2010). New Jersey: IEEE Computer Society, 2010. 1441−1444. [doi: 10.1109/ICCET.2010.5486056]

[14] Xue QS, Cao ZF, Qian HF. A generalized proxy signature scheme based on the RSA cryptosystem. In: Liew KM, *et al*., eds. Proc. of the PDCAT 2004. LNCS 3320, Berlin: Springer-Verlag, 2004. 662−665. [doi: 10.1007/978-3-540-30501-9_127]

[15] Xia XS, Hong F, Cui GH. Security analysis of two forward secure proxy signature schemes. Application Research of Computers, 2009,26(2):709−710 (in Chinese with English abstract).

[16] Shao ZH. Provably secure proxy-protected signature schemes based on RSA. Computers & Electrical Engineering, 2009,35(3): 497−505. [doi: 10.1016/j.compeleceng.2008.11.028]

[17] Huang HF, Chang CC. A novel efficient (*t*,*n*) threshold proxy signature scheme. Information Sciences, 2006,176(10):1338−1349. [doi: 10.1016/j.ins.2005.02.010]

[18] Kim SJ, Park SJ, Won DH. Proxy signatures, revisited. In: Proc. of the ICICS'97. LNCS 1334, Berlin: Springer-Verlag, 1997. 223−232. [doi: 10.1007/BFb0028478]

[19] Cao F, Cao ZF. A secure identity-based multi-proxy signature scheme. Computers and Electrical Engineering, 2009,35(1):86−95. [doi: 10.1016/j.compeleceng.2008.05.005]

[20] Huang XY, Susilo W, Mu Y, Wu W. Proxy signature without random oracles. In: Cao J, *et al*., eds. Proc. of the Mobile Ad-Hoc and Sensor Networks 2006. LNCS 4325, Berlin: Springer-Verlag, 2006. 473−484. [doi: 10.1007/11943952_40]

[21] Jin ZP, Wen QY. Certificateless multi-proxy signature. Computer Communications, 2011,34(3):344−352. [doi: 10.1016/j.comcom.2010.06.013]

[22] Liu ZH, Hu YP, Zhang XS, Ma H. Provably secure multi-proxy signature scheme with revocation in the standard model. Computer Communications, 2011,34(3):494−501. [doi: 10.1016/j.comcom.2010.05.015]

[23] Xu J, Zhang ZF, Feng DG. ID-Based proxy signature using bilinear pairings. http://eprint.iacr.org/2004/206.pdf [doi: 10.1007/11576259_40]

[24] Boneh D, Hanburg M. Generalized identity based and broadcast encryption schemes. In: Pieprzyk J, ed. Proc. of the Advances in Cryptology-ASIACRYPT 2008. LNCS 5350, Berlin: Springer-Verlag, 2008. 455−470. [doi: 10.1007/978-3-540-89255-7_28]

附中文参考文献:

[5] 许峰,崔隽,黄皓.基于双线性配对的可证安全的代理签名方案.电子学报,2009,37(3):439−443.

[6] 明洋,王育民.无随机预言机下的指定验证者代理签名方案.电子与信息学报,2008,30(3):668−671.

[15] 夏祥胜,洪帆,崔国华.两个前向安全的代理签名方案的安全性分析.计算机应用研究,2009,26(2):709−710.

谷科(1980－),男,湖南长沙人,博士生,主要研究领域为信息安全,数字签名,移动电子商务,密码学应用.

王四春(1965－),男,博士,教授,主要研究领域为信息安全,信息系统管理,人工智能理论与方法.

贾维嘉(1957－),男,博士,教授,博士生导师,主要研究领域为移动计算,下一代无线网络通信及协议,异构网络,无线网络安全.

石良武(1960－),男,教授,主要研究领域为信息安全,信息系统管理,计算机网络与多媒体.