软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software, 2022,33(11):4305−4315 [doi: 10.13328/j.cnki.jos.006317]

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

可追溯的广义指定验证者签名证明方案*

唐 飞[1,2], 马 帅[1], 马春亮[1]

[1](重庆邮电大学 计算机科学与技术学院, 重庆 400065)

[2](重庆邮电大学 网络空间安全与信息法学院, 重庆 400065)

通信作者: 唐飞, E-mail: tangfei@cqupt.edu.cn

摘 要: 为了解决传统广义指定验证者签名证明方案中强隐私保护性质对验证者不公平的问题, 提出了可追溯的广义指定验证者签名证明(traceable universal designated verifier signature proof, TUDVSP)方案. 在TUDVSP方案中, 引入一个追溯中心, 可将指定者的转换签名恢复为原始签名, 从而防止签名者与指定者合谋欺骗验证者. 基于现实应用考虑, 从不可伪造性、抗仿冒攻击和可追溯性这3个方面定义了TUDVSP方案的安全模型. 利用双线性映射构造具体的TUDVSP方案, 并证明该方案具有不可伪造性、抗仿冒攻击和可追溯性. 实验结果表明, 完成一次签名追溯仅需21 ms左右的计算开销与120字节的通信开销.

关键词: 广义指定验证者签名证明; 可追溯性; 隐私保护

中图法分类号: TP309

中文引用格式: 唐飞, 马帅, 马春亮. 可追溯的广义指定验证者签名证明方案. 软件学报, 2022, 33(11): 4305−4315. http://www.jos.org.cn/1000-9825/6317.htm

英文引用格式: Tang F, Ma S, Ma CL. Traceable Universal Designated Verifier Signature Proof Scheme. Ruan Jian Xue Bao/Journal of Software, 2022, 33(11): 4305−4315 (in Chinese). http://www.jos.org.cn/1000-9825/6317.htm

Traceable Universal Designated Verifier Signature Proof Scheme

TANG Fei[1,2], MA Shuai[1], MA Chun-Liang[1]

[1](College of Computer Science and Technology, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

[2](School of Cyber Security and Information Law, Chongqing University of Posts and Telecommunications, Chongqing 400065, China)

Abstract: To solve the problem of unfairness for verifier in the traditional universal designated verifier signature proof scheme because of the strong privacy-preserving property, the notion of traceable universal designated verifier signature proof (TUDVSP) was proposed. In this new kind of conditional privacy-preserving authentication scheme, a tracing center was introduced which can recover the transformed signature to the original one, and thus avoid the signer collude the delegator to cheat the verifier. Based on the consideration of real-word applications, security model which contains unforgeability, security against impersonation attack, and traceability for TUDVSP scheme was proposed. By using bilinear map, a concrete TUDVSP scheme was proposed, and the unforgeability, security against impersonation attack, and traceability of the proposed scheme were also proved. The experimental results indicate that it only takes about 21 ms of computation cost and 120 byte of communication overhead.

Key words: universal designated verifier signature proof; traceability; privacy-preserving

数字签名[1]具有完整性校验、不可否认等性质, 适用于数字化文件的公开验证. 然而, 公开验证也意味着隐私泄露, 因为不需要任何秘密信息即可进行签名验证操作. 签名者生成的签名结果一旦公开, 则所有人均可获知该签名.

为了增强数字签名的隐私保护性质, Jakobsson 等人[2]提出了指定验证者签名(designated verifier signature,

* 基金项目: 国家自然科学基金(61702067); 重庆市自然科学基金(cstc2017jcyjAX0201, cstc2020jcyj-msxmX0343)
收稿时间: 2020-12-23; 修改时间: 2021-01-20; 采用时间: 2021-02-04

DVS)的概念. 在 DVS 方案中, 签名者生成签名时指定一个验证者, 该指定的验证者可以利用自己的私钥模拟出一个和原始签名不可区分的副本. 因此, 对任意第三方而言, 无法判断一个签名是原始签名者生成的还是指定验证者模拟的, 只有被指定的验证者才相信签名的有效性, 从而实现“指定验证”的目的. 从定义可以看出, DVS 方案保护了签名者的身份隐私.

在 DVS 方案中, 签名的指定验证操作需要用到原始签名者的私钥, 因此, 该操作只能由原始签名者完成. 这一要求使得 DVS 方案在现实使用中具有一定的局限性. 为使 DVS 方案具有更广泛的用途, Steinfeld 等人[3]提出了广义指定验证者签名(universal designated verifier signature, UDVS)方案这一概念. 在 UDVS 方案中, 共有三方参与者, 分别称为签名者、指定者和验证者. 签名者生成签名并将其秘密发送给指定者; 随后, 指定者利用类似于 DVS 的方法向验证者证明其有效性. 文献[3]之后, 人们提出了许多不同性质的 UDVS 方案, 例如文献[4−7]等.

在 UDVS 方案中, 指定验证者可以利用自己的私钥生成与原始签名不可区分的模拟签名, 从而实现隐私保护性质. 但在现实应用中, 验证者可能拒绝注册获取个人私钥. 针对这一问题, Baek 等人[8]进一步提出了广义指定验证者签名证明(universal designated verifier signature proof, UDVSP)方案的概念. UDVSP 与 UDVS 的区别在于: 指定验证过程采用交互式证明的形式, 在签名证明过程中, 验证者只需要选择随机数而不用注册私钥. 因此, UDVSP 提升了签名指定验证的可用性. 随后, 许多新的 UDVSP 方案[9−11]被陆续提出.

类似于 DVS 和 UDVS 方案, UDVSP 方案也具有签名的不可转移性质, 即只有被指定的验证者才能相信签名的有效性, 签名证明结果无法转移给任何第三方. UDVSP 方案中, 签名验证的隐私保护性质是不可撤销的. 但在现实应用中, 这一强隐私保护性质会使得 UDVSP 方案存在潜在的风险. 以电子病历证明为例[8,12]: 医生(签名者)首先对病人病历数据进行签名认证; 随后, 病人(指定者)利用 UDVSP 方案向医保局(验证者)证明病历真实性, 从而申请保险赔偿. 基于 UDVSP 的不可转移性质, 可以保证病人病历不被恶意泄露给第三方获知. 但正是基于这一强隐私保护性质, 如果医保局事后发现病人联合医生进行恶意骗保, 则无法向任何第三方证明这一事实. 因此, UDVSP 的强隐私保护性质会使得验证者具有潜在的安全风险.

基于上述问题, 本文研究了广义指定验证者证明方案隐私保护性质的可撤销问题. 本文的主要研究工作如下.

(1) 提出了可追溯的广义指定验证者签名证明(traceable universal designated verifier signature proof, TUDVSP)方案的概念. 借鉴群签名的思想[13,14], 在 TUDVSP 方案中引入了一个额外的追溯中心. TUDVSP 方案的指定验证过程与 UDVSP 一样, 区别在于验证结束后, 如果验证者发现证明有问题, 可以请求追溯中心进行仲裁. 追溯中心用追溯私钥将指定签名中蕴含的原始签名恢复出来. 因此, 与 UDVSP 方案的强隐私保护方案不同, TUDVSP 方案提供了一种条件隐私保护机制, 可以保护验证者的权益, 使其免受非法认证的风险.

(2) 形式化定义了 TUDVSP 方案, 并根据方案的定义及应用场景的需求, 提出了方案需满足的安全性质, 具体包括不可伪造性、抗仿冒攻击和可追溯性这 3 个方面; 然后, 基于双线性映射构造一个具体的 TUDVSP 方案. 在随机预言模型下, 证明了所提方案的安全性; 最后, 采用仿真实验的方式分析了方案的效率分析和通信开销.

1 定 义

1.1 双线性映射

令 G_1 和 G_2 是两个阶均为大素数 p 的乘法循环群, g 是群 G_1 的任意一个生成元. 双线性对指的是满足如下性质的一个映射 e:$G_1\times G_1\to G_2$.

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

(2) 非退化性: $e(g,g)\neq 1$.

目前, 基于 Weil 对[15]或 Tate 对[16]及一系列改进的算法, 可以实现双线性映射的高效计算.

定义 1(*k*-CAA 假设)[17]. 对于整数 k, 给定 $\left(g, g^{x}, h_1, \ldots, h_k \in Z_p, g^{\frac{1}{h_1+x}}, \ldots, g^{\frac{1}{h_k+x}}\right)$, 计算 $g^{\frac{1}{h+x}}$ 是困难的, 其中, $h \neq h_i$, $i=1,\ldots,k$.

1.2 TUDVSP方案模型

TUDVSP 方案的模型如图 1 所示. 方案包括签名者、追溯中心、指定者、验证者这 4 个角色, 各角色功能如下.

(1) 签名者(signer): 拥有签名密钥, 可对相关信息进行签名认证操作, 并将认证信息秘密地发送给指定者(即签名拥有者).

(2) 追溯中心(tracing center): 拥有追溯密钥, 可将转换签名恢复为原始签名, 用以证明签名者对消息的确进行过认证操作, 从而避免强隐私保护的指定验证过程对验证者不公平.

(3) 指定者(delegator): 即签名拥有者, 拥有签名者发放的签名认证信息. 指定者向验证者证明签名信息, 但是又不希望直接泄露签名内容. 首先将原始签名进行转换操作, 然后与验证者交互运行指定验证协议, 实现隐私保护认证.

(4) 验证者(verifier): 没有任何秘密信息. 在与指定者交互执行指定验证协议过程中, 只需选择随机数作为挑战值, 然后验证转换签名的有效性. 验证结束后, 验证者如果发现转换签名中的消息是非法的, 则可将转换签名发送给追溯中心请求仲裁.

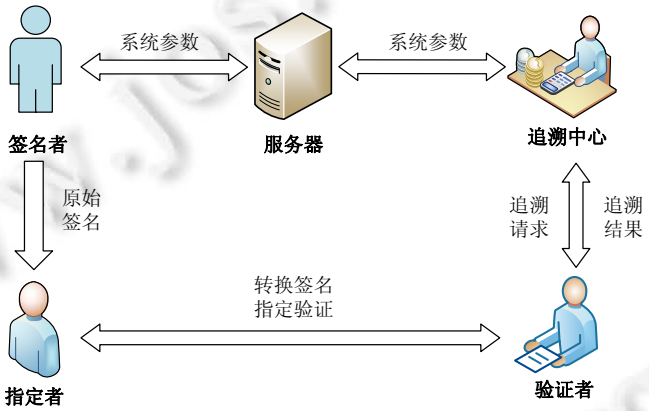

图 1 TUDVSP 方案模型图

观察 TUDVSP 方案模型不难发现: 与传统的 UDVSP 方案相比, 在指定者签名信息的隐私保护认证基础上增加了追溯中心这一角色, 实现转换签名的可追溯性. TUDVSP 方案的可追溯性与群签名[18−20]的匿名撤销性质非常类似. 但是在群签名方案中, 考虑的是签名者身份的隐私保护性质, 追溯的是真实签名者身份. 然而在 TUDVSP 方案中, 保护的是转换签名的隐私性, 追溯的也是对应的签名信息. 此外, TUDVSP 的签名验证采用的是交互式证明协议, 因此, TUDVSP 与群签名是不同性质的隐私保护认证方案. TUDVSP 方案与 UDVSP、群签名方案的性质对比见表 1.

表 1 TUDVSP 方案与群签名、UDVSP 方案性质对比

方案	不可伪造性	签名隐私保护	可追溯性
群签名方案	√	×	√
UDVSP 方案	√	√	×
本文方案	√	√	√

TUDVSP 方案以 UDVSP 方案为基础, 结合群签名中的 Open 算法, 在实现对签名信息隐私保护的同时, 可以实现对转换签名的追溯. TUDVSP 方案同时具备不可伪造性、签名隐私保护性和可追溯性, 在某些应用场景中更具有现实意义. 例如在电子病历认证中, 病历数据对病人而言是敏感信息, 需要隐私保护. 病人用医院

签名认证的电子病历去医保局申请报销, 但他不希望医保局将病历及认证信息转移给第三方获知. 如果采用 TUDVSP 进行病历签署与认证, 则病人可将认证信息进行转换, 使医保局相信自己的确拥有来自医院认证的电子病历. 但是医保局无法将该认证信息转移给第三方. 同时, 如果医保局发现病历有问题, 即医院对非法病历进行了签署认证, 则可以向追溯中心申请仲裁. 追溯中心从转换签名中恢复出原始签名, 从而对医院进行追责, 以保护医保局的权益. 在电子病例证明应用场景中, 用传统 UDVSP 方案和 TUDVSP 方案的区别如图 2 所示, 追溯中心的引入, 有助于构建公平的隐私保护认证方案.

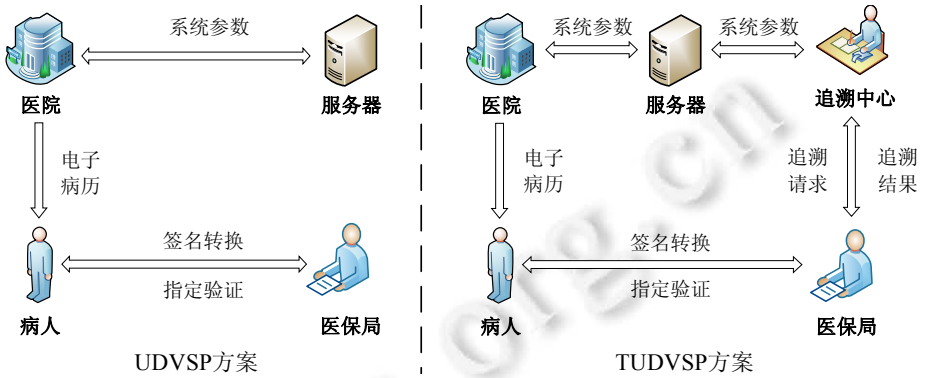

图 2 UDVSP 方案和 TUDVSP 方案在电子病历证明中的应用对比

1.3 TUDVSP方案定义

可追溯的广义指定验证者签名证明方案包括 5 个多项式时间算法和 1 个交互式证明协议, 其中, 交互式证明协议又包含 2 个子算法.

(1) 系统建立(*Setup*): 输入安全参数$\lambda\in N$, 输出系统参数 *params* 和签名者的公私钥对(*pk*,*sk*)和追溯中心的公私钥对(*tpk*,*tsk*).

(2) 签名生成(*Sign*): 签名者输入私钥 *sk* 和消息 *m*, 输出原始签名σ, 并将其秘密地发送给指定者.

(3) 原始签名验证(*Vrfy*): 指定者输入签名者公钥 *pk*、消息 *m* 和签名σ, 验证其有效性: 如果有效, 则输出 1; 否则, 输出 0.

(4) 签名转换(*Trans*): 为了实现隐私保护认证, 指定者在证明其拥有签名前, 先对原始签名进行转换操作. 输入签名者公钥 *pk*、追溯中心公钥 *tpk* 和签名σ, 此外, 指定者还选择一个随机数 k'. 算法输出一个转换签名σ'.

(5) 转换签名交互式验证(*IVrfy*): 该交互验证协议包含两个子算法(*P*,*V*), 由指定者与验证者共同完成, 将指定者记为 *P*, 验证者记为 *V*. *P* 和 *V* 的共同输入包括系统公开参数 *params*、签名者公钥 *pk*、转换签名σ'和消息 *m*. 此外, *P* 还拥有秘密值 k'作为在证明过程中的证据, *V* 不需要额外信息. 在该证明协议中, *P* 向 *V* 证明确实拥有来自原始签名者对消息 *m* 的认证信息σ, 但是并不直接向 *V* 泄露σ. 证明通过, 则输出 1; 否则, 输出 0.

(6) 追溯(*Trace*): 该算法由可信追溯中心运行. 算法输入转换签名σ'和追溯中心的私钥 *tsk*. 如果证明过程中的转换签名σ'的确是由原始签名σ转换计算得到, 则算法输出σ, 证明原始签名者的确对消息 *m* 进行过认证操作; 否则, 算法输出 *null*.

TUDVSP 需要满足如下 3 个方面的正确性.

(1) 原始签名的正确性: 如果签名者诚实执行签名生成算法, 则签名一定有效. 即如果$\sigma\leftarrow Sign(sk,m)$, 则 $1\leftarrow Vrfy(pk,m,\sigma)$.

(2) 转换签名的正确性: 如果指定者诚实执行签名转换算法, 且 *P* 和 *V* 诚实执行交互式验证, 则 *V* 可接受转换签名的有效性. 即如果$\sigma'\leftarrow Trans(pk,tpk,\sigma',k')$, 则 $1\leftarrow IVrfy(P(tsk),V)(params,pk,\sigma',m)$.

(3) 追溯的正确性: 如果指定验证中的转换签名确实由原始签名转换计算得出, 则追踪算法可以正确恢复原始签名. 即如果$\sigma' \leftarrow Trans(pk,tpk,\sigma',k')$, 则$\sigma \leftarrow Trace(tsk,\sigma')$.

1.4 TUDVSP方案安全性

可追溯的广义指定验证者签名证明方案需要满足不可伪造、抗仿冒攻击和可追溯这 3 个方面的安全性质.

(1) 不可伪造性

在 TUDVSP 方案中, 首先需要保证签名算法生成的数字签名是不可伪造的[8]. TUDVSP 方案的不可伪造性与普通数字签名方案的自适应选择消息攻击下的存在性不可伪造性(UF-CMA)[21]是一致的, 该性质由如下实验定义.

定义 2(不可伪造性). 实验 $Exp_{tudvsp}^{uf\text{-}cma}(1^{\lambda})$ 由一个多项式敌手 A 和挑战者 C 共同完成. 敌手首先询问签名预言机多项式次, 随后输出一个关于新消息的伪造签名.

- 训练: 挑战者首先运行系统建立算法 $Setup(1^{\lambda})\rightarrow(params,pk,sk,tpk,tsk)$, 然后将 $params$, pk, tpk 发送给敌手. 随后, 敌手适应性地输入消息 $m_i \in O^{S(sk,\cdot)}$, 挑战者回复签名σ_i.
- 输出: 敌手 A 输出伪造签名(m^*,σ^*), 如果 $Vrfy(params,pk,m^*,\sigma^*)\rightarrow 1$ 且挑战消息 $m^* \notin O^{S(sk,\cdot)}$, 则返回 1, 表示敌手赢得了该游戏; 否则返回 0, 表示敌手失败.

将上述实验中敌手的优势定义为 $Adv_{A,tudvsp}^{uf\text{-}cma}(1^{\lambda}) = \Pr[Exp_{tudvsp}^{uf\text{-}cma}(1^{\lambda}) = 1]$.

如果对任意 PPT 敌手 A 而言, 其攻击优势关于安全参数λ都是可忽略的, 则称 TUDVSP 方案在适应性选择消息攻击下具有不可伪造性.

(2) 抗仿冒攻击

类似于 UDVSP 方案, TUDVSP 方案的第 2 个方面的安全性质要求其能够抵抗仿冒攻击, 即在 TUDVSP 方案中, 任意 PPT 敌手无法仿冒其具有签名者签署的有效签名. 仿冒攻击进一步可以分成“Type-1”和“Type-2”两种类型.

在 Type-1 攻击中, 敌手身份是作为一个恶意的验证者, 他与诚实指定者交互运行 $IVrfy$ 协议, 因此在 Type-1 攻击中, 敌手可以获得转换后的签名. 随后, 敌手向其他诚实的指定验证者仿冒其是指定者. 针对 Type-1 攻击的安全模型定义如下.

定义 3(抗 Type-1 仿冒攻击). 假设$(Setup,Sign,Vrfy,Trans,IVrfy,Trace)$是一个 TUDVSP 方案. PPT 敌手 A 包含两个子算法 V'和 P', 分别代表恶意指定验证者和恶意指定者. 我们用 P 表示 TUDVSP 系统中诚实的指定者, 用 $Conv_{IVrfy}$ 函数表示 P 和 V'交互运行 $IVrfy$ 协议后的会话副本 T. 注意, T 是一个关于 P 和 V'随机选择的变量, 我们将其记为 $Conv_{IVrfy}(P(k'),V')(params,pk,tpk,m,\sigma')\rightarrow T$. 接下来考虑实验 $Exp_{tudvsp}^{im\text{-}type1}(1^{\lambda})$. 首先运行 $Setup(1^{\lambda})\rightarrow(params,pk,sk,tpk,tsk)$. 公开参数 $params$ 和 pk, tpk 公开给诚实的证明者 P 与敌手 $A=(V',P')$. 随后选择任意消息 m 并运行签名算法 $Sign$, 获得一个有效签名σ. 此外, 证明者选择随机值 k'并用其生成转一个换签名σ'. 证明者 P 秘密保存随机值 k', 并将转换签名σ'发给敌手. 接着, V'就与 P 交互式执行 $IVrfy$ 协议 $p(\lambda)$次, 其中, $p(\lambda)$表示关于安全参数λ的一个多项式函数. 协议执行的副本和敌手 V'的随机选择分别记为 T_i 和 $r_i^{V'}$, 其中, $i=1,2,\ldots,p(\lambda)$. 实验中, P'的目的是通过运行 $IVrfy$ 协议向一个诚实的指定验证者仿冒证明其是指定者 P.

实验 $Exp_{tudvsp}^{im\text{-}type1}(1^{\lambda})$ 定义如下.

- 训练:

$$Setup(1^{\lambda}) \rightarrow (params, pk, sk, tpk, tsk)$$
$$m \in \{0,1\}^*,\ Sign(params, sk, m) \rightarrow \sigma$$
$$Trans(params, pk, tpk, k', \sigma) \rightarrow \sigma'$$
$$Conv_{IVrfy}((P(k'),V')(params, pk, m, \sigma') \rightarrow T_i, \text{ 其中, } i = 1,2,\ldots, p(\lambda)$$
$$IVrfy((P'((T_1, r_1^{V'}),\ldots,(T_{p(\lambda)}, r_{p(\lambda)}^{V'})),V')(params, pk, m, \sigma) \rightarrow d$$

- 输出: d.

将上述实验中敌手的优势定义为 $Adv^{im\text{-}type1}_{A,tudvsp}(1^\lambda)=\Pr[Exp^{im\text{-}type1}_{A,tudvsp}(1^\lambda)=1]$. 如果对任意 PPT 敌手 A, 其优势 $Adv^{im\text{-}type1}_{A,tudvsp}(1^\lambda)$ 关于安全参数λ都是可忽略的, 则称该 TUDVSP 方案是抗 Type-1 仿冒攻击的.

接下来考虑 Type-2 攻击. 在这类攻击中, 敌手直接忽略转换签名, 自己尝试构建一个新的转换签名, 并用其运行 *IVrfy* 协议, 向诚实的指定验证者仿冒证明其是合法的指定者.

定义 4(抗 Type-2 仿冒攻击). 假设(*Setup*,*Sign*,*Vrfy*,*Trans*,*IVrfy*,*Trace*)是一个 TUDVSP 方案, A 是一个 PPT 的 Type-2 敌手. 考虑实验 $Exp^{im\text{-}tpye2}_{A,tudvsp}(1^\lambda)$. 首先运行算法 $Setup(1^\lambda)\to(params,pk,sk,tpk,tsk)$, 并将参数 *params* 和 *pk*, *tpk* 发给敌手. 随后选择任意消息 m 并将其发送给敌手 A. 接着, A 自己尝试生成指定者的随机值 k'和转换签名σ', 并用其与一个诚实的指定验证者交互运行 *IVrfy* 协议. 实验 $Exp^{im\text{-}tpye2}_{A,tudvsp}(1^\lambda)$ 的定义如下.

- 训练:

$$Setup(1^\lambda)\to(params,pk,sk,tpk,tsk)$$
$$m\in\{0,1\}^*,\ A(params,pk,tpk,m)\to(k',\sigma')$$
$$IVrfy(A(k'),V)(params,pk,tpk,m,\sigma')\to d$$

- 输出: d.

将上述实验中敌手的优势定义为 $Adv^{im\text{-}type2}_{A,tudvsp}(1^\lambda)=\Pr[Exp^{im\text{-}tpye2}_{A,tudvsp}(1^\lambda)=1]$. 如果对任意 PPT 敌手 A, 其优势 $Exp^{im\text{-}tpye2}_{A,tudvsp}(1^\lambda)$ 关于安全参数λ都是可忽略的, 则称该 TUDVSP 方案是抗 Type-2 仿冒攻击的.

根据上述抗仿冒攻击的定义不难发现: Type-1 敌手首先会充当恶意的验证者与指定者进行交互运行 *IVrfy* 协议, 再充当恶意的指定者欺骗诚实的验证者; 而 Type-2 敌手则直接伪造签名或转换签名欺骗验证者. 因此, Type-1 攻击敌手强于 Type-2 攻击敌手, 即如果一个 TUDVSP 方案能够抗 Type-1 仿冒攻击, 则其一定也能抗 Type-2 仿冒攻击. TUDVSP 的 Type-1 和 Type-2 敌手分别类似于身份识别方案[22,23]中的动态攻击敌手和静态攻击敌手.

(3) 可追溯性

与 UDVSP 方案的强隐私保护性质不同, TUDVSP 方案需要满足可追溯性质. 为了定义 TUDVSP 方案的可追溯性质, 我们借鉴群签名方案中的真实签名者追溯机制. 在 TUDVSP 方案中, 我们要求存在一个追溯中心, 能将转换签名σ'恢复成其对应的原始签名σ. 根据原始签名的不可伪造性可知: 除了原始签名者之外, 没有人能够生成关于其公钥的签名. 因此, 如果签名者对非法消息进行了认证签名, 验证者则可请求追溯中心追究其责任. 在 TUDVSP 方案中, 对于一个有效的转换签名$\sigma\leftarrow Sign(sk,m)$, $\sigma'\leftarrow Trans(pk,tpk,\sigma',k')$, 只有追溯中心才可以追溯真实的签名$\sigma\leftarrow Trace(tsk,\sigma')$. 最后, 追溯中心验证该签名的有效性, 从而判断转换签名是由指定者根据原始签名计算生成, 还是由指定验证者模拟计算生成.

2 TUDVSP 方案

2.1 方案构造

输入安全参数$\lambda\in N$, 构造两个阶为大素数 p 的乘法群环群 G_1 和 G_2, 元素 g 为群 G_1 的生成元, e:$G_1\times G_1\to G_2$ 是一个双线性映射. 选择一个抗碰撞的 hash 函数: $H:\{0,1\}^*\to Z_p^*$.

(1) 系统建立 *Setup*

追溯中心选择两个随机数 $\xi_1,\xi_2\in Z_p^*$和一个群元素 $h\in G_1$, 还选择两个元素 $u,v\in G_1$, 使得 $u^{\xi_1}=v^{\xi_2}=h$. 追溯中心的私钥为 $tsk=(\xi_1,\xi_2)$, 公钥为 $tpk=(u,v,h)$. 签名者随机选择 $x\in Z_p^*$ 作为私钥 sk, 其公钥为 $pk=y=g^x\in G_1$. 系统公开参数为

$$params=(G_1,G_2,p,g,e,H,u,v,h,y) \tag{1}$$

(2) 签名生成 *Sign*

对任意消息 $m\in\{0,1\}^*$, 签名者输入私钥 *sk* 计算签名:

$$\sigma=g^{1/(H(m)+x)}\in G_1 \tag{2}$$

随后, 将σ秘密发送给指定者.

(3) 签名验证 *Vrfy*

指定者收到签名后, 首先验证其有效性, 如果等式:

$$e(\sigma,g^{H(m)}\cdot y)=e(g,g) \tag{3}$$

成立, 则输出 1, 接受该签名; 否则输出 0, 拒绝签名.

(4) 签名转换 *Trans*

指定者在向验证者验证签名前, 对原始签名进行转换操作.

- 首先, 随机选择 $\alpha,\beta\in Z_p^*$ 并计算:

$$T_1=u^{\alpha},\ T_2=v^{\beta},\ T_3=\sigma\cdot h^{\alpha+\beta} \tag{4}$$

- 其次, 还计算两个辅助参数:

$$\delta_1=H(m)\cdot\alpha,\ \delta_2=H(m)\cdot\beta \tag{5}$$

指定者的临时私钥为 $k'=(\alpha,\beta,\delta_1,\delta_2)$, 转换签名为$\sigma'=(T_1,T_2,T_3)$.

(5) 指定验证 *IVrfy*

在指定验证协议中, 指定者为证明者 *P*, 验证者为 *V*. *P* 和 *V* 的共同输入包括系统公开参数 *params*、签名者公钥 *pk*、转换签名σ'和消息 *m*. 此外, *P* 还拥有秘密值 k'作为在验证过程中的私有证据, *V* 不需要额外信息. 指定验证协议如下.

指定者选择随机数 $r_{\alpha},r_{\beta},r_{H(m)},r_{\delta_1},r_{\delta_2}\in Z_p^*$, 并计算:

$$R_1=u^{r_{\alpha}},\ R_2=v^{r_{\beta}} \tag{6}$$

$$R_3=e(T_3,g)^{r_{H(m)}}\cdot e(h,y)^{-r_{\alpha}-r_{\beta}}\cdot e(h,g)^{-r_{\delta_1}-r_{\delta_2}} \tag{7}$$

$$R_4=T_1^{r_{H(m)}}\cdot u^{-r_{\delta_1}},\ R_5=T_2^{r_{H(m)}}\cdot v^{-r_{\delta_2}} \tag{8}$$

随后, 将 $R_1, R_2, R_3, R_4, R_5, r_{H(m)}$发送给验证者.

验证者选择随机数 $c\in Z_p^*$ 作为挑战值, 并将其发送给指定者.

- 指定者计算

$$s_{\alpha}=r_{\alpha}+c\alpha,\ s_{\beta}=r_{\beta}+c\beta,\ s_{\delta_1}=r_{\delta_1}+c\delta_1,\ s_{\delta_2}=r_{\delta_2}+c\delta_2 \tag{9}$$

随后, 将这些值发送给验证者.

最后, 验证者验证如下等式是否成立:

$$u^{s_{\alpha}}=T_1^{c}\cdot R_1,\ v^{s_{\beta}}=T_2^{\beta}\cdot R_2 \tag{10}$$

$$(e(g,g)/e(T_3,y))^{c}\cdot R_3=e(T_3,g)^{r_{H(m)}+cH(m)}\cdot e(h,y)^{-s_{\alpha}-s_{\beta}}\cdot e(h,g)^{-s_{\delta_1}-s_{\delta_2}} \tag{11}$$

$$T_1^{r_{H(m)}+cH(m)}\cdot u^{-s_{\delta_1}}=R_4,\ T_2^{r_{H(m)}+cH(m)}\cdot v^{-s_{\delta_2}}=R_5 \tag{12}$$

如果 5 个等式均成立, 则输出 1, 表示指定者的确拥有来自原始签名者关于消息 *m* 的某个签名.

(6) 签名追溯 *Trace*

当验证者发现签名者与指定者联合作弊对非法消息 *m* 进行认证, 且已完成指定验证操作时, 他将转换签名$\sigma'=(T_1,T_2,T_3)$发送给追溯中心请求仲裁. 收到请求后, 追溯中心用自己的私钥 $tsk=(\xi_1,\xi_2)$对其追溯. 计算过程如下:

$$\sigma=T_3/(T_1^{\xi_1}\cdot T_2^{\xi_2}) \tag{13}$$

最后, 追溯中心可以通过验证签名σ是否有效, 来判断签名者是否对消息 *m* 认证过.

2.2 正确性分析

(1) 原始签名的正确性

$$e(\sigma,g^{H(m)}\cdot y)=e(g^{1/(H(m)+x)},g^{H(m)}\cdot g^{x})=e(g,g).$$

(2) 转换签名的正确性

- 首先, 验证等式(10)的正确性:

$$u^{s_\alpha}=u^{r_\alpha+c\alpha}=u^{r_\alpha}\cdot u^{c\alpha}=T_1^{c}\cdot R_1,\ v^{s_\beta}=v^{r_\beta+c\beta}=v^{r_\beta}\cdot v^{c\beta}=T_2^{c}\cdot R_2.$$

- 其次, 验证等式(11)的正确性:

$$\begin{aligned}
&e(T_3,g)^{r_{H(m)}+cH(m)}\cdot e(h,y)^{-s_\alpha-s_\beta}\cdot e(h,g)^{-s_{\delta_1}-s_{\delta_2}}\\
&=e(T_3,g)^{r_{H(m)}+cH(m)}\cdot e(h,y)^{-r_\alpha-c\alpha-r_\beta-c\beta}\cdot e(h,g)^{-r_{\delta_1}-c\delta_1-r_{\delta_2}-c\delta_2}\\
&=(e(T_3,g)^{H(m)}\cdot e(h,y)^{-\alpha-\beta}\cdot e(h,g)^{-\delta_1-\delta_2})^{c}\cdot e(T_3,g)^{r_{H(m)}}e(h,y)^{-r_\alpha-r_\beta}\cdot e(h,g)^{-r_{\delta_1}-r_{\delta_2}}\\
&=(e(g^{1/(H(m)+x)}\cdot h^{\alpha+\beta},g)^{H(m)}\cdot e(h,g^{x})^{-\alpha+\beta}\cdot e(h,g)^{-H(m)\cdot\alpha-H(m)\cdot\beta})^{c}\cdot R_3\\
&=(e(g^{H(m)/(H(m)+x)}\cdot h^{H(m)\cdot(\alpha+\beta)},g)\cdot e(h^{-H(m)\cdot(\alpha+\beta)},g)/e(h^{\alpha+\beta},g^{x}))^{c}\cdot R_3\\
&=(e(g^{1-x/(H(m)+x)},g)/e(h^{\alpha+\beta},g^{x}))^{c}\cdot R_3\\
&=(e(g,g)/e(g^{1/(H(m)+x)},g^{x})/e(h^{\alpha+\beta},g^{x}))^{c}\cdot R_3\\
&=(e(g,g)/e(T_3,y))^{c}\cdot R_3.
\end{aligned}$$

- 然后, 验证等式(12)的正确性:

$$T_1^{r_{H(m)}+cH(m)}\cdot u^{-s_{\delta_1}}=T_1^{r_{H(m)}}\cdot u^{\alpha\cdot cH(m)}\cdot u^{-r_{\delta_1}-c\delta_1}=T_1^{r_{H(m)}}\cdot u^{\alpha\cdot cH(m)}\cdot u^{-r_{\delta_1}-\alpha\cdot cH(m)}=R_4,$$

$$T_2^{r_{H(m)}+cH(m)}\cdot v^{-s_{\delta_2}}=T_2^{r_{H(m)}}\cdot v^{\beta\cdot cH(m)}\cdot v^{-r_{\delta_2}-c\delta_2}=T_2^{r_{H(m)}}\cdot v^{\beta\cdot cH(m)}\cdot v^{-r_{\delta_2}-\beta\cdot cH(m)}=R_5.$$

(3) 追溯的正确性

$$T_3/(T_1^{\xi_1}\cdot T_2^{\xi_2})=\sigma h^{\alpha+\beta}/(u^{\alpha\xi_1}\cdot v^{\xi_2})=\sigma h^{\alpha+\beta}/(h^{\alpha}\cdot h^{\beta})=\sigma.$$

3 方案安全性与效率分析

3.1 安全性分析

(1) 不可伪造性

不难发现, 本文提出的 TUDVSP 方案的基础签名方案是 ZSS 签名方案[17], 其安全性已经在文献[17]中得以证明. 因此, 本方案的不可伪造性可以直接根据 ZSS 方案的安全性得到.

定理 1. 如果 k-CAA 问题困难, 则本文提出的 TUDVSP 方案具有存在性不可伪造性.

证明: 见附录 1.

(2) 抗仿冒攻击

根据前述 TUDVSP 方案的抗仿冒攻击定义, 我们只需证明本文提出的方案在 Type-1 攻击下是安全的, 则其在 Type-2 攻击下也是安全的.

定理 2. 如果 k-CAA 问题困难, 则本文提出的 TUDVSP 方案具有抗 Tpye-1 仿冒攻击安全性.

证明: 假设 $A=(V',P')$为一个多项式时间的 Type-1 攻击敌手. 令挑战者为 C, 其目的是打破基础签名方案的不可伪造性. 挑战者输入安全参数$\lambda\in N$运行系统建立算法获得公开参数:

$$params=(G_1,G_2,p,g,e,H,u,v,h,y) \tag{14}$$

其中, y 为基础签名方案中签名者的公钥信息, C 没有其对应的私钥. 随后, 挑战者输出公开参数, 并随机选择一个挑战消息 $m\in\{0,1\}^{*}$.

挑战者作为一个真实的指定者 P 与敌手 V'交互运行指定验证协议 *IVrfy*, 敌手获得认证信息$(T_1,T_2,T_3,R_1,R_2,R_3,R_4,R_5,r_{H(m)})$, 敌手 V'选取一个挑战值 c 并将其发送给挑战者. 挑战者返回验证值 $(s_\alpha,s_\beta,s_{\delta_1},s_{\delta_2})$. 根据重置引理[8,22], 则对相同的认证信息, 敌手选取一个不同的挑战值 c', 挑战者返回验证值 $(s'_\alpha,s'_\beta,s'_{\delta_1},s'_{\delta_2})$. 挑战者

可以计算$\Delta c=c-c'$, $\Delta s_\alpha=s_\alpha-s'_\alpha$, Δs_β, Δs_{δ_1}, Δs_{δ_2} 也可用类似的方式计算得到. 由方案中的等式可知 $u^{\Delta s_\alpha}=T_1^{\Delta c}$, 因此可得 $\tilde{\alpha}=\Delta s_\alpha/\Delta c$, 且 $u^{\tilde{\alpha}}=T_1$. 同理可得 $\tilde{\beta}=\Delta s_\beta/\Delta c$, 且 $u^{\tilde{\beta}}=T_2$.

此外, 根据方案中等式可知 $T_1^{\Delta c\cdot H(m)}=u^{\Delta s_{\delta_1}}$, 由 $T_1=u^{\tilde{\alpha}}$ 可得 $u^{\tilde{\alpha}\Delta c\cdot H(m)}=u^{\Delta s_{\delta_1}}$, 其中, $\Delta s_{\delta_1}=\tilde{\alpha}\Delta c\cdot H(m)$. 同理可得 $\Delta s_{\delta_2}=\tilde{\beta}\Delta c\cdot H(m)$. 最后, 根据方案等式可得:

$$\begin{aligned}(e(g,g)/e(T_3,y))^{\Delta c}&=e(T_3,g)^{\Delta c\cdot H(m)}\cdot e(h,y)^{-\Delta s_\alpha-\Delta s_\beta}\cdot e(h,g)^{-\Delta s_{\delta_1}-\Delta s_{\delta_2}}\\&=e(T_3,g)^{\Delta c\cdot H(m)}\cdot e(h,y)^{-\Delta s_\alpha-\Delta s_\beta}\cdot e(h,g)^{-\alpha\Delta c\cdot H(m)-\tilde{\beta}\Delta c\cdot H(m)}\end{aligned}\tag{15}$$

因为 $H(m)=\Delta c\cdot H(m)/\Delta c$, 所以:

$$(e(g,g)/e(T_3,y))=e(T_3,g)^{H(m)}\cdot e(h,y)^{-\tilde{\alpha}-\tilde{\beta}}\cdot e(h,g)^{-H(m)(\tilde{\alpha}+\tilde{\beta})}\tag{16}$$

随后可得 $e(g,g)=e(T_3h^{-\tilde{\alpha}-\tilde{\beta}},yg^{H(m)})$. 因此可以计算 $\sigma'=T_3h^{-\tilde{\alpha}-\tilde{\beta}}$, 并且该签名满足 ZSS 签名方案的验证等式 $e(\sigma',yg^{H(m)})=e(g,g)$. 即挑战者可以根据敌手的仿冒攻击得到一个关于 ZSS 方案的伪造签名. 然而该方案是被证明不可伪造的, 因此不存在关于 TUDVSP 方案的多项式时间 Type-1 仿冒攻击敌手. □

(3) 可追溯性

TUDVSP 方案的可追溯性要求一个合法的转换签名能够被追溯中心正确的打开恢复成一个原始签名.

定理 3. 本文提出的 TUDVSP 方案具有可追溯性.

证明: 根据方案的追溯算法可知:

$$T_3/(T_1^{\xi_1}\cdot T_2^{\xi_2})=\sigma h^{\alpha+\beta}/(u^{\alpha\xi_1}\cdot v^{\xi_2})=\sigma h^{\alpha+\beta}/(h^{\alpha}\cdot h^{\beta})=\sigma\tag{17}$$

因此, 合法的转换签名可以被正确恢复. 此外, 定理 1 的不可伪造性保证了验证者无法自己生成有效的原始签名欺骗追溯中心; 定理 2 的不可仿冒性保证了验证者无法自己生成合法的转换签名欺骗追溯中心. 即验证者提交给追溯中心的合法转换签名只能是来自签名者和指定者. 因此, 本文提出的 TUDVSP 方案具有可追溯性. □

3.2 效率分析

本文的主要工作是提出并实现一种可追溯的条件隐私保护认证方案, 该方案提供了一种不同于现有方案的认证机制, 因此无法与现有方案直接进行效率对比, 仅测试本文提出方案的计算与通信开销. 测试的硬件平台采用 Intel i7-8565U 处理器, 主频为 1.8 GHz, 内存为 8 GB, 代码库使用 Java Pairing-based Cryptography Library (JPBC)[24], 运行环境为 Windows 10 操作系统. 众所周知: 在基于双线性映射的密码方案中, 双线性对运算、hash-to-point 运算、群元素指数运算和群元素乘法运算占了主要的计算开销. 将这 4 个运算的计算开销分别记为 T_{par}, T_{mtp}, $T_{\exp}$, T_{mul}, 并基于这 4 种计算考虑本文提出的 TUDVSP 方案各算法的计算开销. 得到的测试结果如下: T_{par}=14.1565 ms, T_{mtp}=23.3021 ms, $T_{\exp}$=10.4702 ms, T_{mul}=0.1403 ms.

本文提出的 TUDVSP 方案的各算法计算开销见表 2.

表 2 方案计算开销

算法	所需计算	计算时间(ms)
签名生成	$T_{\exp}$	10.470 2
签名验证	$2T_{par}+T_{\exp}+T_{mul}$	38.923 5
签名转换	$3T_{par}+T_{mul}$	31.550 9
指定验证-P	$3T_{par}+9T_{\exp}+2T_{mul}$	136.981 9
指定验证-V	$4T_{par}+12T_{\exp}+8T_{mul}$	183.390 8
签名追溯	$2T_{par}+2T_{mul}$	21.221

接下来分析 TUDVSP 方案在传输带宽方面的需求. 双线性群 G_1 和 G_2 中的元素长度分别为 40 字节和 128 字节, 参数 p 的长度一般为 32 字节[25]. 所提方案的传输带宽需求见表 3.

从表 3 可知: 本文提出的 TUDVSP 方案中, 只有证明算法 P 的传输带宽需求比较大. 但是对指定者而言, 在签名证明过程中需要保证其隐私信息不被泄露. 因此对指定者而言, 消耗更多的计算和传输资源是合理的.

表 3 方案通信开销

传输消息	所需带宽(字节)
原始签名	$\|G_1\|$=40
指定验证-P	$7\cdot\|G_1\|+\|G_2\|+5\cdot\|Z_p\|$=568
指定验证-V	$\|Z_p\|$=32
追溯签名	$3\cdot\|G_1\|$=120

4 结束语

本文在传统广义指定验证者签名证明方案的基础上, 提出了一个新的可追溯广义指定验证者签名证明方案的概念. 相对传统的广义指定验证者签名证明方案, 所提新方案的隐私保护性质可以被追溯中心打破, 从而保护验证者权益, 实现更为公平的隐私保护认证机制. 通过具体的方案构造、安全性证明、效率分析等证明所提方案的可行性.

References:

[1] Diffie W, Hellman M. New directions in cryptography. IEEE Trans. on Information Theory, 1976, 22(6): 644−654.

[2] Jakobsson M, Sako K, Impagliazzo R. Designated verifier proofs and their applications. In: Proc. of the Advances in Cryptology-EUROCRYPT '96. LNCS 1070, Springer, 1996. 143−154.

[3] Steinfeld R, Bull L, Wang HX, Pieprayk J. Universal designated-verifier signatures. In: Proc. of the Advances in Cryptology-ASIACRYPT 2003. LNCS 2894, Springer, 2003. 523−542.

[4] Rastegari P, Berenjkoub M, Dakhilalian M, Susilo W. Universal designated verifier signature scheme with non-delegatability in the standard model. Information Sciences, 2019, 479: 321−334.

[5] Li BH, Liu YZ, Yang S. Lattice-based universal designated verifier signatures. In: Proc. of the 15th IEEE Int'l Conf. on e-business Engineering (ICEBE). IEEE, 2018. 329−334.

[6] Hou SQ, Huang XY, Liu JK, Li J, Xu L. Universal designated verifier transitive signatures for graph-based big data. Information Sciences, 2015, 318: 144−156.

[7] Tang F, Lin CL, Ke PH. Universal designated verifier signcryption. In: Proc. of the NSS 2012. LNCS 7645, 2012. 126−134.

[8] Baek J, Safavi-Naini R, Susilo W. Universal designated verifier signature proof (or how to efficiently prove knowledge of a signature). In: Proc. of the Advances in Cryptology-ASIACRYPT 2005. LNCS 3788, Springer, 2005. 644−661.

[9] Li J, Wang YM. Universal designated verifier ring signature (proof) without random oracles. In: Proc. of the Emerging Directions in Embedded and Ubiquitous Computing (EUC 2006). LNCS 4097, Springer, 2006. 332−341.

[10] Chen GM. A new secure universal designated verifier signature proof system. Journal of Electronics & Information Technology, 2009, 31(2): 489−492.

[11] Chen XF, Chen GM, Zhang FG, Wei BD, Mu Y. Identity-based universal designated verifier signature proof system. Int'l Journal of Network Security, 2009, 52−58.

[12] Tang F, Ma S, Xiang Y, Lin CL. An efficient authentication scheme for blockchain-based electronic health records. IEEE ACCESS, 2019, 7: 41678−41689.

[13] Chaum D, Van HE. Group signatures. In: Proc. of the Workshop on the Theory and Application of Cryptographic Techniques. Berlin, Heidelberg: Springer, 1991. 257−265.

[14] Boneh D, Boyen X. Short signatures without random oracles. In: Proc. of the Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer, 2004. 56−73.

[15] Boneh D, Franklin M. Identity-based encryption from the Weil pairing. In: Proc. of the Advances in Cryptology-CRYPTO 2001. LNCS 2139, Springer, 2001. 213−229.

[16] Frek G, Muller M, Ruck H. The Tate pairing and the discrete logarithm applied to elliptic curve cryptosystems. IEEE Trans. on Information Theory, 1999, 45: 1717−1718.

[17] Zhang FG, Safavi-naini R, Susilo W. An efficient signature scheme from bilinear pairings and its applications. In: Proc. of the Public Key Cryptography 2004. LNCS 2947, Springer, 2004. 277−290.

[18] Eom S, Huh JH. Group signature with restrictive linkability: Minimizing privacy exposure in ubiquitous environment. Journal of Ambient Intelligence and Humanized Computing, 2018, 1−11.

[19] Yue X, Xi M, Chen B, *et al*. A revocable group signatures scheme to provide privacy-preserving authentications. Mobile Networks and Applications, 2020, 1−18.

[20] Zhong H, Huang CL, Xu Y, Cui J. Efficient group signature scheme with revocation. Journal on Communications, 2016, 37(10): 18−24.

[21] Goldwasser S, Micali S, Rivest RL. A digital signature scheme secure against adaptive chosen-message attacks. SIAM Journal on Computing, 1988, 17(2): 281−308.

[22] Bellare M, Palacio A. GQ and Schnorr identification schemes: Proofs of security against impersonation under active and concurrent attack. In: Proc. of the Advances in Cryptology-CRYPTO 2002. LNCS 2442, Springer, 2002. 162−177.

[23] Zhu ZQ, Lin RH, Hu CY. Openstack authentication protocol based on digital certificate. Journal on Communications, 2019, 40(2): 188−196.

[24] Caro DA, Iovino V. JPBC: Java pairing based cryptography. In: Proc. of the IEEE Symp. on Computers and Communications (ISCC). IEEE, 2011. 850−855.

[25] He DB, Zeadally S, Xu BW, Huang XY. An efficient identity-based conditional privacy-preserving authentication scheme for vehicular ad hoc networks. IEEE Trans. on Information Forensics and Security, 2015, 10(12): 2681−2691.

附录 1. 定理 1 的证明

假设存在 PPT 敌手 A 能够以ε的优势打破方案的存在性不可伪造性, 构造挑战者 C 解决 q_s-CAA 问题. 挑战者首先获得 q_s-CAA 问题实例 $(g,g^x,h_1,...,h_k\in Z_p,g^{1/(h_1+x)},...,g^{1/(h_k+x)})$,令 $pk=g^x$.

- 随机预言询问: 挑战者维护一个列表 L, 并准备一个集合 $\{w_1,w_2,...,w_{q_h}\}\in Z_p^{q_h}$, 其中, 随机选择 q_s 个元素令其分别为 $h_1,h_2,...,h_{q_s}$, 其余元素随机选择. 敌手输入 m_i, $1\leqslant i\leqslant q_h$, 挑战者将 w_i 回复给敌手, 并记录$(m_i,w_i)\in L$.
- 签名询问: 敌手输入 m_i 询问其签名. 挑战者查询表格 L, 如果 $w_i\neq h_i$, 则游戏失败; 否则, 挑战者回复 $g^{1/(h_i+x)}$ 给敌手.
- 伪造输出: 敌手输出一个伪造签名(m^*,σ^*), 其中$(m^*,w=H(m^*))\in L$, 且 $w\notin\{h_1,...,h_{q_s}\}$.

最后, 挑战者直接输出(w,σ^*)作为 q_s-CAA 问题实例的解即可. 因为假设敌手输出的是一个合法的伪造签名, 所以其一定满足验证等式 $e(\sigma^*,g^{H(m^*)}y)=e(g,g)$. 因此, (w,σ^*)一定是给定的 q_s-CAA 问题实例的解.

从上述证明过程可以看出, 挑战者的成功取决于敌手的每次签名询问都满足 $w_i=h_i$. 因此, 挑战者成功的概率为 $\varepsilon'\geqslant(q_s/q_h)^{q_s}\cdot\varepsilon$.

唐飞(1986－), 男, 博士, 副教授, CCF 专业会员, 主要研究领域为公钥密码, 隐私保护, 区块链.

马春亮(1998－), 男, 硕士, 主要研究领域为公钥密码, 区块链.

马帅(1993－), 男, 硕士, 主要研究领域为公钥密码, 区块链.