ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.21, No.8, August 2010, pp.2079−2088
doi: 10.3724/SP.J.1001.2010.03634

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

简单的通用可组合代理重签名方案*

洪 璇[1+], 陈克非[2,3], 万中美[2]

[1](上海师范大学 计算机科学与技术系,上海 200234)

[2](上海交通大学 计算机科学与工程系,上海 200240)

[3](现代通信国家重点实验室,四川 成都 610041)

Simplified Universally Composable Proxy Re-Signature

HONG Xuan[1+], CHEN Ke-Fei[2,3], WAN Zhong-Mei[2]

[1](Department of Computer Science and Technology, Shanghai Normal University, Shanghai 200234, China)

[2](Department of Computer Science and Engineering, Shanghai Jiaotong University, Shanghai 200240, China)

[3](National Laboratory of Modern Communications, Chengdu 610041, China)

+ Corresponding author: E-mail: sh.xuanhong@gmail.com

Hong X, Chen KF, Wan ZM. Simplified universally composable proxy re-signature. *Journal of Software*, 2010, 21(8):2079−2088. http://www.jos.org.cn/1000-9825/3634.htm

Abstract: The paper presents a simple proxy re-signature scheme and its two equivalent security model. One is based on the universal composability framework, another is game-based security model. The proposed scheme is bidirectional, multi-use, transitive and key optimal. It is very attractive for its simplicity. Its security can be reduced to the Computational Diffie-Hellman assumption in the Random Oracle Model. It is also secure under the universal composability framework.

Key words: proxy re-signature; universally composable; provable security

摘 要: 给出一个简单的代理重签名方案及其两个等价的安全模型:基于通用可组合安全框架的安全模型和基于游戏的安全模型.提出的方案是双向的、多用的、可传递的和密钥最优化的,其安全性可以规约到随机预言机模型下的计算 Diffie-Hellman 假设.该方案同时也满足通用可组合安全性.

关键词: 代理重签名;通用可组合安全;可证明安全

中图法分类号: TP309 **文献标识码**: A

Blaze 等人[1]在 1998 年的 EUROCRYPT 中首次提出了代理重签名的概念.一个拥有某些额外信息的半可信代理人可以把受理人的签名转换为授权人关于同一个消息的签名,而这个代理人不能得到受理人或授权人的签名密钥,也不能任意生成他们的签名.代理重签名的概念虽然很早就被提出来了,但是由于 Blaze 等人[1]没有

* Supported by the National High-Tech Research and Development Plan of China under Grant No.2007AA01Z456 (国家高技术研究发展计划(863)); the National Basic Research Program of China under Grant No.2007CB311201 (国家重点基础研究发展计划(973)); the National Laboratory of Modern Communications of China under Grant No.9140C1103020803 (现代通信国家重点实验室); the Shanghai Normal University General Project of China under Grant No.SK201037 (上海师范大学一般项目)

Received 2008-09-09; Revised 2009-01-14; Accepted 2009-03-31

给出代理重签名的形式化规范的定义,所以人们不能很好地认识其优点,也很容易把它与其他签名类型(如代理签名、传递签名、群签名、多签名、聚合签名等)相混淆.为了改变这种情况,Ateniese 等人[2]对代理重签名给出了形式化的定义,特别是与上述几种签名类型进行区别.代理重签名方案由于其独特的签名转换功能,可以应用于许多应用场景[3,4],如简化证书管理、路径证明、管理群签名、构造 DRM 的跨域操作系统等.例如,通过代理重签名可以简化不同证书中心(certificate authority,简称 CA)的用户之间建立安全信道的过程.只要 CA1 和 CA2 之间存在一个半可信的代理人,它就能暂时地把 CA1 的证书转换为 CA2 的证书,使得用户之间建立安全信道的过程更为快捷.一般来说,代理重签名可以通过双向性或单向性、多用性或单用性、传递性或非传递性、密钥最优化这几个属性[2,5]来进行分类.

首个代理重签名方案[1]是由 Blaze 等人提出来的.该方案是双向的、多用的.但是,在受理人签名的产生过程和签名转换的过程中,用户需要进行 k 个指数计算,计算效率低下,因此,该方案不适合在实际应用中使用.虽然在 2003 年 Ivan 和 Dodis[6]再次讨论了代理重签名体制,但是他们所给出的方案对于每个授权过程都需要存储一个相应的秘密信息,没有实现密钥最优化.Ateniese 等人[2]首次形式化了代理重签名定义及其安全模型,并提出两个方案,其中,S_{bi} 是双向的和多用的,S_{uni} 是单向的和单用的.但是,他们对安全属性的描述不够简练,所给出的用公式表示的安全模型存在很多冗余.第一个可证明安全的代理重签名方案 S_{mb} 是由 Shao 等人[5]提出来的,同时还提出了第一个基于身份的方案 $S_{id\text{-}mb}$.这两个方案最主要的问题是需要大量的公开参数和耗时的计算.而且他们定义的安全模型有很多限制,影响了方案在实际应用中的推广.

代理重签名是现代密码学的一个新兴研究领域.本文给出了代理重签名的形式化定义和两个等价的安全模型:基于 UC 框架的安全模型和基于游戏的安全模型.这两个安全模型的等价确保了如果方案满足常用的基于游戏的安全模型,那么该方案同时还具有通用可组合性.本文提出的基于游戏的安全模型吸收了以往的工作成果,进行了一些改进,去除了一些限制,简化了一些描述.首先允许攻击者能够控制“corrupted”用户和“honest”用户之间的代理人,其次确保了授权人和受理人的秘密信息的安全性.而本文的基于 UC 框架的安全模型则是首次针对代理重签名概念提出来的.本文还提出了具有 UC 安全的双向代理重签名方案 π_{BPRS}.方案 π_{BPRS} 是在 ElGamal 族三元签名方案的基础上增加再签名密钥生成算法和再签名算法后变形得到的.与其他方案相比,方案 π_{BPRS} 更简单,更容易应用于现有方案.方案 π_{BPRS} 是双向的、多用的、传递的和密钥最优的,其安全性可以规约到随机预言机模型下的计算 Diffie-Hellman 假设.因为已经证明了两个安全模型等价,所以方案 π_{BPRS} 还具有通用可组合安全性.

1 背景知识

1.1 代理重签名

定义 1.1. 一个代理重签名方案由 5 个多项式时间算法(*KeyGen*,*ReKey*,*Sign*,*ReSign*,*Verify*)组成:

- $KeyGen(1^k)\rightarrow(pk,sk)$.输入安全参数 1^k,密钥生成算法 *KeyGen* 输出一对公私钥(pk,sk).
- $KeyGen(sk_a,sk_b)\rightarrow(rk_{a\leftrightarrow b})$.输入用户 *A* 和用户 *B* 的密钥 sk_a,sk_b,再签名密钥生成算法 *ReKey* 输出 *A* 和 *B* 之间的再签名密钥 $rk_{a\leftrightarrow b}$.
- $Sign(sk,m)\rightarrow\sigma$.输入私钥 *sk*、消息 *m*,签名算法 *Sign* 输出消息 *m* 的签名σ.该签名可以用对应 *sk* 的公钥 *pk* 验证.称这种形式的签名为原始签名.
- $Resign(rk_{a\leftrightarrow b},pk_a,m,\sigma)\rightarrow\sigma'$.输入再签名密钥 $rk_{a\leftrightarrow b}$、消息 *m*、用户 *A* 的公钥 pk_a 以及用 pk_a 可以验证的 *m* 的签名σ,再签名算法 *Resign* 输出消息 *m* 的新签名σ'.该签名可以用用户 *B* 的公钥 pk_b 验证.称这种形式的签名为再签名.
- $Verify(pk,m,\sigma)\rightarrow f$.输入消息 *m*、公钥 *pk* 和签名$\sigma$,如果签名$\sigma$确实是可以用 *pk* 验证的 *m* 的签名,则验证算法 *Verify* 输出 1;否则,输出 0.

1.2 通用可组合安全框架

现有的绝大多数协议分析都需要先确立正确的模型,但一个完整的模型需要考虑的状态和情况非常多,难以考虑周全.于是为了简化,许多协议的设计都是简单地考虑单一协议执行的情况.当协议应用到其他环境中时,安全性就需要重新定义,从而带来了安全定义的指数爆炸性增长,给上层协议的安全性证明带来无法估量的难度.于是,研究者们迫切需要一种新的研究方法和路线来应付日益复杂的协议运行环境,比如 Internet.通用可组合安全框架(简称 UC 框架)[7,8]正是适合这个要求的一种协议分析框架.其最优秀的性质就是模块化设计思路:首先单独设计某个协议,只要该协议满足 UC 安全性,就保证了它与其他协议并行运行时的安全性.

UC 框架由 3 个模型搭建而成:现实模型、理想模型以及混合模型.现实模型描述实际协议π执行的情况.理想模型描述协议执行的理想情况,即通过理想功能(ideal functionality)$\mathcal{F}$ 描述希望达到的任务.混合模型则是将现实模型和理想模型进行混合.该模型中的协议可以访问理想功能.建立起这些模型之间的桥梁就是仿真.仿真可以将现实模型的安全规约到理想模型的安全.如果攻击者攻击现实模型下的协议π不比攻击理想模型下的 $\mathcal{F}$ 获得更大的影响或者更多的信息,那么π至少是和 $\mathcal{F}$ 一样安全的.

它与 Bellare 等人[9]于 1998 年提出的模式(modular)方法设计密钥交换协议有相近之处.但是,Bellare 等人的安全模型只是针对密钥交换协议所给出的,其安全在于攻击者与实际系统交互的结果和与理想系统交互的结果一样多.而 UC 框架则是针对任意环境、任意攻击给出协议的描述.UC 框架使用了“不可区分”的思想,通过仿真说明现实模型与理性模型是不可区分的.为了更好地描述仿真的结果,引入环境机 $\mathcal{Z}$ 代表协议运行的整个外部环境.它提供协议所有的输入,并读取协议所有的输出.

定义 1.2(协议π安全实现理想功能 $\mathcal{F}$). 如果对于现实模型中任意的攻击者 A 都存在一个理想模型中的攻击者 S,在任何输入情况下,现实模型中运行 $\mathcal{A}$ 和协议π的环境机 $\mathcal{Z}$ 的输出,与理想模型中运行 S 和理想功能 $\mathcal{F}$ 的环境机 $\mathcal{Z}$ 的输出都是不可区分的,则说协议π安全实现理想功能 $\mathcal{F}$.

定理 1.3(通用可组合定理). 如果协议ρ安全实现了理想功能 $\mathcal{F}$,且协议ρ和理想功能 $\mathcal{F}$ 都是π的子程序,则协议 $\pi^{\rho/\mathcal{F}}$ 安全仿真混合协议π.其中,π表示 $\mathcal{F}$-混合模型中的协议,$\pi^{\rho/\mathcal{F}}$ 则表示π中对 $\mathcal{F}$ 的调用都以对ρ的调用来替代.

1.3 双线性配对

群 G_1 和 G_2 分别表示阶是素数 $p=\Theta(2^k)$的乘法交换群.令群 G_1 和 G_2 上的离散对数问题都是难解的.双线性配对 e 是一个映射 $e=G_1\times G_2\to G_2$,满足 3 条性质:

(1) 双线性性.对于所有 $g,h\in G_1$, $a,b\in\mathbb{Z}_p^*$,都有 $e(g^a,h^b)=e(g,h)^{ab}$.

(2) 可计算性.对于所有 $g,h\in G_1$,存在多项式时间算法计算配对 $e(g,h)\in G_2$.

(3) 非退化性.g 是群 G_1 的生成元,$e(g,g)\neq 1$.

1.4 计算Diffie-Hellman假设问题

群 G 的阶是素数 $p=\Theta(2^k)$,g 是群 G 的生成元.CDH 问题:给定$\langle g,g^a,g^b\rangle$,其中, $a,b\in\mathbb{Z}_p^*$,计算 g^{ab}.

假设算法 $\mathcal{A}$ 能够解决 G 上的 CDH 问题,而 $\mathcal{A}$ 的成功概率可以表示为 $Adv_{\mathcal{A}}=\Pr[\mathcal{A}(g,g^a,g^b)\to g^{ab}]$.如果对于所有的算法都有 $Adv_{\mathcal{A}}$ 是可忽略的,则称 CDH 问题是困难的.

2 代理重签名的安全模型

这一节给出代理重签名方案的基于游戏的安全模型和基于通用可组合安全框架的安全模型,并证明这两个安全模型的等价性.如果方案符合常用的基于游戏的安全定义,那么该方案同时也满足通用可组合安全性.

2.1 基于游戏的安全模型

在吸收了以往工作成果[1,2,5]的基础上,本文定义了代理重签名的基于游戏的安全模型.它对以前的工作进行了一些改进,去除了一些限制,简化了一些描述.首先允许攻击者能够控制在攻破用户和未被攻破用户之间的代理人,其次确保授权人和受理人秘密信息的安全性.本文不仅考虑了外部用户不能得到方案的任何信息,而且

考虑了在代理人和授权人(受理人)都被控制的情况下,如何保证受理人(授权人)的安全.另外,本文采用的仍然是静态攻击的模型,即在这个安全模型中,攻击者必须在游戏开始前就决定需要攻破哪些用户.代理重签名的基于游戏的安全模型是从正确性、一致性和不可伪造性[10]加以说明.

- 正确性.代理重签名方案若满足正确性,就需要满足下面两个条件:首先,对于任意一个在消息空间里的消息 m 和任何从算法 $KeyGen(1^k)$产生的公私钥对(pk,sk),计算$\sigma\leftarrow Sign(m,sk)$,方案必须满足条件 $Verify(pk,m,\sigma)=1$;其次,对于任意一个消息空间内的消息 m,任何从算法 $KeyGen(1^k)$产生的公私钥对(pk_a,sk_a),(pk_b,sk_b)和再签名密钥 $rk_{a\leftrightarrow b}\leftarrow ReKey(sk_a,sk_b)$,若 A 的签名是$\sigma\leftarrow Sign(m,sk_a)$,则方案还必须满足条件 $Verify(pk_b,m,ReSign(rk_{a\leftrightarrow b},m,\sigma))=1$.
- 一致性.如果代理重签名方案满足一致性,那么对于任何在消息空间里的消息 m、公钥 pk 和签名σ,两次调用验证算法 $Verify(pk,m,\sigma)$必定得到相同的结果.
- 不可伪造性.通过挑战者和攻击者之间的游戏来定义代理重签名的不可伪造性.

定义 2.1(PRS CMA game). 游戏由攻击者 $\mathcal{A}$ 查询一系列的随机预言机所组成:

(1) 查询

用户密钥预言机 $O_{CkeyGen}$.攻击者 $\mathcal{A}$ 输入查询某个用户的公钥 pk,且该公钥是由密钥生成算法 $KeyGen$ 合法生成的公钥,则用户密钥预言机 $O_{CKeyGen}$ 返回攻击者 $\mathcal{A}$ 公钥 pk 相应的私钥 sk.

再签名密钥预言机 O_{ReKey}.攻击者 $\mathcal{A}$ 输入查询两个用户的公钥 pk_a,pk_b,其中,pk_a,pk_b 是由密钥生成算法 $KeyGen$ 合法生成的公钥.再签名密钥预言机 O_{ReKey} 返回攻击者 $\mathcal{A}$ 再签名密钥 $rk_{a\leftrightarrow b}\leftarrow ReKey(sk_a,sk_b)$,其中,$sk_a,sk_b$ 是对应于 pk_a,pk_b 的密钥.

签名预言机 O_{Sign}.攻击者 $\mathcal{A}$ 输入查询(pk,m),其中,pk 是由算法 $KeyGen$ 合法生成的公钥,m 是消息空间中任意的消息.签名预言机 O_{Sign} 返回攻击者 $\mathcal{A}$ 签名$\sigma\leftarrow Sign(sk,m)$,其中,$\sigma$可以用对应于 sk 的公钥 pk 验证.

再签名预言机 O_{ReSign}:攻击者 $\mathcal{A}$ 输入查询(pk_a,pk_b,m,σ),其中,pk_a,pk_b 是由算法 $KeyGen$ 合法生成的公钥,σ是消息 m 对应于 pk_a 的签名.再签名预言机 O_{ReSign} 返回 $\mathcal{A}$ 再签名$\sigma'\leftarrow ReSign(ReKey(sk_a,sk_b),m,\sigma)$,其中,$sk_a,sk_b$ 是对应于 pk_a,pk_b 的密钥.

(2) 伪造

原始签名安全性.攻击者 $\mathcal{A}$ 最后输出(pk^*,m^*,σ^*),如果满足下面的条件,就称攻击者在 PRS-CMA 游戏后成功地得到伪造的原始签名:

- σ^*是可以用公钥 pk^*验证的消息 m^*的有效的原始签名;
- pk^*不是密钥预言机 $O_{CKeyGen}$ 的查询;
- (pk^*,m^*)不是签名预言机 O_{Sign} 的查询.

再签名安全性.攻击者 $\mathcal{A}$ 最后输出(pk^*,m^*,σ^*),如果满足下面的条件就称攻击者在 PRS-CMA 游戏后成功地得到伪造的再签名:

- σ^*是可以用公钥 pk^*验证的消息 m^*有效的再签名;
- pk^*不是密钥预言机 $O_{CKeyGen}$ 的查询;
- (pk^*,m^*)不是签名预言机 O_{Sign} 的查询;
- $(\diamond,pk^*)$不是再签名密钥预言机 O_{ReKey} 的查询,其中,$\diamond$代表任何用户的公钥;
- $(\diamond,pk^*,m^*,\blacklozenge)$不是再签名预言机 O_{ReSign} 的查询,其中,$\blacklozenge$代表任意的签名.

攻击者 $\mathcal{A}$ 在游戏后得到一个伪造的原始签名或者再签名的优势可以定义为 $Adv_{\mathcal{A}}=\Pr[\mathcal{A}\ \text{succeeds}]$,这里,概率的大小完全取决于挑战者和攻击者之间的抛币概率.本文称一个代理重签名方案是选择消息攻击下存在性不可伪造的,那么对于任意的攻击者 $\mathcal{A}$,这个游戏中 $Adv_{\mathcal{A}}$ 都是可以忽略的.

2.2 基于通用可组合安全框架的安全模型

为了描述通用可组合安全性,本文定义了代理重签名基于通用可组合安全框架的安全模型.这个安全模型的主要任务是定义一个恰当的代理重签名理想功能 $\mathcal{F}_{PRS}$.Canetti 和 Hohenberger[11]给出了代理重加密的理想功

能,它与代理重签名类似,一个拥有一些额外信息的半可信代理人可以把用户 A 的公钥加密的密文转换为用户 B 的公钥加密的密文.本文仿照 Canetti 和 Hohenberger 的思路,给出代理重签名的理想功能 $\mathcal{F}_{PRS}$ 如下:

- **密钥生成接口**.接收到用户 P 的输入(*KeyGen*,*sid*)后,验证 *sid* 是否等于(P,*sid*′),$\mathcal{F}_{PRS}$ 把(*KeyGen*,*sid*)输入给攻击者 $\mathcal{A}$.$\mathcal{F}_{PRS}$ 并接收到攻击者 $\mathcal{A}$ 的输出(*Algorithms*,*sid*,s_P,v_P)和一个表示"corruption"的比特后(其中 s_P 是一个非确定性多项式时间算法的描述,v_P 是一个确定性多项式时间算法的描述),记录(P,s_P,v_P)并输出(*VerificationAlgorithm*,*sid*,v_P)给用户 P.如果设置了"corrupted"比特,则 $\mathcal{F}_{PRS}$ 记录用户 P 的状态"corrupted".
- **再签名密钥生成接口**.接收到用户 P 和 P'的输入(*ReKey*,*sid*,P,P',X),并验证 *sid* 的有效性后,$\mathcal{F}_{PRS}$ 把(*ReKey*,*sid*,P,P',X)输出给攻击者 $\mathcal{A}$.$\mathcal{F}_{PRS}$ 接受到攻击者 $\mathcal{A}$ 的输出(*ResignAlgorithm*,*sid*,$s_{P,P'}$)和一个表示"corruption"的比特,其中,$s_{P,P'}$是一个多项式时间算法的描述.记录(P,P',X,$s_{P,P'}$),并输出(*Proxy*,*sid*,P,P')给代理人 X.如果设置了"corrupted"比特,则 $\mathcal{F}_{PRS}$ 记录代理人 X 的状态"corrupted".
- **签名接口**.接收到用户 S 的输入(*Sign*,*sid*,m),$\mathcal{F}_{PRS}$ 验证 *sid* 的有效性后,计算σ=$s_P(m)$,并记录(m,σ,v_P).输出(*Signature*,*sid*,m,σ)给用户 S.
- **再签名接口**.接收到代理人 X 的输入(*ReSign*,*sid*,P,P',m,σ),$\mathcal{F}_{PRS}$ 验证 *sid* 的有效性后,判断是否存在记录(P,P',X,$s_{P,P'}$):如果不存在,则输出⊥;否则,计算σ'=$s_{P,P'}(m,\sigma)$,记录(m,σ',v_P),并输出(*ReSignature*,*sid*,m,σ')给代理人 X.
- **签名验证接口**.接收到用户 V 的输入(*Verify*,*sid*,m,σ,v_P^*)后,$\mathcal{F}_{PRS}$ 输出(*Verified*,*sid*,m,σ,f)给用户 V.其中 f 按照如下条件计算:
  - 如果用户 P 的状态是"corrupted",则令 f=1(确保被控制的用户能够"注册"任意的原始签名或者再签名);
  - 如果 $v_P^* \neq v_P$,则令 f=0(确保当给定验证算法不合法时,所有的验证请求结果都是不合法的.$\mathcal{F}_{PRS}$ 只是记录签名和验证算法的关联);
  - 如果 $v_P^* = v_P$ 且存在记录(m,σ,v_P),则令 f=1(确保合法用户合法产生的签名一定可以通过验证请求);
  - 如果 $v_P^* = v_P$ 但不存在记录(m,σ,v_P),则令 f=0(确保合法用户非法产生的签名(即伪造签名)一定不通过验证请求).

代理重签名理想功能 $\mathcal{F}_{PRS}$ 的基本思路是提供"注册机"的功能.签名者可以注册(消息、原始签名)对,代理人可以注册(消息、再签名)对,而任何参与者都能检验某个签名是否已经被注册过.上面给出了代理重签名理想功能 $\mathcal{F}_{PRS}$ 的详细描述.它与签名理想功能 $\mathcal{F}_{SIG}$ 类似.与 $\mathcal{F}_{SIG}$ 不同,$\mathcal{F}_{PRS}$ 允许它的一个实例处理多个用户的密钥和多个代理人的密钥.$\mathcal{F}_{PRS}$ 与 $\mathcal{F}_{SIG}$ 相比还增加了两个模块:再签名密钥生成接口和再签名接口.代理重签名理想功能 $\mathcal{F}_{PRS}$ 共有 5 类输入,这些输入又分别对应 5 个基本模块:密钥生成接口、再签名密钥生成接口、签名接口、再签名接口和签名验证接口.

代理重签名理想功能 $\mathcal{F}_{PRS}$ 的密钥生成接口和再签名密钥生成接口都是由攻击者 $\mathcal{A}$ 决定签名算法、验证算法和再签名算法.同时,攻击者 $\mathcal{A}$ 还可以决定被攻破的用户和代理人的身份.但是,攻击者 $\mathcal{A}$ 需要在使用这些接口之前确定这些被控制的用户和代理人的身份.这就限制了整个模型是静态攻击模型,即在签名接口和再签名接口使用之前已经建立起所有的签名密钥和再签名密钥.

2.3 两种安全模型的等价性

为了证明代理重签名这两种安全模型之间的等价性,首先需要说明如何把代理再签名方案Σ转换成相应的代理再签名协议π_Σ.令Σ=(*KeyGen*,*ReKey*,*Sign*,*ReSign*,*Verify*)为代理再签名方案,协议π_Σ由下面的步骤得到:

1. 接受到输入(*KeyGen*,*sid*)后,用户 P 调用密钥生成算法计算(pk,sk)←*KeyGen*(1^k).算法 $s_P(\cdot)$是签名算法 *Sign*(sk,·).算法 $v_P(\cdot)$是验证算法 *Verify*(pk,·).用户 P 输出(*VerificationAlgorithm*,*sid*,v_P).如果 P 从攻击者 $\mathcal{A}$ 处得到一个"corruption"请求,则把签名算法 $s_P(\cdot)$输出给攻击者 $\mathcal{A}$.
2. 接受到输入(*ReKey*,*sid*,P,P',X)后,代理人 X 调用再签名密钥生成算法计算 $rk_{a\leftrightarrow b}$←*ReKey*(sk_a,sk_b).算法 $s_{P,P'}(\cdot)$是再签名算法 *ReSign*($rk_{a\leftrightarrow b}$).代理人 X 输出(*Proxy*,*sid*,P,P').如果 X 从攻击者 $\mathcal{A}$ 处得到一个

"corrupted"请求,则把再签名算法 $s_{P,P'}(\cdot)$输出给攻击者 $\mathcal{A}$.

3. 接收到输入(*Sign*,*sid*,*m*)后,用户 *S* 调用签名算法计算$\sigma=s_P(m)$,并输出(*Signature*,*sid*,*m*,σ).
4. 接收到输入(*ReSign*,*sid*,*P*,*P'*,*m*,σ)后,代理人 *X* 计算$\sigma'=s_{P,P'}(m,\sigma)$,并输出(*ReSignature*,*sid*,*m*,σ').
5. 接收到输入(*Verify*,*sid*,*m*,σ,v_P)后,验证者 *V* 计算 $f=Verify(pk,m,\sigma)$,并输出(*Verified*,*sid*,*m*,σ,f).

一个代理重签名方案满足基于游戏的安全模型,等价于相应的代理重签名协议满足基于通用可组合安全框架的安全模型.

定理 2.2. 令Σ=(*KeyGen*,*ReKey*,*Sign*,*ReSign*,*Verify*)是代理重签名方案,则相应的协议π_Σ安全地实现了理想功能 $\mathcal{F}_{PRS}$,当且仅当方案Σ满足正确性、一致性、不可伪造性.

证明:首先证明充分性,即需要证明若$\pi=\pi_\Sigma$安全实现了理想功能 $\mathcal{F}_{PRS}$,则Σ是正确的、一致的和不可伪造的.假设当Σ不具有正确性、一致性或不可伪造性时,构造环境机 $\mathcal{Z}$ 和实际攻击者 $\mathcal{A}$,对于任何理想攻击者 $\mathcal{S}$,环境机 $\mathcal{Z}$ 都能分辨它是与运行 $\mathcal{A}$ 和协议π交互,还是与运行 $\mathcal{S}$ 和理想功能 $\mathcal{F}_{PRS}$ 交互.

(1) 假设π不具有正确性,即存在某个消息 *m*,对于公私钥对(*pk*,*sk*)和签名σ,协议π得到验证结果 $Verify(pk,m,\sigma)=0$;或者,对于公私钥对(pk_a,sk_a),(pk_b,sk_b),再签名密钥 $rk_{a\leftrightarrow b}$ 和用户 *A* 的签名σ,协议π验证 $Verify(pk_b,m,ReSign(rk_{a\leftrightarrow b},m,\sigma))=0$.环境机 $\mathcal{Z}$ 首先设 *sid*=(*S*,0),并用请求(*KeyGen*,*sid*)激活 *S*,然后请求对 *m* 签名或再签名操作,最后验证所得原始签名或再签名.因为π不具有正确性,所以当 $\mathcal{Z}$ 与π交互后,$\mathcal{Z}$ 返回的结果是 0.而在理想过程中,$\mathcal{Z}$ 的验证结果一定是 1.所以,当π不具有正确性时,环境机 $\mathcal{Z}$ 一定可以分辨它是与 $\mathcal{A}$ 和协议π交互,还是与 $\mathcal{S}$ 和理想功能 $\mathcal{F}_{PRS}$ 交互.

(2) 假设π不具有一致性,即两次调用签名验证算法 $Verify(pk,m,\sigma)$得到不同的结果.这样,环境机 $\mathcal{Z}$ 用(*pk*,*m*,σ)两次调用 *V*,$\mathcal{Z}$ 的结果就是判断两次验证结果是否一致.因为π不满足一致性,所以 $\mathcal{Z}$ 的结果是 0.而在理想过程中,$\mathcal{Z}$ 的结果是 1.所以,当π不具有正确性时,$\mathcal{Z}$ 一定可以分辨它是与π还是 $\mathcal{F}_{PRS}$ 交互.

(3) 假设π不具有不可伪造性,即存在π的一个成功的伪造者 $\mathcal{A}$.环境机 $\mathcal{Z}$ 首先就激活 $\mathcal{A}$,并如下执行:

- 如果 $\mathcal{A}$ 希望查询 $O_{CKeyGen}$,则 *Z* 调用新的用户并令其是"corrupted",最后返回给 $\mathcal{A}$ 得到的密钥;
- 如果 $\mathcal{A}$ 希望查询 O_{ReKey},则 *Z* 调用两个相关的用户,令其把签名权授权给"corrupted"代理人 *X*,最后返回给 $\mathcal{A}$ 得到的再签名密钥;
- 如果 $\mathcal{A}$ 希望查询 O_{Sign},则 $\mathcal{Z}$ 调用相应的用户,用同一个签名请求其签名,最后返回给 $\mathcal{A}$ 得到的签名;
- 如果 $\mathcal{A}$ 希望查询 O_{ReSign},则 $\mathcal{Z}$ 调用相应的代理人,用同一个再签名请求再签名,最后返回给 $\mathcal{A}$ 得到的结果;
- 若 $\mathcal{A}$ 最后提供某个原始签名或再签名,则环境机 $\mathcal{Z}$ 的结果是"real";否则,$\mathcal{Z}$ 的结果是"ideal".

分析环境机 $\mathcal{Z}$ 的整个过程,在 $\mathcal{Z}$ 与协议π交互时,$\mathcal{Z}$ 调用的攻击者 $\mathcal{A}$ 应该得到协议π所需的所有信息.因为 $\mathcal{A}$ 是π的成功的伪造者,所以 $\mathcal{A}$ 能够以不可忽略的概念ε得到伪造的原始签名或再签名.因此,$\mathcal{Z}$ 就能以同样的概率ε输出"real".而当 $\mathcal{Z}$ 与 $\mathcal{F}_{PRS}$ 交互后,$\mathcal{Z}$ 的输出结果一定是"ideal".所以,$\mathcal{Z}$ 可以分辨它是与实际协议π还是与理想功能 $\mathcal{F}_{PRS}$ 交互.

接来下证明充分性,即证明若π是正确的、一致的和不可伪造的,则π安全实现了 $\mathcal{F}_{PRS}$.假设π不能安全实现 $\mathcal{F}_{PRS}$,即存在环境机 $\mathcal{Z}$ 能够分辨它是与运行 $\mathcal{D}$(dummy adversary,完全按照 $\mathcal{Z}$ 的指令运行,只传递消息)和协议π交互,还是与 $\mathcal{S}$ 和理想功能 $\mathcal{F}_{PRS}$ 交互,其中,由攻击者 $\mathcal{S}$ 提供签名算法和验证算法.即使π具有正确性和一致性,也可以调用 $\mathcal{Z}$ 构造一个成功的针对π的攻击者 $\mathcal{A}$.与攻击者 $\mathcal{A}$ 的交互过程如下所述:

- 当 $\mathcal{Z}$ 用输入(*KeyGen*,*sid*)激活某个"corrupted"的 *P*,则 $\mathcal{A}$ 查询 $O_{CKeyGen}$,将结果(*sk*,*pk*)返回给 $\mathcal{Z}$.
- 当 $\mathcal{Z}$ 用输入(*ReKeyGen*,*sid*,*P*,*P'*,*X*)激活两个用户 *P* 和 *P'*,则 $\mathcal{A}$ 查询 O_{ReKey},将结果 $rk_{P\to P'}$返回给代理人 *X*.若代理人 *X* 的状态是"corrupted",则同时返回 $rk_{P\to P'}$给 $\mathcal{Z}$.
- 当 $\mathcal{Z}$ 用输入(*Sign*,*sid*,*m*)激活用户 *S*,则 $\mathcal{A}$ 查询 O_{Sign} 并得到σ,返回结果(*Signature*,*sid*,*m*,σ)给 *S*.
- 当 $\mathcal{Z}$ 用输入(*ReSign*,*sid*,*P*,*P'*,*m*,σ)激活用户 *S*,则 $\mathcal{A}$ 查询 O_{ReSign},将得到的再签名σ'返回给 *S*.
- 当 $\mathcal{Z}$ 用输入(*Verify*,*sid*,*m*,σ,v_P^*)激活验证者 *V*,则 $\mathcal{A}$ 令 $f=Verify(pk,m,\sigma)$,输出(*Verified*,*sid*,*m*,σ,f)给 *V*.

接下来分析攻击者 $\mathcal{A}$ 成功的概率,即分析环境机 $\mathcal{Z}$ 能够成功地分辨出它是与协议π交互,还是与理想功能

$\mathcal{F}_{PRS}$交互的概率.因为已经假设方案Σ是正确的和一致的,所以最后得到(m,σ,v_P^*),用这个签名查询验证请求时结果是 1.而在理想过程下,用这个签名查询验证请求时结果则是 0.这样,最后必定得到某个伪造的原始签名或再签名.因为环境机 $\mathcal{Z}$ 能以某个不可忽略的概率分辨它是与协议π交互,还是与理想功能 $\mathcal{F}_{PRS}$ 交互,所以构造的攻击者 $\mathcal{A}$ 能够以同样的概率ε赢得 PRS-CMA 游戏. □

3 UC 安全的代理重签名

3.1 具体方案

本文的双向代理重签名方案是在两个阶为 $p=\Theta(2^k)$的群 G_1 和 G_2 上执行.公开参数是(e,p,G_1,G_2,g,h,H),其中,g 和 h 是群 G_1 的两个生成元,$e:G_1\times G_2\to G_2$ 是双线性配对,而 $H(\cdot)$则是一个安全的哈希算法,能够把任意的字符串映射到$\mathbb{Z}_p$中的元素.下面是方案π_{BPRS}的详细描述:

- **密钥的生成**.输入安全参数 1^k 后,随机选择 $a\in\mathbb{Z}_p^*$,输出用户的一对公私钥 $pk=g^a \bmod p$ 和 $sk=a$.
- **再签名密钥的生成**.输入用户 A 和 B 的秘密信息 $h^{sk_a}=h^a$ 和 $h^{sk_b}=h^b$,输出用户 A 和 B 之间的再签名密钥 $rk_{a\leftrightarrow b}=h^a/h^b \bmod p$.
- **签名**.输入私钥 $sk=a$ 和消息 m,随机选择 $r\in\mathbb{Z}_p^*$,并计算

$$R=g^r \bmod p,$$
$$\delta=a+H(m\|R)\cdot r \bmod p-1,$$

最后输出原始签名$\sigma=(\delta,R)$.

- **再签名**.输入再签名密钥 $rk_{a\leftrightarrow b}$,消息 m,用户 A 的公钥 pk_a 和用户 A 的原始签名$\sigma=(\delta,R)$.首先检查σ是否是用户 A 对消息 m 的签名,即判断等式 $Verify(pk_a,m,\sigma)=1$ 是否成立.如果等式不成立,则输出⊥;否则,随机选取 $\Delta r\in\mathbb{Z}_p^*$,并计算

$$\delta'=rk_{a\leftrightarrow b}\cdot h^{\delta}\cdot h^{\Delta r},$$
$$R'=R,$$
$$K=g^{\Delta r} \bmod p,$$

最后输出再签名$\sigma'=(\delta',R',K)$.

- **验证**.输入公钥 pk、消息 m 以及签名σ,令$\omega=H(m\|R)$,然后按照如下条件来判断:
  - 如果σ是原始签名,即签名的格式是$\sigma=(\delta,R)$,则验证 $g^{\delta}\equiv pk\cdot R^{\omega} \bmod p$ 是否成立.如果等式成立,则输出 1;否则,输出 0.
  - 如果σ是再签名,即签名的格式是$\sigma=(\delta,R,K)$,则验证 $e(\delta,g)=e(h,pk)\cdot e(h,R)^{\omega}\cdot e(h,K)$是否成立.如果等式成立,则输出 1;否则,输出 0.

容易看出,方案π_{BPRS}具有双向性、多用性和可传递的.而且因为每个用户只需存储他自己的秘密信息,所以不需要另外的存储空间,方案π_{BPRS}也是密钥最优化的.方案π_{BPRS}是在 ElGamal 族三元签名方案的基础上增加再签名密钥生成算法和再签名算法变形而得到的.相对于其他方案,π_{BPRS} 更容易应用到现有的体制中.而且方案π_{BPRS} 更为简单、有效.在其签名算法和再签名算法中,用户只需进行群 G_1 上的模指数运算和模乘运算.只有当需要验证再签名时,验证者才需要用到配对运算.与第一个可证明安全的方案 S_{mb} 相比(需要用到大量的公开参数),方案π_{BPRS}只需要很少的公开参数.

3.2 安全性证明

这一节将证明方案π_{BPRS} 满足正确性、一致性和选择消息攻击下的不可伪造性.而且根据这两个安全模型之间的等价性(定理 2.2),方案π_{BPRS}满足通用可组合安全性.

定理 3.1. 如果计算 Diffie-Hellman 问题假设在群 G_1 中成立,则方案π_{BPRS}在随机预言机模型下是正确的、一致的和不可伪造的.

证明:方案的正确性和一致性很容易证明.首先看正确性.对于任何消息 m 和算法 $KeyGen(1^k)$产生的公私钥对(pk,sk),如果原始签名$\sigma=Sign(m,sk)=(\delta,R)$,其中,$\delta=a+H(m\|R)\cdot r \bmod p-1$,那么一定有 $g^{\delta}\equiv pk\cdot R^{H(m\|R)} \bmod p$,即 $Verify(pk,m,\sigma)=1$.而对于任何消息空间中的消息 m,公私钥$(pk_a,sk_a),(pk_b,sk_b)$,再签名密钥 $rk_{a\leftrightarrow b}$,用户 A 的签名σ和用户 B 的再签名δ',一定有 $e(\delta',g)=e(h,pk_b)\cdot e(h,R)^{H(m\|R)}\cdot e(h,K)$,即 $Verify(pkb,m,ReSign(rk_{a\leftrightarrow b},m,\sigma))=1$.因此,方案$\pi_{BPRS}$满足正确性.

再看一致性.对于任何消息空间中的消息 m、公钥 pk 和任何签名σ,两次调用 $Verify(pk,m,\sigma)$就是根据是原始签名还是再签名判断 $g^{\delta}\equiv pk\cdot R^{H(m\|R)} \bmod p$,或 $e(\delta,g)=e(h,pk)\cdot e(h,R)^{H(m\|R)}\cdot e(h,K)$.因为条件不变,所以两次的结果一定是相同的.因此,方案π_{BPRS}满足一致性.

接下来看不可伪造性.这个安全属性通过规约来证明.假如存在一个攻击者 A 能够以一个不可忽略的概率ε在时间 t 内破解方案π_{BPRS},而且攻击者 A 能够查询最多 Q_S 次签名预言机 O_{Sign}、Q_{RS} 次再签名预言机 O_{ReSign}、Q_K 次用户密钥预言机 $O_{CKeyGen}$ 和 Q_{RK} 次再签名密钥预言机 O_{ReKey} 以及 Q_H 次哈希查询预言机 O_{Hash},那么可以构造一个攻击者 B,能够以概率 $1/\sqrt{Q_H}$ 在时间 $t'=(3(t+Q_H)+O_B(Q_s\cdot k^3))/\varepsilon$ 内解决群 G_1 上的 CDH 假设问题.

希望解决的 CDH 假设是给定输入(g,g^a,g^b),攻击者 $\mathcal{B}$ 的目的就是求出 g^{ab}.攻击者 $\mathcal{B}$ 先建立了给攻击者 $\mathcal{A}$ 的公开参数:安全参数 $k\geqslant|p|$、群 $G_1=\langle g\rangle$和群 G_2(这两个群的阶是 p)及双线性配对 $e:G_1\times G_2\to G_2$.令 $h=g^b$,系统参数是(e,p,G_1,G_2,g,h,H),其中,$H(\cdot)$是一个随机预言机.

(1) 查询.$\mathcal{B}$ 向 $\mathcal{A}$ 建立了如下的随机预言机:

- O_{Hash}:接受到 $\mathcal{A}$ 查询输入是(m,R)时,攻击者 $\mathcal{B}$ 检查(m,R)是否存储在哈希的数据库 $\mathcal{D}_H$ 中.如果存在这条记录,则只需要直接回答存储的记录即可;如果没有这条记录,则攻击者 $\mathcal{B}$ 随机选取$\omega\in\mathbb{Z}_p$并在数据库 $\mathcal{D}_H$ 中记录(m,R,ω), $\mathcal{B}$ 把ω作为对 $\mathcal{A}$ 的回答.
- $O_{CKeyGen}$:接收到 $\mathcal{A}$ 查询用户 P_i 的密钥查询后,攻击者 $\mathcal{B}$ 随机选取 $x_i\in\mathbb{Z}_p$,并输出 $(pk_i,sk_i)=(g^{x_i},x_i)$.
- O_{ReKey}:接收到 $\mathcal{A}$ 查询输入(pk_i,pk_j)后,攻击者 $\mathcal{B}$ 返回再签名密钥 $rk_{i\leftrightarrow j}=h^{x_i}/h^{x_j} \bmod p$.
- O_{Sign}:接收到 $\mathcal{A}$ 查询输入(pk,m)后,如果 pk 的状态是“corrupted”,那么攻击者 $\mathcal{B}$ 返回直接用签名算法计算的消息 m 的签名结果$\sigma=(\delta,R)$,并随机选取 $r\in Z_p^*$,分别计算:

$$R=g^r \bmod p,$$

$$\delta=a+H(m\|R)\cdot r \bmod p-1;$$

否则,攻击者 $\mathcal{B}$ 伪造签名,随机选取 u,v,令

$$R=g^u\cdot pk^v \bmod p,$$

$$\delta=-u/v \bmod p-1,$$

$$H(m\|R)=-1/v \bmod p-1,$$

把$\omega=H(m\|R)=-1/v$ 作为哈希查询(m,R)的回答记录到哈希数据库 $\mathcal{D}_H$ 中.这样,$\sigma=(\delta,R)$就与实际方案中的签名具有相同的形式,是对 m 的一个有效的原始签名.

- O_{ReSign}:接受到 $\mathcal{A}$ 查询输入(pk_i,pk_j,m,σ)后,首先验证 $Verify(pk_i,m,\sigma)=1$,攻击者 $\mathcal{B}$ 调用 $ReSign(O_{ReKey}(pk_i,pk_j),pk_i,m,\sigma)$计算出再签名$\sigma'$.最后输出 $\mathcal{A}$ 的再签名的结果.如果过程中失败,则输出$\perp$.

(2) 伪造.如果 $\mathcal{B}$ 在一系列的查询之后也不出错,则 $\mathcal{A}$ 能以一个不可忽略的概率ε返回消息 m^*的某个签名σ^*.利用分叉引理可以得到本定理的结论.运行 $1/\varepsilon$次 $\mathcal{A}$,因为 $\mathcal{A}$ 是一个成功的伪造者,所以只要满足这个条件,$\mathcal{B}$ 就能以概率 1 输出消息 m 的一个有效签名.然后,$\mathcal{B}$ 在相同条件下再运行另外两次 $1/\varepsilon$攻击者 $\mathcal{A}$,但是这两次中,$\mathcal{B}$ 重排他对 $\mathcal{A}$ 的 Q_H 个均匀随机哈希查询内容,那么一定得到两个伪造的原始签名,或者两个伪造的再签名.

如果是得到两个伪造的原始签名,那么当两个伪造的原始签名$(m,(\delta,R,e))$和$(m',(\delta',R',e'))$满足条件$(m,R)=(m',R')$时,“成功分叉 $\mathcal{B}$ 的 RO 提问”这个事件就会发生.由生日悖论,这个事件发生的概率大约是 $1/\sqrt{Q_H}$.对于这两个有效的原始签名,$\mathcal{B}$ 可以计算:

$$\delta=a+e\cdot r \bmod p,$$

$$\delta'=a+e'\cdot r' \bmod p.$$

由 $e\neq e'$,推出$\delta\neq\delta'$,所以有

$$a=\frac{e'\cdot\delta-e\cdot\delta'}{e'-e}\bmod p-1.$$

根据这个结论可以计算解决 CDH 假设问题:

$$g^{ab}=(g^b)^{e'\cdot\delta-e\cdot\delta'/e'-e} \bmod p.$$

如果是得到两个伪造的再签名,则当两个伪造的再签名$(m,(\delta,R,K,e))$和$(m',(\delta',R',K',e'))$满足条件$(m,R,K)=(m',R',K')$时,“成功分叉 $\mathcal{B}$ 的 RO 提问”这个事件就会发生.由生日悖论,这个事件发生的概率大约是$1/\sqrt{Q_H}$. 对于这两个有效的原始签名,$\mathcal{B}$ 可以计算:

$$\delta=h^{a+e\cdot r}\cdot K,$$
$$\delta'=h^{a+e'\cdot r}\cdot K'.$$

由 $e\neq e'$,推出$\delta\neq\delta'$,所以有

$$h^a=\left(\frac{\delta^{e'}}{\delta'^{e}}\right)^{e'-e}.$$

因为 $h=g^b$,所以可以解决 CDH 假设问题:

$$g^{ab}=\left(\frac{\delta^{e'}}{\delta'^{e}}\right)^{e'-e} \bmod p.$$

分析攻击者 $\mathcal{B}$ 如何成功完成整个模拟的过程可知,攻击者 $\mathcal{B}$ 成功的概率是 $\Pr[\mathcal{B}\ \text{succeeds}]=1/\sqrt{Q_H}$, 而攻击者 $\mathcal{B}$ 所需的时间是

$$t'=\frac{3(t+Q_H)+O_B(Q_H\cdot k^3)}{\varepsilon}.$$

因此,该定理得证. □

下面的推论可以直接由定理 2.2 和定理 3.1 推导出.

推论 3.2. 如果计算 Diffie-Hellman 假设问题在群 G_1 上成立,则方案π_{BPRS}在随机预言机模型下安全地实现了 $\mathcal{F}_{PRS}$,满足了 UC 安全性.

4 结 论

本文给出了代理重签名的形式化定义和两个等价的安全模型:一个是基于通用可组合安全框架的安全模型,另一个是基于游戏的安全模型.这样就保证了如果方案满足基于游戏的安全性定义,那么方案同时还满足通用可组合安全性.然后,本文还提出了满足 UC 安全的双向代理重签名方案π_{BPRS}.方案π_{BPRS}被证明是正确的、一致的、不可伪造的和通用可组合安全的,其安全性可以规约到随机预言机模型下的计算 Diffie-Hellman 假设.本文给出代理重签名基于UC框架的安全模型,为以后研究UC安全的代理重签名协议铺平了道路.与之前的文献一样,本文考虑的仍然是静态攻击模型,即攻击者必须在一开始就确定这些被控制的用户和代理人的身份.我们以后的工作致力于研究代理重签名动态攻击模型下的安全模型.

致谢 在此,我们向对本文的工作给予支持和建议的上海交通大学密码与信息安全实验室的各位老师和同学表示感谢.

References:

[1] Blaze M, Bleumer G, Strauss M. Divertible protocols and atomic proxy cryptography. In: Proc. of the EUROCRYPT'98. LNCS 1403, Berlin, Heidelberg: Springer-Verlag, 1998. 127−144.

[2] Ateniese G, Hohenberger S. Proxy re-signatures: New definitions, algorithms, and applications. In: Proc. of the ACM CCS 2005. Berlin, Heidelberg: Springer-Verlag, 2005. 310−319.

[3] Ateniese G, Fu K, Green M, Hohenberger S. Improved proxy re-encryption schemes with applications to secure distributed storage. In: Proc. of the 12th Annual Network and Distributed System Security Symp. (NDSS 2005). Berlin, Heidelberg: Springer-Verlag, 2005. 29−43.

[4] Taban G, Cárdenas AA, Gligor VD. Towards a secure and interoperable DRM architecture. In: Proc. of the ACM DRM 2006. Berlin, Heidelberg: Springer-Verlag, 2006. 69−78.

[5] Shao J, Chao ZF, Wang LC, Liang XH. Proxy re-signature schemes without random oracles. In: Proc. of the Indocrypt 2007. LNCS 4859, Berlin, Heidelberg: Springer-Verlag, 2007. 197−209.

[6] Ivan A, Dodis Y. Proxy cryptography revisited. In: Proc. of the 10th Annual Network and Distributed System Security Symp. (NDSS 2003). Berlin, Heidelberg: Springer-Verlag, 2003. 196−204.

[7] Canetti R. Universally composable security: A new paradigm for cryptographic protocols. In: Proc. of the IEEE Symp. on Foundations of Computer Science. Berlin, Heidelberg: Springer-Verlag, 2001. 136−145.

[8] Canetti R. Universally composable signature, certification, and authentication. In: Proc. of the 17th Computer Security Foundations Workshop (CSFW 2004). Berlin, Heidelberg: Springer-Verlag, 2004. 219−233.

[9] Bellare M, Canetti R, Krawczyk H. A modular approach to the design and analysis of authentication and key exchange protocols. In: Proc. of the 30th Annual ACM Symp. on the Theory of Computing. ACM Press, 1998. 419−428.

[10] Goldwasser S, Micali S, Rivest RL. A digital signature scheme secure against adaptive chosen message attacks. SIAM Journal of Computing, 1998,17:281−308. [doi: 10.1137/0217017]

[11] Canetti R, Hohenberger S. Chosen-Ciphertext secure proxy re-encryption. In: Proc. of the ACM CCS 2007. ACM Press, 2007. 185−194.

洪璇(1982－),女,江西宜黄人,博士,讲师,主要研究领域为密码算法与可证明安全,通用可组合安全框架.

陈克非(1959－),男,博士,教授,博士生导师,主要研究领域为密码算法,可证明安全性,数字水印及应用技术.

万中美(1973－),女,博士生,讲师,主要研究领域为信息安全技术.