doi:10.3969/j.issn.1000-436x.2012.z1.009

可证安全的无证书盲代理重签名

冯涛，梁一鑫
(兰州理工大学 计算机与通信学院，甘肃 兰州 730050)

摘 要：利用双线性群，在代理重签名机制和盲签名机制的基础上，提出了一个有效的无证书盲代理重签名方案。方案中解决了密钥托管问题及证书管理带来的额外开销，同时实现了代理者在签名转换中消息隐私特性。基于 NGBDH 问题和 Many-NGBDH 的困难性，证明了新方案具有能够抵抗伪造攻击的特性。该方案满足正确性和消息盲性。

关键词：无证书密码系统；代理重签名；盲签名；双线性映射
中图分类号：TP309 **文献标识码：**A **文章编号：**1000-436X(2012)Z1-0058-12

Provably secure certificate less blind proxy re-signatures

FENG Tao, LIANG Yi-xin
(School of Computer and Communication, Lanzhou University of Technology, Lanzhou 730050, China)

Abstract: Using bilinear groups, an efficient certificate less blind proxy re-signature scheme was proposed based on the proxy re-signature scheme and blind signature scheme. The scheme solves the using of certificate in certificate-based scheme and removes key escrow in ID-based scheme. While achieving message privacy features of the proxy signature conversion. Based on the difficulty of NGBH and Many-NGBH problem, It proves that the new scheme can resist forgery attack characteristics. The scheme satisfies security properties: correctness and message blindness.

Key words: certificateless cryptography; proxy re-signatures; blind signature; bilinear map

1 引言

代理重签名一类特殊的数字签名[1]通过一个可信赖的代理者可以将受托者 Alice 在一个消息 m 上的签名转换为被委托者 Bob 在同一个消息 m 上的签名，代理者并不知道签名的密钥，同时不能代替 Alice 或者 Bob 在任一消息上签名。代理重签名可以在数字版权管理，证书简化管理，路径证明，弱群签名的形成等诸领域应用[2]。

现已构造的代理重签名大多都是在传统密码体制和基于身份密码体制下构造的，而前者的主要问题是证书管理复杂，验证签名时间开销大，这造成多用户系统效率低；后者又存在密钥托管的安全问题。2010 年邓宇乔在文献[3]中首次提出盲代理重签名方案也同样是基于传统密码体制下构造的，其借鉴盲签名体制实现了签名从原始签名者到代理重签名者之间的透明转换，保护了原始签名者的隐私。

相对于传统公钥体制和基于身份的公钥体制下的数字签名而言，无证书签名[4]的优势在于签名

收稿日期：2012-07-24
基金项目：国家自然科学基金资助项目(60972078,61072066)；甘肃省高等学校基本科研业务费基金资助项目(0914ZTB186)；甘肃省自然科学基金资助项目(2007GS04823)；兰州理工大学博士基金资助项目(BS14200901)
Foundation Items: The National Natural Science Foundation of China(60972078,61072066); The Universities Basic Scientific Research Foundation of Gansu Province (0914ZTB186); The Natural Science Foundation of Gansu Province(2007GS04823); The Lanzhou University of Technology PhD Programs of China (BS14200901)

者在验证签名时无须像在传统公钥密码系统下那样验证签名者公钥的有效性;同时也没有基于身份的密码系统中的密钥托管问题。无证书签名系统有两类攻击者，即第 1 类攻击者 A_I 与第 2 类攻击者 A_{II}。第 1 类攻击者不知道系统主密钥，但是可以任意替换用户的公钥；第 2 类攻击者知道系统的主密钥，但是不能替换目标用户的公钥。最早的无证书签名方案的安全模型由 Huang 等[5]人提出。该模型要求：如果第一类攻击者替换了用户 ID 的公钥，那么其在请求 ID 的签名时需要提供 ID 的当前公钥对应的秘密值。在文献[6]中，Zhang 等人给出了一种改进的安全模型，即挑战者回答攻击者的签名请求无须攻击者提供签名者的新公钥对应的秘密值。在文献[7]中，Zhang 等人进一步探讨了无证书签名的安全模型。

在文献[3]中 Deng 方案存在证书存储和管理的问题及验证签名者公钥的有效性的问题，为解决以上不足，本文在现有方案的基础上采用无证书公钥体制，提出了一个无证书盲代理重签名，并根据文献[7]提出来的安全模型， 基于计算 Diffie-Hellman 问题的困难性，证明了新方案具有能够抵抗伪造攻击的特性。

本文结构安排如下：第 2 节介绍了双线性对，NGBDH 及 Many-DH 困难假设的基础知识。第 3 节形式化定义了无证书盲代理重签名及安全模型。第 4 节无证书盲代理重签名方案详细描述。第 5 节基于标准模型下的新方案安全性证明。第 6 节是新方案时间复杂度分析。第 7 节是结束语。

2　预备知识

2.1　双线性对的基本概念

假设 G_1 为由 p 生成的阶，作为素数 q 的循环加法群，G_2 为具有与其相同的阶，作为 q 的循环乘法群。假设 $e:G_1\times G_2\to G_2$ 为满足以下 3 个性质的双线性映射。

1) 双线性：对全体的， $R,Q\in G_1,a,b\in Z_q^*$，都满足等式： $e(aR,bQ)=e(R,Q)^{ab}$。

2) 非退化性：存在 $R,Q\in G_1$，满足 $e(R,Q)^{ab}$。

3) 可计算性：对全体的 $R,Q\in G_1$，都可以用有效的算法，计算出 $e(R,Q)$。

2.2　困难问题的假设

NGBDH 困难假设：在群 G 的阶是素数 $p=\theta(2^k)$，g 是群 G 的生成元。给定 $g^a,g^b\in G$，其中 $a,b\in Z_p^*$，输入 (g^{abc},g^c)。算法 B 至少以优势 ε 解决 NGBDH 问题。假设满足下面不等式

$$\Pr[B(g,g^a,g^b)=(g^{abc},g^c)]\geqslant\varepsilon$$

如果对于任何的算法都无法在运行时间最多 t 同时也能至少优势 ε 解决 NGBDH 问题，则在群 G 中的 (ε,t) – NGBDH 假设成立。

Many-DH 困难假设：在群 G 的阶是素数 $p=\theta(2^k)$，g 是群 G 的生成元。给定 $g^a,g^b,g^c,g^{ab},g^{ac},g^{bc}\in G$，其中 $a,b,c\in Z_p^*$，输入 g^{abc}。算法 B 至少以优势 ε 解决 Many-DH 问题。假设满足下面不等式

$$\Pr[B(g,g^a,g^b,g^c,g^{ab},g^{ac},g^{bc})=g^{abc}]\geqslant\varepsilon$$

如果对于任何的算法都无法在运行时间最多 t 同时也能至少优势 ε 解决 Many-DH 问题，则在群 G 中的 (ε,t)-Many-DH 假设成立。

3　无证书盲代理重签名的定义和安全性

3.1　无证书盲代理重签名定义

无证书盲代理重签名方案 S_{clbprs} 由下面 9 个多项式时间算法构成 (Setup, Partial-Secret-Key-Extract, User-Key-Generation, ReKey, Sign, Blind, ReSign, Unblind, Verify)。

Setup(K)：这个 PPT 算法由 KGC 负责执行，算法需要以安全参数 K 作为输入值，然后计算得到公私密钥对 (mpk,msk)。

Partial-Secret-Key-Extract(*mpk*, *msk*, *ID*): 这个 PPT 算法由 KGC 负责执行，算法需要以 *mpk*, *msk* 还有用户身份 *ID* 作为输入值，然后计算输出对应于用户身份 *ID* 的部分密钥 psk_{ID}。

User-Key-Generation(*mpk*)：这个 PPT 算法由用户身份为 *ID* 的用户负责执行，算法需要 *mpk* 作为输入值，然后计算输出用户身份 *ID* 对应的公私钥密钥对 (pk_{ID},sk_{ID})。

ReKey(psk_A,sk_A,psk_B,sk_B)：这个 PPT 算法由代理者 Peter 负责执行，可以通过安全渠道秘密交换协议输入 psk_A,sk_A,psk_B,sk_B，而使代理者 Peter 无法获得 psk_A,sk_A,psk_B,sk_B。算法为代理者 Peter 生成密钥 $rk_{A\leftrightarrow B}$。这里 (psk_A,sk_A) 为 Alice 的私钥和部分密钥，(psk_B,sk_B) 为 Bob 的私钥和部分密钥。

Sign($m,mpk,ID,psk_{ID},sk_{ID}$)：这个 PPT 算法以

mpk，用户身份 ID，用户私钥 sk_{ID}，用户部分密钥 psk_{ID} 和消息 m 作为输入值，算法生成用户对消息 m 的签名 σ。

Blind(σ_A, m)：输入 m 及 Alice 对 m 的签名 $\sigma_A=(V_A, R_{\pi A}, R_{mA})$，Bob 输出盲化的签名 $\sigma'_A=(V'_A, R'_{\pi A}, R'_{mA})$。

BlindReSign($rk_{A\leftrightarrow B}, \sigma'_A$)：输入重签名密钥 $rk_{A\to B}$ 以及盲化后 Alice 的签名 $\sigma'_A=(V'_A, R'_{\pi A}, R'_{mA})$，首先验证盲化的签名是否有效，如果有效，代理者 Peter 输出盲代理重签名 $\sigma'_B=(V'_B, R'_{\pi B}, R'_{mB})$，否则停止。

Unblind(σ'_B)：输入盲代理重签名 $\sigma'_B=(V'_B, R'_{\pi B}, R'_{mB})$， Bob 验证该代理重签名的有效性，如果有效，则输出脱盲后的签名 $\sigma_B=(V_B, R_{\pi B}, R_{mB})$。

Verify(消息 m，Bob 的公钥 pk_B，转换签名 $\sigma_B(m)$)：确定性算法验证签名 σ_B 是否为消息 m 的有效签名。如果签名 σ_B 有效，输出 l；否则输出 0。

3.2 形式化安全模型

无证书盲代理重签名和之前定义的无证书代理重签名形式化安全模型一样，也分为 2 种类型攻击。

Type I：攻击者 A_I 通过不诚实的用户具有获得用户私钥 sk_{ID} 或者替换用户公钥 pk_{ID} 的能力，但是即无法获得 KGC 的主密钥 msk。具有询问，某用户 ID 的部分密钥 psk_{ID}、私钥 sk_{ID} 和公钥 pk_{ID} 的预言服务。

Type II：攻击者 A_{II} 拥有 KGC 的 msk，同时可以合法生成所有用户的部分密钥 psk_{ID}，私钥 sk_{ID} 和公钥 pk_{ID}，但是不能替换用户的公钥 psk_{ID}。具有询问，某用户 ID 的密钥 sk_{ID} 和公钥 pk_{ID} 的预言服务。

本文定义在静态攻陷模式下，挑战者 C 和攻击者 A 2 个游戏，一个游戏针对攻击者 A_I 而另一个游戏针对攻击者 A_{II}。

游戏 I：在 Type I 攻击类型下挑战者 C 与攻击者 A_I，定义如下游戏基于无证书公钥体制的代理重签名 S_{clprs} 的安全。

初始化：挑战者 C 运行 Setup 算法并且取得 KGC 公私钥对 (mpk, msk)。攻击者 A_I 取得 mpk。

查询：挑战者 C 可以通过以下预言机来回答攻击者 A_I 适应性的询问。

Partial-Secret-Key-Extract 预言机：通过给挑战者 C 输入用户身份 ID，攻击者 A_I 取得部分私钥 psk_{ID}，

Public-Key-Broadcast 预言机：通过给挑战者 C 输入用户身份 ID，挑战者 C 运行 User-Key-Generation 算法，攻击者 A_I 取得用户公钥 pk_{ID}。

Secret-Key-Extract 预言机：通过给挑战者 C 输入用户身份 ID，挑战者 C 运行 User-Key-Generation 算法，攻击者 A_I 取得用户私钥 sk_{ID}。

Replace-Public-Key 预言机：通过给挑战者 C 输入用户身份 ID 新的 (pk', sk')，攻击者 A_I 可以获得挑战者 C 使用新的公私钥对 (pk', sk') 来替换用户身份 ID 之前的公私钥。

ReKey 预言机：通过给挑战者 C 输入 (ID_A, ID_B)，攻击者 A_I 取得代理重签名 $rk_{A\leftrightarrow B}$。前提条件，(ID_A, ID_B) 要么同时没有被攻陷或者都被攻陷了。

Sign 预言机：通过给挑战者 C 输入 $(m, mpk, ID, psk_{ID}, sk_{ID})$，攻击者 A_I 可以取得相应于 A_I 的签名 σ。

BlindReSign 预言机：通过给挑战者 C 输入 (ID_A, ID_B, m, σ)，攻击者 A_I 可以取得重签名 σ。

伪造：攻击者 A_I 输出 $(pk_{ID^*}, \sigma^*, m^*)$。如果满足以下的条件，可以认为攻击者 A_I 赢得了这次游戏。

1) 攻击者 A_I 从未向 Partial-Secret-Key-Extract 预言机询问过 ID^*。

2) 攻击者 A_I 从未向 Sign 预言机询问过 (ID^*, m^*)。

3) $\text{Verify}(m^*, \sigma^*, mpk, ID^*, pk_{ID^*})=1$。

4) $(\Delta, ID^*, m^*, \Diamond)$ 从未做过 BlindReSign 预言机的输入，其中 Δ 表示任何一个用户身份，而 $\Diamond$ 则表示任何一个签名。

定义 $Succ_{A_I}^{cma}(k)$ 为攻击者 A_I 赢得游戏 I 的概率。

游戏 II：在 Type II 攻击类型下挑战者 C 与攻击者 A_{II}，定义如下游戏，基于无证书公钥体制的代理重签名 S_{clprs} 的安全。

初始化：挑战者 C 通过安全参数 k 运行 Setup 算法。攻击者 A_{II} 取得 $mpk \in MPK(k)$。其中 $MPK(k)$ 是所有可能通过 Setup 算法产生的主公钥。

查询：挑战者 C 可以通过以下预言机来回答攻击者 A_{II} 适应性的询问。

Public-Key-Broadcast 预言机：通过给挑战者 C 输入用户身份 ID，挑战者 C 运行 User-Key-

Generation 算法，攻击者 A_I 取得用户公钥 pk_{ID}。

Secret-Key-Extract 预言机:通过给挑战者 C 输入用户身份 ID，挑战者 C 运行 User-Key-Generation 算法，攻击者 A_I 取得用户私钥 sk_{ID}。

ReKey 预言机：通过给挑战者 C 输入 (ID_A, ID_B)，攻击者 A_{II} 取得代理重签名 $rk_{A\leftrightarrow B}$。前提条件，(ID_A, ID_B) 要么同时没有被攻陷或者都被攻陷了。

Sign 预言机：通过给挑战者 C 输入 $(m, mpk, ID, psk_{ID}, sk_{ID})$，攻击者 A_{II} 可以取得相应于 A_I 的签名 σ。

BlindReSign 预言机：通过给挑战者 C 输入 (ID_A, ID_B, m, σ)，攻击者 A_{II} 可以取得重签名 σ。

伪造：攻击者 A_{II} 输出 $(pk_{ID^*}, \sigma^*, m^*)$。如果满足以下的条件，可以认为攻击者 A_{II} 赢得了这次游戏。

攻击者 A_{II} 从未向 User-Key-Generation 预言机询问过 ID^*。

攻击者 A_{II} 从未向 Sign 预言机询问过 (ID^*, m^*)。

$\text{Verify}(m^*, \sigma^*, mpk, ID^*, pk_{ID^*}) = 1$。

$(\Delta, ID^*, m^*, \Diamond)$ 从未做过 BlindReSign 预言机的输入，其中 Δ 表示任何一个用户身份，而 $\Diamond$ 则表示任何一个签名。

本文定义 $Succ_{A_{II}}^{cma}(k)$ 为攻击者 A_{II} 赢得游戏 II 的概率。

4　方案描述

Setup(K)：通过输入安全参数 K，密钥生成中心 KGC 执行以下操作，产生主公钥 mpk 和主密钥 msk。

1) 选择 2 个阶为素数 $p = \Theta(2^k)$ 的有限循环群 G_1 和 G_2，且满足双线性对 $e: G_1 \times G_1 \to G_2$，设 g 为群 G_1 的一个生成元。

2) 随机选择 $a \in_R Z_p^*, g_2 \in G_1$ 并计算 $g_1 = g^a$。

3) 选择 3 个散列函数 $H_u: \{0,1\}^* \to \{0,1\}^{n_u}$ $H_u: \{0,1\}^* \to \{0,1\}^{n_u}$ 和 $H_m: \{0,1\}^* \to \{0,1\}^{n_m}$，设所有的用户身份信息是 n_u 比特长和消息是 n_m 比特长字符串。

4) 随机选择 $u', m' \in_R G_1$，2 个向量 $\hat{u}_i \in_R G_1$，$for\ \ i = 1, \cdots, n_u\ \ \hat{m}_i \in_R G_1\ \ for\ \ i = 1, \cdots, n_m$

同时令 $Let\ \hat{U} = \{H(\hat{u}_i)\}, \hat{M} = \{H(\hat{m}_i)\}$。

最后输出主公钥 $mpk = (G_1, G_2, p, e, g, g_1, u', \hat{U}, m', \hat{M}, H_1, H_2)$，其主密钥 $mk = g_2^a$ 保密。

Partial-Secret-Key-Extract(mpk, msk, ID)：用户提交身份 $ID \in \{0,1\}^*$ 给 KGC。KGC 首先计算 $u = H_u(ID)$，其中 $U \subset \{1, \cdots, n_u\}$ 是 $u[i] = 1$ 的索引 i 的集合，$u[i]$ 是身份 ID 中第 i bit 的值。然后 KGC 随机选择 $r_u \in_R Z_p^*$ 计算部分私钥 $psk = \left(psk^{(1)}, psk^{(2)}\right) = (g_2^a \left(u' \prod_{i \in U} \hat{u}_i\right)^{r_u}, g^{r_u})$。

User-Key-Generation(mpk)：用户随机选择 $x \in_R Z_p^*$ 作为其私钥 sk_{ID}，同时计算出其公钥 $pk_{ID} = \left(pk^{(1)}, pk^{(2)}\right) = \left(g^x, g_1^x\right)$。

ReKey(psk_A, sk_A, psk_B, sk_B)：输出一个重签名密钥

$$rk_{A\to B} = \frac{\left(psk_B\right)^{sk_B}}{\left(psk_A\right)^{sk_A}} = \left(\frac{\left(psk_B^{(1)}\right)^{sk_B}}{\left(psk_A^{(1)}\right)^{sk_A}}, \frac{\left(psk_B^{(2)}\right)^{sk_B}}{\left(psk_A^{(2)}\right)^{sk_A}}\right)$$

Sign(m, mpk, ID, pk_{ID}):由签名用户执行以下操作：首先计算 $m = H_m\left(M, R_\pi, R_m\right)$，接着随机选择 $r_\pi, r_m \in_R Z_p^*$，然后输出签名

$$\sigma = \left(V, R_\pi, R_m\right) = \left(\left(psk^{(1)}\right)^{sk} \left(u' \prod_{i \in U} \hat{u}_i\right)^{r_\pi} \left(m' \prod_{i \in M} \hat{m}_i\right)^{r_m}, \left(psk^{(2)}\right)^{sk} g^{r_\pi}, g^{r_m}\right)$$

其中 $U \subset \{1, \cdots, n_u\}$ 是 $u[i] = 1$ 的索引 i 的集合，$u[i]$ 是 ID 中第 i 比特的值。$M \subset \{1, \cdots, n_m\}$ 是 $m[i] = 1$ 的索引 i 的集合，$m[i]$ 表示的消息 m 第 i bit 的值。

Blind($\sigma_A, w = m' \prod_{i \in M} \hat{m}_i$)：输入 Alice 的签名 $\sigma_A = \left(V_A, R_{\pi A}, R_{mA}\right)$ 及该签名消息对应的值 $w = m' \prod_{i \in M} \hat{m}_i$（$M \subset \{1, \cdots, n_m\}$ 是 $m[i] = 1$ 的索引 i 的集合，$m[i]$ 表示的消息 m 中第 i bit 的值），Bob 选取 $k \in_R Z_p^*$，计算 $w' = w \times g^k, V_A' = V_A \times R_{mA}^k$，$R_{\pi A}' = R_{\pi A}$，$R_{mA}' = R_{mA}$，并把签名 $\sigma_A' = \left(V_A', R_{\pi A}', R_{mA}'\right)$ 发送给代理者 Peter。

BlindReSign($rk_{A\leftrightarrow B}, m' \prod_{i \in M} \hat{m}_i g^k, \sigma_A$)：输入重签

名密钥 $rk_{A\to B}$， w'，签名 $\sigma_A{}'$。

代理者 Peter 验证等式 $e(V_A',g)=e\left(g_2,pk^{(2)}\right)e\left(u'\prod_{i\in U}\hat{u}_i,R'_{\pi A}\right)e\left(m'\prod_{i\in M}\hat{m}_i,R'_{mA}\right)$ 是否成立。如果签名 $\sigma'_A=(V'_A,R'_{\pi A},R'_{mA})$ 是无效的，那么输出 $\perp$；否则输出盲代理重签名

$$\begin{aligned}\sigma'_B&=\sigma'_A rk_{A\leftrightarrow B}\\&=\left(V'_A\frac{\left(psk_B^{(1)}\right)^{sk_B}}{\left(psk_A^{(1)}\right)^{sk_A}},R'_{\pi A}\frac{\left(psk_B^{(2)}\right)^{sk_B}}{\left(psk_A^{(2)}\right)^{sk_A}},R'_{mA}\right)\\&=\left(\left(psk_B^{(1)}\right)^{sk_B}\left(u'\prod_{i\in U}\hat{u}_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}g^{kr_m},\left(psk_B^{(2)}\right)^{sk_B}g^{r_\pi},g^{r_m}\right)\\&=(V'_B,R'_{\pi B},R'_{mB})\end{aligned}$$

UnBlind(σ_B):输入盲代理重签名 $\sigma'_B=(V'_B,R'_{\pi A},R'_{mB})$，代理者 Peter 首先判断等式 $e(V'_B,g)=e\left(g_2,pk^{(2)}\right)e\left(u'\prod_{i\in U}\hat{u}_i,R'_{\pi B}\right)e\left(m'\prod_{i\in M}\hat{m}_i,R'_{mB}\right)$ 是否成立，如果签名 $\sigma'_B=(V'_B,R'_{\pi B},R'_{mB})$ 是无效的，那么输出 $\perp$；否则，Bob 将对它进行脱盲获得

$$\begin{aligned}\sigma_B&=\left(V_B'R'^{-k}_{mB},R'_{\pi A},R'_{mB}\right)\\&=\left(\left(psk_B^{(1)}\right)^{sk_B}\left(u'\prod_{i\in U}\hat{u}_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}g^{kr_m}g^{-kr_m},\left(psk_B^{(2)}\right)^{sk_B}g^{r_\pi},g^{r_m}\right)\\&=(V_B,R_{\pi B},R_{mB})\end{aligned}$$

Verify(m,σ,mpk,ID,pk_{ID})：为了验证消息 $m=H_m(M,R_\pi,R_m)$ 的签名 σ 是否有效，执行如下操作 $e\left(pk^{(1)},g_1\right)=e\left(pk^{(2)},g\right)$ 并且 $e(V,g)=e\left(g_2,pk^{(2)}\right)e\left(u'\prod_{i\in U}\hat{u}_i,R_\pi\right)e\left(m'\prod_{i\in M}\hat{m}_i,R_m\right)$ 验证等式，如等式成立则签名有效，否则签名无效。

5 安全性分析

5.1 协议的安全性分析

定理 1 无证书盲代理重签名方案 S_{clbprs} 满足正确性。

证明

$$\begin{aligned}e(V,g)&=e\left(\left(psk^{(1)}\right)^x\left(u'\prod_{i\in U}\hat{u}_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m},g\right)\\&=e\left(\left(psk^{(1)}\right)^x,g\right)e\left(\left(u'\prod_{i\in U}\hat{u}_i\right)^{r_\pi},g\right)e\left(\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m},g\right)\\&=e\left(g_2,g^{ax}\right)e\left(\left(u'\prod_{i\in U}u_i\right)^{r_u x+r_\pi},g\right)e\left(m'\prod_{i\in M}\hat{m}_i,g^{r_m}\right)\\&=e\left(g_2,g_1^x\right)e\left(u'\prod_{i\in U}u_i,g^{r_u x+r_\pi}\right)e\left(m'\prod_{i\in M}\hat{m}_i,g^{r_m}\right)\\&=e\left(g_2,pk^{(2)}\right)e\left(u'\prod_{i\in U}\hat{u}_i,R_\pi\right)e\left(m'\prod_{i\in M}\hat{m}_i,R_m\right)\end{aligned}$$

5.2 消息盲性

定理 2 无证书盲代理重签名方案 S_{clbprs} 具有消息盲性。

由于本方案 S_{clbprs} 中授权者 Bob 对消息 m 进行 w 值经过如下处理：$w'=w\times g^k$，其中 k 为授权者 Bob 自己秘密选取的，代理者 Peter 要通过 w' 求得 w 并由此辨认出原来的消息明显是不可能的。所以，本方案 S_{clbprs} 具有消息盲性。

5.3 不可伪造性

定理 3 (Type I 存在性不可伪造)基于无证书公钥体制的盲代理重签名方案 S_{clbprs} 在 Type I 类型的攻击下，是 (ε,t) —存在性不可伪造，其中 ε 拥有最多优势，t 拥有最多运行时间。假定 (ε',t')-NGBDH 是在群 G_1 的难解性成立。其中，$\varepsilon'\geqslant\dfrac{\varepsilon}{16(q_E+q_S+q_{BRS}+2q_{RK})(q_S+q_{BRS})(n_{id}+1)(n_m+1)}$ 和 $t+O\left(\begin{matrix}((q_E+q_{RK})n_{id}+(q_S+q_{BRS})(n_{id}+n_m))\rho\\+(q_k+q_E+q_S+q_{BRS}+q_{RK})\tau\end{matrix}\right)$。令 q_E 为询问 Partial-Secret-Key-Extract 预言机最多次数，q_S 为询问 Sign 预言机的最多次数，q_k 为询问 Public-Key-Broadcast 预言机的和 Secret-Key-Extract 预言机最多次数之和，q_{RK} 是询问 ReKey 预言机最多次数，q_{BRS} 是询问 BlindReSign 预言机最多次数，ρ 是在群 G_1 的进行乘法是在群 G_1 进行乘法运算和 τ 为在群 G_1 的进行指数运算的时间。

证明 假设在 Type I 类型的攻击者 A 存在。构造另一个攻击者 B 利用攻击者 A 最少在 ε' 的概率下最多 t' 的时间解决 NGBDH 问题。攻击者 B 给出如下问题实例：群 G_1 的阶是素数 $p=\theta(2^k)$，其中 g 是群 G_1 的生成元且 $g^a,g^b\in G_1$，同时输出 g^{abc}，$g^c\in G_1$。为了利用攻击者 A 解决这个问题，攻击

者 B 需要模拟一个挑战者和所有预言机，并将按照以下步骤做。

初始化：为了准备这个无证书盲代理重签名 S_{clbprs} 的仿真游戏，进行如下参数设置。

1) $l_{id}=2\left(q_E+q_S+q_{BRS}+2q_{RK}\right)$；

2) $l_m=2\left(q_S+q_{BRS}\right)$；

3) k_{id} 为随机整数，满足 $0\leqslant k_{id}\leqslant n_{id}$、$l_{id}\left(n_{id}+1\right)<p$；

4) k_m 为随机整数，满足 $0\leqslant k_m\leqslant n_m$、$l_m\left(n_m+1\right)<p$；

5) 随机选择 $n_{id}+1$ 个不大于 l_{id} 的整数 x'、$x_i\left(i=1,\cdots,n_{id}\right)$；

6) 随机选择 n_m+1 个不大于 l_m 的整数 z'、$z_i\left(i=1,\cdots,n_m\right)$；

7) 随机选择 $n_{id}+n_m+2$ 个数 y'、$y_i(i=1,\cdots,n_{id})$、w'、$w_i(i=1,\cdots,n_m)$。

根据以上参数设置，设置如下式子

$$F(ID)=x'+\sum_{i\in U}x_i-l_{id}k_{id}\text{ 和 }J(ID)=y'+\sum_{i\in U}y_i$$

$$K(m)=z'+\sum_{i\in M}z_i-l_mk_m\text{ 和 }L(m)=w'+\sum_{i\in M}w_i$$

攻击者 B 构造了如下一些公开参数为

1) $g_1=g^a,g_2=g^b$；

2) $u'=g_2^{x'-l_{id}k_{id}}g^{y'},u_i=g_2^{x_i}g^{y_i}\left(1\leqslant i\leqslant n_{id}\right)$；

3) $m'=g_2^{z'-l_mk_m}g^{w'},m_i=g_2^{z_i}g^{w_i}\left(1\leqslant i\leqslant n_m\right)$。

根据以上参数设置，对于任意用户身份 *ID* 和任意消息 *m*，本文拥有下面的等式。

$$u'\prod_{i\in u}u_i=g_2^{F(ID)}g^{J(ID)}\quad\text{和}\quad m'\prod_{i\in M}m_i=g_2^{F(m)}g^{J(m)}$$

并且将所有公开参数传递给攻击者 A。

查询：攻击者 B 模拟以下所有预言机 $O_{\text{Public-Key-Broadcast}}$、$O_{\text{Secret-Key-Extract}}$、$O_{\text{Partial-Secret-Key-Extract}}$、$O_{\text{ReKey}}$、$O_{\text{Sign}}$ 及 $O_{\text{BlindReSign}}$ 预言服务。

$O_{\text{Public-Key-Broadcast}}$：攻击者 B 把用户的公私钥对保存在数据库中。当接受到用户身份 *ID* 关于公钥查询时，攻击者 B 首先在数据库中查询与用户身份 *ID* 相对应的记录。如果查询不到，攻击者 B 运行 User-Key-Generation 算法计算出用户私钥 sk_{ID} 和公钥 pk_{ID}，并且把公私钥对保存在数据库中，同时把公钥 pk_{ID} 查询结果返回。

$O_{\text{Secret-Key-Extract}}$：当接受到用户身份 *ID* 关于私钥查询时，攻击者 B 首先在数据库中查询与用户身份 *ID* 相对应的记录。如果查询不到，攻击者 B 运行 User-Key-Generation 算法计算出用户私钥 sk_{ID} 和公钥 pk_{ID}，并且把这对公私钥对存放在数据库中，同时把公钥 sk_{ID} 查询结果返回。

$O_{\text{Partial-Secret-Key-Extract}}$：当接受到用户身份 *ID* 进行部分密钥查询时

Function Partial_Secret_Key_Extract(参数:用户身份 *ID*)

if (*ID* 是未被攻陷的　OR　*ID* 是被攻陷)

CDH 攻击者 B，首先计算 $F(ID)$ 的值，然后进行如下步骤。

if　$F(ID)=0 \bmod p$　then

CDH 攻击者 *B* 模拟游戏结束并宣告失败；

else if　$F(ID)\neq 0 \bmod p$　then

CDH 攻击者 B，随机选择 $r_{id}\in Z_p^*$ 并计算出部分密钥

$$\left(psk^{(1)},psk^{(2)}\right)=\left(g_1^{-J(ID)/F(ID)}\left(u'\prod_{i\in U}u_i\right)^{r_{id}},g_1^{-1/F(ID)}g^{r_{id}}\right)$$

通过设置 $\tilde{r}_{id}=r_{id}-a/F(ID)$，验证 *psk* 是否为有效的部分密钥

$$\begin{aligned}psk^{(1)}&=g_1^{-J(ID)/F(ID)}\left(u'\prod\nolimits_{i\in u}u_i\right)^{r_{id}}\\&=g_1^{-J(ID)/F(ID)}\left(g^{J(ID)}g_2^{F(ID)}\right)^{r_{id}}\\&=g_2^{a}\left(g_2^{F(ID)}g^{J(ID)}\right)^{-a/F(ID)}\left(g_2^{F(ID)}g^{J(ID)}\right)^{r_{id}}\\&=g_2^{a}\left(g_2^{F(ID)}g^{J(ID)}\right)^{r_{id}-a/F(ID)}\\&=g^{a}\left(u'\prod\nolimits_{i\in U}u_i\right)^{\tilde{r}_{id}}\end{aligned}$$

和 $psk^{(2)}=g_1^{-1/F(ID)}g^{r_{id}}=g^{r_{id}-a/F(ID)}=g^{\tilde{r}}$

end if

end if

end function

通过执行上面函数，对攻击者 A 而言，无法区分所有由攻击者 B 产生的部分密钥和 KGC 产生的部分密钥。

$O_{\text{Public-Key-Replace}}$：当接受到用户身份 *ID* 替换公钥查询时，攻击者 B 首先在数据库中查询与用户身份 *ID* 相对应的记录。如果查询不到，攻击者 B 运行为用户身份 *ID* 创建新的实体。

O_{Sign}：当接受到用户身份 *ID* 关于消息 *m* 的签

名查询时

Function Sign(参数 1：用户身份 ID，参数 2：消息 m)

攻击者 B 在数据库中查询用户身份 ID 的公钥是否被替换

if 已经用 $(pk^{(1)}, pk^{(2)})$ 进行了公钥替换 then

攻击者 B 先获得 $K(m)$ 的值，然后执行如下判断

if $K(m)=0 \bmod p$ then

攻击者 B 模拟游戏结束并且宣告失败；

else if $K(m)\neq 0 \bmod p$ then

攻击者 B 随机选择 2 个数 $r_\pi, r_m \in Z_p^*$ 并且 $\tilde{r}_m = r_m - \frac{ax}{K(m)}$，构造并计算签名

$$\sigma = ((pk^2)^{-\frac{L(m)}{K(m)}}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}, g^{r_\pi}, (pk^{(2)})^{-\frac{1}{K(m)}}g^{r_m})$$
$$= (g_2^{ax}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{\tilde{r}_m}, g^{r_\pi}, g^{\tilde{r}_m})$$
$$= (V, R_\pi, R_m)$$

end if

else if 没有进行公钥替换 then

首先计算出 $u = H_u(ID), m = H_m(m)$

if $F(ID)\neq 0 \bmod p$ then

攻击者 B 尝试用 $O_{\text{Partial-Secret-Key-Extract}}$ 生成部分密钥，然后检查数据库中该用户身份 ID 的私钥是否被创建；如果没有创建，则运行 User-Key-Generation 算法生成公私密钥对并保存在数据库中。如果创建了，则用签名算法生成 ID 和 m 上的签名。

else if $F(ID)=0 \bmod p$ then

攻击者 B 先获得 $K(m)$ 的值，然后执行如下判断

if $K(m)=0 \bmod p$ then

攻击者 B 模拟游戏结束并且宣告失败；

else if $K(m)\neq 0 \bmod p$ then

攻击者 B 随机选择 $r_\pi, r_m \in Z_p^*$ 并且 $\tilde{r}_m = r_m x - \frac{a}{K(m)}x$，其中 x 取自数据库，如果数据库中不存在就使用 User-Key-Generation 算法生成先。然后构造并计算签名。

$$\sigma = (g_1^{-\frac{L(m)}{K(m)}x}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m x},$$
$$g^{r_\pi}, g_1^{-\frac{x}{K(m)}}g^{r_m x})$$
$$= (g_2^{ax}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{\tilde{r}_m}, g^{r_\pi}, g^{\tilde{r}_m})$$
$$= (V, R_\pi, R_m)$$

end if

end if

end if

end function

通过执行上面的函数，从攻击者 A 无法区分攻击者 B 产生的签名和由真实用户产生的签名。

O_{ReKey}：当接受到重签名密钥查询时：

Function ReKey (参数 (ID_i, ID_j))

if (ID_i, ID_j) 用户身份都未被攻破 then

根据用户身份 ID_i, ID_j 分别在数据库中查找对应部分密钥，如不存在分别随机选择 r_{id_i}, r_{id_j}，并与其对应的身份 ID_i, ID_j 存在数据库中。

然后计算重签名密钥

$$rk_{i\leftrightarrow j} = (\frac{\left(u'\prod_{I\in ID_i}u_i\right)^{r_{id_i}}}{\left(u'\prod_{I\in ID_j}u_i\right)^{r_{id_j}}}, g^{r_{id_i}-r_{id_j}})$$ ($I\in ID_i$ 其中 ID_i 的第 I 位的值是 1)

else if (ID_i, ID_j) 都是被攻陷的 then

$rk_{i\leftrightarrow j} = \text{ReKey}(O_{\text{Partial-Secret-Key-Extract}}(ID_i), O_{\text{Partial-Secret-Key-Extract}}(ID_j))$

(通过部分用户密钥询问预言机 $O_{\text{Partial-Secret-Key-Extract}}$)。

end if

end function

$O_{\text{BlindReSign}}$：当接受到重签名询时输入 (ID_i, ID_j, m, σ')，执行如下算法。

Function BlindReSign(参数: (ID_i, ID_j, m, σ'))

if Verify$(ID_i, m, \sigma)\neq 1$ then

攻击者 B 输出 $\perp$。

else

if ID_i 和 ID_j 被攻陷或者未被攻陷 then

BLindReSign$(O_{\text{ReKey}}(ID_i, ID_j), ID_i, m, \sigma')$

else

输出 $O_{\text{Sign}}(ID_j, m)$

end if

end if

end function

伪造：

if 攻击者 B 没有退出模拟游戏且模拟查询成功 then

攻击者 A 将会至少以概率 ε，取得一个消息 m^*，一个用户身份 ID^* 和一个在 pk_{ID^*} 对消息 m^* 的伪造签名 $\sigma^* = (V, R_\pi, R_m)$。

if $F(ID^*) \neq 0 \bmod p$ or $K(m^*) \neq 0 \bmod p$ then 攻击者 B 结束模拟游戏并宣告失败；

else

对于某些 $r_{id}^*, r_m^* \in Z_P$，σ^* 一定满足下面的等式

$$\frac{V}{R_\pi^{J(ID^*)} R_m^{L(m^*)}} = \frac{g_2^{ax}\left(u'\prod_{i\in U} u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}}{g^{J(u^*)r_\pi} g^{L(m^*)r_m}}$$

$$= \frac{g_2^{ax}\left(g_2^{F(ID^*)} g^{J(ID^*)}\right)^{r_\pi}\left(g_2^{K(m^*)} g^{L(m^*)}\right)^{r_m}}{g^{J(u^*)r_\pi} g^{L(m^*)r_m}}$$

$$= g_2^{ax}$$

$$= g^{abx}$$

攻击者 B 输出 $(g^{abx}, pk_{ID^*}^{(1)}) = (g^{abx}, g^x)$ 作为解决 NGBDH 问题的实例；

end if

概率分析：

为了成功完成模拟游戏，要求以下条件都必须满足：

预言机 $O_{\text{Partial-Secret-Key-Extract}}$ 的输入的 ID 满足 $F(ID^*) \neq 0 \bmod p$；

预言机 O_{Sign} 和预言机 $O_{\text{BlindReSign}}$ 的输入的 (ID, m) 当发生公钥替换的时候满足 $F(ID) \neq 0 \bmod p$ 或者 $K(m) \neq 0 \bmod p$；当没有发生公钥替换的时候满足 $K(m) \neq 0 \bmod p$；

预言机 O_{ReKey} 的所有输入的 (ID_i, ID_j) 满足

$$F(ID_i) \neq 0 \bmod p \text{、} F(ID_j) \neq 0 \bmod p$$

$$F(ID^*) = 0 \bmod p \text{ 和 } K(m^*) = 0 \bmod p$$

为了使分析概率简单，设 $ID_1, ID_2, \cdots, ID_{q_I}$ 是出现在查询 $O_{\text{Partial-Secret-Key-Extract}}$、$O_{Sign}$ 和 $O_{\text{BlindReSign}}$ 中的用户身份当然 ID^* 不在其中。同时设 $m_1, m_2, \cdots, m_{q_M}$ 是出现在 $O_{\text{BlindReSign}}$ 和 O_{Sign} 中的消息。显然，有 $q_I \leqslant q_E + q_S + q_{BRS} + 2q_{RK}$ 和 $q_M \leqslant q_S + q_{BRS}$。

定义事件 E_i^F、$E_i^{'F}$、E_F^*、E_i^K、$E_i^{'K}$ 和 E_K^* 如下。

$$E_i^F : F(ID_i) \neq 0 \bmod p, E_i'^F : F(ID_i) \neq 0 \bmod l_{id},$$

$$E_F^* : F(ID^*) = 0 \bmod p,$$

$$E_i^K : K(m_i) \neq 0 \bmod p, E_i'^K : F(m_i) \neq 0 \bmod l_m,$$

$$E_K^* : K(m^*) = 0 \bmod p$$

显然事件 $\wedge_{i=1}^{q_I} E_i^F$、E_F^*、$\wedge_{i=1}^{q_M} E_i^K$、E_K^* 是独立的，因此，攻击者 B 不退出的概率 $\Pr[\neg abort] \geqslant \Pr\left[\wedge_{i=1}^{q_I} E_i^F \wedge E_F^* \wedge_{i=1}^{q_M} E_i^K \wedge E_K^*\right]$。

由于 $l_{id}(n_{id}+1) < q$、$x', x_i (i = 1, \cdots, n_m)$ 都是不大于 l_{id} 的正整数，因此得出 $0 \leqslant l_{id} k_{id} < q$ 和 $0 \leqslant x' + \sum_{i\in U} x_i < q$。

由 $l_{id}(n_{id}+1) < q$ 假设可知 $F(ID) = 0 \bmod q$ 则得出 $F(ID) = 0 \bmod l_m$，从 $F(m) \neq 0 \bmod l_{id}$ 得出 $F(m) \neq 0 \bmod q$。因此，可以推出 $\Pr[E_i^F] \geqslant \Pr[E_i^F]$，及

$$\Pr[E_F^*] = \Pr[F(ID^*) = 0 \bmod q \wedge F(ID^*) = 0 \bmod l_{id}]$$

$$= \Pr[F(ID^*) = 0 \bmod l_{id}] \Pr\left[\begin{array}{l} F(ID^*) = 0 \bmod p \\ | F(ID^*) = 0 \bmod l_{id} \end{array}\right]$$

$$= \frac{1}{l_{id}} \frac{1}{n_{id}+1}$$

和

$$\Pr[\wedge_{i=1}^{q_I} E_i^F] \geqslant \Pr\left[\wedge_{i=1}^{q_I} E_i^{'}\right]$$

$$= 1 - \Pr\left[\vee_{i=1}^{q_I} \neg E_i^{'F}\right]$$

$$\geqslant 1 - \sum_{i=1}^{q_I} \Pr\left[\neg E_i^{'F}\right]$$

$$= 1 - \frac{q_{ID}}{l_{id}}$$

$$= 1 - \frac{q_E + q_S + q_{BRS} + q_{RK}}{l_{id}}$$

$$\geqslant 1/2$$

本文用类似的分析技术，得到 $\Pr[E_K^*] \geqslant \frac{1}{2(q_S + q_{BRS})} \cdot \frac{1}{n_m + 1}$ 和 $\Pr[\wedge_{i=1}^{q_M} E_i^K] \geqslant 1/2$

通过整理上述结果，可以得到

$$\Pr[\neg abort] \geqslant \Pr[\wedge_{i=1}^{q_I} E_i^F \wedge E_F^* \wedge_{i=1}^{q_M} E_i^K \wedge E_K^*]$$
$$\geqslant \frac{1}{16(q_E+q_S+q_{BRS}+2q_{RK})(q_S+q_{BRS})(n_{id}+1)(n_m+1)}$$

如果模拟不放弃，就会产生一个伪造的签名概率至少 ε。因此，攻击者 B 可以解决 NGBDH 问题实例的概率为

$$\varepsilon' \geqslant \frac{\varepsilon}{16(q_E+q_S+q_{BRS}+2q_{RK})(q_S+q_{BRS})(n_{id}+1)(n_m+1)}$$

(注意：由于 $O_{\text{Public-Key-Broadcast}}$ 查询、$O_{\text{Secret-Key-Extract}}$ 查询和 $O_{\text{Public-Key-Replace}}$ 查询不导致模拟放弃，因此他们被排除在概率分析。)

定理 4 (Type II 存在性不可伪造) 基于无证书公钥体制的盲代理重签名方案 S_{clbprs} 在 Type II 类型的攻击下，是 (ε,t) —存在性不可伪造，其中 ε 拥有最多优势，t 拥有最多运行时间。设 (ε',t')-Many-DH 是在群 G_1 的难解性成立。其中 $\varepsilon' \geqslant \frac{\varepsilon}{16q_K(q_E+q_S+q_{BRS}+2q_{RK})(q_S+q_{BRS})(n_{id}+1)(n_m+1)}$ 和 $t+O((q_{RK}n_{id}+(q_S+q_{BRS})(n_{id}+n_m))\rho+(q_k+q_S+q_{BRS}+q_{RK})\tau)$。令 q_S 为询问 Sign 预言机的最多次数，q_k 为询问 Public-Key-Broadcast 预言机的和 Secret-Key-Extract 预言机最多次数之和，q_{RK} 是询问 ReKey 预言机最多次数，q_{BRS} 是询问 BlindReSign 预言机最多次数，ρ 是在群 G_1 的进行乘法是在群 G_1 进行乘法运算和 τ 为在群 G_1 的进行指数运算的时间。

证明 假设在 Type II 类型的攻击者 A 存在。本文构造另一个攻击者 B 利用攻击者 A 最少在 ε' 的概率下最多 t' 的时间解决 Many-DH 问题。攻击者 B 给出如下问题实例：群 G_1 的阶是素数 $p=\theta(2^k)$，其中 g 是群 G_1 的生成元且 $g^a, g^b, g^x, g^{ab}, g^{ax}, g^{bx} \in G_1$，同时输出 $g^{abx} \in G_1$。为了利用攻击者 A 解决这个问题，攻击者 B 需要模拟一个挑战者和所有预言机，并将按照以下步骤做。

初始化：为了准备这个无证书盲代理重签名 S_{clbprs} 的仿真游戏，进行如下参数设置。

1) $l_{id}=2(q_E+q_S+q_{BRS}+2q_{RK})$；

2) $l_m=2(q_S+q_{BRS})$；

3) k_{id} 为随机整数，满足 $0 \leqslant k_{id} \leqslant n_{id}$、$l_{id}(n_{id}+1)<p$；

4) k_m 为随机整数，满足 $0 \leqslant k_m \leqslant n_m$、$l_m(n_m+1)<p$；

5) 随机选择 $n_{id}+1$ 个不大于 l_{id} 的整数 x'、$x_i(i=1,\cdots,n_{id})$；

6) 随机选择 n_m+1 个不大于 l_m 的整数 z_i'、$z_i(i=1,\cdots,n_m)$；

7) 随机选择 $n_{id}+n_m+2$ 个数 y'、$y_i(i=1,\cdots,n_{id})$、w'、$w_i(i=1,\cdots,n_m)$；

根据以上参数设置，设置如下式子。

$F(ID)=x'+\sum_{i\in U}x_i-l_{id}k_{id}$ 和 $J(ID)=y'+\sum_{i\in U}y_i$

$K(m)=z'+\sum_{i\in M}z_i-l_mk_m$ 和 $L(m)=w'+\sum_{i\in M}w_i$

攻击者 B 构造了如下一些公开参数为

1) $g_1=g^a, g_2=g^b, (g_2)^a=g^{ab}$；

2) $pk_{ID^*}^{(1)}=g^x, pk_{ID^*}^{(2)}=g^{ax}$；

3) $u'=g_2^{x'-l_{id}k_{id}}g^{y'}, u_i=g_2^{x_i}g^{y_i}(1\leqslant i\leqslant n_{id})$；

4) $m'=g_2^{z'-l_mk_m}g^{w'}, m_i=g_2^{z_i}g^{w_i}(1\leqslant i\leqslant n_m)$。

根据以上参数设置，对于任意用户身份 ID 和任意消息 m，本文拥有下面的等式

$$u'\prod_{i\in u}u_i=g_2^{F(ID)}g^{J(ID)} \text{ 和 } m'\prod_{i\in M}m_i=g_2^{F(m)}g^{J(m)}$$

并且将所有公开参数和主密钥 $g_2^a=g^{ab}$ 传递给攻击者 A。

查询：攻击者 B 模拟以下所有预言机 $O_{\text{Public-Key-Broadcast}}$、$O_{\text{Secret-Key-Extract}}$、$O_{\text{Partial-Secret-Key-Extract}}$、$O_{\text{ReKey}}$、$O_{\text{Sign}}$ 及 $O_{\text{BlindReSign}}$ 预言服务。

$O_{\text{Public-Key-Broadcast}}$：攻击者 B 把用户的公私钥对保存在数据库中，首先把身份 ID^* 的公开密钥保存在数据库中。当接受到用户身份 ID 关于公钥查询时，攻击者 B 首先在数据库中查询与用户身份 ID 相对应的记录。如果查询不到，攻击者 B 运行 User-Key-Generation 算法计算出用户私钥 sk_{ID} 和公钥 pk_{ID}，并且把公私钥对保存在数据库中，同时把公钥 pk_{ID} 查询结果返回。

$O_{\text{Secret-Key-Extract}}$：当接受到用户身份 ID 关于私钥查询时，攻击者 B 首先在数据库中查询与用户身份 ID 相对应的记录。如果查询不到，攻击者 B 运行 User-Key-Generation 算法计算出用户私钥 sk_{ID} 和公钥 pk_{ID}，并且把这对公私钥对存放在数据库中，同时把公钥 sk_{ID} 查询结果返回。

O_{Sign}：当接受到用户身份 ID 对消息 m 的签名查询时

Function Sign(参数 1：用户身份 ID，参数 2:消息 m)

攻击者 B 首先检查用户身份是否等于 ID^*

if 用户身份等于 ID^* then

攻击者 B 先获得 $K(m)$ 的值，然后执行如下判断

if $K(m)=0 \bmod p$ then

攻击者 B 模拟游戏结束并且宣告失败；

else if $K(m)\neq 0 \bmod p$ then

攻击者 B 随机选择二个数 $r_\pi, r_m \in Z_p^*$ 并且 $\bar{r}_m = r_m - \dfrac{ax}{K(m)}$，构造并计算签名。

$$\sigma = ((pk_{ID^*}^{(2)})^{-\frac{L(m)}{K(m)}}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}, g^{r_\pi}, (pk_{ID^*}^{(2)})^{-\frac{1}{K(m)}} g^{r_m})$$

$$= (g_2^{ax}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{\tilde{r}_m}, g^{r_\pi}, g^{\tilde{r}_m})$$

$$= (V, R_\pi, R_m)$$

end if

else if 用户身份不等于 ID^* then

首先计算出 $u = H_u(ID), m = H_m(m)$

if $F(ID)\neq 0 \bmod p$ then

攻击者 B 尝试用 $O_{\text{Partial-Secret-Key-Extract}}$ 生成部分密钥，然后检查数据库中该用户身份 ID 的私钥是否被创建；如果没有创建，则运行 User-Key-Generation 算法生成公私密钥对并保存在数据库中。如果创建了，则用签名算法生成 ID 和 m 上的签名。

else if $F(ID)=0 \bmod p$ then

攻击者 B 先获得 $K(m)$ 的值，然后执行如下判断

If $K(m)=0 \bmod p$ then

攻击者 B 模拟游戏结束并且宣告失败；

else if $K(m)\neq 0 \bmod p$ then

攻击者 B 随机选择 $r_\pi, r_m \in Z_p^*$ 并且 $\tilde{r}_m = r_m x - \dfrac{a}{K(m)}x$，其中 x 取自数据库，如果数据库中不存在就使用 User-Key-Generation 算法生成先。然后构造并计算签名。

$$\sigma = (g_1^{-\frac{L(m)}{K(m)}x}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m x},$$

$$g^{r_\pi}, g_1^{-\frac{x}{K(m)}} g^{r_m x})$$

$$= (g_2^{ax}\left(u'\prod_{i\in U}u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{\tilde{r}_m}, g^{r_\pi}, g^{\tilde{r}_m})$$

$$= (V, R_\pi, R_m)$$

end if

end if

end if

end function

通过执行上面的函数，从攻击者 A 无法区分攻击者 B 产生的签名和由真实用户产生的签名。

O_{ReKey}：当接受到重签名密钥查询时：

Function ReKey(参数 (ID_i, ID_j))

if (ID_i, ID_j) 用户身份都未被攻破 then

根据用户身份 ID_i, ID_j 分别在数据库中查找对应部分密钥，如不存在分别随机选择 r_{id_i}, r_{id_j}，并与其对应的身份 ID_i, ID_j 存在数据库中。

然后计算重签名密钥

$$rk_{i\leftrightarrow j} = \left(\frac{\left(u'\prod_{I\in ID_i}u_i\right)^{r_{id_i}}}{\left(u'\prod_{I\in ID_j}u_i\right)^{r_{id_j}}}, g^{r_{id_i}-r_{id_j}}\right)$$ ($I \in ID_i$ 其中 ID_i 的第 I 位的值是 1)

else if (ID_i, ID_j) 都是被攻陷的 then

$rk_{i\leftrightarrow j} = \text{ReKey}(O_{\text{Partial-Secret-Key-Extract}}(ID_i), O_{\text{Partial-Secret-Key-Extract}}(ID_j))$ (通过部分用户密钥询问预言机 $O_{\text{Partial-Secret-Key-Extract}}$)。

end if

end function

$O_{\text{BlindReSign}}$：当接受到重签名询时输入 (ID_i, ID_j, m, σ)，执行如下算法。

Function BlindReSign(参数：(ID_i, ID_j, m, σ))

if $Verify(ID_i, m, \sigma)\neq 1$ then

攻击者 B 输出 $\perp$。

else

if ID_i 和 ID_j 被攻陷或者未被攻陷 then Blind ReSign$(O_{\text{ReKey}}(ID_i, ID_j), ID_i, m, \sigma)$

else

输出 $O_{\text{Sign}}(ID_j, m)$

end if

end if

end function

伪造：

if 攻击者 B 没有退出模拟游戏且模拟查询成功 then

攻击者 A 将会至少以概率 ε，取得一个消息 m^*，一个用户身份 ID^* 和一个在 pk_{ID^*} 对消息 m^* 的伪造签名 $\sigma^* = (V, R_\pi, R_m)$。

if $F(ID^*) \neq 0 \bmod p$ or $K(m^*) \neq 0 \bmod p$ then 攻击者 B 结束模拟游戏并宣告失败；

else

对于某些 $r_{id}^*, r_m^* \in Z_P$，σ^* 一定满足下面的等式。

$$\frac{V}{R_\pi^{J(ID^*)} R_m^{L(m^*)}} = \frac{g_2^{ax}\left(u'\prod_{i\in U} u_i\right)^{r_\pi}\left(m'\prod_{i\in M}\hat{m}_i\right)^{r_m}}{g^{J(u^*)r_\pi} g^{L(m^*)r_m}}$$

$$= \frac{g_2^{ax}\left(g_2^{F(ID^*)} g^{J(ID^*)}\right)^{r_\pi}\left(g_2^{K(m^*)} g^{L(m^*)}\right)^{r_m}}{g^{J(u^*)r_\pi} g^{L(m^*)r_m}}$$

$$= g_2^{ax}$$

$$= g^{abx}$$

攻击者 B 输出 g^{abx} 作为解决 Many-DH 问题的实例；

end if

概率分析：

为了成功完成模拟游戏，本文要求以下条件都必须满足。

1) 预言机 O_{Sign} 和预言机 $O_{\text{BlindReSign}}$ 的输入的 (ID, m) 当 $ID \neq ID^*$ 要求满足 $F(ID) \neq 0 \bmod p$ 或者 $K(m) \neq 0 \bmod p$；当 $ID = ID^*$ 要求满足 $K(m) \neq 0 \bmod p$；

2) 预言机 O_{ReKey} 的所有输入的 (ID_i, ID_j) 满足 $F(ID_i) \neq 0 \bmod p$、$F(ID_j) \neq 0 \bmod p$；

3) $F(ID^*) = 0 \bmod p$ 和 $K(m^*) = 0 \bmod p$。

此外，为了获得所需的结果，它需要攻击者 A 选择身份 ID^* 的一个伪造签名。为了使分析概率简单，设 $ID_1, ID_2, \cdots, ID_{q_I}$ 是出现在查询 O_{Sign} 和 $O_{\text{BlindReSign}}$ 中的用户身份当然 ID^* 不在其中。同时设 $m_1, m_2, \cdots, m_{q_M}$ 是出现在 $O_{\text{BlindReSign}}$ 和 O_{Sign} 中的消息。显然，有 $q_I \leqslant q_E + q_S + q_{BRS} + 2q_{RK}$ 和 $q_M \leqslant q_S + q_{BRS}$。

定义事件 E_i^F、$E_i^{'F}$、E_F^*、E_i^K、$E_i^{'K}$ 和 E_K^* 如下。

$E_i^F : F(ID_i) \neq 0 \bmod p$,

$E_i^{'F} : F(ID_i) \neq 0 \bmod l_{id}, E_F^* : F(ID^*) = 0 \bmod p$,

$E_i^K : K(m_i) \neq 0 \bmod p, E_i^{'K} : F(m_i) \neq 0 \bmod l_m$,

$E_K^* : K(m^*) = 0 \bmod p$

显然事件 $\wedge_{i=1}^{q_I} E_i^F$、E_F^*、$\wedge_{i=1}^{q_M} E_i^K$、E_K^* 是独立的，因此，攻击者 B 不退出的概率 $\Pr[\neg abort] \geqslant \Pr\left[\wedge_{i=1}^{q_I} E_i^F \wedge E_F^* \wedge_{i=1}^{q_M} E_i^K \wedge E_K^*\right]$。

由于 $l_{id}(n_{id}+1) < q$、$x', x_i (i = 1, \cdots, n_m)$ 都是不大于 l_{id} 的正整数，因此得出 $0 \leqslant l_{id} k_{id} < q$ 和 $0 \leqslant x' + \sum_{i\in U} x_i < q$。

由 $l_{id}(n_{id}+1) < q$ 假设可知 $F(ID) = 0 \bmod q$ 则得出 $F(ID) = 0 \bmod l_m$，从 $F(m) \neq 0 \bmod l_{id}$ 得出 $F(m) \neq 0 \bmod q$。因此，可以推出 $\Pr\left[E_i^F\right] \geqslant \Pr\left[E_i^F\right]$，及

$$\Pr\left[E_F^*\right] = \Pr\left[F(ID^*) = 0 \bmod q \wedge F(ID^*) = 0 \bmod l_{id}\right]$$

$$= \Pr\left[F(ID^*) = 0 \bmod l_{id}\right] \Pr\left[\begin{array}{l} F(ID^*) = 0 \bmod p \\ \left| F(ID^*) = 0 \bmod l_{id} \right. \end{array}\right]$$

$$= \frac{1}{l_{id}} \frac{1}{n_{id}+1}$$

和

$$\Pr[\wedge_{i=1}^{q_I} E_i^F] \geqslant \Pr\left[\wedge_{i=1}^{q_I} E_i^{'}\right]$$

$$= 1 - \Pr\left[\vee_{i=1}^{q_I} \neg E_i^{'F}\right]$$

$$\geqslant 1 - \sum_{i=1}^{q_I} \Pr\left[\neg E_i^{'F}\right]$$

$$= 1 - \frac{q_{ID}}{l_{id}}$$

$$= 1 - \frac{q_E + q_S + q_{BRS} + q_{RK}}{l_{id}}$$

$$\geqslant 1/2$$

本文用类似的分析技术，得到 $\Pr[E_K^*] \geqslant \frac{1}{2(q_S + q_{BRS})} \frac{1}{n_m + 1}$ 和 $\Pr[\wedge_{i=1}^{q_M} E_i^K] \geqslant 1/2$

通过整理上述结果，可以得到

$$\Pr[\neg abort] \geqslant \Pr[\wedge_{i=1}^{q_I} E_i^F \wedge E_F^* \wedge_{i=1}^{q_M} E_i^K \wedge E_K^*]$$

$$\geqslant \frac{1}{16(q_E + q_S + q_{BRS} + 2q_{RK})(q_S + q_{BRS})(n_{id}+1)(n_m+1)}$$

如果模拟不放弃，就会产生一个伪造的签名概率至少 ε。另外，由于攻击者 B 需要猜测哪一个用户身份 ID 是攻击者 A 要伪造的并且把这个身份的

公钥分配问题的实例元素，而猜测正确的概率为 $1/q_k$。因此，攻击者 B 可以解决 Many-DH 问题实例的概率为

$$\varepsilon' \geqslant \frac{\varepsilon}{16q_K(q_E+q_S+q_{BRS}+2q_{RK})(q_S+q_{BRS})(n_{id}+1)(n_m+1)}$$

(注意：由于 $O_{\text{Secret-Key-Extract}}$ 查询和 $O_{\text{Public-Key-Replace}}$ 查询不导致模拟放弃，因此他们被排除在概率分析。)

6　新方案时间复杂度分析

在 Type I 类型攻击下时间复杂度分析：

攻击者 B 的时间复杂度主要是由乘法和指数运算组成的。

1) $O_{\text{Partial-Secret-Key-Extract}}$ 查询有 $O(n_{id})$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

2) O_{ReKey} 查询有 $O(n_{id})$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

3) O_{Sign} 查询有 $O(n_{id}+n_m)$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

4) $O_{\text{BlindReSign}}$ 查询有 $O(n_{id}+n_m)$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

5) $O_{\text{Public-Key-Broadcast}}$ 查询有 $O(1)$ 复杂度的指数运算；

6) $O_{\text{Secret-Key-Extract}}$ 查询有 $O(1)$ 复杂度的指数运算；

因此，攻击者 B 的时间复杂度为

$$t+O\begin{pmatrix}((q_E+q_{RK})n_{id}+(q_S+q_{BRS})(n_{id}+n_m))\rho\\+(q_k+q_E+q_S+q_{BRS}+q_{RK})\tau\end{pmatrix}$$

其中，ρ 是在群 G_1 的进行乘法是在群 G_1 进行乘法和 τ 为在群 G_1 的进行指数运算的时间。

时间复杂度分析：

攻击者 B 的时间复杂度主要是由乘法和指数运算组成的。

1) O_{ReKey} 查询有 $O(n_{id})$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

2) O_{Sign} 查询有 $O(n_{id}+n_m)$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

3) $O_{\text{BlindReSign}}$ 查询有 $O(n_{id}+n_m)$ 复杂度的乘法运算和 $O(1)$ 复杂度的指数运算；

4) $O_{\text{Public-Key-Broadcast}}$ 查询有 $O(1)$ 复杂度的指数运算；

5) $O_{\text{Secret-Key-Extract}}$ 查询有 $O(1)$ 复杂度的指数运算；

因此，攻击者 B 的时间复杂度为 $t+O\begin{pmatrix}(q_{RK}n_{id}+(q_S+q_{BRS})(n_{id}+n_m))\rho\\+(q_k+q_S+q_{BRS}+q_{RK})\tau\end{pmatrix}$，其中 ρ 是在群 G_1 的进行乘法是在群 G_1 进行乘法和 τ 为在群 G_1 的进行指数运算的时间。

7　结束语

本文分析了现有的基于身份密钥体制的代理重签名及基于传统密钥体制的代理重签名，发现这些方案要么存在密钥托管问题，要么存在证书管理问题。论文结合无证书公钥体制、代理重签名机制和盲签名机制，通过利用双线群，提出了一种有效的无证书盲代理签名方案。方案中解决了密钥托管问题及证书管理带来的额外开销，同时实现了代理者在签名转换中消息隐私特性。基于 NGBDH 和 Many-DH 的困难性，在标准模型下证明了新方案具有能够抵抗伪造攻击的特性。该方案满足正确性和消息盲性，与 Deng 方案相比具有较高的效率。

参考文献：

[1] BLAZE M, BLEUMER G, STRAUSS M. Divertible protocols and atomic proxy cryptography[A]. advances in Cryptology EUROCRYPT'98[C]. Helsinki, Finland, 1998.127-144.

[2] SHAMIR A. Identity based cryptosystems and signature schemes[A]. Advances in Cryptology-Crypto'84[C]. Sahta Baybara, California, USA, 1984.47-53

[3] AL-RIYAMI S S, PATERSON K G. Certificateless public key cryptography[A]. Advances in Cryptography-Asiacryt 2003[C]. Taipei , China, 2003.452-273.

[4] 邓宇乔，杜明辉，尤再来. 一种基于标准模型的盲代理重签名方案[J]. 电子与信息学报, 2010, 32(5): 1219-1223.
DENG Y Q, DU M H, YOU Z L. A blind proxy re-signatures scheme based on standard model[J]. Journal of Electronics & Information Technology, 2010, 32(5): 1219-1223.

[5] CANETTI R, GOLDREICH O, HALEVI S. The random oracle methodology[J]. Journal of the ACM, 2004. 577-594.

[6] CHAUM D. Blind signatures for untraceable payments[A]. Advances in Crypto'82[C]. Plenum, NY, USA, 1982. 199-203.

[7] 张福泰,孙银霞, 张磊等.无证书公钥密码体制研究[J]. 软件学报, 2011,22(6):1316-1332.
ZHANG F T, SUN Y X, ZHANG L, *et al*. Research on certificateless public key cryptography[J]. Journal of Software. 2011,22(6): 1316-1332.

（下转第 78 页）

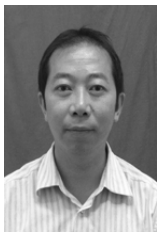
[9] GOKHALE A. Principles for optimizing CORBA internet Inter-ORB protocol performance system sciences[A]. Proceedings of the Thirty-First Hawaii International Conference[C]. Hawai, USA, 1998. 376-385.

[10] SENIVONGSE T, SURIYENTRAKORN P. A CORBA-based architecture for service change notification[A]. IEEE Enterprise Distributed Object Computing Conference[C]. Seattle, WA, USA, 2001. 22-33.

[11] MA C, BACON J. CORBA:a CORBA-based event architecture[A]. Proceedings of the 4th USENIX Conference on Object-Oriented Technologies and Systems[C]. Santa Fe, New Mexico, USA, 1998. 21-24

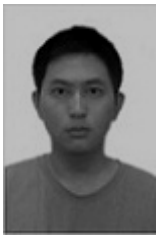
[12] 李祎.J2EE 平台下消息中间件及其安全性的研究[D]. 武汉: 武汉理工大学，2007
LI Y. The Study on Message Middleware and Its Security in J2EE platform[D]. Wuhan: Wuhan University of Technology,2007.

作者简介：



陈文宇（1968-），男，浙江兰溪人，博士，电子科技大学教授，主要研究方向为软件理论、模式识别、编译技术。

曾茹（1986-），女，四川宜宾人，电子科技大学硕士生，主要研究方向为软件理论、计算理论、编译技术。



皮维（1987-），男，重庆人，电子科技大学硕士生，主要研究方向为软件理论、模式识别、编译技术。

李文（1987-），男，四川资阳人，电子科技大学硕士生，主要研究方向为软件理论、模式识别、编译技术。

（上接第 69 页）

[8] ATENIESE G, HOHENBERGER S. Proxy re-signatures:new definitions algorithms, and applications[A]. ACM CCS 2005[C]. New York, NY, USA, 2005.310-319.

[9] SHAO J, CAO Z F, WANG L C, *et al.* Proxy re-signature schemes without random Oracles[A]. INDOCRYPT 2007[C]. Chennai, India, 2007.197-209.

作者简介：

冯涛（1970-），男，甘肃临洮人，博士，兰州理工大学计算机与通信学院副院长、研究员，主要研究方向为可证明安全协议理论、无线和移动网络安全。

梁一鑫（1980-），男，江苏扬州人，硕士，兰州理工大学计算机与通信学院讲师，主要研究方向网络安全、代理重签名。