安全加密的门限签名混淆

李亚红[1]，王彩芬[2,3]，张玉磊[2]，杨小东[2]，黄海燕[1]
（1. 兰州交通大学电子与信息工程学院，甘肃 兰州 730070；2. 西北师范大学计算机科学与工程学院，甘肃 兰州 730070；
3. 深圳技术大学大数据与互联网学院，广东 深圳 518118）

摘 要：针对门限签名密钥泄露的安全问题，首先提出了一个加密门限签名功能，并对所提功能混淆，混淆电路的输出可交给任意第三方执行，不会泄露门限签名密钥的信息。然后定义了加密门限签名功能和混淆器的安全模型，存在不可伪造性和平均情况虚拟黑盒性质，并对其正确性和安全性进行证明。理论和仿真实验分析表明，对加密门限签名的混淆具有可行性。

关键词：混淆；加密门限签名；加密私钥；平均情况虚拟黑盒性质
中图分类号：TP309
文献标识码：A
doi: 10.11959/j.issn.1000−436x.2020129

Secure obfuscation for encrypted threshold signatures

LI Yahong[1], WANG Caifen[2,3], ZHANG Yulei[2], YANG Xiaodong[2], HUANG Haiyan[1]
1. College of Telecommunication Engineering, Lanzhou Jiaotong University, Lanzhou 730070, China
2. College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China
3. College of Big Data and Internet, Shenzhen Technology University, Shenzhen 518118, China

Abstract: Aiming at the key leakage security problem of the threshold signature, an encrypted threshold signature functionality was firstly proposed and securely obfuscated. The output of obfuscated circuit could be implemented by any third party without revealing the private key of threshold signature. Secondly, the security models of encrypted threshold signature functionality and the obfuscator were defined, such as the existential unforgeablity and the average case virtual black box property, and its correctness and security were proved. The analyses of theory and simulation experiment show that the obfuscation for encrypted threshold signature has feasibility.

Key words: obfuscation, encrypted threshold signature, encrypted private key, average case virtual black box property

1 引言

门限签名的概念由 Desmedt[1]提出。在 (k,n) 门限签名方案中，当参与者数等于或大于门限值 k 时，才可以生成有效的签名。门限签名具有权力分散、风险分担等特点，因此门限签名方案被广泛研究和讨论[2-6]。为了满足不同的应用需求，研究者提出了不同性质的门限签名，如适合移动互联网的门限群签名[7]、实现区块链技术的门限环签名[8]、解决船舶自组网认证问题的门限代理签名[9]等。然而，门限密码体制的安全性建立在私钥安全的前提下，一旦私钥泄露系统将存在较大的安全隐患。例如，在白盒攻击环境中，攻击者通过观察或者执行密码软件很容易获得密钥信息，因此需要对其中的私密信息，特别是密钥信息进行保护。

混淆可以有效解决密钥泄露的问题，混淆是将一段程序转换成另一段不可理解的程序的过程，但不泄露其秘密信息。混淆具有极好的安全性和保密性，可应用于云计算[10]和隐私保护领域[11]。混淆的形式化定义由 Barak 等[12]提出，该定义包括以下几

收稿日期：2019−10−28；**修回日期**：2020−04−02
基金项目：国家自然科学基金资助项目（No.61063041, No.61901201）；兰州交通大学青年科学基金资助项目（No.2018002）
Foundation Items: The National Natural Science Foundation of China (No.61063041, No.61901201), The Youth Science Foundation of Lanzhou Jiaotong University (No.2018002)

个方面：1) 功能性，混淆后的程序与原程序具有同样的功能；2) 效率性，混淆的程序运行时间是原程序运行时间的多项式级别；3)安全性，任何敌手在获得混淆的程序后，不能从中获得任何有用的信息。2007 年，Hohenberger 等[13]提出对传统密码学功能的混淆，实现了对重加密功能的混淆，解决了解密密钥的暴露问题，同时提出能够应用密码学的重要安全性质——平均情况虚拟黑盒性质（ACVBP, average case virtual black box property），并在判定性线性假设下证明了混淆的安全性。受文献[13]的启发，2010 年，Hada[14]在欧洲密码会议上利用混淆为增加签名安全性提出了一种有效的方法——加密签名功能的混淆，即用 Waters 签名[15]和线性加密方案[16]构造了一个加密签名功能，对其私密信息签名私钥进行安全混淆，实现了密钥保护。利用同样的方法，不同的加密签名的混淆方案被提出，如加密群签名的混淆[17]、加密环签名的混淆[18]、加密基于身份签名的混淆[19]等，上述方案从不同的签名性质研究了混淆的可能性。

针对门限签名（TS, threshold signature）密钥安全存储的问题，本文对门限签名的密钥进行混淆，密钥信息以密文形式隐藏在算法中，有效地保护了门限签名的密钥。所提混淆算法保证可信第三方不能伪造该签名，从而降低了可信第三方的权限。最后，在标准模型下证明了加密门限签名（ETS, encrypted threshold signature）的混淆器的安全性。

2 预备知识

2.1 基础知识和双线性映射

记 $\{C_t\}_{t\in\mathbb{N}}$ 为一类概率电路，其中，C_t 是一个多项式电路的集合，C_t 输入长度为 t 的多项式 $l_{\text{in}}(t)$，输出为 $l_{\text{out}}(t)$。$y\leftarrow A^C(x)$ 表示预言机 A 访问电路 C 的情况下，以 x 为输入，输出 y。$M^{\ll C\gg}$ 表示算法 M 抽样访问电路 C，其输入值只包含常规值。电路 $\mathbb{C}=\{C_t\}_{t\in\mathbb{N}}$ 的混淆器（Obf）是一个概率多项式的编译器，以一个概率电路 $C\in C_t$ 为输入，输出一个难以理解的概率电路 $C'=\text{Obf}(C)$，且 $\text{Obf}(C)$ 实现的功能和 C 相同。

定义 1 双线性映射。设 q 是一个大素数，G和G_T 是 2 个有着相同素数阶 q 的乘法循环群，g 是 G 的生成元。一个双线性映射 $e:G\times G\rightarrow G_T$ 在 G 上满足以下性质。

1) 双线性。对任意的 $g,h\in G$，存在 $a,b\in Z_q$，使 $e(g^a,h^b)=e(g,h)^{ab}$。

2) 非退化性。存在 $a,b\in Z_q$，使 $e(g^a,g^b)\neq 1$。

3) 可计算性。对所有的 $g,h\in G$，存在有效算法计算 $e(g,h)$。

2.2 线性加密方案

为了构造加密门限签名的混淆，本节详细描述线性加密方案[16]和门限签名方案[6]。线性加密方案包含以下 3 个算法。

1)密钥生成算法 KG。设 g 是 G 的生成元，随机选择 $a,b\in Z_q$，输出私钥 $\text{sk}_e=(a,b)$ 和公钥 $\text{pk}_e=(\text{pk}_{e1},\text{pk}_{e2})=(g^a,g^b)$。

2)加密算法 Enc。输入公钥 pk_e 和消息 M，随机选择 $x_1,x_2\in Z_q$，输出密文 $C=(C_1,C_2,C_3)=(\text{pk}_{e1}^{x_1},\text{pk}_{e2}^{x_2},g^{x_1+x_2}M)$。

3) 解密算法 Dec。输入私钥 $\text{sk}_e=(a,b)$ 和密文 C，输出 $M=\dfrac{C_3}{C_1^{\frac{1}{a}}C_2^{\frac{1}{b}}}$。

为了保证 ETS 的混淆算法输出的概率性，要确保 pk_e 关于密文可重随机化，即重随机化算法。

重随机化 ReRand 算法。给定公钥 pk_e 和密文 C，随机选择 $x_1',x_2'\in Z_q$，密文重随机化为 $C'=(C_1',C_2',C_3')=(C_1\text{pk}_{e1}^{x_1'},C_2\text{pk}_{e2}^{x_2'},C_3g^{x_1'+x_2'})=(\text{pk}_{e1}^{x_1+x_1'},\text{pk}_{e2}^{x_2+x_2'},g^{x_1+x_2+x_1'+x_2'}M)$。

文献[16]证明了在判定性线性假设成立的条件下，线性加密方案满足选择明文攻击下具有不可区分性(IND-CPA, indistinguish ability under chosen plaintext attack)。

2.3 (*k*, *n*)门限签名方案

1) 系统建立Setup算法。输入安全参数 1^λ、k 和 n，其中，k 表示门限值，n 表示参与者数，且 $1\leqslant k\leqslant n$。输出素数 q 阶循环乘法群 G 和 G_T，g 为群 G 的生成元，$e:G\times G\rightarrow G_T$ 为一个双线性映射。设群体成员的集合 $P=(P_1,P_2,\cdots,P_n)$，可信中心随机选择 $a_0\in Z_q$，$g_2,u',\boldsymbol{U}=(u_1,u_2,\cdots,u_n)\in G$，计算 $g_1=g^{a_0}$，执行以下步骤。

① 随机选取 $a_1,a_2,\cdots,a_{k-1}\in Z_q$，构造次数为 $k-1$的多项式 $f(x)=a_0+a_1x+\cdots+a_{k-1}x^{k-1},f(x)\in Z_q(x)$，计算私值分片 $\text{sk}_i=g_2^{f(i)}$，将 sk_i 秘密分发给参与者 P_i，其中 $i\in\{1,2,\cdots,n\}$。

② 计算公共验证密钥 $\mathbf{VK}=(g^{f(1)},g^{f(2)},\cdots,g^{f(n)})$，公开 $\mathbf{VK}$。

输出公钥 $p=\{q,G,G_T,e,g,g_1,g_2,u',\boldsymbol{U},\mathbf{VK}\}$，保密 a_0、$f(x)$ 和 $\mathbf{SK}=(\mathrm{sk}_1,\mathrm{sk}_2,\cdots,\mathrm{sk}_n)$。

2) 部分签名生成 SS 算法。输入私钥分片 sk_i、公钥 p 和 $m=m_1\cdots m_n\in\{0,1\}^n$，$P_i$ 随机选择 $r_i\in Z_q$，计算并输出部分签名 $\sigma_i=(\sigma_{i1},\sigma_{i2})=(\mathrm{sk}_i(u'\prod_{i=1}^{n}u_i^{m_i})^{r_i},g^{r_i})$，其中 $i\in\{1,2,\cdots,n\}$。

3) 部分签名验证 SV 算法。输入部分签名 $\sigma_i=(\sigma_{i1},\sigma_{i2})$、$m=m_1\cdots m_n\in\{0,1\}^n$ 和公钥 p，验证等式 $e(\sigma_{i1},g)=e(g_2,\mathrm{vk}_i)e(u'\prod_{i=1}^{n}u_i^{m_i},\sigma_{i2})$ 是否成立。若成立，表示签名有效；否则签名无效。

4) 合成签名 SC 算法。输入 k 份有效的部分签名 $\sigma_j=(\sigma_{j1},\sigma_{j2})_{i\in\Phi}$，其中，集合 $\Phi\subset\{1,2,\cdots,n\}$，且 $|\Phi|=k$，计算 $\lambda_j=\prod_{i,j\in\Phi,j\neq i}\frac{-i}{j-i}$，输出合成签名 $\sigma=(\sigma_1,\sigma_2)=(\prod_{j\in\Phi}(\sigma_{j1})^{\lambda_j},\prod_{j\in\Phi}(\sigma_{j2})^{\lambda_j})$。

5) 签名验证 Verify 算法。给定公钥 p、$m=m_1\cdots m_n\in\{0,1\}^n$ 和签名 $\sigma=(\sigma_1,\sigma_2)$，验证等式 $e(\sigma_1,g)=e(g_2,g_1)e(u'\prod_{i=1}^{n}u_i^{m_i},\sigma_2)$ 是否成立。若成立，表示签名有效；否则签名无效。

文献[6]证明了门限签名方案是存在性不可伪造的。

3　加密门限签名的混淆

ETS 构造混淆的思想是“先对消息进行门限签名，然后加密该门限签名”等价于“先加密门限签名的私钥分片，再用加密的私钥分片生成门限签名”。前者是 ETS 功能，“加密门限签名的私钥分片”是对 ETS 功能的混淆，“用加密的私钥分片生成门限签名”是执行混淆程序。

3.1　ETS 功能

为了实现 ETS 功能，定义一类电路 $\mathbb{C}_{\mathrm{ETS}}=\{C_\lambda\}_{\lambda\in\mathbb{N}}$，并对其混淆。$C_\lambda$ 是有关 $C_{p,\mathbf{SK},\mathrm{pk}_e}$ 的电路集合，可从 $C_{p,\mathrm{SK},\mathrm{pk}_e}$ 提取参数 $(p,\mathbf{SK},\mathrm{pk}_e)$，其中 $C_{p,\mathbf{SK},\mathrm{pk}_e}$ 能实现 ETS 功能 $\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$。给定 p、pk_e 和门限签名私钥 $\mathbf{SK}$。$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 功能定义如下，若输入的 m 为特殊值 keys 时，则输出 (p,pk_e)；否则对消息 m 进行加密门限签名，即先门限签名后加密，具体如下。

1) 按照 SS 算法、SV 算法和 SC 算法生成门限签名 (σ_1,σ_2)。

2) 加密门限签名。运行算法 $(S_1,S_2)\leftarrow(\mathrm{Enc}(\mathrm{pk}_e,\sigma_1),\mathrm{Enc}(\mathrm{pk}_e,\sigma_2))$，计算 $S_1=(S_{11},S_{12},S_{13})=(\mathrm{pk}_{e1}{}^{x_1},\mathrm{pk}_{e2}{}^{x_2},g^{x_1+x_2}\sigma_1)$，$S_2=(S_{21},S_{22},S_{23})=(\mathrm{pk}_{e1}{}^{y_1},\mathrm{pk}_{e2}{}^{y_2},g^{y_1+y_2}\sigma_2)$，其中 $x_1,x_2,y_1,y_2\in Z_q$，输出 (S_1,S_2)。

通过验证算法 ETS.Verify 验证 ETS 功能 $\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 的正确性。

验证算法 ETS.Verify。给定公钥 p、$m=m_1\cdots m_n\in\{0,1\}^n$、$(S_1,S_2)$ 和 $\mathrm{sk}_e=(a,b)$，首先解密 $\sigma_i=\frac{S_{i3}}{S_{i1}^{\frac{1}{a}}S_{i2}^{\frac{1}{b}}}$，$i=1,2$。然后验证等式 $e(\sigma_1,g)=e(g_2,g_1)e(u'\prod_{i=1}^{n}u_i^{m_i},\sigma_2)$ 是否成立。若成立，表示签名有效；否则签名无效。

实现 ETS 功能的电路包含门限签名私钥分片 $\mathbf{SK}$，得到原始电路的恶意操作者可从中提取签名私钥 $\mathbf{SK}$。3.2 节将利用混淆器解决保护 $\mathbf{SK}$ 的问题。

3.2　ETS 功能混淆器的构造

ETS 混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的输出与 $C_{p,\mathbf{SK},\mathrm{pk}_e}$ 相同。$\mathrm{Obf}_{\mathrm{ETS}}$ 算法先对私钥 $\mathbf{SK}=(\mathrm{sk}_1,\mathrm{sk}_2,\cdots,\mathrm{sk}_n)$ 进行线性加密生成一个新的私钥 $\overline{\mathbf{SK}}=(\overline{\mathrm{sk}_1},\overline{\mathrm{sk}_2},\cdots,\overline{\mathrm{sk}_n})$，然后利用 $\overline{\mathrm{sk}_i}$ 生成签名。给定 $C_{p,\mathbf{SK},\mathrm{pk}_e}$，包含 $(p,\mathbf{SK},\mathrm{pk}_e)$，构造的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 定义如下。

1) 提取 pk_e、sk_i 和 p，运行 $C_i=(C_{i1},C_{i2},C_{i3})=(\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},\overline{\mathrm{sk}_i})\leftarrow\mathrm{Enc}(\mathrm{pk}_e,\mathrm{sk}_i)$，得到 $\overline{\mathrm{sk}_i}=g^{x_{i1}+x_{i2}}\mathrm{sk}_i$，$C_i$ 是对 sk_i 的加密，计算 $\overline{\mathrm{vk}_i}=g^{x_{i1}+x_{i2}}\mathrm{vk}_i$，$x_{i1},x_{i2}\in Z_q$。令 $z=(\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},\overline{\mathrm{sk}_i})$，其中 $i\in\{1,2,\cdots,n\}$。

2) 构造并输出混淆电路 R_{p,z,pk_e}。若输入的 m 为函数 keys 时，则输出 (p,pk_e)；否则给定 $m=m_1\cdots m_n\in\{0,1\}^n$、$\mathrm{pk}_e=(\mathrm{pk}_{e1},\mathrm{pk}_{e2})$、$p$、$\overline{\mathrm{vk}_i}$ 和 $z=(c_{i1},c_{i2},c_{i3})=(\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},\overline{\mathrm{sk}_i})$，其中 $i\in\{1,2,\cdots,n\}$，执行如下算法。

① 部分签名算法。随机选择 $r_i'\in Z_q$，计算部分签名 $\overline{\sigma_i}=(\overline{\sigma_{i1}},\overline{\sigma_{i2}})=(\overline{\mathrm{sk}_i}(u'\prod_{i=1}^{n}u_i^{m_i})^{r_i'},g^{r_i'})$。验证等式

$e(\overline{\sigma_{i1}}, g) = e(g_2, \overline{\mathrm{vk}_i}) e(u' \prod_{i=1}^{n} u_i^{m_i}, \overline{\sigma_{i2}})$ 是否成立。若成立，签名有效；否则签名无效。

② 合成签名算法。输入 k 份有效的部分签名 $\overline{\sigma_j} = (\overline{\sigma_{j1}}, \overline{\sigma_{j2}})_{j \in \Phi}$，其中集合 $\Phi \subset \{1,2,\cdots,n\}$，且 $|\Phi| = k$，计算 $\lambda_j' = \prod_{i,j \in \Phi, j \neq i} \frac{-i}{j-i}$，输出合成签名 $\overline{\sigma} = (\overline{\sigma_1}, \overline{\sigma_2}) = (\prod_{j \in \Phi} (\overline{\sigma_{j1}})^{\lambda_j'}, \prod_{j \in \Phi} (\overline{\sigma_{j2}})^{\lambda_j'})$。

③ 重随机化 ReRand 算法。对密文 $\overline{\sigma} = (\overline{\sigma_1}, \overline{\sigma_2})$ 重随机化，计算 $c_1 = \prod_{j \in \Phi} (c_{j1})^{\lambda_j'} = \mathrm{pk}_{e1}^{\sum_{j \in \Phi} \lambda_j' x_{j1}}$，$c_2 = \prod_{j \in \Phi} (c_{j2})^{\lambda_j'} = \mathrm{pk}_{e2}^{\sum_{j \in \Phi} \lambda_j' x_{j2}}$，随机选择 $x_1', x_2', y_1', y_2' \in Z_q$，运行算法 $\overline{\overline{\sigma_1}} \leftarrow \mathrm{ReRand}(\mathrm{pk}_e, c_1, c_2, \overline{\sigma_1})$，$\overline{\overline{\sigma_2}} \leftarrow \mathrm{Enc}(\mathrm{pk}_e, \overline{\sigma_2})$，计算得到 $\overline{\overline{\sigma_2}} = (\mathrm{pk}_{e1}^{y_1'}, \mathrm{pk}_{e2}^{y_2'}, g^{y_1'+y_2'} \overline{\sigma_2})$ 和 $\overline{\overline{\sigma_1}} = (c_1 \mathrm{pk}_{e1}^{x_1'}, c_2 \mathrm{pk}_{e2}^{x_2'}, g^{x_1'+x_2'} \overline{\sigma_1}) = (\mathrm{pk}_{e1}^{\sum_{j \in \Phi} \lambda_j' x_{j1} + x_1'}, \mathrm{pk}_{e2}^{\sum_{j \in \Phi} \lambda_j' x_{j2} + x_2'}, g^{x_1'+x_2'} \overline{\sigma_1})$，输出 $(\overline{\overline{\sigma_1}}, \overline{\overline{\sigma_2}})$。

在实现 ETS 功能的混淆电路中，可信中心没有用分发给参与者 P_i 的私钥分片 sk_i 生成门限签名，而是用加密后的私钥分片 $\overline{\mathrm{sk}_i}$ 进行签名，这样可信中心不能伪造该签名，从而降低了其权限。

4 ETS 混淆器的安全性定义和分析

本节主要分析 ETS 混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的安全性。ETS 混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的安全定义包括 ACVBP 和存在不可伪造性。对混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的 ACVBP 对预言询问做了一定的限制，即满足有关联预言机集 $T(C)=\{\mathrm{SS}(p, \mathrm{sk}_i, m), i=1,2,\cdots,n\}=\mathrm{SS}_{\mathrm{sk}_i}$ 和 $R(C)=\{\mathrm{Corruption}, |\Phi| \leqslant k-1\} = \mathrm{Corruption}^{|\Phi| \leqslant k-1}$ 的 ACVBP，其中 $\mathrm{Corruption}^{|\Phi| \leqslant k-1}$ 表示敌手预言询问 Corruption 最多只能获得 $k-1$ 份私钥分片。

4.1 ETS 混淆器的安全性定义

定义 2 功能保护性。设 Obf 是一个概率多项式混淆器，$\mathcal{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$ 为概率电路，若对任意的 $C \in C_\lambda$，使下式成立。

$$\Pr[C' \leftarrow \mathrm{Obf}(C) : \forall t, \triangle(C(t), C'(t)) = 0] = 1$$

其中，$\triangle(C(t), C'(t)) = \sum_{y \in \{0,1\}^{l_{\mathrm{out}}(\lambda)}} \left| \Pr[o \leftarrow C(t) : o = y] - \Pr[o \leftarrow C'(t) : o = y] \right|$ 表示 $C(t)$ 与 $C'(t)$ 的统计距离。

定义 3 关于有关联预言机 $T(C)$ 和 $R(C)$ 的 ACVBP。设概率电路 $\mathbb{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$，令 $T(C)$ 和 $R(C)$ 为一组与电路 C 有关联的预言机集合，Obf 是电路 $\mathbb{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$ 的混淆器。对任意的 $C \in C_\lambda$，若以下条件成立，则称 C 的混淆器 Obf 满足有关联预言机 $T(C)$ 和 $R(C)$ 的 ACVBP：存在概率多项式算法 S (模拟者)，概率多项式算法 D (区分者)，充分大的数 $\lambda \in \mathbb{N}$，任意的多项式 $p(\cdot)$，任意的 $z \in \{0,1\}^{\mathrm{poly}(\lambda)}$，有

$$\left| \Pr\left[\begin{array}{l} C \leftarrow C_\lambda; \\ C' \leftarrow \mathrm{Obf}(C); \\ b \leftarrow D^{\ll C, T(C), R(C) \gg}(C', z); \end{array} : b = 1 \right] - \Pr\left[\begin{array}{l} C \leftarrow C_\lambda; \\ C'' \leftarrow S^{\ll C \gg}(1^\lambda, z); \\ b \leftarrow D^{\ll C, T(C), R(C) \gg}(C'', z); \end{array} : b = 1 \right] \right| < \frac{1}{p(\lambda)}$$

其中，$D^{\ll C, T(C), R(C) \gg}$ 表示以抽样预言的方式询问 C、$T(C)$、$R(C)$。

定义 4 ETS 功能的存在不可伪造性。设 Obf 是电路 $\mathcal{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$ 的混淆器。称 TS 关于 ETS 功能满足存在不可伪造性，若以下条件成立：即对多项式时间的敌手 A，充分大 $\lambda \in \mathbb{N}$，任意多项式 $p(\cdot)$，任意的 $z \in \{0,1\}^{\mathrm{poly}(\lambda)}$，有

$$\left| \Pr\left[\begin{array}{c} (p, \mathbf{SK}) \leftarrow \mathrm{Setup}(1^\lambda, k, n), (\mathrm{sk}_e, \mathrm{pk}_e) \leftarrow \mathrm{KG}(p); \\ (m, \sigma, Q) \leftarrow A^{\ll \mathrm{SS}_{\mathrm{sk}_i}, \mathrm{Corruption}^{|\Phi| \leqslant k-1} \gg}(p, \boldsymbol{VK}, \mathrm{pk}_e, z); \\ \mathrm{Veriy}(m, \sigma, p) = 1, m \notin Q; \end{array} \right] \right| < \frac{1}{p(\lambda)}$$

其中，$A^{\ll \mathrm{SS}_{\mathrm{sk}_i}, \mathrm{Corruption}^{|\Phi| \leqslant k-1} \gg}$ 表示 A 进行抽样预言询问 SS_{sk_i} 和 $\mathrm{Corruption}^{|\Phi| \leqslant k-1}$，$Q$ 是敌手 A 进行适应性询问 $\mathrm{SS}_{\mathrm{sk}_i}$ 的一组消息集合。

定义 5 ETS 混淆器的存在不可伪造性。设 Obf 是电路 $\mathcal{C} = \{C_\lambda\}_{\lambda \in \mathbb{N}}$ 的混淆器。称 TS 关于 ETS 混淆器 Obf 满足存在不可伪造性，若以下条件成立：即对多项式时间的敌手 A，充分大 $\lambda \in \mathbb{N}$，任意多项式 $p(\cdot)$，任意的 $z \in \{0,1\}^{\mathrm{poly}(\lambda)}$，有

$$\left| \Pr\left[\begin{array}{c} (p, \mathbf{SK}) \leftarrow \mathrm{Setup}(1^\lambda, k, n), (\mathrm{sk}_e, \mathrm{pk}_e) \leftarrow \mathrm{KG}(p); \\ C' \leftarrow \mathrm{Obf}(C); \\ (m, \sigma, Q) \leftarrow A^{\ll \mathrm{SS}_{\mathrm{sk}_i}, \mathrm{Corruption}^{|\Phi| \leqslant k-1} \gg}(p, C', \mathbf{VK}, \mathrm{pk}_e, z); \\ \mathrm{Verify}(m, \sigma, p) = 1, m \notin Q; \end{array} \right] \right| < \frac{1}{p(\lambda)}$$

其中，表示 A 进行预言询问 $\mathrm{SS}_{\mathrm{sk}_i}$ 和 $\mathrm{Corruption}^{|\Phi|\leqslant k-1}$，$Q$ 是敌手 A 进行适应性询问 $\mathrm{SS}_{\mathrm{sk}_i}$ 的一组消息集合。

4.2 ETS 混淆器的安全性分析

定理 1 功能保护性。设 $C_{p,\mathbf{SK},\mathrm{pk}_e}\in C_\lambda$ 和 $R_{p,z,\mathrm{pk}_e}=\mathrm{Obf}_{\mathrm{ETS}}(C_{p,\mathbf{SK},\mathrm{pk}_e})$，则对任意的输入，$C_{p,\mathbf{SK},\mathrm{pk}_e}$ 和 R_{p,z,pk_e} 的输出分布是相同的。

证明 设 $\mathrm{pk}_e=(g^a,g^b)$，若输入消息 m 为特殊值 keys 时，则输出 (p,pk_e)；否则对 $m=m_1\cdots m_n\in\{0,1\}^n$，运行 $\sigma_i\leftarrow\mathrm{SS}(p,\mathrm{sk}_i,m)(i\in\{1,2,\cdots,n\})$ 和 $(\sigma_1,\sigma_2)\leftarrow\mathrm{SC}(\Phi,\sigma_j)(j\in\Phi)$，得到 $(\sigma_1,\sigma_2)=(\prod_{j\in\Phi}(\sigma_{j1})^{\lambda_j},\prod_{j\in\Phi}(\sigma_{j2})^{\lambda_j})$。运行算法 $(S_1,S_2)\leftarrow(\mathrm{Enc}(\mathrm{pk}_e,\sigma_1),\mathrm{Enc}(\mathrm{pk}_e,\sigma_2))$，随机选取 $x_1,x_2,y_1,y_2\in Z_q$，计算 S_1 和 S_2 如下

$$
\begin{aligned}
S_1&=(\mathrm{pk}_{e1}{}^{x_1},\mathrm{pk}_{e2}{}^{x_2},g^{x_1+x_2}\sigma_1)=\\
&(\mathrm{pk}_{e1}{}^{x_1},\mathrm{pk}_{e2}{}^{x_2},g^{x_1+x_2}\prod_{j\in\Phi}(\sigma_{j1})^{\lambda_j})=\\
&(\mathrm{pk}_{e1}{}^{x_1},\mathrm{pk}_{e2}{}^{x_2},g^{x_1+x_2}\prod_{j\in\Phi}(\mathrm{sk}_j(u'\prod_{i=1}^{n}u_i^{m_i})^{r_j})^{\lambda_j})
\end{aligned}
$$

$$
\begin{aligned}
S_2&=(\mathrm{pk}_{e1}{}^{y_1},\mathrm{pk}_{e2}{}^{y_2},g^{y_1+y_2}\sigma_2)=\\
&(\mathrm{pk}_{e1}{}^{y_1},\mathrm{pk}_{e2}{}^{y_2},g^{y_1+y_2}\prod_{j\in\Phi}(g^{r_j})^{\lambda_j})
\end{aligned}
$$

$C_{p,\mathbf{SK},\mathrm{pk}_e}$ 的输出为 (S_1,S_2)。同理，计算 $R_{p,z,\mathrm{pk}_e}=\mathrm{Obf}_{\mathrm{ETS}}(C_{p,\mathbf{SK},\mathrm{pk}_e})=(\overline{\overline{\sigma}}_1,\overline{\overline{\sigma}}_2)$，随机选取 $x_1',x_2',y_1',y_2'\in Z_q$，计算 $(\overline{\overline{\sigma}}_1,\overline{\overline{\sigma}}_2)$ 如下

$$
\begin{aligned}
&\overline{\overline{\sigma}}_1=(C_{11},C_{21},C_{31})=(\mathrm{pk}_{e1}^{\sum_{j\in\Phi}\lambda_j'x_{j1}+x_1'},\mathrm{pk}_{e2}^{\sum_{j\in\Phi}\lambda_j'x_{j2}+x_2'},\\
&g^{x_1'+x_2'}\prod_{j\in\Phi}(\overline{\sigma_{j1}})^{\lambda_j'})=(\mathrm{pk}_{e1}^{\sum_{j\in\Phi}\lambda_j'x_{j1}+x_1'},\mathrm{pk}_{e2}^{\sum_{j\in\Phi}\lambda_j'x_{j2}+x_2'},\\
&g^{x_1'+x_2'}\prod_{j\in\Phi}(\overline{\mathrm{sk}_j}(u'\prod_{i=1}^{n}u_i^{m_i})^{r_j'})^{\lambda_j'})=(\mathrm{pk}_{e1}^{\sum_{j\in\Phi}\lambda_j'x_{j1}+x_1'},\\
&\mathrm{pk}_{e2}^{\sum_{j\in\Phi}\lambda_j'x_{j2}+x_2'},g^{x_1'+x_2'}\prod_{j\in\Phi}(g^{x_{j1}+x_{j2}}\mathrm{sk}_j(u'\prod_{i=1}^{n}u_i^{m_i})^{r_j'})^{\lambda_j'})=\\
&(\mathrm{pk}_{e1}^{\sum_{j\in\Phi}\lambda_j'x_{j1}+x_1'},\mathrm{pk}_{e2}^{\sum_{j\in\Phi}\lambda_j'x_{j2}+x_2'},g^{\sum_{j\in\Phi}\lambda_j'(x_{j1}+x_{j2})+x_1'+x_2'}\cdot\\
&\prod_{j\in\Phi}(\mathrm{sk}_j(u'\prod_{i=1}^{n}u_i^{m_i})^{r_j'})^{\lambda_j'})=(\mathrm{pk}_{e1}^{\sum_{j\in\Phi}\lambda_j'x_{j1}+x_1'},\mathrm{pk}_{e2}^{\sum_{j\in\Phi}\lambda_j'x_{j2}+x_2'},\\
&g^{\sum_{j\in\Phi}\lambda_j'(x_{j1}+x_{j2})+x_1'+x_2'}\prod_{j\in\Phi}(\mathrm{sk}_j(u'\prod_{i=1}^{n}u_i^{m_i})^{r_j'})^{\lambda_j'})
\end{aligned}
$$

$$
\begin{aligned}
&\overline{\overline{\sigma}}_2=(C_{21},C_{22},C_{32})=(\mathrm{pk}_{e1}^{y_1'},\mathrm{pk}_{e2}^{y_2'},g^{y_1'+y_2'}\prod_{j\in\Phi}(\overline{\sigma_{j2}})^{\lambda_j})=\\
&(\mathrm{pk}_{e1}^{y_1'},\mathrm{pk}_{e2}^{y_2'},g^{y_1'+y_2'}\prod_{j\in\Phi}(g^{r_j'})^{\lambda_j})
\end{aligned}
$$

显然，(S_1,S_2) 与 $(\overline{\overline{\sigma}}_1,\overline{\overline{\sigma}}_2)$ 的分布相同。证毕。

ETS 功能的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的正确性可通过以下算法验证。给定公钥 p、$m=m_1\cdots m_n\in\{0,1\}^n$、$(\overline{\overline{\sigma}}_1,\overline{\overline{\sigma}}_2)$ 和 $\mathrm{sk}_e=(a,b)$，首先，解密 $\sigma_i=\dfrac{C_{i3}}{C_{i1}^{\frac{1}{a}}C_{i2}^{\frac{1}{b}}}$ $(i=1,2)$；其次验证等式 $e(\sigma_1,g)=e(g_2,g_1)e(u'\prod_{i=1}^{n}u_i^{m_i},\sigma_2)$ 是否成立，成立表示签名有效，否则签名无效。

定理 2 如果判定性线性假设成立，那么 ETS 功能的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 满足有关预 $R(C)=\mathrm{Corruption}^{|\Phi|\leqslant k-1}$ 和 $T(C)$=$\mathrm{SS}_{\mathrm{sk}_i}$ 的 ACVBP。

证明 设 $C=C_{p,\mathbf{SK},\mathrm{pk}_e}$，下面将说明任何一个多项式时间的区分器 D 在可以询问预言 $\{C_{p,\mathbf{SK},\mathrm{pk}_e},\mathrm{SS}_{\mathrm{sk}_i},\mathrm{Corruption}^{|\Phi|\leqslant k-1}\}$ 的情况下都无法区分模拟器 S 输出的分布与真正的 $\mathrm{Obf}_{\mathrm{ETS}}$ 输出的分布。

用反证法证明。假设存在一个区分器 $D^{\ll C,T(C),R(C)\gg}$ 以不可忽略的概率 δ 来区分 C' 和 C''，即 $\mathrm{Pr}_{\mathrm{Nick}}-\mathrm{Pr}_{\mathrm{Junk}}>\delta$，其中，$\mathrm{Pr}_{\mathrm{Nick}}$ 是对真实 sk_i 加密混淆输出的概率，$\mathrm{Pr}_{\mathrm{Junk}}$ 是模拟器 S 对 sk_i' 加密输出的概率，具体如下。

$$
\mathrm{Pr}_{\mathrm{Nick}}=\mathrm{Pr}\left[\begin{array}{c}(p,\mathbf{SK})\leftarrow\mathrm{Setup}(1^\lambda,k,n);\\(\mathrm{pk}_e,\mathrm{sk}_e)\leftarrow\mathrm{KG}(p);\\z=(C_{i1},C_{i2},C_{i3})\leftarrow\mathrm{Enc}(\mathrm{pk}_e,\mathrm{sk}_i)_{i\in\{1,2,\cdots,n\}};:b=1\\C'\leftarrow R_{p,z,\mathrm{pk}_e};\\b\leftarrow D^{\ll C,T(C),R(C)\gg}(C');\end{array}\right]
$$

$$
\mathrm{Pr}_{\mathrm{Junk}}=\mathrm{Pr}\left[\begin{array}{c}(p,\mathbf{SK})\leftarrow\mathrm{Setup}(1^\lambda,k,n);\\(\mathrm{pk}_e,\mathrm{sk}_e)\leftarrow\mathrm{KG}(p);\\C''\leftarrow S^{\ll C_{p,\mathbf{SK},pk_e}\gg}();\\b\leftarrow D^{\ll C,T(C),R(C)\gg}(C'');\end{array}:b=1\right]
$$

$\mathrm{Pr}_{\mathrm{Junk}}$ 中构造的模拟器 $S^{\ll C_{p,\mathbf{SK},\mathrm{pk}_e}\gg}$ 将按以下方式定义。

从 $C_{p,\mathbf{SK},\mathrm{pk}_e}$ 解析 $\mathrm{pk}_e=(\mathrm{pk}_{e1},\mathrm{pk}_{e2})$ 和 p，随机选择 $x_{i1},x_{i2}\in Z_q$，$\mathbf{SK}'=(\mathrm{sk}_1',\mathrm{sk}_2',\cdots,\mathrm{sk}_n')\in G$，计算 $(\mathrm{pk}_{e1}^{x_{i1}},\mathrm{pk}_{e2}^{x_{i2}},d_i)\leftarrow\mathrm{Enc}(\mathrm{pk}_e,\mathrm{sk}_i')$，其中 $d_i=g^{x_{i1}+x_{i2}}\mathrm{sk}_i'$

表示对 sk_i' 的加密，令 $\mathrm{Junk}=\{\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},d_i\}_{i\in\{1,2,\cdots,n\}}$，输出 $C''=R_{p,\mathrm{Junk},\mathrm{pk}_e}$。显然 $R_{p,\mathrm{Junk},\mathrm{pk}_e}$ 与 R_{p,z,pk_e} 分布相同。

为了攻破线性加密方案的 IND-CPA。模拟器 D 与敌手(A，B)交互进行以下游戏。

1) 系统初始化。A 运行算法 $(q,G,G_T,e,g,p,\mathbf{SK}=(\mathrm{sk}_1,\mathrm{sk}_2,\cdots,\mathrm{sk}_n),a_0)\leftarrow\mathrm{Setup}(1^\lambda,k,n)$，随机选取 $\mathbf{SK}'=(\mathrm{sk}_1',\mathrm{sk}_2',\cdots,\mathrm{sk}_n')\in G$，设 $c=(q,G,G_T,e,g,\mathbf{SK},\mathbf{SK}')$，输出 c。

2) Corruption 询问。D 随机选择 $i\in\{1,2,\cdots,n\}$，A 返回私钥 sk_i 给 D。在整个游戏中，D 最多只能获得 $k-1$ 份私钥。

3) 部分签名询问。输入消息 m、私钥 $\mathrm{sk}_i(i\in\{1,2,\cdots,n\}$，运行 $\sigma_i\leftarrow\mathrm{SS}(p,\mathrm{sk}_i,m)$ 和 $\sigma\leftarrow\mathrm{SC}(\Phi,\sigma_j)(j\in\Phi)$，返回 σ 结果给 D。

4)加密门限签名询问。输入消息 m、pk_e 和公钥 p，运行算法 $\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 返回 $C_{p,\mathbf{SK},\mathrm{pk}_e}(m)$ 给 D。

5) 挑战阶段。输入 c，B，运行算法 $(\mathrm{pk}_e,\mathrm{sk}_e)\leftarrow\mathrm{KG}(p)$，随机选取 $b\in\{0,1\}$，$x_{i1},x_{i2}\in Z_q$，计算 $\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}}$。若 $b=0$，设 $\mathrm{sk}_i=d_i$，否则令 $\mathrm{sk}_i'=d_i$，之后运行 $(\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},\mathrm{pk}_{e1}{}^{x_{i1}+x_{i2}}d_i)\leftarrow\mathrm{Enc}(\mathrm{pk}_e,d_i)$，输出 $\mathrm{ct}=(\mathrm{pk}_{e1}{}^{x_{i1}},\mathrm{pk}_{e2}{}^{x_{i2}},\mathrm{pk}_{e1}{}^{x_{i1}+x_{i2}}d_i)$ 和 pk_e 给 A，其中 $i\in\{1,2,\cdots,n\}$。若 $b=1$，则 $R_{p,\mathrm{ct},\mathrm{pk}_e}$ 等价于由模拟器 S 生成的 $R_{p,\mathrm{Junk},\mathrm{pk}_e}$，否则 $R_{p,\mathrm{ct},\mathrm{pk}_e}$ 是 $\mathrm{Obf}_{\mathrm{ETS}}$ 算法真实有效的输出。

6)猜测阶段。输入公钥 p 和 ct，A 按照 ETS 功能生成 $C_{p,\mathbf{SK},\mathrm{pk}_e}$，按照 $\mathrm{Obf}_{\mathrm{ETS}}$ 算法生成 $R_{p,\mathrm{ct},\mathrm{pk}_e}$。若 $D^{\ll C,T(C),R(C)\gg}(R_{p,\mathrm{ct},\mathrm{pk}_e})=1$，则 $b'=0$，否则随机选取 $b'\in\{0,1\}$，输出 b'。

在上述安全游戏中，计算 $\Pr[b=b']$ 如下

$$\Pr[b=0\wedge b'=0]=\Pr[b=0]\cdot\Pr[D^{\ll C,T(C),R(C)\gg}(R_{p,\mathrm{ct},\mathrm{pk}_e})=1\,|\,b=0]+\Pr[b=0]\times\frac{1}{2}\cdot\Pr[D^{\ll C,T(C),R(C)\gg}(R_{p,\mathrm{ct},\mathrm{pk}_e})\neq1\,|\,b=0]=\frac{1}{2}\Pr_{\mathrm{Nick}}+\frac{1}{2}\left(\frac{1}{2}(1-\Pr_{\mathrm{Nick}})\right)=\frac{1+\Pr_{\mathrm{Nick}}}{4}$$

$$\Pr[b=1\wedge b'=1]=\Pr[b=1]\cdot\frac{1}{2}\Pr[D^{\ll C,T(C),R(C)\gg}(R_{p,\mathrm{ct},\mathrm{pk}_e})\neq1\,|\,b=1]=\frac{1}{2}\left(\frac{1}{2}(1-\Pr_{\mathrm{Junk}})\right)=\frac{1-\Pr_{\mathrm{Junk}}}{4}$$

通过上述的分析，敌手 A 抵抗 IND-CPA 赢得优势 $\mathrm{Adv}_A^{\text{IND-CPA}}$ 为

$$\mathrm{Adv}_A^{\text{IND-CPA}}=2\Pr[b=b']-1=2(\Pr[b=0\wedge b'=0]+\Pr[b=1\wedge b'=1])-1=2\left(\frac{1+\Pr_{\mathrm{Nick}}}{4}+\frac{1-\Pr_{\mathrm{Junk}}}{4}\right)-1=\frac{\Pr_{\mathrm{Nick}}-\Pr_{\mathrm{Junk}}}{4}=\frac{\delta}{2}$$

由前面假设可知 $\Pr_{\mathrm{Nick}}-\Pr_{\mathrm{Junk}}>\delta$，其中 δ 是不可忽略的，故敌手赢得 IND-CPA 的优势 $\mathrm{Adv}_A^{\text{IND-CPA}}$ 亦不可忽略，这与假设相矛盾。证明构造的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 满足有关预言 $T(C)$=SS_{sk_i} 和 $R(C)=\mathrm{Corruption}^{|\Phi|\leqslant k-1}$ 的 ACVBP，因此定理 2 成立。证毕。

定理 3 设 $T(C)$ 和 $R(C)$ 分别表示为 $\mathrm{SS}_{\mathrm{sk}_i}$ 和 $\mathrm{Corruption}^{|\Phi|\leqslant k-1}$，其中 $C=C_{p,\mathbf{SK},\mathrm{pk}_e}$。如果 ETS 的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 满足有关联预言 $T(C)$ 和 $R(C)$ 的 ACVBP，那么关于 ETS 功能的存在不可伪造性意味着关于混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的存在不可伪造性。

证明 若 TS 方案满足 ETS 功能的存在不可伪造性，但不满足混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的存在不可伪造性，将证明与定理 2 矛盾。给定一个概率多项式时间的区分器 D 可预言询问 $C_{p,\mathbf{SK},\mathrm{pk}_e}$、$\mathrm{SS}_{\mathrm{sk}_i}$ 和 $\mathrm{Corruption}^{|\Phi|\leqslant k-1}$，验证敌手 A 是否能够攻破混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的存在不可伪造性。

1) 输入电路 C(混淆电路或模拟电路)和辅助输入 z，抽样访问 $C_{p,\mathbf{SK},\mathrm{pk}_e}$ 提取 (p,pk_e)。

2) 抽样访问 $T(C)$ 和 $R(C)$，得到签名 $(m,\sigma,Q)\leftarrow A^{\ll\mathrm{SS}_{\mathrm{sk}_i},\mathrm{Corruption}^{|\Phi|\leqslant k-1}\gg}(p,\mathbf{VK},\mathrm{pk}_e,z)$。

3) 若 $(m,\sigma,p),m\notin Q$ 是一个有效的签名，输出 1。

若 C 是混淆电路，则区分者 D 输出 1 的概率等于敌手 A 攻破混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 的存在不可伪造性的概率，由前文假设可知，这个概率是不可忽略的。若 C 是模拟电路，则 D 输出 1 的概率可忽略的，否则敌手 A 将攻破 ETS 功能的存在不可伪造性，从而与假设矛盾，定理 3 得证。

由定理 2 和定理 3 可得，即使敌手可以获得混淆电路，门限签名方案是满足存在性不可伪造的。对 ETS 进行混淆，主要是增强安全性，而且混淆后的电路交给任意代理方执行是安全的，代理方无法从中得到任何有用的信息。

推论 1 在判定性线性假设下，门限签名方案

关于混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 是存在性不可伪造的。

证明　由文献[6]可知门限签名方案满足存在性不可伪造的，即

$$\left|\Pr\left[\begin{array}{c}(p,\mathbf{SK})\leftarrow\mathrm{Setup}(1^{\lambda},k,n);\\(m,\sigma,Q)\leftarrow A^{\ll \mathrm{SS}_{sk_i},\mathrm{Corruption}^{|\Phi|\leqslant k-1}\gg}(p,\mathbf{VK},z);\\\mathrm{Veriy}(m,\sigma,p)=1,m\notin Q;\end{array}\right]\right|<\frac{1}{p(\lambda)}$$

通过分析定义 4 的 ETS 功能的存在性不可伪造与上述定义不同之处仅在于敌手 A 给定了公钥 pk_e，不影响算法的安全性，因此 TS 与 ETS 功能的存在性不可伪造的定义一致，且定理 2 中证明 ETS 功能的混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 满足有关 $R(C)=\mathrm{Corruption}^{|\Phi|\leqslant k-1}$ 和 $T(C)=\mathrm{SS}_{\mathrm{sk}_i}$ 的 ACVBP。从而由定理 3 可知，门限签名方案关于混淆器 $\mathrm{Obf}_{\mathrm{ETS}}$ 是存在性不可伪造的。证毕。

5　加密门限方案的混淆的效率分析

5.1　理论分析

表 1 分别列出了算法 Setup、$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$、$\mathrm{Obf}_{\mathrm{ETS}}$ 和 R_{p,z,pk_e} 的计算复杂度，其中在 Z_q 上定义的运算为 Rand、Add 和 Mult，在 G 上定义的运算为 Rand、Mult 和 Exp，在 G_T 上定义的运算 Mult，Pair 为 $e:G\times G\to G_T$ 上的双线性对运算。Rand、Add、Mult 和 Exp 分别表示生成随机元的数、加法运算、乘法运算和指数运算，Pair 表示对运算。其中，由于 $m=m_1\cdots m_n\in\{0,1\}^n$，故 $\prod_{i=1}^{n}u_i^{m_i}=\prod_{i=1}^{n}x_i$，当 $m_i=1$ 时，$x_i=u_i$，否则 $x_i=1$，因此 $\prod_{i=1}^{n}u_i^{m_i}$ 的运算为 G 上的乘法运算。从表 1 可看出，算法的时间代价随 n 和 k 个数的增加呈线性增长关系，显然混淆电路与原电路相比，在计算性能上没有明显优势。

5.2　数值实验分析

通过实验仿真分析了算法的计算成本。本次仿真是在 Linux 平台上进行，使用 PBC 函数库[20]，用 C 语言编程，主机 CPU 主频为 2.9 GHz，内存为 4 GB。

1) 图 1 给出了 n=10、k=5 时，Setup、$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$、$\mathrm{Obf}_{\mathrm{ETS}}$ 和 R_{p,z,pk_e} 算法的平均运行时间，分别为 389 ms、622 ms、477 ms 和 781 ms，上述操作的平均运行时间维持在 800 ms 以内，R_{p,z,pk_e} 平均运行时间比 $\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 多 159 ms。

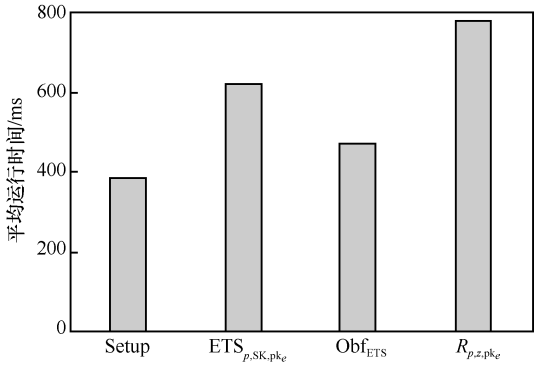

图 1　消息 $n=10,k=5$ 时，算法的运行时间

2) 图 2 给出当 $n=7$、10、15，$k=4$ 时，Setup、R_{p,z,pk_e}、$\mathrm{Obf}_{\mathrm{ETS}}$ 和 $\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 算法的运行时间。显然 Setup、$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$、$\mathrm{Obf}_{\mathrm{ETS}}$ 和 R_{p,z,pk_e} 的计算效率随着 n 的增加平均时间也逐渐增大，对比结果与理论分析结果相一致。Setup 平均运行时间分别为 282 ms、384 ms 和 548 ms，$\mathrm{Obf}_{\mathrm{ETS}}$ 平均运行时间分别为 345 ms、501 ms 和 730 ms，$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 平

表 1　　**计算复杂度**

空间	运算	Setup	$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$	$\mathrm{Obf}_{\mathrm{ETS}}$	R_{p,z,pk_e}
Z_q	Rand	k	$n+4$	$2n$	$n+4$
	Add	$(n+1)(k-1)$	2	$2n$	$2k$
	Mult	$\frac{(n+1)k(k-1)}{2}$	0	0	$2k$
G	Rand	$n+2$	0	0	0
	Mult	0	$2k+n+1$	$n+1$	$4k+n+1$
	Exp	$2n+1$	$2n+2k+6$	$3n+1$	$2n+4k+6$
G_T	Mult	0	1	0	1
$e:G\times G\to G_T$	Pair	0	3	0	3

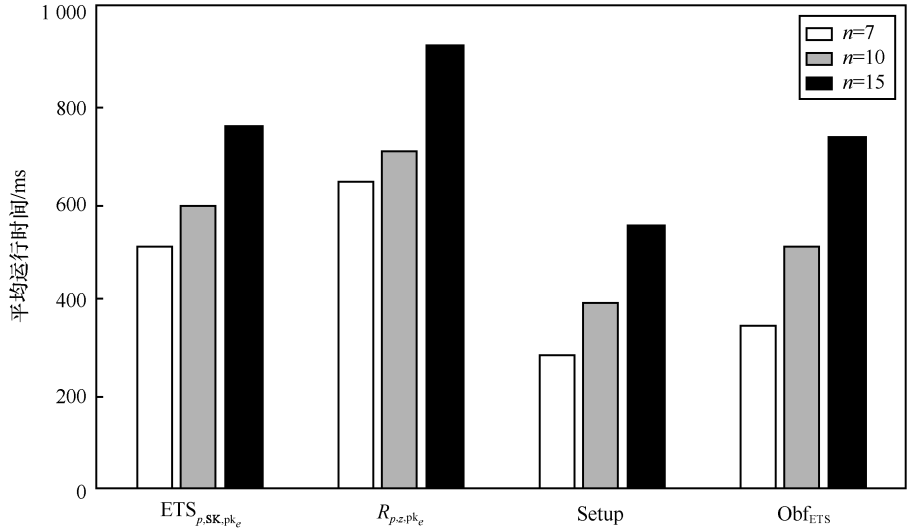

图 2 当 $k=4$ 时算法的计算效率

均运行时间分别为 498 ms、585 ms 和 749 ms，而 R_{p,z,pk_e} 平均运行时间分别是 635 ms、700 ms 和 920 ms。$\mathrm{ETS}_{p,\mathbf{SK},\mathrm{pk}_e}$ 比 R_{p,z,pk_e} 快的原因是，前者比后者少 G 上的 $2k$ 个指数运算和 $2k$ 个乘法运算。

5.3 混淆器的应用

在分布式认证中，n 个移动节点中任意 k 个移动节点通过合作共同提供认证服务。如果一个认证节点被攻破，那么在这个认证节点上的私钥分片会被攻击者通过计算来获得，在第 k 个认证节点被攻破后，攻击者可用窃取的 k 个私钥分量，生成认证私钥或者直接用窃取的认证私钥进行非法认证。为了避免攻击造成节点的私钥分量和认证私钥的泄露，因此使用加密方法将私钥分片“隐藏”在被混淆的程序中。

6 结束语

传统密码学建立在黑盒模型上，并假设运行环境是安全的。但现实中，软件的执行过程对攻击者是可见的，攻击者通过观察或者执行密码软件很容易就可以获得密钥信息。为了保护 TS 中参与者的私钥，首先提出了一个 ETS 功能的混淆算法，其次对 ACVBP 进行推广，并证明所提混淆器是安全的。理论和实验分析证明本文所提算法适合于分布式系统的应用。

参考文献：

[1] DESMEDT Y. Threshold cryptosystems[C]//Advances in Cryptology-CRYPTO'89. Berlin: Springer, 1989:1-14.

[2] YANG W, LUO W, LUO X, et al. Fully distributed certificateless threshold signature without random oracles[J]. Science China(Information Sciences), 2018, 61(9):259-269.

[3] 张艳硕,李文敬,陈雷,等. 基于特征值的可验证特殊门限秘密共享方案[J]. 通信学报, 2018, 39(8):169-175.
ZHANG Y S, LI W J, CHEN L, et al. Verifiable special threshold secret sharing scheme based on eigenvalue, Journal on Communications, 2018, 39(8):169-175.

[4] LIBERT B, YUNG M. Adaptively secure non-interactive threshold cryptosystems[C]// International Conference on Automata. Berlin: Springer, 2011: 588-600.

[5] ZHOU G, ZENG P, YUAN X, et al. An efficient code-based threshold ring signature scheme with a leader-participant model[J]. Security & Communication Networks, 2017, 2017:1.

[6] LI J, YUEN T H, KIM K. Practical threshold signatures without random oracles[C]// International Conference on Provable Security. Berlin: Springer, 2007: 198-207.

[7] 陈立全, 朱政, 王慕阳,等. 适用于移动互联网的门限群签名方案[J]. 计算机学报, 2018, 41(5): 86-101.
CHEN L Q, ZHU Z, WANG M Y, et al. A threshold group signature scheme for mobile Internet application[J]. Chinese Journal of Computers, 2018, 41(5): 86-101.

[8] 任艳丽, 徐丹婷, 张新鹏, 等. 基于门限环签名的可删除区块链[J]. 通信学报, 2019, 40(4): 75-86.
REN Y L, XU D T, ZHANG X P, et al. Deletable blockchain based on threshold ring signature[J]. Journal on Communications, 2019, 40(4): 75-86.

[9] 徐明, 李旭如, 刘朝斌,等. 基于双重代理密钥的船舶自组网门限签名方案[J]. 通信学报, 2018, 39(7): 170-179.
XU M, LI X R, LIU C B, et al. Dual-proxy key-based threshold signature scheme for ship ad-hoc network[J]. Journal on Communications,

2018, 39(7): 170-179.

[10] MOWBRAY M, PEARSON S, SHEN Y. Enhancing privacy in cloud computing via policy-based obfuscation[J]. Journal of Supercomputing, 2012, 61(2): 267-291.

[11] SHI Y, ZHANG Q, LIANG J W, et al. Obfuscatable anonymous authentication scheme for mobile crowd sensing[J]. IEEE Systems Journal, 2018, PP(99): 1-12.

[12] BARAK B, GOLDREICH O, IMPSGLIAZZO R, et al. On the (im)possibility of obfuscating programs[J]. Lecture Notes in Computer Science, 2001, 2139(2): 1-18.

[13] HOHENBERGER S, ROTHBLUM G N, SHELAT A, et al. Securely obfuscating re-encryption[M]. Berlin: Springer, 2007.

[14] HADA S. Secure obfuscation for encrypted signatures[C]// International Conference on Theory & Applications of Cryptographic Techniques. Berlin: Springer, 2010: 92-112.

[15] WATERS B. Efficient identity-based encryption without random oracles[C]// International Conference on the Theory & Applications of Cryptographic Techniques. Berlin: Springer, 2005:14-127.

[16] BONEH D, BOYEN X, SHACHAM H. Short group signatures[C]// 24th Annual International Cryptology Conference. Springer Berlin, 2004: 41-55.

[17] SHI Y, ZHAO Q P, FAN H F, et al. Secure obfuscation for encrypted group signatures[J]. Plos One, 2015, 10(7):1.

[18] 陈兴发, 高崇志, 姚正安, 等.安全加密的环签名混淆器[J]. 中山大学学报(自然科学版), 2014, 53(1):8-17.
CHEN X F, GAO C Z, YAO Z A, et al. Secure obfuscation for encrypted ring signatures[J]. Acta Scientiarum Naturalium Universitatis Sunyatseni, 2014, 53(1):8-17.

[19] SHI Y, HAN J X, FAN H F, et al. Protecting encrypted signature functions against intrusions on computing devices by obfuscation[J]. IEEE Access, 2016, 4: 6401-6415.

[20] TAKAGI T, OKAMOTO T, OKAMOTO E, et al. Pairing-based cryptography–pairing 2007[M]. Berlin: Springer, 2007.

[作者简介]

李亚红（1984– ），女，甘肃定西人，博士，兰州交通大学副教授，主要研究方向为密码学和信息安全。

王彩芬（1963– ），女，河北安国人，博士，西北师范大学教授、博士生导师，主要研究方向为密码学、网络安全和信息安全。

张玉磊（1979– ），男，甘肃靖远人，博士，西北师范大学副教授，主要研究方向为信息安全。

杨小东（1981– ），男，甘肃甘谷人，博士，西北师范大学副教授，主要研究方向为信息安全学及云计算安全。

黄海燕（1988– ），女，甘肃张掖人，博士，兰州交通大学副教授，主要研究方向为认知协作传输。