软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2023,34(6):2892−2905 [doi: 10.13328/j.cnki.jos.006516]

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

ECDSA 签名方案的颠覆攻击与改进*

严都力[1,2], 禹 勇[1,2], 李艳楠[3], 李慧琳[1], 赵艳琦[1], 田爱奎[4]

[1](陕西师范大学 计算机科学学院, 陕西 西安 710119)

[2](密码科学技术国家重点实验室, 北京 100878)

[3](School of Computer and Information Technology, University of Wollongong, Wollongong 2522, Australia)

[4](山东理工大学 计算机科学与技术学院, 山东 淄博 255049)

通信作者: 禹勇, E-mail: yuyongxy@163.com; 田爱奎, E-mail: takui@sdut.edu.cn

摘 要: 斯诺登事件揭露了某些密码体制的确存在被颠覆的事实. 椭圆曲线数字签名算法 (elliptic curve digital signature algorithm, ECDSA) 在同等安全强度下, 因其签名长度短而被广泛应用, 如被用于比特币交易单的签名. ECDSA 签名算法是否会被颠覆且存在修复方法仍是一个挑战. 正面回答了这一问题: 首先利用伪随机函数 (pseudorandom function, PRF) 计算$\tilde{k}$替换 ECDSA 签名中使用的随机数k, 实现了对 ECDSA 签名的颠覆, 使得敌手只需获得至多 3 个连续签名就能够提取出签名私钥; 然后, 将签名私钥、签名消息与其他随机签名组件的哈希值作为签名算法的第 2 个随机数, 对 ECDSA 签名进行了改进, 提出了抗颠覆攻击的 ECDSA 签名, 即使敌手替换新签名算法的某个组件, 也无法提取签名私钥的任何信息; 最后, 对提出的算法与已有算法进行了效率测试, 实验结果证明了提出的算法在计算复杂度与算法执行效率方面都具备优势.

关键词: 斯诺登事件; ECDSA 签名; 比特币; 颠覆攻击; 哈希函数

中图法分类号: TP309

中文引用格式: 严都力, 禹勇, 李艳楠, 李慧琳, 赵艳琦, 田爱奎. ECDSA签名方案的颠覆攻击与改进. 软件学报, 2023, 34(6): 2892–2905. http://www.jos.org.cn/1000-9825/6516.htm

英文引用格式: Yan DL, Yu Y, Li YN, Li HL, Zhao YQ, Tian AK. Subversion Attack and Improvement of ECDSA Signature Scheme. Ruan Jian Xue Bao/Journal of Software, 2023, 34(6): 2892–2905 (in Chinese). http://www.jos.org.cn/1000-9825/6516.htm

Subversion Attack and Improvement of ECDSA Signature Scheme

YAN Du-Li[1,2], YU Yong[1,2], LI Yan-Nan[3], LI Hui-Lin[1], ZHAO Yan-Qi[1], TIAN Ai-Kui[4]

[1](School of Computer Science, Shaanxi Normal University, Xi'an 710119, China)

[2](State Key Laboratory of Cryptology, Beijing 100878, China)

[3](School of Computer and Information Technology, University of Wollongong, Wollongong 2522, Australia)

[4](School of Computer Science and Technology, Shandong University of Technology, Zibo 255049, China)

Abstract: The Snowden incident revealed the fact that certain cryptosystems were indeed subverted. Elliptic curve digital signature algorithm (ECDSA) has been widely used due to its short signature length advantage under the same security level, for example, signing bitcoin transactions. However, whether the ECDSA can be subverted and how to resist this attack remain a challenge. This study answers this question positively. Firstly, it is shown that how to use a pseudorandom function (PRF) to calculate a random value to replace the randomness used in the ECDSA. The subverted ECDSA enables an adversary to extract signing private key by obtaining at most three consecutive signatures. Secondly, the hash value of private key, message, and the random signature component are used as the second random number to improve the ECDSA scheme, and as a result, the signature scheme against subversion-resistant attack is proposed. Even

* 基金项目: 国家自然科学基金 (61872229, U19B2021); 教育部 2020 年度区块链核心技术战略研究项目 (2020KJ010301); 陕西省重点研发计划 (2020ZDLGY09-06, 2021ZDLGY06-04)
收稿时间: 2020-02-08; 修改时间: 2020-04-28; 采用时间: 2020-06-06; jos 在线出版时间: 2022-11-30
CNKI 网络首发时间: 2022-12-01

an adversary replaces the component of the new signature algorithm, it cannot extract any information of the signing key. Finally, the proposed algorithm and existing algorithm are implemented, and the implementation demonstrates that the proposed scheme has advantages in terms of computational complexity and efficiency.

Key words: Snowden incident; ECDSA signature; bitcoin; subversion attack; hash function

2013 年 6 月, 美国前中央情报局职员爱德华·斯诺登爆料了一项由美国国家安全局 (National Security Agency, NSA) 于 2007 年起开始实施的绝密电子监听计划, 被称为“棱镜计划 (PRISM)”[1]. 据斯诺登爆料, NSA 通过潜入互联网服务商的服务器和数据库, 窃取了数以亿计的用户信息, 并汇聚成庞大的数据库. 此外, NSA 还秘密入侵微软、谷歌等企业在各国数据中心之间的通信网络, 收集情报、挖掘数据[2]. 此事件涉及的许多窃密方法与密码学密切相关. NSA 甚至操纵国际标准化组织, 强行推行与密码学伪随机发生器有相同作用的 Dual_EC_DRBG[3]成为相关标准并广泛推广. Dual_EC_DRBG 由 NSA 内部职员设计并在设计过程中省略了形式化证明, 设计者在其标准推荐的参数中嵌入后门, 通过后门和已知的伪随机发生器输出结果, 预测后续随机发生器产生的随机数, 这导致伪随机数发生器的安全性不复存在[4]. 另外, NSA 职员通过对密码算法进行分析、篡改和替换等非常规攻击手段, 攻破密码系统窃取秘密信息, 导致一些在传统密码分析方法下安全的密码体制不再安全.

ECDSA[5]是基于离散对数困难问题的数字签名体制, 在同等安全级别下, 签名长度短的特性让其在众多实际应用中被广泛采用, 如在比特币系统中[6], 交易单的数字签名算法采用的就是 ECDSA 签名算法, 其签名私钥确保了比特币资产的所有权. 若签名私钥被盗或丢失, 比特币安全将受到巨大威胁. 据区块链最新数据统计, 比特币交易中奖励无人认领、盗窃等形式遗失的比特币近 170 万个, 根据当前比特币价格 (1 个比特币约 8 000 美元), 170 万个比特币价值约 136 亿美元[7]比特币遗失被盗的原因之一是其交易授权采用的 ECDSA 签名算法容易遭受各类攻击引起签名私钥泄露. 一旦比特币拥有者的私钥丢失, 便失去了比特币的所有权, 所以, 保护比特币很重要的措施之一是保护用户的 ECDSA 签名私钥. 然而自斯诺登事件后, ECDSA 签名也随之面临被颠覆攻击等手段窃取签名私钥的威胁.

颠覆攻击起源于 Simmons 等人[8]提出的阈下信道, 通信双方借助密码技术构建一条隐蔽信道将传递的秘密信息隐藏, 除特定接收者外其他人无法知道传递的秘密信息. 颠覆攻击[9]指攻击者利用篡改过的密码方案或协议替换正常方案或协议的某些部分, 通过对密码方案、协议或原语的破坏获取私钥等秘密信息, 达到攻击目的. 在颠覆过程中, 即使拥有私钥的用户也无法检测到正常签名方案被恶意替换, 仍然正常执行被颠覆后的签名方案. 实现颠覆攻击的手段包括算法颠覆与公开参数颠覆[4]. Young 等人[10]把这种攻击定义为“Kleptography”, 并对 RSA 加密、Diffie-Hellman 密钥交换、ElGamal 签名等算法给出了具体的攻击方法[11−14]. Bellare 等人[9]给出了颠覆攻击的形式化定义, 并针对无状态、随机的对称密钥加密方案提出了一种通用的、不可检测的攻击模型, 同时采用有状态、确定的唯一密文来防御攻击者对算法的颠覆攻击. Degabriele 等人[15]克服了文献 [9] 安全模型中设置攻击者颠覆能力限制的局限性, 提出了适用范围更广泛的输入触发颠覆攻击. Ateniese 等人[16]针对随机化数字签名的颠覆攻击, 设计了唯一签名和不可篡改的密码逆向防火墙来抵抗颠覆攻击带来的安全威胁. Liu 等人[17]针对可拆分的特殊签名体制提出了非对称颠覆攻击, 该攻击的优势是: 拥有攻击者所持有的秘密后门信息无需嵌入颠覆算法便可提取签名私钥. Baek 等人[18]研究了 DSA[19]签名算法的颠覆攻击, 并给出了具体的颠覆方法和抵抗用户利用签名时间分析检测颠覆攻击的对策. Bellare 等人[20] 提出了参数颠覆的概念, 并分析了非交互式零知识证明协议在公共参考串模型下遭受颠覆的安全性. Dodis 等人[21]和 Degabriele 等人[22]基于参数颠覆研究了伪随机数生成器的颠覆攻击问题.

针对加密体制的颠覆攻击, Bellare 等人[9]设计了唯一密文防御措施来抵制颠覆攻击. Ateniese 等人[16]设计了唯一签名方案抵制签名体制的算法颠覆攻击. Russell 等人[23]基于检测安全思想提出了看门狗 (Watchdog) 的安全模型, 该模型具备直接检测并抵制算法颠覆的优势. Chow 等人[24]结合 Watchdog 模型的分割融合技术, 对签名体制提出了一种通用性防御方法, 并利用全域哈希设计了完全颠覆模型下安全的签名体制. Russell 等人[25]基于分割融合技术, 提出了一种对随机化密码算法的颠覆攻击防范方法. Mironov 等人[26]提出了一种功能保留、安全保留、抗泄漏的密码逆向防火墙 (cryptographic reverse firewall, CRF) 来抵御颠覆攻击. Fischlin 等人[27]提出了区别于

CRF 与 Watchdog 防御机制的自防御方法, 同时为随机化单钥加密体制和具有同态性质的公钥加密体制设计了相应的安全防护方法. 上述颠覆攻击研究虽涉及到一般性签名体制、加密体制、密码参数及密码原语, 针对相应攻击的应对防御措施也被提出, 但是针对某些特定签名算法的攻击方法与防御措施还不完善.

本文的主要工作如下.

(1) 提出了针对 ECDSA 签名算法的颠覆方法, 攻击者利用这一颠覆方法能够提取出签名私钥, 此外, 除攻击者外, 任何人无法区分正常 ECDSA 签名算法与颠覆的 ECDSA 签名算法.

(2) 利用哈希函数将签名私钥、签名消息与其他某个随机签名组件的哈希结果作为第二个随机数, 对正常 ECDSA 签名算法改进, 构造了具备抗颠覆特性的签名方案. 在提出的抗颠覆 ECDSA 签名方案中, 即使攻击者替换算法某些组件得到有效签名, 也无法获得签名私钥的任何信息.

(3) 将抗颠覆的 ECDSA 签名算法与已有签名算法进行了效率测试, 实验结果验证了抗颠覆的 ECDSA 签名算法在计算复杂度与算法执行效率方面都具备优势.

本文第 1 节简要介绍伪随机函数、哈希函数等预备知识. 第 2 节对颠覆签名及其安全目标进行介绍. 第 3 节提出了颠覆的 ECDSA 签名方案, 并对其安全目标进行分析与证明. 第 4 节利用哈希函数对 ECDSA 签名方案进行改进, 并对提出的抗颠覆 ECDSA 签名方案的存在性不可伪造和抗颠覆安全特性进行分析. 第 5 节对抗颠覆的 ECDSA 签名方案与已有签名方案的性能进行比较. 第 6 节对本文工作进行总结.

1 预备知识

1.1 伪随机函数

定义 1. 函数 $F:0,1^{\lambda}\times 0,1^{\rho}\to\{0,1\}^{n}$ 记作一个有效的带密钥函数, 函数 $Func_{\rho}^{n}$ 表示所有 f: $\{0,1\}^{\rho}\to\{0,1\}^{n}$ 函数的集合. 如果对所有概率多项式时间区分器 D, 存在一个可忽略的函数 $\varepsilon(\lambda)$, 满足:

$$|\Pr[D^{F(\cdot)}(1^{\lambda})=1]-\Pr[D^{f(\cdot)}(1^{\lambda})=1]|\leqslant\varepsilon(\lambda),$$

则称 F 是一个伪随机函数 (pseudorandom function, PRF)[28]. 区分器 D 区分函数 F 与 f 优势定义为:

$$Adv_{D}^{\mathrm{prf}}(\lambda)=|\Pr[D^{F(\cdot)}(1^{\lambda})=1]-\Pr[D^{f(\cdot)}(1^{\lambda})=1]|.$$

1.2 哈希函数

定义 2. 一个安全的密码学哈希函数[29] $H:\{0,1\}^{*}\to\{0,1\}^{n}$, 满足以下 3 个性质.

(1) 抗碰撞性: 找到两个不同输入 x 与 x', 满足 $H(x)=H(x')$ 在计算上是不可行的.

(2) 抗原像性: 给定 $y=H(x)$, 对于概率多项式时间敌手找到一个 x' 使得 $H(x')=y$ 在计算上不可行的.

(3) 抗第二原像性: 给定输入 x, 对于概率多项式时间敌手找到一个 $x'(x'\neq x)$ 使得 $H(x')=H(x)$ 在计算上是不可行的.

1.3 数字签名

定义 3. 一个数字签名方案[30] $SS=(KeyGen,Sign,Verify)$ 由 3 个多项式时间算法组成, 具体描述如下.

(1) $KeyGen(1^{\lambda})$: 输入安全参数 λ, 输出密钥对 (pk,sk), 其中 sk 是签名私钥, pk 是签名公钥.

(2) $Sign(sk,m)$: 输入私钥 sk 和签名消息 $m\in M$, 输出签名 σ, 其中 M 表示签名的消息空间.

(3) $Verify(pk,m,\sigma)$: 输入公钥 pk, 消息 $m\in M$ 和签名 σ, 该算法输出一个比特 b. 若 $b=1$ 表示签名通过验证; 否则, 签名无效.

签名方案 SS 需满足正确性要求:

$$\Pr[Verify(pk,m,Sign(sk,m))=1:(pk,sk)\leftarrow KeyGen(1^{\lambda}),m\in M]=1.$$

定义 4. 如果对于任意多项式 $\ell(\cdot)$ 以及任意概率多项式时间敌手 $\mathcal{A}$, $\mathcal{A}$ 在下述实验中成功的概率 $\varepsilon_{\mathcal{A}}(\lambda)$ 是可忽略的, 那么, 称签名方案 $SS=(KeyGen,Sign,Verify)$ 满足适应性选择消息攻击下存在性不可伪造 (existential unforgeability adaptive chosen-message attack, EUF-CMA)[31].

(1) 初始化: 挑战者 C 生成系统参数, 执行密钥生成算法 $KeyGen(1^{\lambda})$, 生成签名密钥对 (pk,sk), $\mathcal{S}$ 将公钥 pk 发送给敌手 $\mathcal{A}$.

(2) 签名询问: 敌手 $\mathcal{A}$ 选择任意消息 $m_i \in \{m_1,\ldots,m_\ell\}$ 进行签名询问, 挑战者 C 计算签名 $\sigma_i = Sign(m_i)$ 返回给 $\mathcal{A}$.

(3) 签名伪造: 敌手 $\mathcal{A}$ 返回一个签名 (m,σ), 如果 $Verify(pk,m,\sigma)=1$ 且 $m \notin \{m_1,\ldots,m_\ell\}$, 则 $\mathcal{A}$ 伪造签名成功, 否则失败.

敌手 $\mathcal{A}$ 成功的概率定义为安全参数 λ 的函数:

$$\varepsilon_A(\lambda)=\Pr\left[\begin{array}{l}\{m_i\}_{i=1}^{\ell}\leftarrow M;(pk,sk)\leftarrow KeyGen(1^{\lambda});\forall i\in[\ell]:\sigma_i\leftarrow Sign_{sk}(m_i);\\ (m,\sigma)\leftarrow A\left(pk,\{(m_i,\sigma_i)\}_{i=1}^{\ell}\right):Verify(pk,m,\sigma)=1\wedge m\notin\{m_1,\ldots,m_\ell\}\end{array}\right].$$

1.4 离散对数

定义 5. 令 $GroupGen$ 是一个多项式时间算法, 其输入为安全参数 λ, 输出为一个 q 阶循环群 G 以及群 G 的生成元 g, 其中 $g\in G$. 给定群 G 的生成元 g 和群中的随机元素 h, 计算 $\log_g(h)$ 被称为群 G 上的离散对数问题[30]. 离散对数假设是指对于任意多项式时间敌手 $\mathcal{A}$, 求解 $GroupGen$ 的离散对数问题的概率 $\varepsilon(\lambda)$ 是可忽略的.

敌手 $\mathcal{A}$ 求解离散对数问题的概率定义为安全参数 λ 的函数:

$$\varepsilon(\lambda)=\Pr[(G,g)\leftarrow GroupGen(1^{\lambda});h\leftarrow_R G;x\leftarrow A(G,g,h):h=g^x].$$

2 颠覆签名

颠覆签名[18]指颠覆者利用篡改的算法替换正常签名方案的密钥生成算法或签名算法中的某些组件, 最终颠覆者利用已知签名信息和颠覆密钥达到提取正常签名私钥的目的. 整个颠覆过程中, 除拥有颠覆密钥的颠覆者外, 任何人无法检测签名算法被颠覆. 颠覆签名的形式化定义如下.

定义 6. $SS=(KeyGen,Sign,Verify)$ 记为正常签名方案, 颠覆者通过某种手段破坏 SS, 得到颠覆的签名方案 $\overline{SS}=(\overline{KeyGen},\overline{Sign},\overline{Verify})$. 颠覆签名方案 $\overline{SS}$ 具体算法描述如下.

(1) $\overline{KeyGen}(1^{\ell})$: 输入安全参数 ℓ, 输出颠覆密钥 $subk$.

(2) $\overline{Sign}(subk,m,sk,\eta)$: 输入颠覆签名密钥 $subk$ 、正常签名私钥 sk 、签名消息 $m\in M$ 和签名状态 η, 输出签名 $\bar{\sigma}$ 和签名状态 η', 其中签名状态主要用作计数;

(3) $\overline{Verify}(pk,m,\bar{\sigma})$: 输入公钥 pk 、消息 m 和颠覆签名 $\bar{\sigma}$, 输出一个比特 b, 若 $b=1$ 表示颠覆签名有效, 否则无效.

颠覆的签名方案需满足正确性要求, 即:

$$\Pr\left[Verify(pk,m,\overline{Sign}(subk,m,sk,\eta))=1:(pk,sk)\leftarrow KeyGen(1^{\lambda}),subk\leftarrow\overline{KeyGen}(1^{\ell}),m\in M\right]=1.$$

一个颠覆签名方案除满足正确性要求外, 还需同时满足“密钥提取”和“不可检测性”. 其定义如下.

● 密钥提取 ($KeyExtract$)

通常情况下, 颠覆者从正常签名中分析获得签名私钥是计算上不可行的. 假设存在颠覆者可通过破坏签名方案提取签名私钥或签名中其他秘密信息, 颠覆者首先需要对签名方案进行颠覆获得有效的颠覆签名, 然后利用若干个颠覆签名与颠覆密钥提取签名私钥, 这类攻击是当前最为普遍的颠覆攻击方式. 其形式化定义如下.

定义 7. 令 $\mathcal{A}$ 是一个概率多项式时间敌手 (颠覆者), $SS=(KeyGen,Sign,Verify)$ 记为正常签名案, $\overline{SS}=(\overline{KeyGen},\overline{Sign},\overline{Verify})$ 记作颠覆的签名方案, 敌手 $\mathcal{A}$ 在时间 t 内最多进行 q_s 次签名询问, $\mathcal{A}$ 与挑战者 C 进行如下游戏.

(1) 初始化: C 执行 $KeyGen(1^{\lambda})$ 与 $\overline{KeyGen}(1^{\ell})$ 算法, 生成正常签名密钥对 (pk,sk) 和颠覆密钥 $subk$, C 将公钥 pk 发送给 $\mathcal{A}$.

(2) 签名询问: $\mathcal{A}$ 任意选择消息 m_i 进行签名询问, C 将 m_i 的签名 $\bar{\sigma}_i$ 应答给 $\mathcal{A}$.

(3) 密钥提取: $\mathcal{A}$ 通过一系列询问得到消息签名对 $(m_1,\bar{\sigma}_1),\ldots,(m_{q_s},\bar{\sigma}_{q_s})$, 利用签名公钥 pk 和颠覆密钥 $subk$ 计

算出 sk'. 若密钥提取成功, 即 $sk'=sk$, 输出 b=1, 否则输出 b=0.

敌手 $\mathcal{A}$ 成功提取密钥的优势定义为安全参数 λ 的函数:

$$Adv^{KeyExtract}_{A,SS,\overline{SS}}(\lambda)=\Pr[KeyExtract_{A,SS,\overline{SS}}(\lambda)=1].$$

● 不可检测性

① 状态重置 (state reset): 状态重置指通过某种操作将记录状态的参数值初始化. 签名方案分为有状态和无状态两种. 通常有状态的签名方案在程序重新启动或克隆创建虚拟机时会状态重置; 无状态签名方案不需要维护任何状态, 不必考虑状态重置. 一般情况, 状态重置会向敌手或用户泄露一些信息. 从签名者视角考虑, 无状态签名相对于有状态签名泄露信息更少. 本文中用户作为检测者, 颠覆的签名方案进行状态重置可能会泄露某些信息给检测者, 本文签名状态用于对签名计数, 即使状态重置后签名状态初始化, 也不会泄漏签名的任何秘密信息.

② 状态重置情况下不可检测性 (undetectability): 该安全目标从敌手 $\mathcal{A}$ 的视角定义, 一般情况下拥有签名私钥的用户对检测颠覆签名更加感兴趣, 即在不可检测性定义中, 用户被赋予拥有签名私钥能力的检测者来检测签名. 此外, 用户可以通过访问状态预言机来实现状态重置, 当用户调用状态预言机进行状态重置询问, 颠覆签名状态恢复到初始化, 这增强了用户检测签名的能力. 对于没有颠覆密钥的用户, 他们无法检测签名是正常签名算法还是颠覆签名算法生成的签名. 用户与挑战者之间刻画检测颠覆签名的游戏如下.

定义 8. 令 B 是拥有签名私钥的用户, $SS=(KeyGen,Sign,Verify)$ 是正常签案, $\overline{SS}=(\overline{KeyGen},\overline{Sign},\overline{Verify})$ 为颠覆签名方案, B 在时间 t 内最多进行 q_s 次签名询问和 q_r 次状态重置询问, B 在多项式时间内检测 $\overline{SS}$ 与挑战者 C 游戏如下:

(1) 初始化: 挑战者 C 执行 $KeyGen(1^{\lambda})$ 和 $\overline{KeyGen}(1^{\ell})$ 算法生成正常签名密钥对 (pk,sk) 和颠覆密钥 $subk$, S 将公钥 pk 发送给 B.

(2) 训练: B 询问 C 关于消息 m_i 的签名和状态重置, C 将对应的签名 σ_i 与状态 η_i 返回给 B.

(3) 挑战: B 输入消息 m_i, C 选择 $b\in\{0,1\}$, 若 $b=0$, C 执行 $\sigma_i=Sign(sk,m_i)$ 得到签名 σ_i 作为签名询问的应答; 否则, C 执行 $(\bar{\sigma}_i,\eta_i)=\overline{Sign}(sk,m_i,subk,\eta_{i-1})$ 将签名 $\bar{\sigma}_i$ 作为应答; B 进行状态重置查询, C 将颠覆签名状态 η_i 初始化, 设置为 Φ.

(4) 猜测: B 输出一个比特 b', 如果 $b'=b$, B 检测颠覆签名成功.

B 检测颠覆签名方案 $\overline{SS}$ 优的势定义为参数 λ 的函数:

$$Adv^{detect}_{B,SS,\overline{SS}}(\lambda)=\Pr[Detect_{B,SS,\overline{SS}}(\lambda)=1]-1/2.$$

3 颠覆的 ECDSA 签名方案

受文献 [18] 颠覆思想的启发, 本节针对 ECDSA 签名算法进行颠覆, 最终达到密钥提取和不可检测性的攻击目标. 第 3.1 节回顾了正常的 ECDSA 签名算法, 第 3.2 节针对提出的颠覆 ECDSA 签名算法进行详细描述, 第 3.3 节对颠覆的 ECDSA 签名算法的安全目标进行分析与证明.

3.1 ECDSA 签名算法

定义ECDSA = $(KeyGen_{\mathrm{ECDSA}},Sign_{\mathrm{ECDSA}},Verify_{\mathrm{ECDSA}})$为正常的 ECDSA 签名方案. ECC 参数记为 $PP=(p,F_p,a,b,G,n,H,I)$, 其中 p 为 F_p 特征, 参数 $a,b\in F_p$ 确定了安全的椭圆曲线 $E(F_p):y^2=x^3+ax+b$, 其中 a,b 满足 $4a^3+27b^2\neq 0$; 定义点 G 为椭圆曲线基点, 素数 n 表示基点 G 的阶; $H:\{0,1\}^*\to Z_n$ 记作安全的哈希函数, 函数 $I:<G>\to Z_n$ 表示将椭圆曲线上坐标点转化为整数. 正常 ECDSA 签名算法描述如下.

● 密钥生成算法 ($KeyGen_{\mathrm{ECDSA}}$)

① 随机选取整数 d, 其中 $1\leqslant d\leqslant n-1$;

② 计算 $Q=dG$, Q 表示签名公钥, d 表示签名私钥.

● 签名算法 ($Sign_{\mathrm{ECDSA}}$)

① 随机选取整数 k, 其中 $1\leqslant k\leqslant n-1$;

② 计算 $R=kG=(x_1,y_1)$, 令 $r=I(R)$, 若 $r=0$ 返回到①;

③ 计算 $e=H(m)$;

④ 计算 $k^{-1} \bmod n$;

⑤ 计算 $s=k^{-1}(e+dr) \bmod n$, 若 $s=0$ 返回到①;

⑥ 签名者对消息 m 的签名为 $\sigma=(r,s)$, 发送签名 σ 给验证者.

● 验证算法 ($Verify_{\mathrm{ECDSA}}$)

① 验证者收到签名 $\sigma=(r,s)$ 后, 验证 r 和 s 是否满足 $1\leqslant r,s\leqslant n-1$;

② 计算 $e=H(m)$;

③ 计算 $w=s^{-1} \bmod n$;

④ 计算 $u_1=ew \bmod n$ 和 $u_2=rw \bmod n$;

⑤ 计算 $X=u_1G+u_2Q=(x_1',y_2')$;

⑥ 若 $X=O$, 拒绝签名, 否则, 令 $v=I(X)$;

⑦ 如果 $v=r$, 签名验证通过, 否则, 该签名无效.

ECDSA 签名算法已在一般群模型下被证明 EUF-CMA, 详细证明过程见参考文献 [5].

3.2 颠覆的 ECDSA 签名算法

定义 $\overline{\mathrm{ECDSA}}=(\overline{KeyGen_{\mathrm{ECDSA}}},\overline{Sign_{\mathrm{ECDSA}}},\overline{Verify_{\mathrm{ECDSA}}})$ 为颠覆的 ECDSA 签名方案, $\overline{\mathrm{ECDSA}}$ 中参数设置与正常 ECDSA 签名方案参数设置相同, $\overline{\mathrm{ECDSA}}$ 签名算法具体描述如下.

● 密钥生成算法 ($\overline{KeyGen_{\mathrm{ECDSA}}}$)

输入参数 1^{ℓ}, 颠覆者任意选取 $\kappa\in\{0,1\}^{\ell}$ 与伪随机函数 $F:\{0,1\}^{\ell}\times\{0,1\}^{*}\to\{0,1\}^{n}$, 输出颠覆密钥 $subk=(F,\kappa)$.

● 签名算法 ($\overline{Sign_{\mathrm{ECDSA}}}$)

输入 ECDSA 签名的公共参数 PP、消息 m、私钥 d 和颠覆密钥 $subk$, 设置签名状态 $\eta=(i,\tau)$ 对签名计数, 设置为 $i=0,\tau=\Phi$. $\overline{\mathrm{ECDSA}}$ 具体签名算法如算法 1.

算法 1. $\overline{\mathrm{ECDSA}}$ 签名算法.

If $i \bmod 2=0$

① 随机选择 $k\in Z_n^*$;

② 计算 $r=I(kG)$;

③ 计算 $e=H(m)$;

④ 计算 $s=k^{-1}(e+dr) \bmod n$;

⑤ 设置 $\tau=r$;

⑥ m 的签名为 $\bar{\sigma}=(r,s)$.

Else

① 计算 $\tilde{k}=F(\kappa,\tau) \bmod n$;

② 计算 $r'=I\left(\tilde{k}G\right)$;

③ 计算 $e=H(m)$;

④ 计算 $s'=\tilde{k}^{-1}(e+dr') \bmod n$;

⑤ m 的签名为 $\bar{\sigma}=(r',s')$.

令 $i=i+1$;

令 $\eta=(i,\tau)$;

返回消息 m 的签名 $\bar{\sigma}$ 和状态 η.

上述 $\overline{\text{ECDSA}}$ 签名算法中, 颠覆者只替换了正常 ECDSA 签名算法中的随机数 k, 整个颠覆的 ECDSA 签名算法的执行, 由用户正常执行并获得消息 m 的有效签名, 所以在 $\overline{\text{ECDSA}}$ 签名算法的参数输入与签名过程中都有用户签名私钥 d 的参与. 该算法即使有私钥 d 参与, 颠覆者也无法直接获取签名私钥. 颠覆者若想达到颠覆目的, 必须利用用户执行签名算法得到的有效签名 $\bar{\sigma}$ 和自身的颠覆密钥 $subk$ 才能够提取出私钥 d. 此外, 颠覆者通过判断签名消息的索引 i 对正常 ECDSA 算法破坏, 当 $i \bmod 2 = 0$ 时, 颠覆者对 ECDSA 签名算法不做任何替换; 当 $i \bmod 2 = 1$ 时, $\mathcal{A}$ 利用伪随机函数 F 计算 $\tilde{k}$ 替换正常 ECDSA 签名中的随机数 k, 其中 $k \in Z_n^*$, $\tilde{k} = F(\kappa,\tau) \bmod n$. 假设 i 为偶数, $\overline{\text{ECDSA}}$ 签名算法具体执行过程如表 1 所示.

表 1 $\overline{\text{ECDSA}}$ 签名算法执行过程

i	0	1	…	i	$i+1$
$i \bmod 2$	0	1	…	0	1
m_i	m_0	m_1	…	m_i	m_{i+1}
k_i	$k_0 \in Z_n^*$	$k_1 = F(\kappa,\tau_0) \bmod n$	…	$k_i \in Z_n^*$	$k_{i+1} = F(\kappa,\tau_i) \bmod n$
r_i	$r_0 = I(k_0 G)$	$r_1 = I(k_1 G)$	…	$r_i = I(k_i G)$	$r_{i+1} = I(k_{i+1} G)$
τ	$\tau_0 = r_0$	r_0	…	$\tau_i = r_i$	r_i
s_i	$s_0 = k_0^{-1}(H(m_0) + x r_0) \bmod n$	…	…	s_i	s_{i+1}
η_i	$(1, r_0)$	$(2, r_0)$	…	$(i+1, r_i)$	$(i+2, r_i)$
$\bar{\sigma}_i$	(r_0, s_0)	(r_1, s_1)	…	(r_i, s_i)	(r_{i+1}, s_{i+1})

- 验证算法 ($\overline{Verify_{\text{ECDSA}}}$)

$\overline{\text{ECDSA}}$ 验证算法与 ECDSA 验证算法相同, 验证者输入消息 m、签名 $\bar{\sigma}$ 和公钥 Q 执行验证算法 $\overline{Verify_{\text{ECDSA}}}$, 若签名满足 $\overline{Verify}(Q,\bar{\sigma},m) = 1$, 则颠覆的 ECDSA 签名 $\bar{\sigma}$ 为消息 m 的有效签名.

3.3 安全性分析

利用上述颠覆方法, 敌手 $\mathcal{A}$ (颠覆者) 最多只需 3 个连续有效的颠覆签名, 即可成功提取 ECDSA 签名私钥 d. 若 $\mathcal{A}$ 收集的 3 个连续颠覆的 ECDSA 签名索引以偶数 ($i \bmod 2 = 0$) 开始, 只需前两个连续签名便可以提取私钥; 若签名索引以奇数 ($i \bmod 2 = 1$) 开始, 则 $\mathcal{A}$ 选择奇数索引 i 后的第 $i+1$ 与第 $i+2$ 个签名进行私钥提取, $\mathcal{A}$ 最多只需 3 个连续颠覆的 ECDSA 签名便可提取签名私钥.

定理 1. 假设敌手 $\mathcal{A}$ 在 t 时间内最多进行 q_s 次签名询问, 得到任意 3 个连续有效的 $\overline{\text{ECDSA}}$ 签名, 则 $\mathcal{A}$ 可以以 100% 的概率提取 ECDSA 的签名私钥 d.

证明: 若敌手 A 通过对消息 m_i, m_{i+1}, m_{i+2} 的签名进行问询, 获得 3 个连续对应的 $\overline{\text{ECDSA}}$ 签名, 分别为 $\bar{\sigma}_i = (r_i, s_i)$, $\bar{\sigma}_{i+1} = (r_{i+1}, s_{i+1})$ 与 $\bar{\sigma}_{i+2} = (r_{i+2}, s_{i+2})$. 假设 $i \bmod 2 = 0$, 敌手 $\mathcal{A}$ 利用颠覆密钥 $subk = (F,\kappa)$ 和已知签名信息提取 ECDSA 签名私钥 d 过程如下.

① 计算 $\tilde{k}_{i+1} = F(\kappa, r_i)$;

② 计算 $\tilde{d} = \dfrac{\tilde{k}_{i+1} s_{i+1} - H(m_{i+1})}{r_{i+1}} \bmod n$;

③ 验证 $Q' = \tilde{d}G = Q$ 是否成立.

若上述步骤③成立, 则 $\tilde{d} = d$, 敌手 A 成功提取到正确的签名私钥 d.

同理, 假设 $i \bmod 2 \neq 0$, 即 $(i+1) \bmod 2 = 0$, $\mathcal{A}$ 通过 $\bar{\sigma}_{i+1} = (r_{i+1}, s_{i+1})$ 和 $\bar{\sigma}_{i+2} = (r_{i+2}, s_{i+2})$, 利用 $i \bmod 2 = 0$ 密钥提取的方法提取签名私钥 d 的概率仍为 100%.

因此, 敌手 $\mathcal{A}$ 在 t 时间内经过最多 q_s 次签名询问, 提取 ECDSA 签名私钥 d 的概率为 100%, 提取优势为 $Adv_{A,\overline{\text{ECDSA}}}^{\text{extract}}(\lambda) = 1$.

定理 2. 若函数 F 是一个伪随机函数 PRF, 用户 B 在时间 t 内经过 q_s 次签名询问与 q_r 次状态重置询问, B 检测出 $\overline{\text{ECDSA}}$ 的优势是可忽略的.

证明: 若 B 想检测颠覆的 ECDSA 签名方案 $\overline{\text{ECDSA}}$, 则需要区分 PRF 和真正的随机函数, 通过一系列游戏刻画完成状态重置情况下的不可检测性证明.

$Game_{\text{real}}$ 游戏表示用户 B 与模拟者 $\mathcal{S}$ 之间模拟真实的不可检测性游戏. 当 B 对消息 m_i 进行签名询问, $\mathcal{S}$ 任意选择 $b\in\{0,1\}$. 若 $b=0$, 模拟者 $\mathcal{S}$ 执行正常的 ECDSA 签名算法, 得到签名 $\sigma_i=Sign_{\text{ECDSA}}(d,m_i)$ 作为应答; 若 $b=1$, $\mathcal{S}$ 应答 B 的签名询问有两种情况: ① 若 B 询问的消息 m_i 的索引满足 $i \bmod 2=0$, 则 $\mathcal{S}$ 执行 $\sigma_i=Sign_{\text{ECDSA}}(d,m_i)$ 应答 B; ② 若 B 询问的消息 m_i 的索引满足 $i \bmod 2=1$, 则 S 执行 $(\bar{\sigma}_i,\eta_i)=\overline{Sign}_{\text{ECDSA}}(d,m_i,subk,\eta_{i-1})$ 应答 B. $Game_{\text{real}}$ 游戏中若 B 进行状态重置询问, S 将签名状态 $\eta_i=(i,\tau)$ 初始化, 设置为 $i=0,\tau=\Phi$. 此游戏中 $\mathcal{S}$ 对 B 签名询问的应答 σ_i 与 $\bar{\sigma}_i$, 区别在于签名算法中随机数 k 的来源, 正常 ECDSA 签名 σ_i 使用的随机数为 $k\in Z_n$, 而 $\overline{\text{ECDSA}}$ 签名 $\bar{\sigma}_i$ 中使用的随机数为 $\tilde{k}=F(\kappa,\tau) \bmod n$.

$Game_{\text{random}}$ 游戏中, 当 B 签名询问的消息 m_i 的索引满足 $i \bmod 2=1$, 且 $\mathcal{S}$ 选择 $b=1$ 对应的签名应答 B 时, $\mathcal{S}$ 执行 $\overline{\text{ECDSA}}$ 签名算法使用的随机数是随机选择的 k', 并非像 $Game_{\text{real}}$ 游戏中利用伪随机函数 F 计算. 首先令 $T=\Phi$, B 提交签名询问后, $\mathcal{S}$ 依据签名索引 i 寻找 T 中是否存在元组 (τ,π), 其中 $\tau=r$, r 表示当 $i \bmod 2=0$ 时消息 m_i 签名的第 1 部分. 如果 $(\tau,\pi)\in T$, 则 $\mathcal{S}$ 直接利用 τ 对应的 π 计算当前签名使用的随机数 k', 即 $k'=\pi \bmod n$; 如果 $(\tau,\pi)\notin T$, $\mathcal{S}$ 随机选择 $\pi\in\{0,1\}^n$, 计算 $k'=\pi \bmod n$, 然后更新列表 T 存储 (τ,π), 即 $T=T\cup\{(\tau,\pi)\}$. $\mathcal{S}$ 利用 k' 替换 $\bar{\sigma}_i$ 中使用的随机数 $\tilde{k}=F(\kappa,\tau) \bmod n$. $Game_{\text{random}}$ 游戏中 B 与 $\mathcal{S}$ 在其他算法的执行与 $Game_{\text{real}}$ 游戏算法完全相同.

$Game'_{\text{random}}$ 游戏在 $Game_{\text{random}}$ 游戏基础上, $\mathcal{S}$ 在计算 $\overline{\text{ECDSA}}$ 签名算法中使用的随机数时, 省略了判断 $(\tau,\pi)\in T$ 的情况, 仅考虑 $(\tau,\pi)\notin T$ 情况. 即 $\mathcal{S}$ 直接选择 $\pi\in\{0,1\}^n$, 计算 $k'=\pi \bmod n$ 作为 $\overline{\text{ECDSA}}$ 签名算法中使用的随机数 $\tilde{k}$.

最后, $Game_{\text{reset}}$ 游戏是在 $Game'_{\text{random}}$ 游戏的基础上, B 未进行状态重置询问, 当 B 在 $Game_{\text{reset}}$ 游戏中签名询问得到 $\mathcal{S}$ 的应答后, B 继续询问其他消息的签名.

$Game_{\text{real}}$ 游戏与 $Game_{\text{random}}$ 游戏的区别在于, 当 B 签名询问的消息 m_i 的索引满足 $i \bmod 2=1$, 且 $\mathcal{S}$ 选择 $b=1$ 对应的签名应答 B 时, $\mathcal{S}$ 执行 $\overline{\text{ECDSA}}$ 返回消息 m_i 的签名 $\bar{\sigma}_i$ 使用的随机数 k 的来源不同. $Game_{\text{real}}$ 游戏中签名 $\bar{\sigma}_i$ 使用的随机数 $\tilde{k}=F(\kappa,\tau) \bmod n$, 而 $Game_{\text{random}}$ 游戏中签名 $\bar{\sigma}_i$ 使用的随机数 $k'=\pi \bmod n$. 若存在 B 可以区分 $Game_{\text{random}}$ 游戏与 $Game_{\text{real}}$ 游戏, 必然存在一个区分器 D 可以区分 PRF 与真随机函数. 通过定义 1 可知, 对所有概率多项式时间内区分器 D, 区分 PRF 与真随机函数的优势 $Adv_D^{\text{prf}}(\lambda)$ 是可忽略的, 则 B 区分 $Game_{\text{random}}$ 游戏与 $Game_{\text{real}}$ 游戏的优势 $Adv_B^{\text{prf}}(\lambda)$ 也是可忽略的.

$Game_{\text{random}}$ 游戏与 $Game'_{\text{random}}$ 游戏的区别在于, 当 B 签名询问的消息 m_i 的索引满足 $i \bmod 2=1$, 且 $\mathcal{S}$ 选择 $b=1$ 对应的签名应答 B 时, $\mathcal{S}$ 在执行 $\overline{\text{ECDSA}}$ 计算随机数 $k'=\pi \bmod n$ 时, 是否判断 $(\tau,\pi)\in T$. $Game_{\text{random}}$ 游戏中 $\mathcal{S}$ 分别考虑 $(\tau,\pi)\in T$ 与 $(\tau,\pi)\notin T$ 的情况, 然后计算签名中使用的随机数 k'; $Game'_{\text{random}}$ 游戏仅考虑 $(\tau,\pi)\notin T$ 情况. $Game_{\text{random}}$ 游戏比 $Game'_{\text{random}}$ 游戏多考虑 $(\tau,\pi)\in T$ 的情况, 若令 $\tau_i=\tau_j$ 的概率表示 $(\tau,\pi)\in T$ 的情况, 其中 $j\leqslant q_s$. 若存在 $\tau_i=\tau_j$, 则表示 B 询问的消息 m_j 的签名之前被问询过, 其概率最大为 q_s/n, 由于 q_s/n 远小于 q_s^2/n, 且 q_s^2/n 大小可忽略, 所以 q_s/n 的大小是可忽略的.

$Game'_{\text{random}}$ 游戏与 $Game_{\text{reset}}$ 游戏的区别在于 B 在游戏过程中, 是否进行状态重置询问, $Game_{\text{reset}}$ 游戏中 B 未进行状态重置询问. 若 $Game'_{\text{random}}$ 游戏中 B 签名询问后未进行状态重置询问, B 对消息 m_i 签名询问后, $\mathcal{S}$ 选择 $b=1$ 对应的签名作为应答, 无论签名索引 $i \bmod 2=1$ 或 $i \bmod 2=0$, 签名使用的随机数 k' 与 k 具有相同的分布, 所以 $\mathcal{S}$ 应答的签名 σ_i 与 $\bar{\sigma}_i$ 具有相同的分布. 若 B 在 $Game'_{\text{random}}$ 游戏中进行状态重置询问, 签名状态 $\eta_i=(i,\tau)$ 被初始化设置为 $i=0,\tau=\Phi$, 因其状态重置后索引 i 满足 $i \bmod 2=0$, $\mathcal{S}$ 仍以正常签名 σ_i 应答 B. 所以整个 $Game'_{\text{random}}$ 游戏中, 无论 B 是否进行状态重置询问, $\mathcal{S}$ 针对其签名询问的应答都是正常的 ECDSA 签名 σ_i. $Game_{\text{reset}}$ 游戏是在 $Game'_{\text{random}}$ 游戏基础上, B 未进行状态重置询问. 由于 B 在 $Game'_{\text{random}}$ 游戏中, 状态重置询问并不影响 $\mathcal{S}$ 对 B 签名询问的应答, 所以 B 在 $Game_{\text{reset}}$ 游戏与 $Game'_{\text{random}}$ 游戏中检测 $\overline{\text{ECDSA}}$ 的概率相同.

$Game_{\text{reset}}$ 游戏中, 无论 $\mathcal{S}$ 选择 $b=0$ 或 $b=1$ 来应答 B 对消息 m_i 的询问, 应答的签名中使用的随机数 k 与 k' 都

满足 $k,k' \in Z_n^*$, 其分布相同, 所以签名 σ_i 与 $\bar{\sigma}_i$ 分布相同, 即 B 在 $Game_{\text{reset}}$ 游戏中检测 $\overline{\text{ECDSA}}$ 没有任何优势, 即 B 在 $Game_{\text{reset}}$ 游戏中检测 $\overline{\text{ECDSA}}$ 概率为 1/2.

通过上述 $Game_{\text{real}}$, $Game_{\text{random}}$, $Game'_{\text{random}}$ 与 $Game_{\text{reset}}$ 一系列游戏, 分析可得 B 检测 $\overline{\text{ECDSA}}$ 的优势:

$$Adv^{\text{detect}}_{B,\text{ECDSA},\overline{\text{ECDSA}}}(\lambda) = \Pr[Detect_{B,\text{ECDSA},\overline{\text{ECDSA}}}(\lambda) = 1] - 1/2 \leqslant Adv^{\text{prf}}_{B}(\lambda) + q_s^2/n.$$

综上得证, 若函数 F 是一个伪随机函数 PRF, 则 B 区分正常 ECDSA 签名方案与颠覆的 ECDSA 签名方案 $\overline{\text{ECDSA}}$ 的优势是可忽略的.

4 抗颠覆的 ECDSA 签名方案

$\overline{\text{ECDSA}}$ 中颠覆者借助伪随机函数 PRF 计算随机数 $\tilde{k}$, 替换正常 ECDSA 签名方案中的随机数 k, 然后利用已知签名信息提取正常 ECDSA 签名私钥, 拥有签名私钥的签名者也无法检测正常 ECDSA 签名方案已被颠覆. 为了抵抗对正常 ECDSA 签名算法的颠覆, 本节对正常 ECDSA 签名算法进行改进, 利用哈希函数计算签名的另一个随机数 $\alpha = H(d\|m\|E)$, 其中 d 是签名私钥, m 是签名消息, E 表示随机数 k 的承诺值. 针对不同签名消息 m 计算得到不同的 α, 且 α 只能被签名者计算, 然后将 α 作为签名组件应用于生成签名. 签名中随机组件 α 的计算与签名消息 m 和签名私钥 d 绑定目的为了保证 α 值的唯一性. 由于抗颠覆的 ECDSA 签名方案中使用的随机数 α 由特定方法计算, 颠覆者若尝试通过替换 α 达到攻击目的, 签名者可以通过计算和判断 α 来确定是否继续执行签名算法. 若签名者计算 $\alpha = H(d\|m\|E)$, 则继续执行签名算法; 否则, 终止. 此判断过程保证了签名算法中使用的随机数 α 的正确有效性, 抗颠覆的 ECDSA 签名方案描述如下.

4.1 抗颠覆的 ECDSA 签名算法

$\text{ECDSA}' = (KeyGen'_{\text{ECDSA}}, Sign'_{\text{ECDSA}}, Verify'_{\text{ECDSA}})$ 记为抗颠覆的 ECDSA 签名方案, ECDSA′ 中参数设置与 ECDSA 签名方案参数设置相同, 公共参数 $PP = (p, F_p, a, b, G, n, H, I)$. 其中 $H: 0,1^* \to Z_n$ 记作安全的哈希函数, 函数 $I: <G> \to Z_n$ 表示将椭圆曲线上坐标点转化为整数. 密钥生成算法 $KeyGen'_{\text{ECDSA}}$ 生成签名密钥对 (d,Q), 其中 Q 是公钥, d 是私钥. ECDSA′ 签名方案的签名算法与验证算法流程如图 1 所示, 虚线内表示对正常 ECDSA 签名算法的改进.

● 签名算法 ($Sign'_{\text{ECDSA}}$)

① 随机选取整数 k, 其中 $1 \leqslant k \leqslant n-1$;

② 计算 $E = kG = (x_1, y_1)$;

③ 计算 $\alpha = H(d\|m\|E)$;

④ 判断 α, 若 $\alpha \neq H(d\|m\|E)$ 返回到①;

⑤ 计算 $\alpha E = (x_2, y_2)$;

⑥ 令 $r = I(\alpha E)$, 如果 $r = 0$ 返回到①;

⑦ 计算 $e = H(m\|r)$;

⑧ 计算 $s = \alpha k e + rd \pmod n$, 如果 $s = 0$ 返回到①;

⑨ 签名者对消息 m 的签名 $\sigma' = (r,s)$, 发送签名 σ' 给验证者.

● 验证算法 ($Verify'_{\text{ECDSA}}$)

① 验证者接收到签名 $\sigma' = (r,s)$ 后, 验证 r 和 s 是否满足 $1 \leqslant r,s \leqslant n-1$;

② 计算 $e = H(m\|r)$;

③ 计算 $w = e^{-1} \bmod n$;

④ 计算 $u_1 = ws \bmod n$ 和 $u_2 = wr \bmod n$;

⑤ 计算 $X = u_1 G - u_2 Q = (\tilde{x}, \tilde{y})$;

⑥ 如果 $X = O$, 拒绝签名, 否则, 令 $v = I(X)$;

⑦ 如果 $v = r$, 签名验证通过, 否则, 该签名无效.

方案的正确性如下:

由 $s=\alpha ke+rd(\bmod n)$ 得:

$$\alpha k=e^{-1}(s-rd)=e^{-1}s-e^{-1}rd=ws-wrd=u_1-u_2d(\bmod n).$$

因此 $X=u_1G-u_2Q=e^{-1}sG-e^{-1}rdG=e^{-1}(s-rd)G=\alpha kG=(\tilde{x},\tilde{y})$.

令 $v=I(X)=\tilde{x}$;

验证者验证若 $v=r$, 则签名 $\sigma'=(r,s)$ 验证通过, 该签名方案具备正确性要求.

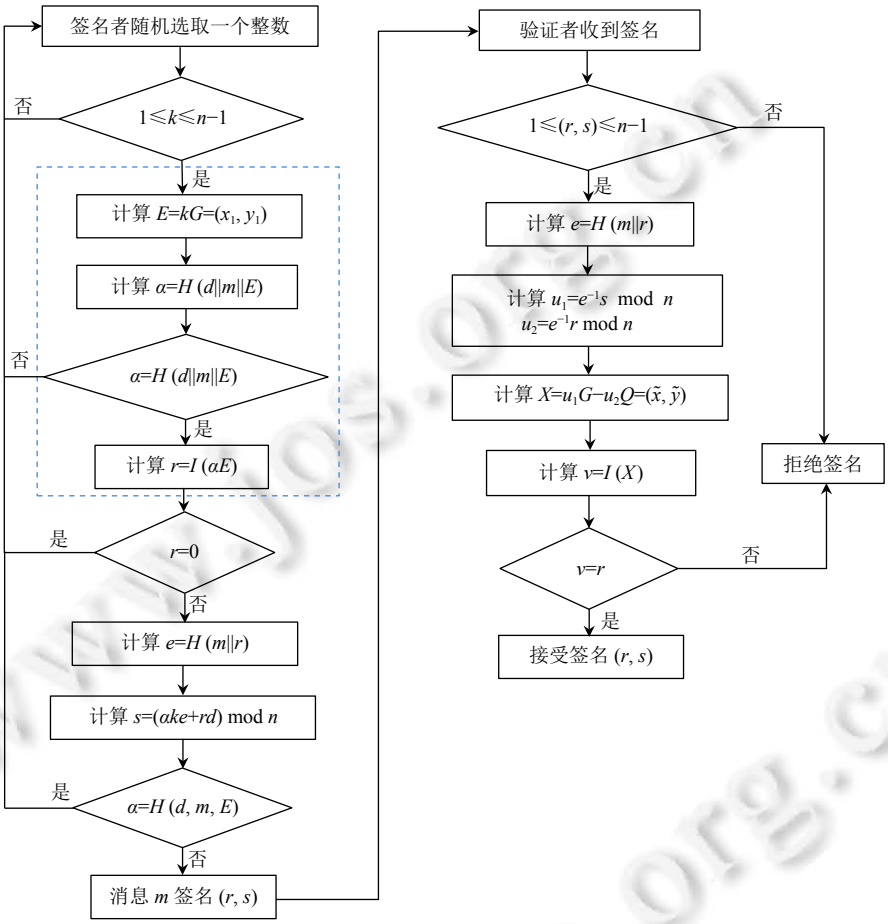


图 1 抗颠覆的 ECDSA 签名方案算法流程图

4.2 安全性分析

(1) 不可伪造性

定理 3. 在随机预言机模型下, 如果离散对数问题是困难的, 则抗颠覆的 ECDSA 签名是 EUF-CMA 安全.

证明: 若存在一个多项式时间敌手 $\mathcal{A}$, 在随机预言机模型中能够以不可忽略的概率 $\varepsilon(\lambda)$ (λ 是安全参数) 输出一个抗颠覆的 ECDSA 签名的有效伪造, 则一定存在一个模拟者 $\mathcal{S}$ 能够求解椭圆曲线离散对数困难问题. 具体证明如下.

模拟者 $\mathcal{S}$ 收到离散对数困难问题实例 $Q=dG$, 将 Q 作为签名公钥发送给敌手 $\mathcal{A}$. $\mathcal{A}$ 可以进行哈希询问、签名询问和函数 I 的询问. 我们借鉴文献 [32](Theorem 14) 的证明思路, 仅需证明本方案在随机预言机模型下, 无需签名者私钥的参与, 即可产生一个合法的签名元组, 且产生的签名元组与真实签名具有同样的分布. 然后根据分叉引理[32], 若敌手 $\mathcal{A}$ 输出一个有效伪造, 通过回滚 (rewind), 可以输出两个伪造签名元组, 以解决困难问题实例. 下面我们说明产生有效签名元组的方式具体操作: 给定一个消息 m, $\mathcal{S}$ 随机选择 $e,s,r\in Z_n^*$, 设置 $R=e^{-1}sG-e^{-1}rQ$, 并设

置函数 I, 使得 $I(R)=r$. 一个有效的签名元组即可表示为 (m,R,e,σ), 其中 $\sigma=(r,s)$, 且这一签名元组的各个元素均与真实签名同分布.

伪造: 敌手 $\mathcal{A}$ 输出一个关于消息 m 的伪造签名 (m,R,e,r,s), 根据分叉引理[32], $\mathcal{A}$ 可以输出另一个伪造 (m,R,e',r',s'), 其中 $e\neq e'$. $\mathcal{S}$ 根据 $\mathcal{A}$ 伪造的两组有效签名 (m,R,e,r,s) 和 (m,R,e',r',s') 建立方程:

$$e^{-1}sG-e^{-1}rQ=R=e'^{-1}s'G-e'^{-1}r'Q.$$

$\mathcal{S}$ 可以很容易地求解出上述方程中 Q 关于 G 的离散对数 d.

$$d=\frac{e^{-1}s-e'^{-1}s'}{e^{-1}r-e'^{-1}r'}.$$

综上, 如果存在敌手 $\mathcal{A}$ 能够以不可忽略的概率 $\varepsilon(\lambda)$ 有效伪造一个抗颠覆的 ECDSA 签名, 那么一定存在一个 $\mathcal{S}$ 解决离散对数问题, 证毕.

(2) 抗颠覆性

定理 4. 假设敌手 $\mathcal{A}$ 在概率多项式时间 t 内, 经过 q_s 次签名询问得到 3 个或若干个连续有效的抗颠覆的 ECDSA 签名, $\mathcal{A}$ 利用定义 7 密钥提取的方法提取出私钥 d 的概率是可忽略的.

证明: 若敌手 $\mathcal{A}$ 获得任意 3 个连续消息 m_i, m_{i+1}, m_{i+2} 的抗颠覆 ECDSA 签名, 分别为 $\sigma'_i=(r_i,s_i)$, $\sigma'_{i+1}=(r_{i+1},s_{i+1})$ 与 $\sigma'_{i+2}=(r_{i+2},s_{i+2})$. 若索引 $i \bmod 2=0$, $\mathcal{A}$ 按照以下方法步骤提取签名私钥.

① 计算 $k_{i+1}=F(\kappa,r_i)$;

② 依据等式 $s_{i+1}=\alpha_{i+1}k_{i+1}e_{i+1}+r_{i+1}d'(\bmod n)$;

③ 计算 $d'=\dfrac{s_{i+1}-\alpha_{i+1}k_{i+1}e_{i+1}}{r_{i+1}} \bmod n$.

上述步骤③等式中, 因 $\alpha_{i+1}=H(d\|m_{i+1}\|E_{i+1})$ 中 d 是签名私钥 $\mathcal{A}$ 未知, $\mathcal{A}$ 无法计算出 α_{i+1}, 即使 $\mathcal{A}$ 已知 k_{i+1}, e_{i+1}, s_{i+1} 与 r_{i+1}, 也无法计算 d', 无法判断 $d'=d$ 是否成立, 所以 $\mathcal{A}$ 提取私钥 d 失败.

同理, 若签名消息 m_i 的索引 $i \bmod 2\neq 0$, $\mathcal{A}$ 获得 3 个连续有效的抗颠覆 ECDSA 签名后, 首先舍弃第 i 个签名, 利用 $\sigma_{i+1}=(r_{i+1},s_{i+1})$ 与 $\sigma_{i+2}=(r_{i+2},s_{i+2})$ 依据上述签名索引 $i \bmod 2=0$ 密钥提取方法, 也无法提取签名私钥 d, 所以 $\mathcal{A}$ 提取出签名私钥 d 的概率也是可忽略的.

此外, 若敌手 $\mathcal{A}$ 具备更强的攻击能力, 可以替换每个抗颠覆的 ECDSA 签名中使用的随机数 k_i, 当 $\mathcal{A}$ 获取 x 个有效签名, 其中 $x\leqslant q_s$.

$$\begin{cases} s_1=\alpha_1k_1e_1+r_1d(\bmod n) \\ \quad\vdots \\ s_i=\alpha_ik_ie_i+r_id(\bmod n) \\ \quad\vdots \\ s_x=\alpha_xk_xe_x+r_xd(\bmod n) \end{cases}.$$

上述方程组中, 敌手 $\mathcal{A}$ 已知签名相关参数 e_i,r_i,s_i 和 k_i, 但 $\mathcal{A}$ 在概率多项式时间内无法计算签名中的另一个随机数 α_i, x 个方程等式中有 $x+1$ 个未知数, $\mathcal{A}$ 无法通过已知签名信息提取出签名私钥 d.

综上得证, 即使敌手 $\mathcal{A}$ 在 t 时间内经过 q_s 次签名询问得到 3 个或若干个连续有效的抗颠覆 ECDSA 签名, $\mathcal{A}$ 利用定义 7 密钥提取的方法提取出签名密钥 d 的概率是可忽略的, 证毕.

5 性能分析

本节从算法计算复杂度与算法执行效率两方面对抗颠覆的 ECDSA 签名方案与已有的签名方案进行对比分析.

(1) 计算复杂度分析

签名方案中算法执行效率的快慢受算法运算量影响, 一般算法运算包括加法、模乘、模逆及点积运算, 其中加法对算法执行效率影响可忽略不计, 模逆运算影响最大, 一次模逆运算约等于算法执行 9 次点积运算. 设模乘运算的数据规模为 n, 则一次点积运算复杂度为 $O(n^2)$, 一次模逆运算复杂度 $O(9n^2)$, 一次模乘法运算复杂度为 $O(n^2\log_2 n)$. 本文用 $[mc]$、$[mn]$ 和 $[dj]$ 分别表示模乘、模逆与点积运算, 其中 $[mn]=9[dj]$, T_1、T_2 与 T_3 分别表示

ECDSA 签名方案、文献 [33] 方案与本文签名方案算法的总运算量, 3 种签名方案运算量比较如表 2 所示.

表 2 中 $T_1=\mathrm{O}[(4\log_2 n+22)n^2]$, $T_2=\mathrm{O}[(8\log_2 n+32)n^2]$, $T_3=\mathrm{O}[(5\log_2 n+14)n^2]$, 其中文献 [33] 方案算法计算复杂度最大, 本文方案略大于 ECDSA 签名方案计算复杂度, 具体对比如图 2 所示.

(2) 算法执行效率分析

本节分别利用抗颠覆的 ECDSA 签名方案、文献 [33] 签名方案和 ECDSA 签名方案对一个大小为 5 KB 文件进行签名及验证测试, 同时将这 3 种签名方案执行效率对比分析. 本文中算法效率测试的实验环境为 Inter(R) Core(TM)i5-7400CPU@3.00 GHz 的 64 位 Win10 操作系统, 内存容量为 8.00 GB 的台式机. 实验工具采用 Visual Studio 2012 编译工具编译程序, 调用椭圆曲线密码体制的 Miracl 函数库辅助算法实现. 3 种签名方案的密钥生成、签名及验证算法执行效率分别对比如图 3 所示.

表 2 本文与已有的签名方案算法运算量比较

方案	密钥生成			签名			验证			总计
	[mc]	[mn]	[dj]	[mc]	[mn]	[dj]	[mc]	[mn]	[dj]	
ECDSA[5]	0	0	1	2	1	1	2	1	2	T_1
文献[33]方案	0	1	2	6	1	1	2	1	2	T_2
本文方案	0	0	1	3	0	2	2	1	2	T_3

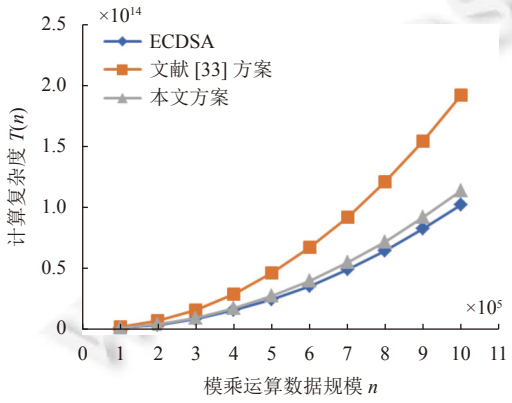


图 2 3 种签名方案算法计算复杂度比较

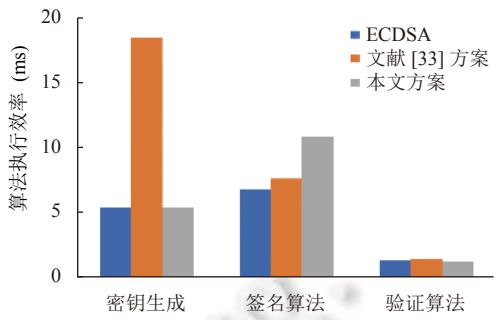


图 3 3 种签名方案算法执行效率比较

由图 3 分析可知, 本文抗颠覆的 ECDSA 签名方案与 ECDSA 签名方案在密钥生成阶段和签名验证阶段算法耗时基本相同, 由于本文方案在计算签名的第 2 个随机数 α 耗时的原因, 签名算法耗时比另外两种签名算法耗时长. 但本文签名方案中密钥生成时间比文献 [33] 密钥生成时间快约 3.5 倍, 算法总执行效率比文献 [33] 算法执行效率快约 1.6 倍, 签名长度相同情况下, 本文签名方案的算法执行效率相比文献 [33] 有所提高, 同时本文方案满足抗颠覆的安全特性.

6 总 结

本文首先提出了针对 ECDSA 签名算法的颠覆方法, 同时对颠覆的 ECDSA 签名的安全目标进行分析与证明. 然后, 利用哈希函数对 ECDSA 签名方案改进, 构造具备抗颠覆特性的签名方案, 对抗颠覆的 ECDSA 签名的 EUF-CMA 和抗颠覆性进行分析. 本文提出的具备抗颠覆特性的签名方案仅针对 ECDSA 签名算法遭受的随机数颠覆攻击, 对于其他颠覆攻击方法的抗颠覆性后续将继续研究. 最后, 通过对本文抗颠覆的 ECDSA 签名算法与已有签名算法效率测试, 实验结果验证了抗颠覆的 ECDSA 签名算法在计算复杂度与算法执行效率方面都具备优势.

References:

[1] Support the Guardian. Revealed: How US and UK spy agencies defeat internet privacy and security. 2013. https://www.theguardian.com/world/2013/sep/05/nsa-gchq-encryption-codes-security

[2] Tang Q, Yung M. Cliptography: Post-snowden cryptography. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security (CCS). Dallas: ACM, 2017. 2615–2616. [doi: 10.1145/3133956.3136065]

[3] Bernstein DJ, Lange T, Niederhagen R. Dual EC DRBG. 2015. https://projectbullrun.org/dual-ec/

[4] Li G, Liu JW, Zhang ZY. An overview on cryptography against mass surveillance. Journal of Cryptologic Research, 2019, 6(3): 269–282 (in Chinese with English abstract). [doi: 10.13868/j.cnki.jcr.000301]

[5] Brown DRL. The exact security of ECDSA. In: Advances in Elliptic Curve Cryptography. 2000. https://www.doc88.com/p-7778768574187.html

[6] Nakamoto S. Bitcoin: A peer-to-peer electronic cash system. 2008. https://bitcoin.org/bitcoin.pdf

[7] Zhang ZX, Wang MW. Survey on blockchain wallet scheme. Computer Engineering and Applications, 2020, 56(6): 28–38 (in Chinese with English abstract). [doi: 10.3778/j.issn.1002-8331.1910-0044]

[8] Simmons GJ. The Prisoners' problem and the subliminal channel. In: Chaum D, ed. Advances in Cryptology. Boston: Springer, 1984. 51–67. [doi: 10.1007/978-1-4684-4730-9_5]

[9] Bellare M, Paterson KG, Rogaway P. Security of symmetric encryption against mass Surveillance. In: Proc. of the 34th Annual Cryptology Conf. Santa Barbara: Springer, 2014. 1–19. [doi: 10.1007/978-3-662-44371-2_1]

[10] Young A, Yung M. The dark side of "Black-Box" cryptography or: Should we trust capstone? In: Proc. of the 16th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 1996. 89–103. [doi: 10.1007/3-540-68697-5_8]

[11] Young A, Yung M. Kleptography: Using cryptography against cryptography. In: Proc. of the 1997 Int'l Conf. on the Theory and Application of Cryptographic Techniques. Konstanz: Springer, 1997. 62–74. [doi: 10.1007/3-540-69053-0_6]

[12] Young A, Yung M. The prevalence of kleptographic attacks on discrete-log based cryptosystems. In: Proc. of the 17th Annual Int'l Cryptology Conf. Santa Barbara: Springer, 1997. 264–276. [doi: 10.1007/BFb0052241]

[13] Young A, Yung M. Malicious cryptography: Exposing cryptovirology. Topics in Cryptology—CT-RSA. 2005. 7–18.

[14] Young A, Yung M. A space efficient backdoor in RSA and its applications. In: Proc. of the 12th Int'l Workshop Selected Areas in Cryptography. Kingston: Springer, 2006. 128–143. [doi: 10.1007/11693383_9]

[15] Degabriele JP, Farshim P, Poettering B. A more cautious approach to security against mass surveillance. In: Proc. of the 22nd Int'l Workshop Fast Software Encryption. Istanbul: Springer, 2015. 579–598. [doi: 10.1007/978-3-662-48116-5_28]

[16] Ateniese G, Magri B, Venturi D. Subversion-resilient signature schemes. In: Proc. of the 22nd ACM SIGSAC Conf. on Computer and Communications Security. Denver: ACM, 2015. 364–375. [doi: 10.1145/2810103.2813635]

[17] Liu C, Chen RM, Wang Y, Wang YJ. Asymmetric subversion attacks on signature schemes. In: Proc. of the 23rd Australasian Conf. on Information Security and Privacy. Wollongong: Springer, 2018. 376–395. [doi: 10.1007/978-3-319-93638-3_22]

[18] Baek J, Susilo W, Kim J, Chow YW. Subversion in practice: How to efficiently undermine signatures. IEEE Access, 2019, 7: 68799–68811. [doi: 10.1109/ACCESS.2019.2918550]

[19] Al-Absi MA, Abdullaev A, Al-Absi AA, Sain M, Lee HJ. Cryptography survey of DSS and DSA. In: Li L, Pratihar D, Chakrabarty S, Mishra P, eds. Advances in Materials and Manufacturing Engineering. Lecture Notes in Mechanical Engineering, Singapore: Springer. 2020. 661–669. [doi: 10.1007/978-981-15-1307-7_75]

[20] Bellare M, Fuchsbauer G, Scafuro A. NIZKs with an untrusted CRS: Security in the face of parameter subversion. In: Proc. of the 22nd Int'l Conf. on the Theory and Application of Cryptology and Information Security. Hanoi: Springer, 2016. 777–804. [doi: 10.1007/978-3-662-53890-6_26]

[21] Dodis Y, Ganesh C, Golovnev A, Juels A, Ristenpart T. A formal treatment of backdoored pseudorandom generators. In: Proc. of the 34th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Sofia: Springer, 2015. 101–126. [doi: 10.1007/978-3-662-46800-5_5]

[22] Degabriele JP, Paterson KG, Schuldt JCN, Woodage J. Backdoors in pseudorandom number generators: Possibility and impossibility results. In: Proc. of the 36th Annual Int'l Cryptology Confe. Santa Barbara: Springer, 2016. 403–432. [doi: 10.1007/978-3-662-53018-4_15]

[23] Russell A, Tang Q, Yung M, Zhou HS. Cliptography: Clipping the power of kleptographic attacks. In: Proc. of the 22nd Int'l Conf. on the Theory and Application of Cryptology and Information Security. Hanoi: Springer, 2016. 34–64. [doi: 10.1007/978-3-662-53890-6_2]

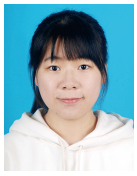
[24] Chow SSM, Russell A, Tang Q, Yung M, Zhao YJ, Zhou HS. Let a non-barking watchdog bite: Cliptographic signatures with an offline watchdog. In: Proc. of the 22nd IACR Int'l Conf. on Practice and Theory of Public-key Cryptograph. Beijing: Springer, 2019. 221–251.

[doi: 10.1007/978-3-030-17253-4_8]
[25] Russell A, Tang Q, Yung M, Zhou HS. Generic semantic security against a kleptographic adversary. In: Proc. of the 2017 ACM SIGSAC Conf. on Computer and Communications Security. Dallas: ACM, 2017. 907–922. [doi: 10.1145/3133956.3133993]
[26] Mironov I, Stephens-Davidowitz N. Cryptographic reverse firewalls. In: Proc. of the 34th Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Sofia: Springer, 2015. 657–686. [doi: 10.1007/978-3-662-46803-6_22]
[27] Fischlin M, Mazaheri S. Self-guarding cryptographic protocols against algorithm substitution attacks. In: Proc. of the 31st IEEE Computer Security Foundations Symp. Oxford: IEEE, 2018. 76–90. [doi: 10.1109/CSF.2018.00013]
[28] Bellare M, Cash D. Pseudorandom functions and permutations provably secure against related-key attacks. In: Proc. of the 30th Annual Cryptology Conf. Santa Barbara: Springer, 2010. 666–684. [doi: 10.1007/978-3-642-14623-7_36]
[29] Yu Y, Ding YJ, Zhao YQ, Li YN, Zhao Y, Du XJ, Guizani M. LRCoin: Leakage-resilient cryptocurrency based on bitcoin for data trading in IoT. IEEE Internet of Things Journal, 2019, 6(3): 4702–4710. [doi: 10.1109/JIOT.2018.2878406]
[30] Yang B. Cyberspace Security. 4th ed., Beijing: Tsinghua University Press, 2017. 161–165 (in Chinese).
[31] 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. [doi: 10.1137/0217017]
[32] Pointcheval D, Stern J. Security arguments for digital signatures and blind signatures. Journal of Cryptology, 2000, 13(3): 361–396. [doi: 10.1007/s001450010003]
[33] Li YM, Zhang P. Security analysis and improvement of elliptic curve digital signature scheme. In: Proc. of the 5th Int'l Conf. on Artificial Intelligence and Security. New York: Springer, 2019. 609–617. [doi: 10.1007/978-3-030-24271-8_54]

附中文参考文献:

[4] 李耕, 刘建伟, 张宗洋. 抗大规模监视密码学研究综述. 密码学报, 2019, 6(3): 269–282. [doi: 10.13868/j.cnki.jcr.000301]
[7] 张中霞, 王明文. 区块链钱包方案研究综述. 计算机工程与应用, 2020, 56(6): 28–38. [doi: 10.3778/j.issn.1002-8331.1910-0044]
[30] 杨波. 现代密码学. 第4版, 北京: 清华大学出版社, 2017. 182–200.

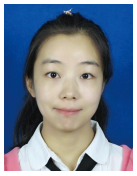
严都力(1995－), 女, 助教, 主要研究领域为密码学, 区块链安全.



李慧琳(1996－), 女, 博士生, 主要研究领域为区块链与智能合约.

禹勇(1980－), 男, 博士, 教授, CCF 高级会员, 主要研究领域为公钥密码理论及应用, 区块链与密码货币, 云计算安全.

赵艳琦(1992－), 男, 博士, 副教授, 主要研究领域为密码学, 区块链安全.



李艳楠(1991－), 女, 博士生, 主要研究领域为区块链, 云存储安全.

田爱奎(1963－), 男, 博士, 教授, 主要研究领域为信息安全.