标准模型下的服务器辅助验证代理重签名方案

杨小东* 李亚楠 高国娟 王彩芬 鲁小勇
(西北师范大学计算机科学与工程学院 兰州 730070)

摘 要：代理重签名具有转换签名的功能，在云存储、数据交换、跨域身份认证等领域有广泛的应用前景。目前大多数代理重签名方案需要复杂的双线性对运算，无法适用于计算能力较弱的低端计算设备。为了提高代理重签名的签名验证效率，该文给出了双向服务器辅助验证代理重签名的安全性定义，并提出一个高效的服务器辅助验证代理重签名方案，在标准模型下证明新方案在合谋攻击和选择消息攻击下是安全的。分析结果表明，新方案有效减少了双线性对的计算量，大大降低了签名验证算法的计算复杂度，在效率上优于已有的代理重签名方案。

关键词：密码学；服务器辅助验证代理重签名；合谋攻击；不可伪造性；标准模型
中图分类号：TP309 **文献标识码**：A **文章编号**：1009-5896(2016)05-1151-07
DOI: 10.11999/JEIT150966

Sever-aided Verification Proxy Re-signature Scheme in the Standard Model

YANG Xiaodong LI Yanan GAO Guojuan WANG Caifen LU Xiaoyong
(*College of Computer Science and Engineering*, *Northwest Normal University*, *Lanzhou* 730070, *China*)

Abstract: Proxy re-signature has the function of converting signature, and has extensive application prospects, such as cloud storage, data exchange, cross-domain identity authentication and so on. However, most proxy re-signature schemes require expensive bilinear pairing operations, which are not suitable for low-power devices. To improve the performance of proxy re-signature schemes, the security model of a bidirectional sever-aided verification proxy re-signature is presented. Furthermore, a sever-aided verification proxy re-signature scheme is proposed. This scheme is proven to be secure under collusion attacks and adaptive chosen message attacks in the standard model. Analysis results show that the proposed scheme effectively reduces the computation cost of pairing operation, and it greatly reduces computational complexity of signature verification algorithm. The proposed scheme is more efficient than the existing proxy re-signature schemes.

Key words: Cryptography; Sever-aided verification proxy re-signature; Collusion attack; Unforgeability; Standard model

1 引言

代理重签名是现代密码学的一个热点研究方向。 在代理重签名体制中，受托者(delegatee)生成消息的原始签名，一个半可信代理者将原始签名转换为委托者(delegator)对同一个消息的有效签名，但代理者无法获得受托者或委托者的签名密钥[1]。代理重签名具有签名转换的功能，可以实现云存储环境下用户数据的完整性证明、跨域透明认证、提供遍历的路径证明、分散代理的签名权利、简化证书管理等[2−4]。

文献[1]首次提出了代理重签名的概念，但没有给出形式化的安全性定义。文献[5]首次提出了代理重签名的安全模型，并构造了一个双向多用方案和一个单向单用方案，在随机预言模型下给出了严格的安全性证明。文献[6]进一步完善了代理重签名的安全属性定义，基于文献[7]构造了第 1 个标准模型下安全的代理重签名方案；但文献[8]发现该方案存在安全缺陷，并提出了一个改进方案。文献[9]提出了一个单向多用的代理重签名方案，但签名的验证

收稿日期：2015-08-20；改回日期：2016-01-04；网络出版：2016-03-30
*通信作者：杨小东 y200888@163.com
基金项目：国家自然科学基金(61262057, 61063041)，甘肃省科技计划 (145RJDA325, 1308RJYA039)，国家档案局科技项目(2014-X-33)，兰州市科技计划项目(2013-4-22，2014-1-256)，甘肃省高等学校科研项目(2015A-011)，西北师范大学青年教师科研能力提升计划(NWNU-LKQN-13-23)
Foundation Items: The National Natural Science Foundation of China (61262057, 61063041), Science and Technology Project of Gansu Province of China (145RJDA325, 1308RJYA039), Science and Technology Foundation of National Archives of China (2014-X-33), Science and Technology Project of Lanzhou (2013-4-22, 2014-1-256), Gansu Province Higher Educational Science and Technology Program (2015A-011), Youth Foundation of Northwest Normal University (NWNU-LKQN-13-23)

开销随着重签名层数的增加呈线性增长。近年来，代理重签名引起了广泛的关注，一些具有特殊性质的代理重密码方案及其应用被提出，比如基于身份的代理重签名[10]、门限代理重签名[11,12]、基于格的代理重签名[13,14]等。随着云计算和大数据的迅速发展，具有强大计算能力的云服务提供商将成为代理重签名方案中的代理者，低端计算设备(如智能手机、无线传感器)是非常重要的云计算终端[15]。这些设备具有较弱的计算能力、有限的能源供应和较短的响应时间。然而，目前大部分代理重签名方案的签名验证算法需要复杂的双线性运算，无法适用于计算能力较弱的低端计算设备。因此，如何降低代理重签名方案的签名验证计算量，是一个非常有意义且非常迫切的研究课题。

服务器辅助验证代理重签名将大部分签名验证的计算任务转移给一个计算能力很强的服务器执行，大大降低了验证者的计算量，非常适用于低端计算设备。文献[16]给出了服务器辅助验证代理重签名的安全模型，但该模型只考虑了服务器与合法代理者的合谋攻击。同时，文献[16]构造了两个单向服务器辅助验证代理重签名方案，但这两个方案是在随机预言模型中可证明安全的，并且第 2 个方案无法抵抗服务器和非法代理者的合谋攻击。文献[17]指出在随机预言模型下安全的方案，在现实中不一定是安全的。为了克服这些安全缺陷，很有必要研究标准模型下可证明安全的服务器辅助验证代理重签名方案。本文结合服务器辅助验证签名和代理重签名的安全属性，给出了双向服务器辅助验证代理重签名的形式化安全模型。基于改进的 SHAO 方案[8]，构造了一个能有效抵抗合谋攻击的双向服务器辅助验证代理重签名方案。验证者通过与服务器执行服务器辅助验证协议，以较小的计算代价完成签名的验证，提高了签名的验证速度。新方案的签名验证算法减少了复杂的双线性对运算，与现有的同类方案相比，验证者具有更低的计算时间开销，可应用于签名验证时间受限或计算能力有限的设备，如无线传感器、智能手机、PDA 等。通过对相关领域的数据搜索，目前还没有关于标准模型下的服务器辅助验证代理重签名研究的公开文献。

2 预备知识

2.1 双线性映射

令p是一个大素数，G_1和G_2是两个阶为p的循环群，g是G_1的一个生成元，$e:G_1\times G_1\to G_2$是满足以下条件的双线性映射。

(1)双线性：对任意的$a,b\in Z_p^*$，满足$e(g^a,g^b)=e(g,g)^{ab}$。

(2)非退化性：存在$g,h\in G_1$，使得$e(g,h)\neq 1$。

(3)可计算性：存在一个有效的算法$e(g,h)$，其中$g,h\in G_1$。

2.2 CDH 假设

定义 1 (CDH 问题) 计算性 CDH (Computational Diffie-Hellman) 问题：对于任意的$a,b\in Z_p^*$，已知$(g,g^a,g^b)\in G_1^3$，计算$g^{ab}\in G_1$。

定义 2 (CDH 假设) 不存在一个算法在多项式时间内以不可忽略的概率求解群G_1中的 CDH 问题。

3 双向服务器辅助验证代理重签名的安全性定义

一个双向服务器辅助验证代理重签名方案 SAVPRS=(setup, keygen, rekey, sign, resign, verify, SAV-setup, SAV-verify)包括如下 8 个算法。

(1)系统参数生成算法 setup $(1^\eta)\to$ cp：给定安全参数η，运行该算法生成公开参数 cp。

(2)密钥生成算法 keygen (cp)$\to$(pk,sk)：给定系统参数 cp，生成用户的公钥/私钥对(pk,sk)。

(3)重签名密钥生成算法 rekey ($\mathrm{pk_A}$,$\mathrm{sk_A}$,$\mathrm{pk_B}$,$\mathrm{sk_B}$)：给定受托者 Alice 和委托者 Bob 的公私钥对($\mathrm{pk_A}$,$\mathrm{sk_A}$)和($\mathrm{pk_B}$,$\mathrm{sk_B}$)，利用安全的通信信道为代理者生成一个重签名密钥$\mathrm{rk_{A\to B}}$。

(4)签名生成算法 Sign (m,$\mathrm{sk_A}$)：给定一个消息m和受托者 Alice 的私钥$\mathrm{sk_A}$，生成一个对应于公钥$\mathrm{pk_A}$的消息m的原始签名σ。

(5)重签名生成算法 resign ($\mathrm{rk_{A\to B}}$,m,$\mathrm{pk_A}$,σ_A)：给定重签名密钥$\mathrm{rk_{A\to B}}$，受托者 Alice 的公钥$\mathrm{pk_A}$，消息m，受托者对m的签名σ_A，若 Verify ($\mathrm{pk_A}$, m, σ_A)=0，输出$\perp$；否则，生成一个对应于委托者公钥$\mathrm{pk_B}$的消息m的重签名σ_B。

(6)签名验证算法 verify (m,pk,σ)：给定公钥 pk，消息m和签名σ，若σ是对应于 pk 的消息m的有效签名，输出 1；否则，输出 0。

(7)服务器辅助验证参数生成算法 SAV-setup (cp)：给定参数 cp，为验证者生成一个字符串 VString。

(8)服务器辅助验证协议 SAV-verify (VString, m,pk,σ)：对于字符串 VString，公钥 pk 和消息签名对(m,σ)，如果服务器让验证者确信σ是一个有效签名，输出 1；否则，输出 0。由于签名验证者的计算能力较弱，不能执行复杂的密码运算，因此可通过与服务器之间的交互协议，借助服务器来完成签名的验证。

定义 3　令 Φ-Verify 和 Φ-SAV-Verify 分别表示验证者在 Verify 算法和 SAV-Verify 协议中的计算量，如果 Φ-SAV-Verify < Φ-Verify，则称 SAVPRS 方案是节约计算的。

服务器辅助验证代理重签名的安全性，至少包括普通代理重签名的存在不可伪造性和服务器辅助验证协议 SAV-Verify 的完备性。存在不可伪造性保证攻击者不能生成一个新消息的合法签名，完备性保证服务器不能让验证者确信一个非法签名是合法的。由于服务器能获得任何消息的原始签名或重签名，因此无法给出一个统一的安全定义来刻画代理重签名的存在不可伪造性和服务器辅助验证协议的完备性。

文献[5,6]已经定义了代理重签名的存在不可伪造性，但在文献[16]给出的服务器辅助验证代理重签名的完备性定义中，仅考虑了服务器与合法代理者的合谋，未考虑服务器与非法代理者的合谋。由于双向代理重签名允许受托者和委托者的角色相互转换，但不允许代理者与受托者(或委托者)之间的合谋，因此双向服务器辅助验证代理重签名不允许代理者、服务器、受托者(或委托者)三者之间的合谋。鉴于此，针对双向服务器辅助验证代理重签名方案的合谋攻击可分为两类：一类是针对原始签名的受托者和服务器的合谋攻击，另一类是针对重签名的代理者和服务器的合谋攻击。下面基于文献[16]的完备性定义，通过挑战者和攻击者之间的两个游戏，定义在合谋攻击和自适应性选择消息攻击下双向服务器辅助验证代理重签名的完备性。

在第 1 个游戏 **Game1** 中，由于允许服务器和受托者合谋，因此攻击者拥有受托者的私钥，能生成任意消息的原始签名。经过与挑战者的有限次服务器辅助验证询问后，攻击者的目标是让验证者确信一个非法的原始签名是合法的。在第 2 个游戏 **Game2** 中，由于允许服务器和代理者合谋，因此攻击者拥有代理者的重签名密钥，能转换任意消息的原始签名为相应的重签名。经过与挑战者的有限次服务器辅助验证询问后，攻击者的目标是让验证者确信一个非法的重签名是合法的。验证者的字符串 VString 对攻击者是保密的，如果攻击者赢得了第 1 个游戏，说明验证者无法判断攻击者提供的原始签名是否合法，从而表明签名方案无法抵抗服务器和受托者的合谋攻击；如果攻击者赢得了第 2 个游戏，说明验证者无法判断攻击者提供的重签名是否合法，从而表明签名方案无法抵抗服务器和代理者的合谋攻击。因此，如果攻击者能赢得任何一个游戏，则表明签名方案不满足完备性。两个游戏的具体描述如下：

Game1：在这个游戏中，攻击者 $\mathcal{A}_1$ 拥有受托者的公私钥对 $(\mathrm{pk_A}, \mathrm{sk_A})$，可以代表受托者生成任意一个消息的原始签名。

建立：挑战者 C 首先运行 setup, keygen 和 SAV-setup 3 个算法，获得系统参数 cp，受托者的公私钥对 $(\mathrm{pk_A}, \mathrm{sk_A})$ 和字符串 VString，然后将 cp 和 $(\mathrm{pk_A}, \mathrm{sk_A})$ 发送给 $\mathcal{A}_1$。

查询：攻击者 $\mathcal{A}_1$ 可以自适应性地进行 q_s 次服务器辅助验证询问。对于每次询问 $(m_i, \boldsymbol{\sigma}_i)$，挑战者 C 扮演验证者，攻击者 $\mathcal{A}_1$ 扮演服务器，C 和 $\mathcal{A}_1$ 首先执行 SAV-Verify 协议，然后将输出的结果作为响应返回给 $\mathcal{A}_1$。

输出：攻击者 $\mathcal{A}_1$ 输出一个消息 m^* 和字符串 σ^*，令 Ω_{m^*} 是 m^* 的所有合法签名集合，$\sigma^* \notin \Omega_{m^*}$。如果 Verify$(m^*, \mathrm{pk_A}, \sigma^*)$=0 且 SAV-Verify(VString, m^*, $\mathrm{pk_A}, \sigma^*$)=1，即攻击者 $\mathcal{A}_1$ 让挑战者 C 确信 σ^* 是 m^* 对应于公钥 $\mathrm{pk_A}$ 的合法签名，则攻击者 $\mathcal{A}_1$ 获胜。

Game2：在这个游戏中，攻击者 $\mathcal{A}_2$ 拥有受托者和委托者之间的重签名密钥 $\mathrm{rk_{A\to B}}$，可以代表代理者将消息的原始签名转换为同一个消息的重签名。

建立：挑战者 C 运行 setup, keygen, rekey 和 SAV-setup 4 个算法，得到系统参数 cp、受托者和委托者的公私钥对 $(\mathrm{pk_A}, \mathrm{sk_A})$ 和 $(\mathrm{pk_B}, \mathrm{sk_B})$、重签名密钥 $\mathrm{rk_{A\to B}}$ 及字符串 VString，并将 cp, $\mathrm{pk_A}$, $\mathrm{pk_B}$ 和 $\mathrm{rk_{A\to B}}$ 发送给 $\mathcal{A}_2$。

查询：攻击者 $\mathcal{A}_2$ 可以自适应性地进行 q_v 次服务器辅助验证询问，与 Game1 中的应答方式相同。

输出：攻击者 $\mathcal{A}_2$ 最后输出一个消息 m^* 和字符串 σ^*，令 Ω_{m^*} 是 m^* 对应于公钥 $\mathrm{pk_B}$ 的所有合法签名集合，$\sigma^* \notin \Omega_{m^*}$。如果 Verify $(m^*, \mathrm{pk_B}, \sigma^*)$=0 且 SAV-Verify (VString, m^*, $\mathrm{pk_B}$, σ^*)=1，即攻击者 $\mathcal{A}_2$ 让挑战者 C 确信 σ^* 是 m^* 的合法签名，则攻击者 $\mathcal{A}_2$ 获胜。

攻击者赢得上述游戏的概率完全取决于挑战者和攻击者之间的抛币概率。

定义 4　如果攻击者在上述两个游戏中获胜的概率是可忽略的，则称双向服务器辅助验证代理重签名方案中的 SAV-Verify 协议是完备的。

定义 5　如果一个代理重签名方案在自适应选择消息攻击下是存在不可伪造的，服务器辅助验证协议是完备的，则称相应的服务器辅助验证代理重签名方案在合谋攻击和选择消息攻击下是安全的[16]。

4　新的服务器辅助验证代理重签名方案

本节基于改进的 SHAO 方案[8]，构造一个安全高效的双向服务器辅助验证代理重签名方案。新方

案的参与实体由受托者、委托者、验证者、代理者和服务器组成，受托者负责生成消息的原始签名，半可信代理者将原始签名转换为委托者的重签名，验证者在半可信服务器的协助下完成签名的有效性验证。

(1)系统参数生成算法(setup)：令 p 是一个大素数，G_1 和 G_2 是两个阶为 p 的循环群，选取 G_1 的一个生成元 g 和双线性配对映射 $e: G_1 \times G_1 \to G_2$。在群 G_1 中随机选取 n_m+2 个元素 $(g_2, u, u_1, \cdots, u_{n_m})$，公开系统参数 $\mathrm{cp}=(G_1, G_2, p, e, g, g_2, u, \{u_i\}_{i=1}^{n_m})$。

(2)密钥生成算法(keygen)：给定系统参数 cp，随机选取 $a \in Z_p^*$，生成用户的公钥/私钥对 (pk,sk) $=(e(g_2, g^a), a)$。

(3)重签名密钥生成算法(rekey)：给定受托者 Alice 和委托者 Bob 的私钥 $(\mathrm{sk}_\mathrm{A}=\alpha, \mathrm{sk}_\mathrm{B}=\beta)$，采用与文献[5]相同方法为半可信的代理者生成一个重签名密钥 $\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}=\beta/\alpha(\mathrm{mod}\, p)$。假设受托者、委托者和代理者之间有安全的通信信道，生成重签名密钥的具体过程如下：

(a)代理者随机选取 $k \in Z_p^*$，并将 k 发送给受托者 Alice；

(b)受托者 Alice 通过私钥 $\mathrm{sk}_\mathrm{A}=\alpha$ 计算 $k_1=\mathrm{sk}_\mathrm{A} k=k\alpha(\mathrm{mod}\, p)$，然后将 k_1 发送给委托者 Bob；

(c)委托者 Bob 通过私钥 $\mathrm{sk}_\mathrm{B}=\beta$ 计算 $k_2=\mathrm{sk}_\mathrm{B}/k_1=\beta/k\alpha(\mathrm{mod}\, p)$，并将 k_2 发送给代理者；

(d)代理者收到 k_2 后，计算受托者 Alice 与委托者 Bob 之间重签名密钥 $\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}=k_2 k=\beta/\alpha(\mathrm{mod}\, p)$。

(4)签名生成算法(sign)：给定受托者的私钥 $\mathrm{sk}_\mathrm{A}=\alpha$ 和一个 n_m bit 长的消息 $m=(m_1, \cdots, m_{n_m}) \in \{0,1\}^{n_m}$，输出一个对应于公钥 pk_A 的消息 m 的原始签名 $\sigma_\mathrm{A}=(\sigma_{\mathrm{A},1}, \sigma_{\mathrm{A},2})=(g_2^\alpha \varpi^t, g^t)$，这里 $t \in_R Z_p$ 且 $\varpi=u\prod_{i=1}^{n_m}(u_i)^{m_i}$。

(5)重签名生成算法(resign)：给定重签名密钥 $\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}$，一个 n_m bit 长的消息 $m=(m_1, \cdots, m_{n_m}) \in \{0,1\}^{n_m}$，受托者的公钥 pk_A 和签名 $\sigma_\mathrm{A}=(\sigma_{\mathrm{A},1}, \sigma_{\mathrm{A},2})$，如果 Verify $(\mathrm{pk}_\mathrm{A}, m, \sigma_\mathrm{A})=0$，则原始签名无效，输出 $\perp$；否则，随机选取 $\tilde{r} \in Z_p^*$，输出一个对应于公钥 pk_B 的消息 m 的重签名 $\sigma_\mathrm{B}=(\sigma_{\mathrm{B},1}, \sigma_{\mathrm{B},2})=((\sigma_{\mathrm{A},1})^{\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}}\varpi^{\tilde{r}}, (\sigma_{\mathrm{A},2})^{\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}} g^{\tilde{r}})$。

(6)签名验证算法(verify)：验证者收到消息 m 的签名 $\sigma=(\sigma_1, \sigma_2)$ 后，利用公钥 pk 计算等式 $e(\sigma_1, g)=\mathrm{pk}\cdot e(\varpi, \sigma_2)$ 是否成立。如果相等，说明签名有效，输出 1；否则，输出 0。

(7)服务器辅助验证参数生成算法(SAV-setup)：给定系统参数 cp，验证者随机选取一个元素 $x \in Z_p^*$，令字符串 $\mathrm{VString}=x$。

(8)服务器辅助验证协议(SAV-verify)：给定 $\mathrm{VString}=x$，一个公钥 pk 和一个消息签名对 $(m, \sigma=(\sigma_1, \sigma_2))$，验证者和服务器之间的服务器辅助验证交互协议如下：

(a)验证者计算 $\sigma'=(\sigma_1', \sigma_2')=((\sigma_1)^x, (\sigma_2)^x)$，将 (m, σ') 发送给服务器。

(b)服务器计算 $K_1=e(\sigma_1', g)$ 和 $K_2=e(\varpi, \sigma_2')$，将 (K_1, K_2) 发送给验证者。

(c)验证者计算等式 $K_1=(\mathrm{pk})^x K_2$ 是否成立。如果等式成立，验证者确信 σ 是消息 m 的合法签名，输出 1；否则，验证者确信 σ 是无效签名，输出 0。

5 安全性证明与有效性分析

5.1 正确性分析

对于委托者的签名 $\sigma_\mathrm{B}=(\sigma_{\mathrm{B},1}, \sigma_{\mathrm{B},2})=(g_2^\beta \varpi^r, g^r)$ 和字符串 $\mathrm{VString}=x$，则有

$$\begin{aligned} K_1 &= e\left(\sigma_{\mathrm{B},1}', g\right)=e\left(\left(\sigma_{\mathrm{B},1}\right)^x, g\right)=e\left(\left(g_2^\beta \varpi^r\right)^x, g\right) \\ &= e\left(\left(g_2^\beta, g\right)^x e\left(\varpi^{rx}, g\right)\right)=e\left(\left(g_2, g^\beta\right)^x e\left(\varpi, g^{rx}\right)\right) \\ &= (pk_\mathrm{B})^x e\left(\varpi, g^{rx}\right)=(pk_\mathrm{B})^x e\left(\varpi, (g^r)^x\right) \\ &= (pk_\mathrm{B})^x e\left(\varpi, (\sigma_{\mathrm{B},2})^x\right)=(pk_\mathrm{B})^x e\left(\varpi, \sigma_{\mathrm{B},2}'\right)=(\mathrm{pk}_\mathrm{B})^x K_2 \end{aligned}$$

以上推导过程证明了服务器辅助验证协议的正确性。因为原始签名与重签名的长度相同，所以本方案满足透明性和多用性。由于重签名密钥 $\mathrm{rk}_{\mathrm{A}\to\mathrm{B}}=\beta/\alpha=1/\mathrm{rk}_{\mathrm{B}\to\mathrm{A}}$，所以本方案满足双向性。受托者的私钥、委托者的私钥和代理者的重签名密钥，都是 Z_p^* 的一个元素，因此新方案满足密钥最优性。

5.2 安全性分析

针对文献[18]提出的服务器辅助验证签名方案的安全缺陷，文献[19]给出了两类服务器和签名者合谋的攻击方案。但在本文新方案的服务器辅助验证协议 SAV-verify 中，验证者利用字符串 VString 对签名 $\sigma=(\sigma_1, \sigma_2)$ 进行了幂运算处理，使得服务器无法恢复出 VString，因而可有效抵抗这两类攻击[19]。本文提出的新方案基于改进的 SHAO 方案[8]，而文献[8]在标准模型下证明该方案满足存在不可伪造性，其安全性可归约到 CDH 假设。由第 3 节的定义 5 可知，为了证明新方案的安全性，只需证明新方案的服务器辅助验证协议 SAV-verify 满足完备性。

引理 1 假定 $\mathcal{A}_1$ 代表服务器和受托者合谋的攻击者，则 $\mathcal{A}_1$ 让挑战者 C 确信一个非法原始签名是合法的概率是可忽略的。

证明 令 $\mathcal{A}_1$ 扮演 SAV-Verify 协议中服务器的角色，C 扮演 SAV-Verify 协议中验证者的角色。给定一个消息的非法原始签名后，$\mathcal{A}_1$ 的任务是让 C 确信这个

非法签名是合法的。挑战者 C 与攻击者 $\mathcal{A}_1$ 的交互过程如下：

建立：挑战者 C 执行 Setup 算法生成系统参数 cp，随机选取 2 个元素 $x^*, a \in Z_p^*$，令 $\text{VString} = x^*$，计算受托者的公私钥对 $(\text{pk}_\text{A}, \text{sk}_\text{A}) = (e(g_2, g^a), a)$，将 $\{\text{cp}, \text{pk}_\text{A}, \text{sk}_\text{A}\}$ 发送给攻击者 $\mathcal{A}_1$。

查询：攻击者 $\mathcal{A}_1$ 可以自适应性地进行有限次服务器辅助验证询问。对于每次询问 (m_i, σ_i)，挑战者 C 和 $\mathcal{A}_1$ 执行 SAV-verify 协议，然后将协议的输出结果作为响应返回给 $\mathcal{A}_1$。

输出：攻击者 $\mathcal{A}_1$ 最后输出消息 m^* 和字符串 $\sigma^* = (\sigma_1^*, \sigma_2^*)$，令 Ω_{m^*} 是消息 m^* 对应于公钥 pk_A 的所有合法签名的集合，$\sigma^* \notin \Omega_{m^*}$。挑战者 C 收到 (m^*, σ^*) 后，利用 VString 计算 $(\sigma^*)' = ((\sigma_1^*)', (\sigma_2^*)') = ((\sigma_1^*)^{x^*}, (\sigma_2^*)^{x^*})$，将 $(\sigma^*)' = ((\sigma_1^*)', (\sigma_2^*)')$ 发送给攻击者 $\mathcal{A}_1$。$\mathcal{A}_1$ 然后计算 $K_1^* = \text{e}((\sigma_1^*)', g)$ 和 $K_2^* = e(\varpi^*, (\sigma_2^*)')$，并将 (K_1^*, K_2^*) 返回给 C。下面分析等式 $K_1^* = (\text{pk}_\text{A})^{x^*} K_2^*$ 成立的概率是 $1/(p-1)$。

(1)由于 $(\sigma^*)' = (\sigma^*)^{x^*}$ 且 $x^* \in_R Z_p^*$，因此攻击者 $\mathcal{A}_1$ 通过 σ^* 成功伪造 $(\sigma^*)'$ 的概率是 $1/(p-1)$。

(2)假设攻击者 $\mathcal{A}_1$ 返回 (K_1^*, K_2^*)，使得 $K_1^* = (\text{pk}_\text{A})^{x^*} K_2^*$，则有

$$\log_{\text{pk}^*} K_1^* = x^* + \log_{\text{pk}_\text{A}} K_2^*。$$

因为 x^* 是从 Z_p^* 中随机选取的，所以攻击者 $\mathcal{A}_1$ 寻找 x^* 使得上述等式成立的概率是 $1/(p-1)$。综上所述，攻击者 $\mathcal{A}_1$ 是让挑战者 C 确信 (m^*, σ^*) 是合法签名概率是 $1/(p-1)$。由于 p 是一个大素数，因此 $\mathcal{A}_1$ 让挑战者 C 确信一个非法原始签名是合法的概率是可忽略的。　　证毕

引理 2　假定 $\mathcal{A}_2$ 代表服务器和代理者合谋的攻击者，则 $\mathcal{A}_2$ 让挑战者 C 确信一个非法重签名是合法的概率是可忽略的。

证明　令 $\mathcal{A}_2$ 扮演 SAV-Verify 协议中服务器的角色，C 扮演 SAV-verify 协议中验证者的角色。给定一个消息的非法重签名后，$\mathcal{A}_2$ 的任务是让 C 确信这个非法签名是合法的。挑战者 C 与攻击者 $\mathcal{A}_2$ 的交互过程如下：

建立：挑战者 C 运行 Setup 算法得到系统参数 cp，随机选取 3 个元素 $x^*, a, b \in Z_p^*$，令 $\text{VString} = x^*$，计算受托者的公私钥对 $(\text{pk}_\text{A}, \text{sk}_\text{A}) = (e(g_2, g^a), a)$ 和委托者的公私钥对 $(\text{pk}_\text{B}, \text{sk}_\text{B}) = (e(g_2, g^b), b)$，以及一个重签名密钥 $\text{rk}_{\text{A}\to\text{B}} = b/a (\text{mod}\, p)$，将 cp，$\text{pk}_\text{A}$，$\text{pk}_\text{B}$ 和 $\text{rk}_{\text{A}\to\text{B}}$ 发送给攻击者 $\mathcal{A}_2$。

查询：与引理 1 中的询问应答过程相同。

输出：最后攻击者 $\mathcal{A}_2$ 输出消息 m^* 和字符串 $\sigma^* = (\sigma_1^*, \sigma_2^*)$，令 Ω_{m^*} 是消息 m^* 对应于公钥 pk_B 的所有合法签名的集合，$\sigma^* \notin \Omega_{m^*}$。与引理 1 中的分析过程相似，攻击者 $\mathcal{A}_2$ 是让挑战者 C 确信 (m^*, σ^*) 是合法签名概率是 $1/(p-1)$，即 $\mathcal{A}_2$ 让挑战者 C 确信一个非法重签名是合法的概率是可忽略的。　　证毕

由定义 4，引理 1 和引理 2 可推导出定理 1。

定理 1　新方案的服务器辅助验证协议 SAV-verify 在合谋攻击和自适应性选择消息攻击下是完备的。

定理 2　在标准模型下，改进的 SHAO 方案在自适应性选择消息下是存在不可伪造的[8]。

根据定义 5、定理 1 和定理 2，可得如下定理 3。

定理 3　在标准模型下，本文提出的双向服务器辅助验证代理重签名方案在合谋攻击和自适应选择消息攻击下是安全的。

5.3 有效性分析

为了表述方便，用 WANG-SAVPRS-1 方案和 WANG-SAVPRS-2 方案分别表示文献[16]提出的第 1 个和第 2 个服务器辅助验证代理重签名方案；用 SHAO 方案表示文献[6]提出的第 1 个标准模型下的双向代理重签名方案，改进的 SHAO 方案表示文献[8]提出的改进代理重签名方案。假定所有方案选择相同长度的素数 p，以及相同阶的群 G_1 和 G_2。由于乘法、加法以及哈希函数的计算量比较小，因此计算开销仅考虑计量算较大的双线性对和幂运算。下面将 SHAO 方案[6]、改进的 SHAO 方案[8]、WANG-SAVPRS-1 方案[16]、WANG-SAVPRS-2 方案[16]与本文新方案的验证者计算开销进行比较，其结果如表 1 所示。

表 1 验证者的计算开销比较

方案	G_1 中的幂运算	G_2 中的幂运算	双线性对运算	标准模型
SHAO 方案[6]	0	0	3	是
改进的 SHAO 方案[8]	0	0	3	是
WANG-SAVPRS-1 方案[16]	2	1	1	否
WANG-SAVPRS-2 方案[16]	1	1	0	否
本文新方案	2	1	0	是

从表 1 可以看出，在本文提出的新方案中，签名验证算法 Verify 需要 2 个双线性对运算，服务器辅助验证协议 SAV-Verify 仅需执行 3 次幂运算，因此新方案是节约计算的。

由于在本文新方案中，验证者通过服务器辅助验证协议将复杂的双线对计算任务转移给服务器执行，因此验证签名时不需要执行计算量很大的双线性对运算。新方案比 WANG-SAVPRS-2 方案[16]多一次幂运算，但新方案是在标准模型下可证明安全的，并且能有效抵抗合谋攻击[19]，具有更高的安全性。与其他 3 个方案相比，验证者在新方案中具有较低的计算时间开销，能以较小的计算代价完成签名的验证。

由于 SHAO 方案[6]与改进的 SHAO 方案[8]具有相同的签名验证算法，下面仅对本文新方案与 SHAO 方案中验证者的时间开销、验证效率与不同数量级长度的签名消息进行实验分析比较，结果如图 1 和图 2 所示。本次实验运行的硬件环境：CPU 为英特尔酷睿 i7-3770 处理器，主频 3.4 GHz，内存 8 GB；软件环境：64 位的 Windows 7 操作系统，Microsoft Virtual PC 和 PBC-0.4.7-VC。选取 PBC 库自带的类型 A 配对，其中基域 q 的阶为 512 bit，素数 p 的阶为 1024 bit，配对构造在有限域 F_q 中椭圆曲线 $y^2 = x^3 + x$ 上，群 G_1 的生成元 g 为点 (5642230551766631144183492372235849714883088637656482766155437411313171234436297973495214557682097275064172063196724838464185930350849865592007824116003187, 2455553984234305369666618352851565100398220941107533453314835887815996373594693584530588487829065962570984905233456553133966262727853389232334228115638311)，受托者的私钥 sk_A 为 19950382392086678948357536379043009033900478635 5，委托者私钥 sk_B 为 53959389856738787801917629826326997076765366645 1。

从图 1 可知，对于相同长度的签名消息，验证者在新方案中的计算时间开销低于 SHAO 方案。由于验证者在 SHAO 方案中执行 3 次双线对，所以当签名消息长度增大时，验证者的签名验证时间开销增速比较快。但在新方案中，服务器执行了双线对运算，验证者仅需要执行 3 次幂运算，因此当签名消息长度增大时，验证者的签名验证时间开销增速比较慢。图 2 表明新方案大大减少了验证者的计算开销，降低了签名验证的时间；与 SHAO 方案相比，新方案的验证效率至少提高了 62%。

6 结束语

随着云计算和大数据的重要性越来越突出，智能手机等低端计算设备将是非常重要的云计算终端。为了研究适用于低端计算设备的代理重签名方案，结合代理重签名和服务器辅助验证签名体制，本文提出了双向服务器辅助验证代理重签名的形式化安全模型，构造了一个具体的实现方案，并在新的安全模型中给出了安全性证明。验证者在新方案的服务器辅助验证协议中，不需要执行复杂的双线性对运算，极大地节约了验证者的计算量，提高了签名验证效率，很好地满足了低端计算设备的计算能力弱、能源供应有限的需求。下一步的工作是设计高效的基于身份的服务器辅助验证代理重签名方案。

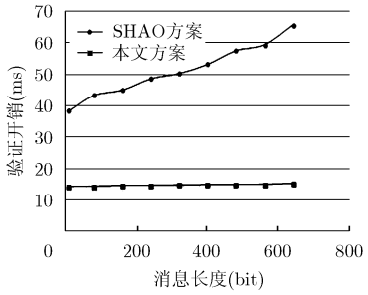

图 1 验证者的时间开销与消息长度关系图

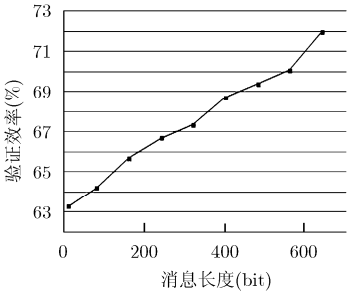

图 2 验证效率与消息长度关系图

参 考 文 献

[1] BLAZE M, BLEUMER G, and STRAUSS M. Divertible protocols and atomic proxy cryptography[C]. Proceedings of EUROCRYPT'98, Helsinki, Finland, 1998: 127–144. doi: 10.1.1.81.8246.

[2] HAO S G, ZHANG L, and MUHAMMAD G. A union authentication protocol of cross-domain based on bilinear pairing[J]. *Journal of Software*, 2013, 8(5): 1094–1100. doi: 10.4304/jsw.8.5.1094-1100.

[3] NGUYEN T C, SHEN W, LUO Z, *et al*. Novel Data Integrity Verification Schemes in Cloud Storage[M]. Switzerland: Springer International Publishing, 2015: 115–125. doi: 10.1007/BFb0054122.

[4] 孙奕, 陈性元, 杜学绘, 等. 一种用于流交换的代理重签名方案[J]. 软件学报, 2015, 26(1): 129–144. doi: 10.13328/j.cnki.jos.004553.
SUN Yi, CHEN X Y, DU X H, *et al*. Proxy re-signature scheme for stream exchange[J]. *Journal of Software*, 2015, 26(1): 129–144. doi: 10.13328/j.cnki.jos.004553.

[5] ATENIESE G and HOHENBERGER S. Proxy re-signatures: new definitions, algorithms, and applications[C]. Proceedings of the 12th ACM CCS, Alexandria, USA, 2005: 310–319. doi: 10.1145/1102120.1102161.

[6] SHAO J, CAO Z, WANG L, *et al*. Proxy re-signature schemes without random oracles[C]. Proceedings of INDO-CRYPT 2007, Chennai, India, 2007: 197–209. doi: 10.1007/978-3-540-77026-8_15.

[7] WATERS B. Efficient identity-based encryption without random oracles[C]. Proceedings of EuroCrypt 2005, Aarhus, 2005: 114–127. doi: 10.1007/11426639_7.

[8] KiIATE K, IKKWON Y, and SECOGAN L. Remark on shao et al's bidirectional proxy re-signature scheme in indocrypt'07[J]. *International Journal of Network Security*, 2009, 8(3): 308–311. doi: 10.4304/jcp.7.7.1796-1800.

[9] LIBERT B and VERGNAUD D. Multi-use unidirectional proxy re-signatures[C]. Proceedings of the 15th ACM Conference on Computer and Communications Security, Alexandria, USA, 2008: 511–520. doi: 10.1145/1455770.1455835.

[10] WANG W P. An identity-based blind proxy re-signature scheme[J]. *Computer Applications and Software*, 2012, 29(10): 308–313. doi: 10.3969/j.issn.1000.

[11] YANG X, LI C, LI Y, *et al*. Divisible on-line/off-line proxy re-signature[J]. *Applied Mathematics & Information Sciences*, 2015, 9(2): 759–767. doi: 10.1007/978-3-642-00862-7_10.

[12] YANG X, WANG C, ZHANG L, *et al*. On-line/off-line threshold proxy re-signatures[J]. *Chinese Journal of Electronics*, 2014, 23(2): 248–253. doi: 10.4156/jcit.vol7.issue23.7.

[13] TIAN M M. Identity-based proxy re-signatures from lattices [J]. *Information Processing Letters*, 2015, 115(4): 462–467. doi: 10.1016/j.ipl.2014.12.002.

[14] 江明明, 胡予濮, 王保仓, 等. 格上基于身份的单向代理重签名[J]. 电子与信息学报, 2014, 36(3): 645–649. doi: 10.3724/SP.J.1146.2013. 00818.
JIANG M M, HU Y P, WANG B C, *et al*. Identity-based unidirectional proxy re-signature over lattice[J]. *Journal of Electronics & Information Technology*, 2014, 36(3): 645–649. doi: 10.3724/SP.J.1146.2013. 00818.

[15] 龙昭华, 龚俊, 王波, 等. 无线传感器网络中分簇安全路由协议保密通信方法的能效研究[J]. 电子与信息学报, 2015, 37(8): 2000–2006. doi: 10.11999/JEIT141284.
LONG Z H, GONG J, WANG B, *et al*. Energy efficiency study of secret communication method on clustering[J]. *Journal of Electronics & Information Technology*, 2015, 37(8): 2000–2006. doi: 10.11999/JEIT141284.

[16] WANG Z and LÜ W. Server-aided verification proxy re-signature[C]. Proceedings of Trust, Security and Privacy in Computing and Communications, Melbourne, Australia, 2013: 1704–1707. doi: 10.1109/TrustCom.2013.211.

[17] CANETTI R, GOLDREICH O, and HALEVI S. The random oracle methodology, revisited[J]. *Journal of the ACM*, 2004, 51(4): 557–594. doi: 10.1145/1008731.1008734.

[18] WU W, MU Y, SUSILO W, *et al*. Server-aided verification signatures: definitions and new constructions[C]. Proceedings of Provable Security, Shanghai, China, 2008: 141–155. doi: 10.1007/978-3-540-88733-1_10.

[19] WANG Zh W, WANG L Ch , YANG Y X, *et al*. Comment on Wu et al.'s server-aided verification signature schemes[J]. *International Journal of Network Security*, 2010, 10(2): 158–160. doi: 10.1.1.592.231.

杨小东：　男，1981 年生，副教授，研究方向为代理重密码、云计算安全等.

李亚楠：　男，1990 年生，硕士生，研究方向为网络安全.

高国娟：　女，1991 年生，硕士生，研究方向为信息安全.

王彩芬：　女，1963 年生，教授，研究方向为同态签名、多变量密码等.

鲁小勇：　男，1981 年生，博士生，研究方向为信息系统安全、隐私保护等.