软件学报 ISSN 1000-9825, CODEN RUXUEW
*Journal of Software* [doi: 10.13328/j.cnki.jos.007210]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

# 基于代数关系的轻量级密码 DEFAULT 统计故障分析*

李 玮[1,2,3,4], 秦梦洋[1], 谷大武[2], 连 晟[1], 温云华[1]

[1](东华大学 计算机科学与技术学院, 上海 201620)

[2](上海交通大学 计算机科学与工程系, 上海 200240)

[3](上海交通大学 上海市可扩展计算与系统重点实验室, 上海 200240)

[4](上海交通大学 上海市信息安全综合管理技术研究重点实验室, 上海 200240)

通信作者: 李玮, E-mail: weili@dhu.edu.cn

**摘 要**: DEFAULT 是于 2021 年亚洲密码学年会中提出的一种新型轻量级密码算法, 适用于保护物联网中的微型芯片、微控制器和传感器等设备的信息安全. 本文基于唯密文的基本假设, 针对 DEFAULT 密码提出了一种基于代数关系的统计故障分析方法. 该方法使用随机半字节故障模型, 通过对代数关系的构造分析并结合故障注入前后中间状态的统计分布变化来破译密码. 此外, 本文采用 AD 检验—平方欧氏距离、AD 检验—极大似然估计和 AD 检验—汉明重量等新型区分器, 最少仅需 1 344 个故障即可以 99% 及以上的成功率破解该算法的 128 比特原始密钥. 理论分析和实验结果表明, DEFAULT 密码不能抵抗基于代数关系的统计故障分析的攻击. 该研究为其它轻量级分组密码算法的安全性分析提供了有价值的参考.

**关键词**: DEFAULT; 轻量级密码; 密码分析; 统计故障分析; 代数关系

**中图法分类号**: TP309

中文引用格式: 李玮, 秦梦洋, 谷大武, 连晟, 温云华. 基于代数关系的轻量级密码DEFAULT统计故障分析. 软件学报. http://www.jos.org.cn/1000-9825/7210.htm

英文引用格式: Li W, Qin M, Gu D, Lian S, Wen Y. DEFAULT Lightweight Cryptosystem Against Statistical Fault Analysis Based on Algebraic Relationship. Ruan Jian Xue Bao/Journal of Software (in Chinese). http://www.jos.org.cn/1000-9825/7210.htm

## DEFAULT Lightweight Cryptosystem Against Statistical Fault Analysis Based on Algebraic Relationship

LI Wei[1,2,3,4], QIN Meng-Yang[1], GU Da-Wu[2], LIAN Sheng[1], WEN Yun-Hua[1]

[1](School of Computer Science and Technology, Donghua University, Shanghai 201620, China)

[2](Department of Computer and Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China)

[3](Shanghai Key Laboratory of Scalable Computing and System, Shanghai Jiao Tong University, Shanghai 200240, China)

[4](Shanghai Key Laboratory of Integrate Administration Technologies for Information Security, Shanghai Jiao Tong University, Shanghai 200240, China)

**Abstract**: DEFAULT, a new lightweight cryptosystem presented at Asiacrypt in 2021, is designed to protect the information security of Internet of Things (IoT) devices, such as microchips, microcontrollers, and sensors. Based on the ciphertext-only attack assumption, the statistical fault analysis of the DEFAULT cipher with the algebraic relationship is proposed. The statistical fault analysis uses the random nibble-oriented fault model. It not only combines statistical distributions of the intermediate states before and after the fault injections but also takes advantage of the algebraic relationship and novel distinguishers, including Anderson Darling test–Square Euclidean imbalance, Anderson Darling test–Maximum likelihood estimate, and Anderson Darling test–Hamming weight. The analysis requires at least 1344 faults to achieve the reliability of 99% in the recovery of the 128-bit secret key of DEFAULT. The theoretical analysis and experimental

* 基金项目: 国家重点研发计划 (2020YFA0712300); 国家自然科学基金 (62172395, 62102077, 62072307); 信息安全国家重点实验室开放课题 (2021-MS-05); 上海市扬帆计划 (21YF1401200); 中央高校基本科研业基金 (223202D-25)
收稿时间: 2023-12-26; 修改时间: 2024-02-07; 采用时间: 2024-04-12; jos 在线出版时间: 2024-06-20

results show that the DEFAULT lightweight cryptosystem is not resistant to the statistical fault attack based on the algebraic relationship. This study provides an important reference for the security analysis of the other lightweight cryptosystems.

**Key words**: DEFAULT; lightweight cryptosystem; cryptanalysis; statistical fault analysis; algebraic relationship

随着 5G 通信、嵌入式和云计算等技术的快速发展, 越来越多的设备被赋予万物互联和智能化的能力. 物联网设备已经广泛融入了人们生活的各个领域, 智能家居、智慧监测设备和智能传输设备极大地方便了人们的生活, 智能制造、智慧医疗和智能农业等极大地提高了工作效率, 但同时也带来了数据隐私和信息安全的挑战[1−3]. 这些物联网设备通常专注于采集和记录传感器信息, 并将这些信息上传至云服务器. 因此在使用过程中, 这些设备难免会收集到用户的敏感信息和隐私数据. 如果这些信息和数据不加防护地通过公开信道传输至云服务器, 很可能被攻击者截获、篡改和伪造, 造成严重的信息泄露威胁, 因此保证物联网设备的信息安全十分重要. 在信息安全领域, 密码算法常用于来保护信息的安全性. 但是, 物联网设备如射频识别标签 (RFID)、无线传感器和微控制器等, 通常体积小巧且算力有限, 传统密码算法难以保证效率和能耗, 因此轻量级密码算法应运而生[4−5]. 这些轻量级密码算法具有资源消耗低、安全等级高和易于软硬件实现等特点, 可以兼顾安全、高效和节能, 并且易在物联网设备上实现, 因此被广泛用于保证物联网设备信息的保密性、完整性和可认证性[6−10].

侧信道分析是指攻击者利用密码设备运行过程中的功耗、耗时和故障等信息来破译密码算法[11−13]. 在物联网环境中, 攻击者可以相对轻松地接近物理设备, 获取其使用权限, 甚至破坏这些设备, 从而迫使设备中的密码在运行过程中产生故障并且输出错误信息. 攻击者通过收集这些错误信息进行分析, 进而可以快速破译出密码的密钥, 该攻击方法称为故障分析[14]. 随着故障分析技术的不断发展, 国内外学者逐渐提出了差分故障分析、统计故障分析和代数故障分析等多种方法[15−22], 这些故障分析方法对密码的安全性造成了极大的威胁. 随着微电子技术的发展, 异常电流、时钟干扰和激光照射等故障注入的方式日益多样化, 故障注入的位置也愈发精细. 因此, 故障分析已经成为检测密码算法实现安全性的重要方法之一.

常见的密码分析方法根据攻击者的不同能力可以划分为: 选择明文攻击、已知明文攻击、选择密文攻击和唯密文攻击等. 其中, 差分分析、线性分析、代数分析等经典密码分析, 以及差分故障分析、功耗分析等侧信道分析均属于已知或选择明文攻击的范畴. 统计故障分析基于唯密文的基本假设, 仅需攻击者截获密码设备的密文, 然后利用中间状态的统计变化和区分器实现错误密钥的筛除从而破译密钥, 对攻击者的能力要求最弱, 所以在实际中更容易实施, 常用于检测物联网设备的实现安全性[23−25].

DEFAULT 是 Baksi 等学者在 2021 年亚洲密码学年会中提出的一种新型轻量级分组密码, 采用 128 比特的分组长度和密钥长度, 具有良好的安全性和兼容性. 并且它采用特别的 S 盒设计, 可以有效地抵抗常见的经典密码分析和侧信道分析, 增加分析难度[26−29]. 设计者仅用 2 377 个等效电路门即在台积电 65 纳米标准单元库上进行了物理实现, 较低的实现代价和良好的安全性使其可以应用于 RFID、无线传感器和微控制器等物联网设备[27]. 目前, 国内外尚未有针对 DEFAULT 密码实现统计故障分析的公开成果.

本文基于唯密文攻击, 结合统计故障分析和代数关系构造, 使用随机半字节故障模型, 提出了基于代数关系的新型统计故障分析方法. 鉴于该密码中 S 盒的特殊设计, 若采用传统的统计故障分析方法, 仅能完成不低于 $2^{89.84}$ 候选密钥空间的恢复, 难以破译唯一的原始密钥. 本文构造基于代数关系的统计故障分析方法, 能够有效降低候选密钥空间, 从而快速破解 DEFAULT 密码的唯一密钥. 并且, 本文提出了基于安德森达林拟合度检验 (Anderson Darling test, AD 检验) 的新型区分器, 譬如 AD 检验—平方欧氏距离 (AD–SEI)、AD 检验—极大似然估计 (AD–MLE) 以及 AD 检验—汉明重量 (AD–HW). 理论分析和实验结果表明, 新方法能够以 99% 及以上成功率恢复 DEFAULT 密码的 128 比特原始密钥, 在提升攻击的成功率, 降低故障数、耗时和复杂度等方面均有较佳表现.

本文第 1 节简述 DEFAULT 密码的安全性分析、代数分析和统计故障分析的相关工作. 第 2 节介绍了 DEFAULT 密码算法的参数和加解密过程. 第 3 节提出了基于代数关系的统计故障分析, 具体包括基本假设、故障模型、S 盒分析及主要步骤. 第 4 节从故障数和时间等多个指标分析实验结果. 第 5 节总结全文.

## 1 相关工作

自 DEFAULT 密码于亚洲密码学年会公布以来, 研究学者对其安全性进行了深入研究[26−29]. 设计者 Baksi 等学者指出, DEFAULT 密码不仅可以抵御线性分析、差分分析等经典密码分析方法, 而且能够抵抗常见侧信道分析[27]. 2021 年, Dey 等学者针对该密码实现了差分故障攻击, 通过在最后一轮注入故障进行筛选, 利用差分关系, 将密钥空间缩减到 $2^{16}$ [28]. 后来, 设计者升级了密钥编排方案的构造, 将每轮运算采用相同的原始密钥, 更新为基于原始密钥产生四个轮密钥, 循环参与到每轮运算中[27]. 2022 年, Nageler 等学者在最后一轮注入半字节故障, 结合最后六轮的逐轮故障注入, 并建立归一化方程, 利用差分故障分析进一步降低了分析代价[29]. 次年, Jang 等学者提出了 DEFAULT 密码的量子实现并证明其安全性[30]. 表 1 给出了 DEFAULT 密码的现有安全性分析方法比较.

表 1 DEFAULT 密码的安全性分析对比

| 分析类型 | 基本假设 | 攻击轮数 | 首次故障注入轮 | 文献 |
|---|---|---|---|---|
| 线性分析 | 已知明文攻击 | 11 | – | [27] |
| 不可能差分分析 | 选择明文攻击 | 7 | – | [27] |
| 差分分析 | 选择明文攻击 | 8 | – | [27] |
| 代数分析 | 选择明文攻击 | 8 | – | [27] |
| 积分分析 | 选择明文攻击 | 12 | – | [27] |
| 差分故障分析 | 选择明文攻击 | 80 | 倒数第一轮 | [28,29] |
| 统计故障分析 | 唯密文攻击 | 80 | 倒数第三轮 | 本文 |

在基于代数关系的密码分析发展历程中, 国内外学者提出了多种攻击方法, 例如代数分析、代数故障分析等, 用于检验密码的设计和实现安全性. 2003 年, Courtois 等学者首次提出了代数分析, 通过将密码表示为包含若干变量的方程组, 并将搜集到的明文密文对代入方程求解, 进而破译了 Toyocrypt 和 LILI 等流密码[31]. 2007 年, Courtois 等学者针对分组密码 DES 算法的组成部件分别建模为代数方程, 利用代数求解器为方程求解, 有效提升了分析效率[32]. 2009 年, Courtois 等学者针对流密码 Hitag2, 使用代数求解器实现了代数攻击, 可以在 6 小时内恢复完整密钥[33]. 2013 年, Zhang 等学者针对轻量级分组密码 Piccolo 算法采用代数故障分析方法, 对加解密过程和密钥编排算法建立代数方程, 仅需单个故障破译出全部密钥[34]. 2014 年, Zhao 等学者仅用 8 个故障实现了针对分组密码 GOST 的代数故障分析, 提高了攻击效率[35]. 2017 年, Chen 等学者针对分组密码 HIGHT, 提出用一组代数方程组来表示密码和注入故障的方法, 实现了代数故障分析[21]. 2019 年, Le 等学者通过引入差分轨迹, 借助求解器实现了针对分组密码 SIMON 的代数差分故障分析[22]. 2021 年, Gruber 等学者通过注入单比特翻转故障, 实现了认证加密算法 Subterranean 2.0 的代数故障分析[36]. 近两年, Fang 等学者基于 S 盒分解, 使用代数持续故障分析, 在 9 秒内破译了分组密码 SKINNY[37,38]. 2023 年, Qiu 等学者改进代数系统, 加快了求解效率, 将代数故障分析应用于 AES、Serpent 和 SPEEDY 等经典分组密码中, 拓展了代数故障分析的攻击范围[39]. 上述分析方法均在选择明文攻击下实现, 利用代数关系对加密过程或密钥编排算法建立代数方程, 再求解并筛选密钥.

2013 年, Fuhr 等学者首次提出了统计故障分析[18], 该方法利用故障注入产生的随机故障密文, 根据加密过程可以计算部分中间状态值, 然后结合统计关系, 利用区分器对密钥进行筛选, 成功破译了 AES 分组密码. 统计故障分析基于唯密文的基本假设, 仅要求攻击者能够截获密文即可, 对攻击者的能力要求最弱, 因此在实际中更容易实现. 2017 年, Nozaki 等学者使用统计故障分析, 通过引入时钟故障并对最后两轮加密过程进行分析, 成功破译了 Midori 轻量级分组密码算法[40]. 2019 年, Ramezanpour 等学者通过在最后一轮的一对 S 盒中注入故障, 实现了对 ASCON 认证加密算法的统计故障分析[41]. 2021 年, Li 等学者通过在倒数第二轮注入半字节随机故障, 实现了对轻量级分组密码 Piccolo 的统计故障分析[19]. 2022 年, Bagheri 等学者通过引入持续故障, 实现了对 DES 和 Camellia 等分组密码的统计无效持久故障分析[25].

本文针对 DEFAULT 密码的内部结构, 将代数关系和统计故障分析相结合, 提出了基于代数关系的新型统计故障分析方法, 将故障注入位置扩展至更深的轮数, 并且实现了唯密文基本假设下的密钥破解. 同时, 本文提出基

于 AD 检验的新型区分器设计, 在成功率、故障数和耗时等方面优于经典区分器, 表 2 以 AES 密码、Piccolo 密码以及 DEFAULT 密码的密钥恢复为例, 总结了各区分器破译密码的结果对比.

表 2　针对 AES、Piccolo 和 DEFAULT 的统计故障分析结果比较

| 区分器 | 密码 | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| | 指标 | | | | | | | | |
| | AES[18] | | | Piccolo[19] | | | DEFAULT | | |
| | 故障注入轮 | 故障数(个) | 成功率(%) | 故障注入轮 | 故障数(个) | 成功率(%) | 故障注入轮 | 故障数(个) | 成功率(%) |
| SEI | 倒数第一轮 | 320 | 99 | 倒数第二轮 | ∞ | 28 | 倒数第三轮 | ∞ | 6 |
| MLE | 倒数第一轮 | 224 | 99 | 倒数第二轮 | 313 | 99 | 倒数第三轮 | 2 080 | 84 |
| HW | 倒数第一轮 | 288 | 99 | 倒数第二轮 | 364 | 99 | 倒数第三轮 | 1 472 | 99 |
| AD–SEI | – | – | – | – | – | – | 倒数第三轮 | 2048 | 99 |
| AD–MLE | – | – | – | – | – | – | 倒数第三轮 | 1 408 | 99 |
| AD–HW | – | – | – | – | – | – | 倒数第三轮 | 1 344 | 99 |

## 2　DEFAULT 密码简介

### 2.1　符号说明

记 $Z_2^e$ 为 $e$ 比特的二进制向量集合.

记 $X \in (Z_2^4)^{32}$ 和 $Y \in (Z_2^4)^{32}$ 分别为 128 比特的明文和密文.

记 $K \in (Z_2^4)^{32}$ 和 $RK_r \in (Z_2^4)^{32}$ 分别为 128 比特的原始密钥和第 $r$+1 轮密钥, 其中 $r \in [0,79]$.

记 $SC_C$ 和 $SC_L$ 分别为内外核轮的 S 盒替换, $AC_C$ 和 $AC_L$ 分别为内外核轮的轮常数加, $PB$ 和 $AK$ 分别为比特置换和轮密钥加. 记 $SC_C^{-1}$、$SC_L^{-1}$、$AC_C^{-1}$、$AC_L^{-1}$、$PB^{-1}$ 和 $AK^{-1}$ 分别为以上操作的逆运算.

记 $A_r$、$B_r$、$C_r$ 和 $D_r$ 分别为第 $r$+1 轮 S 盒替换、比特置换、轮常数加和轮密钥加的输出状态值, 其中, $r \in [0,79]$.

记 $\oplus$ 和 $\boxplus$ 分别为轮密钥加和轮常数加, mod、/和 $\|$ 分别为模、除法和级联运算.

### 2.2　算法参数

DEFAULT 是于 2021 年亚洲密码学年会中提出的一种新型轻量级分组密码[27]. 该算法采用代换置换网络结构, 分组长度和密钥长度均为 128 比特, 总轮数为 80 轮, 每轮的轮函数均包括 4 种运算: S 盒替换、比特置换、轮常数加和轮密钥加, 如图 1 所示, 加密过程如算法 1 所示. 解密过程同加密过程, 轮密钥采用相反的顺序.

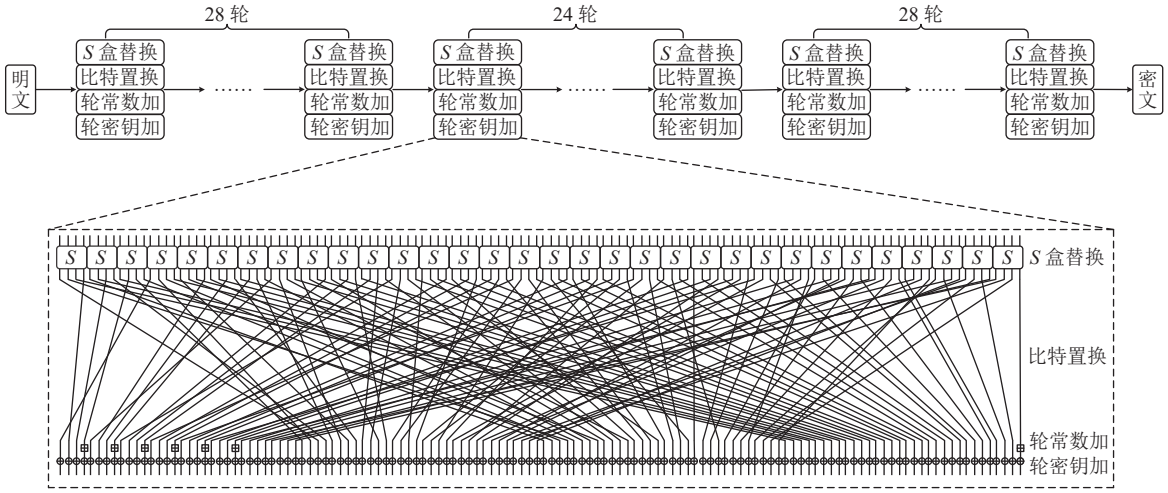


图 1　DEFAULT 密码示意图

算法 1. DEFAULT 密码的加密算法

输入: 明文 $X$ 和密钥 $K$

输出: 密文 $Y$

1. $S = X$
2. FOR $r = 0$ TO 27 DO
3. $\quad S = AK(AC_L(PB(SC_L(S))))$
4. END FOR
5. FOR $r = 28$ TO 51 DO
6. $\quad S = AK(AC_C(PB(SC_C(S))))$
7. END FOR
8. FOR $r = 52$ TO 79 DO
9. $\quad S = AK(AC_L(PB(SC_L(S))))$
10. END FOR
11. $Y = S$
12. RETURN $Y$

在轮数设计中, 整体 80 轮分别采用前 28 轮和后 28 轮为外核轮, 中间 24 轮为内核轮. 内外核轮采用不同的 S 盒和轮常数设计, 目的是分别抵御经典密码分析和常见侧信道分析, 如表 3 和 4 所示.

表 3 DEFAULT 密码使用的 S 盒

| 位置 | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 外核轮 | 0 | 3 | 7 | 14 | 13 | 4 | 10 | 9 | 12 | 15 | 1 | 8 | 11 | 2 | 6 | 5 |
| 内核轮 | 1 | 9 | 6 | 15 | 7 | 12 | 8 | 2 | 10 | 14 | 13 | 0 | 4 | 3 | 11 | 5 |
| 密钥编排 | 0 | 3 | 7 | 14 | 13 | 4 | 10 | 9 | 12 | 15 | 1 | 8 | 11 | 2 | 6 | 5 |

表 4 DEFAULT 密码使用的轮常数

| 位置 | 轮常数 | | | | | | | | | | | | | | 轮数 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 外核轮 | 1 | 3 | 7 | 15 | 31 | 62 | 61 | 59 | 55 | 47 | 30 | 60 | 57 | 71 | 第1–28轮 |
| | 39 | 14 | 29 | 58 | 53 | 43 | 22 | 44 | 24 | 48 | 33 | 2 | 5 | 11 | 第53–80轮 |
| 内核轮 | | 1 | 3 | 7 | 15 | 31 | 62 | 61 | 59 | 55 | 47 | 30 | 60 | | 第29–52轮 |
| | | 57 | 51 | 39 | 14 | 29 | 58 | 53 | 43 | 22 | 44 | 24 | 48 | | |
| 密钥编排 | 1 | | | | | | | | | | | | | | 第1–4轮 |

S 盒是常见分组密码中唯一的非线性组件, 其强度直接影响密码整体的安全性. 为了增强抵御侧信道分析的强度, 外核轮 S 盒采用包含 3 个特殊的差分输入和差分输出对的设计. 在外核轮中, 对于 S 盒的任意两个输入, 如果它们的差分输入等于特定的差分输入值 6、9 或 15, 那么它们的差分输出也是相同的, 并且结果分别为 10、15 或 5.

算法 2 给出了密钥编排算法[27]. 原始密钥 $K$ 经过 4 轮 S 盒替换、比特置换和轮常数加变换后, 得到 4 个初始轮密钥 $RK_0$、$RK_1$、$RK_2$ 和 $RK_3$, 其中, S 盒替换、比特置换运算与外核轮相同, 轮常数加变换使用的轮常数为 1, 如表 3 和表 4 所示. 初始轮密钥循环参与每轮运算中, 每轮轮密钥 $RK_r$ 依次对应初始轮密钥 $RK_{r \bmod 4}$, 其中 $r \in [0, 79]$.

算法 2. DEFAULT 密码的密钥编排算法

输入: 原始密钥 $K$

输出: 轮密钥 $RK_0, RK_1, \ldots, RK_{78}, RK_{79}$

1. $RK_0$=$K$
2. FOR $r$ = 1 TO 3 DO
3. $T = RK_{r-1}$
4. FOR $t$ = 1 TO 4 DO
5. $T = AC(PB(SC_L(T)))$
6. END FOR
7. $RK_r = T$
8. END FOR
9. FOR $r$ = 4 TO 79 DO
10. $RK_r = RK_{r \bmod 4}$
11. END FOR
12. RETURN $RK_0$, $RK_1$,…, $RK_{78}$, $RK_{79}$

## 3 基于代数关系的统计故障分析

在唯密文攻击下, 经典的统计故障分析方法一般需要从错误密文倒推中间状态, 并利用中间状态值的统计分布信息, 选择合适的区分器来筛选正确密钥. 由于 DEFAULT 密码的外核轮选用了特别的 S 盒设计, 攻击者在进行攻击倒推时, 会出现相同的中间状态分布对应不同的轮密钥, 待搜索的密钥空间达到 $2^{89.84}$ 及以上, 难以缩减到唯一破译密钥的有限范围. 本文提出一种结合代数关系的统计故障分析方法, 利用唯密文攻击获得的随机密文, 通过构造密钥编排方案中不同轮密钥之间的代数关系对密钥进行筛选, 并采用新型区分器去破译密码.

### 3.1 基本假设和故障模型

本文采用唯密文攻击的基本假设, 攻击者仅需获得多个由相同原始密钥加密过的密文. 在正确加密过程中, 半字节中间状态值通常服从均匀分布, 每个值出现的概率为:

$$100\% \div 16 = 6.25\%. \tag{1}$$

本文采用随机半字节的故障模型, 攻击者在密码设备加密过程中, 通过按位“与”的方式注入半字节故障. 中间状态值受故障影响后的分布会变得不均匀, 分布概率满足:

$$\left(\frac{3}{4}\right)^{4-hw(\varphi)} \cdot \left(\frac{1}{4}\right)^{hw(\varphi)} = \frac{3^{4-hw(\varphi)}}{4^4}, \tag{2}$$

其中, $\varphi$ 为中间状态的所有可能值, $hw(\varphi)$ 表示 $\varphi$ 的汉明重量. 本文采用随机半字节模型, $\varphi \in [0,15]$ 且 $hw(\varphi) \in [0,3]$. 图 2 给出了半字节中间状态正常加密时的均匀分布和受故障影响后的非均匀分布的分布率, 非均匀分布和均匀分布的概率之和分别为 100%.

### 3.2 S 盒及代数关系的分析构造

根据外核轮函数的设计, 若 S 盒替换运算的任意两个输入的差分值等于为 $\Delta in \in (Z_2^4)^{32}$, 那么该差分输出值必定为确定的 $\Delta out \in (Z_2^4)^{32}$. 结合 S 盒替换运算和 S 盒的差分特性, 有:

$$SC_L(C_{78} \oplus RK_{78}) \oplus SC_L(C_{78} \oplus RK_{78} \oplus \Delta in) = \Delta out, \tag{3}$$

可得:

$$SC_L(C_{78} \oplus RK_{78}) = SC_L(C_{78} \oplus RK_{78} \oplus \Delta in) \oplus \Delta out. \tag{4}$$

结合算法最后两轮运算, 有:

$$\begin{aligned} Y &= AK(AC_L(PB(SC_L(AK(C_{78}))))) \\ &= AC_L(PB(SC_L(C_{78}\oplus RK_{78})))\oplus RK_{79} \\ &= AC_L(PB(SC_L(C_{78}\oplus RK_{78}\oplus \Delta in)\oplus \Delta out))\oplus RK_{79} \\ &= AC_L(PB(SC_L(C_{78}\oplus RK_{78}\oplus \Delta in)))\oplus PB(\Delta out)\oplus RK_{79} \\ &= AC_L(PB(SC_L(C_{78}\oplus RK_{78}\oplus \Delta in)))\oplus (RK_{79}\oplus PB(\Delta out)). \end{aligned} \tag{5}$$

所以对于任意给定的输入输出差分对 $(\Delta in, \Delta out)$, 最后两轮密钥 $(RK_{78}, RK_{79})$ 均有对应的等价轮密钥 $(RK_{78}\oplus \Delta in, RK_{79}\oplus PB(\Delta out))$. 攻击者使用正确轮密钥、等价轮密钥分别对同一 S 盒输入进行加密, 均能够得到相同的输出.

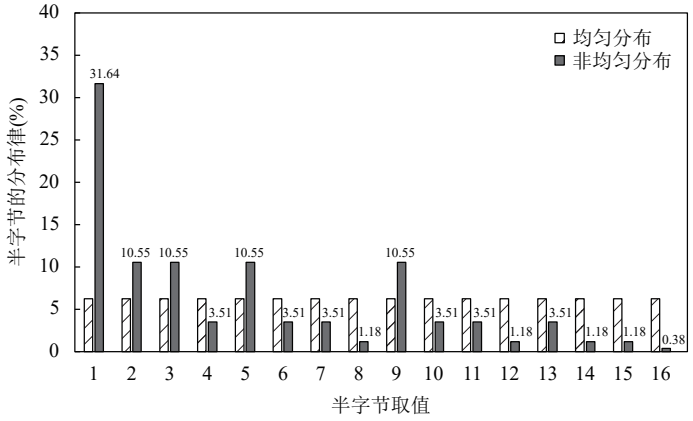


图 2　半字节分布律

在外核轮的设计中, 每个 S 盒含有 3 组特殊的差分输入和差分输出对, 分别为: (6, 10)、(9, 15) 和 (15, 5). 对于单个 S 盒, 若产生 4 组密钥解, 则这 4 组密钥解均能从 S 盒的输出倒推回正确的输入, 导致现有的统计故障分析难以对等价密钥进行筛选. 当故障注入轮数越深, 涉及的轮密钥数量越多, 相应的等价密钥空间越大, 进一步增加恢复唯一原始密钥的难度. 如表 5 所示, $rk_{77}$、$rk_{78}$ 和 $rk_{79}$ 分别表示最后三轮中运算中, 某个 S 盒替换运算对应的半字节轮密钥, $pb$ 表示对应位置半字节的比特置换操作.

表 5　最后两轮及最后三轮的密钥解

| 序号 | 最后两轮等价密钥 | 最后三轮等价密钥 |
|---|---|---|
| 1 | $(rk_{78}, rk_{79})$ | $(rk_{77}, rk_{78}, rk_{79})$ |
| 2 | $(rk_{78}\oplus 6, rk_{79}\oplus pb(10))$ | $(rk_{77}, rk_{78}\oplus 6, rk_{79}\oplus pb(10))$ |
| 3 | $(rk_{78}\oplus 9, rk_{79}\oplus pb(15))$ | $(rk_{77}, rk_{78}\oplus 9, rk_{79}\oplus pb(15))$ |
| 4 | $(rk_{78}\oplus 15, rk_{79}\oplus pb(5))$ | $(rk_{77}, rk_{78}\oplus 15, rk_{79}\oplus pb(5))$ |
| 5 | – | $(rk_{77}\oplus 6, rk_{78}\oplus pb(10), rk_{79})$ |
| 6 | – | $(rk_{77}\oplus 9, rk_{78}\oplus pb(15), rk_{79})$ |
| 7 | – | $(rk_{77}\oplus 15, rk_{78}\oplus pb(5), rk_{79})$ |

攻击者在进行统计故障分析时, 即使区分器能筛选出全部的等价密钥, 但仍然不能正确区分等价密钥与正确密钥. 对最后两轮加密过程进行分析时, 若每个 S 盒产生 4 组密钥解, 则对于全部 32 个 S 盒, 最终会产生 $4^{32} = 2^{64}$ 个等价密钥. 若分析过程扩展到最后三轮计算, 则会出现以下两种情况:

情况一. 若等价轮密钥出现在倒数第一轮和倒数第二轮时, 则

$$\begin{aligned} Y &= AC_L(PB(SC_L(AC_L(PB(SC_L(C_{77}\oplus RK_{77})))\oplus RK_{78}\oplus \Delta in)\oplus \Delta out))\oplus RK_{79} \\ &= AC_L(PB(SC_L(AC_L(PB(SC_L(C_{77}\oplus RK_{77})))\oplus RK_{78}\oplus \Delta in)))\oplus (RK_{79}\oplus PB(\Delta out)). \end{aligned} \tag{6}$$

情况二. 若等价轮密钥出现在倒数第二轮和倒数第三轮时, 则

$$\begin{aligned} Y &= AC_L(PB(SC_L(AC_L(PB(SC_L(C_{77}\oplus RK_{77}\oplus \Delta in)\oplus \Delta out))\oplus RK_{78})))\oplus RK_{79} \\ &= AC_L(PB(SC_L(AC_L(PB(SC_L(C_{77}\oplus RK_{77}\oplus \Delta in)))\oplus (RK_{78}\oplus PB(\Delta out)))))\oplus RK_{79}. \end{aligned} \tag{7}$$

若利用特殊差分输入输出求解方程, 会得到 7 组密钥解, 如表 5 所示. 对于全部的 32 个 S 盒, 等价密钥数量会增加到 $7^{32}\approx 2^{89.84}$ 个. 在对 DEFAULT 密码进行统计故障分析时, 只能得到一个包含大量等价密钥的候选空间, 无法找到正确的原始密钥, 也难以对候选密钥空间进行穷尽搜索.

鉴于多轮密钥之间存在变换关系, 譬如, 第 $r$ 轮密钥经密钥编排算法可以推导得到第 $r$+1 轮密钥, 其中 $r\in[0,79]$. 攻击者可以先利用统计故障分析恢复全部的等价密钥集合, 再依据密钥编排算法中不同轮密钥的关系建立代数方程, 利用代数关系分析并筛选多轮密钥, 从而缩小候选密钥空间.

分析过程由统计故障分析获取全部等价密钥和利用代数分析筛选密钥两部分组成. 若故障导入轮数位置较浅, 在实际实施时易被硬件防护, 且涉及密钥较少, 代数方程相应数量变少, 难以进行有效筛选. 但是, 若故障位置导入较深, 则故障路径较长, 统计故障分析的复杂度将被大幅提升. 本文采用故障注入的位置为算法倒数第三轮的比特置换之后, 达到代数方程求解和计算攻击复杂度之间的平衡.

### 3.3 攻击步骤

攻击过程包括有以下五个步骤:

步骤一. 攻击者利用搭载 DEFAULT 密码的设备使用同一密钥进行加密, 并在加密运算的倒数第三轮注入故障, 收集被故障影响后的密文, 故障注入后的扩散路径如图 3 所示.

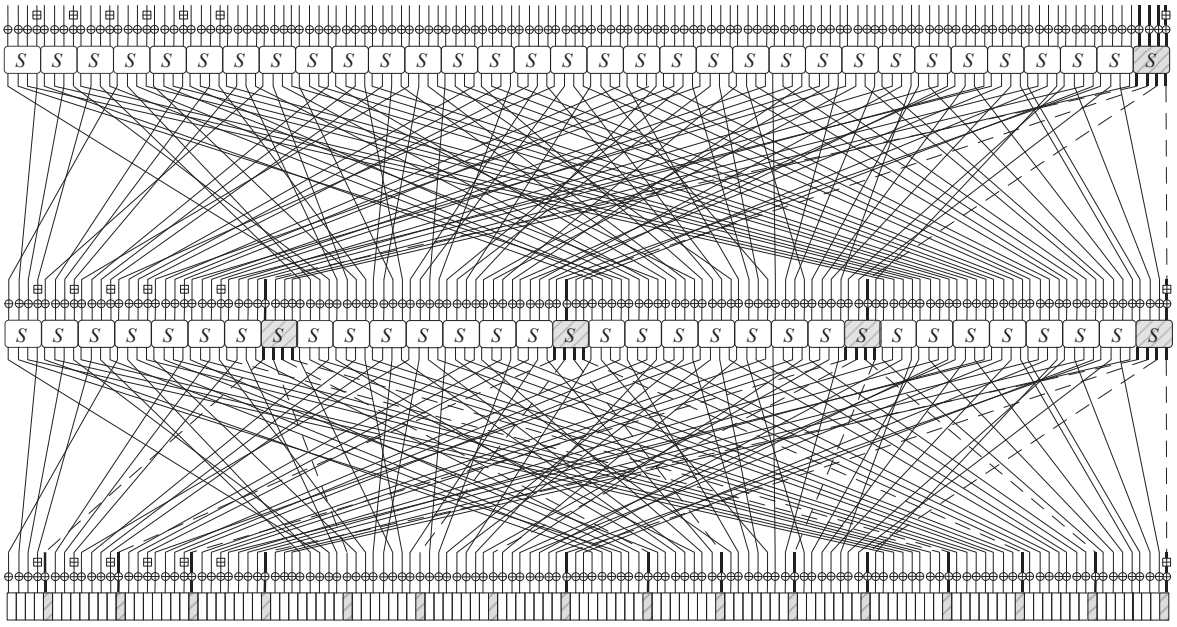

图 3 DEFAULT 密码最后两轮加密时的半字节故障扩散路径图

步骤二. 攻击者获得导入故障后输出的故障密文后, 结合最后三轮的轮密钥, 可以推导倒数第三轮比特置换运算的输出 $B_{77}$. 其中, 每次枚举参与计算的最后三轮部分密钥有 24 比特, 包括 $RK_{79}$ 的 16 比特、$RK_{78}$ 的 4 比特和 $RK_{77}$ 的 4 比特. 表达式为:

$$\begin{aligned} B_{77} &= AC_L^{-1}(AK^{-1}(D_{77})) \\ &= AC_L^{-1}(AK^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(AK^{-1}(D_{78})))))) \\ &= AC_L^{-1}(AK^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(AK^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(AK^{-1}(Y)))))))))), \end{aligned} \tag{8}$$

步骤三. 攻击者选择 4.4 节中合适的区分器, 对于每个中间状态值, 均可获得其对应的区分器值, 利用该区分

器的最值求出所对应的候选密钥, 即为正确密钥. 注入故障位置与可恢复密钥位置对应关系如图 4 所示, 若在 $B_{77}$ 的第 0、1、2、3 个半字节注入故障, 均可恢复最后一轮密钥相同位置的 16 比特. 攻击者通过对最后一轮密钥取交集可以减少候选密钥数量.

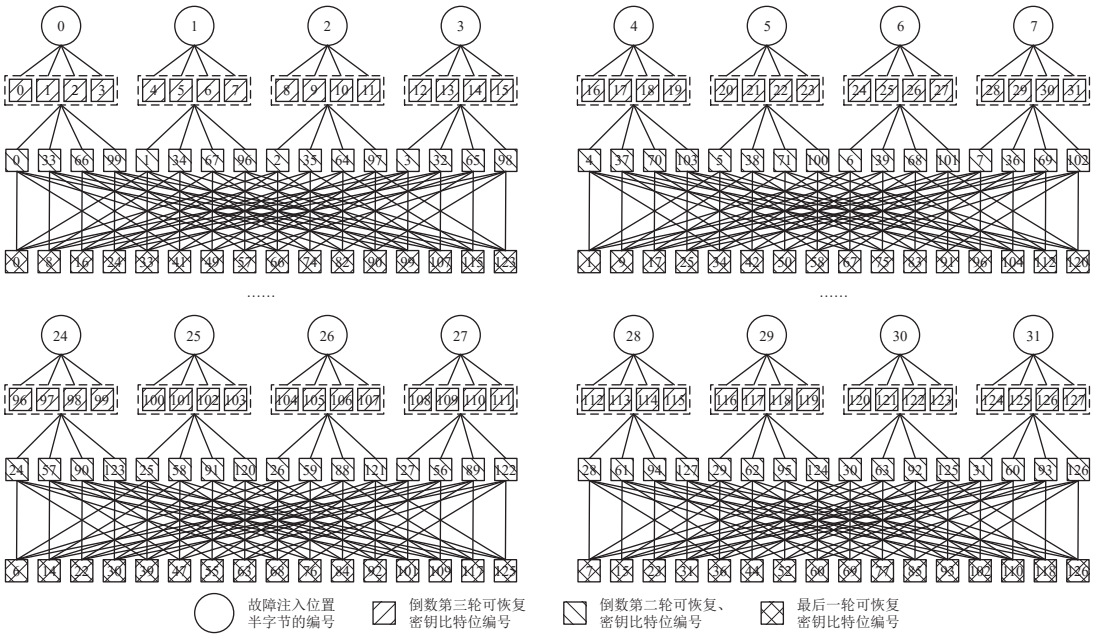


图 4　注入故障位置与可恢复密钥位置对应关系

步骤四. 攻击者多次在 $B_{77}$ 进行随机半字节故障注入, 通过重复步骤一到三, 每次故障均可以获得 $RK_{79}$ 的 16 比特、 $RK_{78}$ 的 4 比特和 $RK_{77}$ 的 4 比特. 根据图 4 所示的每次故障可恢复最后三轮密钥比特位的关系, 组合每次获取的最后三轮密钥比特信息, 逐步获取 $RK_{77}$ 、 $RK_{78}$ 和 $RK_{79}$ 轮密钥的全部 128 比特等价密钥组.

步骤五. 攻击者建立代数方程, 并对等价密钥集合进行代数关系分析. DEFAULT 密码不同轮所使用的轮密钥之间存在着变换关系, 由密钥循环关系可知, 最后三轮所使用的密钥分别为 $RK_1$ 、 $RK_2$ 和 $RK_3$ , 由算法 2 中的密钥编排算法可知, 三个轮密钥间存在关系:

$$RK_2 = AC_L(PB(SC_L(AC_L(PB(SC_L(AC_L(PB(SC_L(AC_L(PB(SC_L(RK_1)))))))))))), \tag{9}$$

$$RK_3 = AC_L(PB(SC_L(AC_L(PB(SC_L(AC_L(PB(SC_L(AC_L(PB(SC_L(RK_2)))))))))))), \tag{10}$$

密钥编排算法总共进行 4 轮 S 盒替换、比特置换和轮常数加运算. 为了加快求解速度, 攻击者根据不同操作内的代数关系, 将密钥编排算法第 $t$+1 轮的 S 盒替换、比特置换和轮密钥加运算分别表示为如下代数方程:

(1) S 盒替换

根据 S 盒内部关系将 S 盒替换运算表示为下列代数方程:

$$\begin{cases} b_t^{4j} = a_t^{4j}a_t^{4j+1} \oplus a_t^{4j}a_t^{4j+2} \oplus a_t^{4j} \oplus a_t^{4j+1}a_t^{4j+3} \oplus a_t^{4j+1} \oplus a_t^{4j+2}a_t^{4j+3} \\ b_t^{4j+1} = a_t^{4j} \oplus a_t^{4j+1} \oplus a_t^{4j+2} \\ b_t^{4j+2} = a_t^{4j}a_t^{4j+1} \oplus a_t^{4j}a_t^{4j+2} \oplus a_t^{4j+2} \oplus a_t^{4j+1}a_t^{4j+3} \oplus a_t^{4j+3} \oplus a_t^{4j+2}a_t^{4j+3} \\ b_t^{4j+3} = a_t^{4j+1} \oplus a_t^{4j+2} \oplus a_t^{4j+3} \end{cases}, \tag{11}$$

其中, $a_t^{4j+k}$ 和 $b_t^{4j+k}$ 分别表示密钥编排算法中第 $t$+1 轮 S 盒替换输入状态和输出状态中第 $j$ 个 S 盒的第 $k$ 比特, 并且 $t \in [0, 3], j \in [0, 31], k \in [0, 3]$.

(2) 比特置换

根据置换矩阵的特性建立如下代数方程:

$$b_t^{32((i+4-i/4) \bmod 4)+i \bmod 4+4(i/16)} \oplus c_t^i = 0, \tag{12}$$

其中, $b_t^i$ 和 $c_t^i$ 分别表示密钥编排算法中第 $t$+1 轮比特置换输入状态和输出状态的第 $i$ 比特, 并且 $t \in [0,3]$, $i \in [0,127]$.

(3) 轮常数加

密钥编排算法的轮常数加运算仅在密钥第 127 位异或“1”, 则轮常数加的代数方程为:

$$c_t^{127} \oplus d_t^{127} \oplus 1 = 0, \tag{13}$$

其中, $c_t^{127}$ 和 $d_t^{127}$ 分别表示密钥编排算法中第 $t$+1 轮常数加输入状态和输出状态的第 127 比特, 并且 $t \in [0,3]$.

只有正确的轮密钥 $(RK_1, RK_2, RK_3)$ 才会满足上述方程组, 当代数方程建立后, 每组等价密钥可代入方程并判断方程是否成立, 从而对等价密钥进行筛选, 利用不断缩小的密钥候选空间, 筛选出正确轮密钥 $(RK_1, RK_2, RK_3)$. 最后, 攻击者通过密钥编排算法, 恢复原始密钥 $K$ 为:

$$K = SC_L^{-1}(PB^{-1}(AC_L^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(SC_L^{-1}(PB^{-1}(AC_L^{-1}(RK_1)))))))))))). \tag{14}$$

### 3.4 区分器

本文使用了平方欧式距离、极大似然估计、汉明重量等经典区分器, 并且基于安德森达林拟合度检验 (Anderson Darling test), 提出了三种新型组合区分器 AD 检验—平方欧式距离、AD 检验—极大似然估计和 AD 检验—汉明重量.

#### 3.4.1 经典区分器

(1) 平方欧式距离 (Square Euclidean imbalance, SEI)

欧式距离又称欧几里得距离, 用于描述空间中两点的距离, 其平方值为平方欧式距离, 由数学家 Euclid 提出. 2013 年, Fuhr 等学者在 AES 密码的唯密文故障分析中首次使用该区分器[18]. SEI 区分器表达式为:

$$\mathrm{SEI} = \sum_{\varphi=0}^{w-1} \left( \frac{n(\varphi)}{m} - \frac{1}{w} \right)^2, \tag{15}$$

其中, $\varphi$ 为故障注入后所有可能的中间状态值, $w$ 表示半字节所有可能取值的个数, $m$ 为注入故障数量, $n(\varphi)$ 表示中间状态值为 $\varphi$ 的个数. SEI 区分器用于计算实际统计的样本值分布与理论均匀分布之间的距离, 在故障注入后, 实际中间状态的分布会偏离均匀分布, 所以正确密钥对应的 SEI 统计量值应为最大值.

(2) 极大似然估计 (Maximum likelihood estimate, MLE)

极大似然估计由 Gauss 等数学家于 1821 年提出, 并最早被 Fuhr 等学者应用于唯密文故障分析中[18]. 该区分器通过似然函数计算每一组样本值理论应该出现的概率. MLE 区分器表示为:

$$\mathrm{MLE} = \prod_{\lambda=1}^{m} p(\varepsilon_\lambda), \tag{16}$$

其中 $m$ 为注入故障数量, $\varepsilon_\lambda$ 表示第 $\lambda$ 个故障推导出来的中间状态值, $p(\varepsilon_\lambda)$ 表示中间状态值为 $\varepsilon_\lambda$ 的理论概率. 当 MLE 取最大值时, 表示使用该密钥倒推回的中间状态样本值的出现概率最大, 所以最大值情况下对应的候选密钥即为正确密钥.

(3) 汉明重量 (Hamming weight, HW)

汉明重量由 Reed 学者于 1954 年提出, 用于计算二进制字符串中‘1’的个数[42]. 2013 年, Fuhr 等学者将其应用于 AES 算法的统计故障分析中[18]. 正常情况下, 中间状态值中‘0’和‘1’是均匀分布的, 故障通过按位“与”的方式注入后, 会导致中间状态值中‘0’的个数比‘1’的个数多, 所以由正确密钥推导的中间状态值中‘1’的个数较少, 则正确密钥对应的汉明重量值应为最小值. HW 表达式为:

$$\mathrm{HW} = \sum_{\lambda=1}^{m} hw(\varepsilon_\lambda), \tag{17}$$

其中, $m$ 为注入故障数量, $\varepsilon_\lambda$ 表示由第 $\lambda$ 个故障推导出来的中间状态值, $hw(\varepsilon_\lambda)$ 表示 $\varepsilon_\lambda$ 的汉明重量.

3.4.2 新型区分器

AD 检验 (Anderson Darling test) 是由 Anderson 和 Darling 于 1954 年提出的一种非参数检验方法, 通常用于评估数据集与特定分布之间的拟合程度[43]. 对于给定的数据集和分布, 如果分布与数据的拟合越好, AD 检验就将产生较小的统计量. 该方法被广泛应用于统计学、可靠性检验以及异常检测等领域. AD 检验的表达式为:

$$\mathrm{AD}=\sum_{\varphi=0}^{w-1}[u(\varphi)-v(\varphi)]^2 p(\varphi), \tag{18}$$

其中 $w$ 表示半字节所有可能取值个数, $\varphi$ 为故障注入后所有可能中间状态值, $u(\varphi)$ 表示中间状态值小于 $\varphi$ 的实际概率, $v(\varphi)$ 表示中间状态值小于 $\varphi$ 的理论概率, $p(\varphi)$ 表示中间状态值等于 $\varphi$ 的理论概率.

AD 检验统计量用于衡量中间状态值与理论分布之间的拟合程度. 只有通过正确密钥推导得到的中间状态值的分布才能接近理论概率分布. 因此, 当使用 AD 检验时, 正确密钥即对应统计量最小值.

(1) AD 检验—平方欧式距离 (Anderson Darling test–Square Euclidean imbalance, AD–SEI)

结合该密码的设计, 本文提出 SEI 区分器的改进表达式, 计算样本偏离中间状态值理论分布的程度. 改进 SEI 的公式为:

$$\mathrm{SEI}=\sum_{\varphi=0}^{w-1}\left(\frac{n(\varphi)}{m}-p(\varphi)\right)^2, \tag{19}$$

其中, $p(\varphi)$ 表示中间状态值为 $\varphi$ 的理论概率. 当 SEI 取最小值时, 代表该密钥对应的中间状态值与理论分布契合程度最高, 所以 SEI 最小值对应的候选密钥为正确密钥. AD–SEI 区分器将 AD 检验和 SEI 区分器融合, 首先对得到的中间状态值计算其对应的 AD 检验统计量, 对密钥进行初步筛选, 然后对剩下的密钥计算其改进后的 SEI 统计量, 最小值对应的密钥即为正确密钥.

(2) AD 检验—极大似然估计 (Anderson Darling test–Maximum likelihood estimate, AD–MLE)

AD–MLE 区分器融合了 AD 检验和极大似然估计两者的特点. 首先, 对于实验获得的每组中间状态值, 攻击者计算相应的 AD 检验统计量, 并保留那些具有较小统计量值的密钥, 以进行密钥粗筛. 接下来, 对于剩余的候选密钥, 分别计算它们对应的极大似然估计统计量值, 最大值对应的候选密钥即为正确密钥. AD–MLE 区分器可以通过 AD 检验初步筛选候选密钥, 缩小极大似然估计区分器的筛选范围.

(3) AD 检验—汉明重量 (Anderson Darling test–Hamming weight, AD–HW)

AD–HW 区分器将 AD 检验和汉明重量区分器相结合, 先使用 AD 检验, 对每组候选密钥得到的中间状态值进行筛选, 保留较小 AD 检验统计量值对应的候选密钥, 再使用汉明重量区分器对得到的候选密钥进行筛选, 并选择最小的汉明重量值对应的密钥, 即为正确密钥. 表 6 总结了本文使用的区分器取值情况以及筛选过程说明.

表 6 不同区分器的取值和说明

| 区分器 | 取值范围 | 筛选过程 |
|---|---|---|
| SEI | 最大值 | 评估中间状态偏离平均分布的程度, 选择偏离平均程度最大的样本 |
| MLE | 最大值 | 评估中间状态的出现概率, 选择概率最大的统计样本 |
| HW | 最小值 | 评估中间状态的汉明重量, 选择汉明重量最小的统计样本 |
| AD–SEI | AD最小值<br>SEI最小值 | 先使用AD和SEI, 再选择最小AD值且中间状态偏离理论分布最小的统计样本 |
| AD–MLE | AD最小值<br>MLE最大值 | 先使用AD和MLE, 再选择最小AD值且中间状态出现概率最大的统计样本 |
| AD–HW | AD最小值<br>HW最小值 | 先使用AD和HW, 再选择出最小AD值且最小汉明重量的统计样本 |

## 4 实验分析

本文使用计算机模拟实现 DEFAULT 密码故障分析过程, 所使用的设备 CPU 为 Intel(R) Core(TM) i5–10400F,

并且编程语言为 JAVA 语言. 本文共进行了 10 000 次实验, 包括故障注入、中间状态推导、区分器筛选、等价密钥恢复和原始密钥恢复等过程. 当区分器达到最高成功率时, 该方法可以筛选出所有等价密钥, 且区分器得到的等价密钥值和数量均相同. 代数方程的求解时间和复杂度仅取决于等价密钥的数量, 因此在步骤 5 中代数方程求解所需的时间及时间复杂度均为 1 980.26 小时和 $2^{66.58}$ . 如下为获取全部等价密钥的各项指标, 具体如附录表 A1 和 A2 所示.

### 4.1 故障数

故障数是评估故障分析效果的主要标准, 指攻击者为了恢复密钥所需要使用的故障总数. 若所需故障数量越少, 攻击者越容易在物联网环境下实施攻击. 如图 5 所示, SEI 和 MLE 区分器在所有的区分器中, 随着故障数的增加, 只能够达到 6% 和 84% 的成功率. 如表 2 所示, 与经典区分器相比, 新型区分器 AD–SEI、AD–MLE 和 AD–HW 仅需 2048、1 408 和 1 344 个故障, 即可实现以 99% 的成功率筛选出 128 比特密钥. 在降低故障数方面, 新型区分器具有较大的优势, 其中, AD–HW 区分器所需故障数量最少.

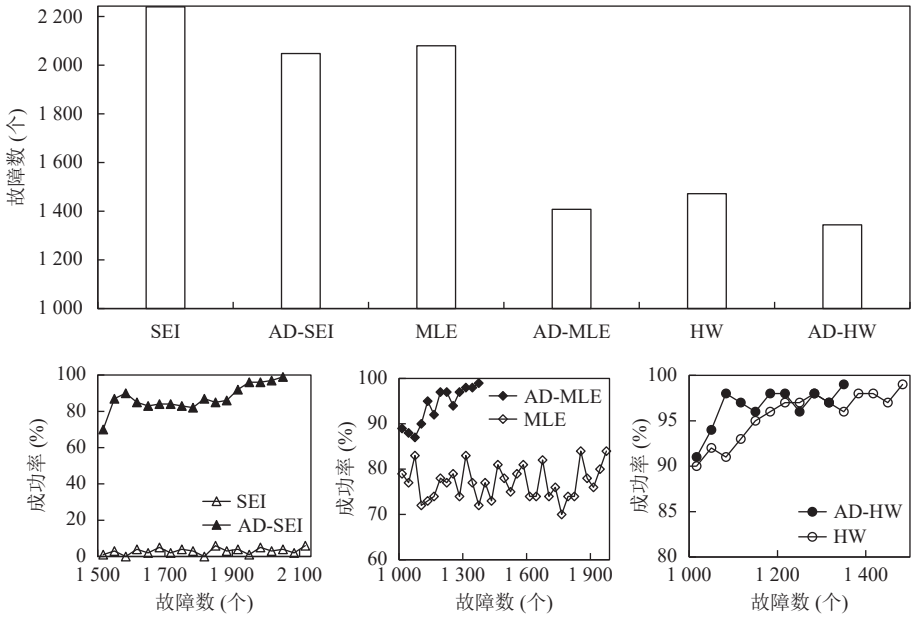


图 5 各区分器恢复原始密钥所需故障数及成功率

### 4.2 成功率

成功率是指在指定的故障数下成功破解密钥的概率. 图 6 为各个区分器恢复原始密钥的成功率对比, 其中横纵坐标分别为故障数和恢复密钥的成功率. 分析可知, SEI 区分器的成功率一直较低, 最高仅为 6%; 当故障数达到 2048 个, AD–SEI 区分器能实现 99% 的成功率. 同理, MLE 区分器的成功率一直在 80% 上下波动, 而 AD–MLE 通过结合 AD 检验, 当故障数为 1 408 个故障, 成功率达到 99% 及以上. 在所有区分器中, AD–HW 区分器在相同故障数下具有更好的表现, 能以 1 344 个故障实现 99% 及以上的成功率.

### 4.3 耗 时

耗时是指从注入故障到推导中间状态值并使用区分器对其分析, 进而获得所有等价密钥所需要的时间. 图 7 展示了在不同故障数下各区分器恢复 128 比特密钥所需的时间, 其中横纵坐标分别表示注入的总故障数和在指定故障数下各区分器所需的累计时间. 在 AD–SEI、AD–MLE 和 AD–HW 区分器以 99% 以上的成功率恢复 128 比特密钥的情况下, 耗时分别为 52.16 分钟、52.47 分钟和 48.38 分钟. 与经典区分器相比, 新型区分器的耗时均更少.

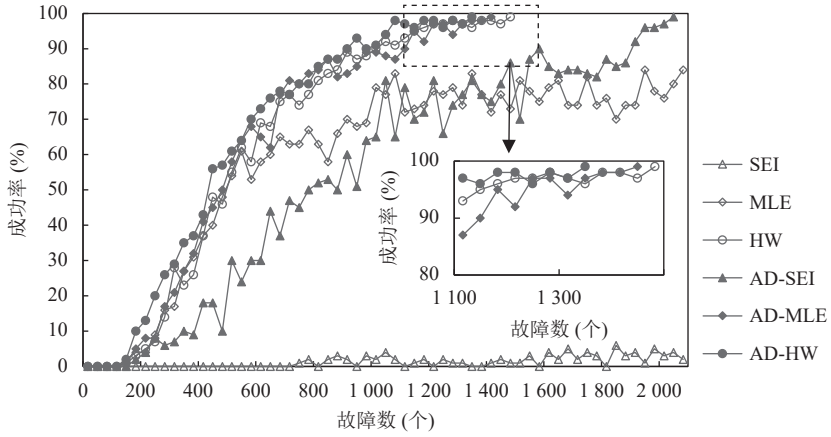


图 6　各区分器不同故障数下恢复原始密钥的成功率

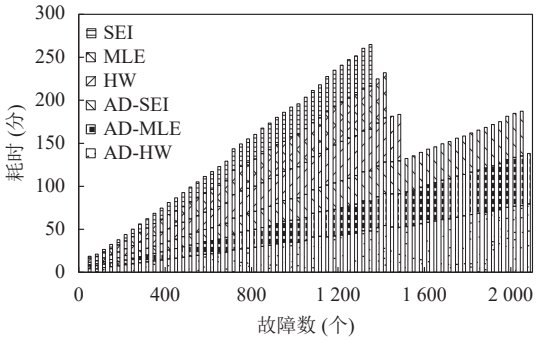


图 7　不同故障数下各区分器恢复原始密钥的耗时累加

### 4.4　复杂度

时间复杂度通常用加解密次数或内存访问次数来衡量, 即在密钥恢复过程中处理所有数据所需的时间资源. 数据复杂度指在恢复密钥过程中所需的数据资源. 计算公式如下:

$$mn+2^{u}m+2^{u}w, \tag{20}$$

$$(2^{u}+1)mv, \tag{21}$$

其中, $m$ 为区分器达到最大成功率时所需的总故障数, $n$ 为密码的变换轮数, $u$ 为实验需要枚举的密钥长度, $2^{u}m$ 为实验过程中对所有密文进行处理的次数, $w$ 表示半字节中间状态的可能取值个数, $2^{u}w$ 为区分器对中间状态进行分析的次数, $v$ 为密码的分组长度, $mv$ 表示获取受故障影响的密文的数据量. 各区分器的时间复杂度和数据复杂度如表 7 所示. 在所有的区分器中, 本文提出 AD–SEI、AD–HW 和 AD–MLE 区分器所需的时间复杂度和数据复杂度更低.

表 7　各区分器恢复 128 比特原始密钥的参数比较

| 区分器 | 成功率 | 故障数 | 时间复杂度 | 数据复杂度 |
|---|---|---|---|---|
| SEI : AD–SEI | 6 : 99 | ∞: 2048 | ∞: $2^{35.01}$ | ∞: $2^{42.00}$ |
| MLE : AD–MLE | 84 : 99 | 2 080 : 1 408 | $2^{34.92}$ : $2^{34.48}$ | $2^{41.91}$ : $2^{41.46}$ |
| HW : AD–HW | 99 : 99 | 1 472 : 1 408 | $2^{34.54}$ : $2^{34.41}$ | $2^{41.52}$ : $2^{41.39}$ |

从图 5–7 和表 7 的实验结果来看, 新型组合区分器 AD–SEI 和 AD–MLE 能提升 SEI 和 MLE 经典区分器的较低成功率, 达到 99% 及以上的成功率. 因此, 新型组合区分器能在较短的时间内以更少的故障数、更高的成功

率破解 DEFAULT 密码.

## 5 总 结

本文研究了 DEFAULT 密码的中间状态统计分布和多轮密钥之间的代数关系, 提出了基于代数关系的统计故障分析方法, 并结合 AD 检验, 讨论了新型组合区分器 AD–SEI、AD–MLE 和 AD–HW 的性能. 该研究不仅能够以 99% 及以上的成功率破译该 DEFAULT 密码, 而且能够降低破译密钥的故障数和复杂度, 减少破译时间. 研究结果表明, 基于代数关系的统计故障分析可以对 DEFAULT 密码的安全性产生威胁. 下一步的研究将结合该密码内部更深轮数进行安全分析.

**References:**


[1] Ahmed T, Samima S, Zuhair M, Ghayvat H, Khan MA, Kumar N. *FIMBISAE*: A multimodal biometric secured data access framework for Internet of Medical Things ecosystem. IEEE Internet of Things Journal, 2023, 10(7): 6259–6270. [doi: 10.1109/JIOT.2022.3225518]

[2] Wang CY, Wang D, Duan YH, Tao XF. Secure and lightweight user authentication scheme for cloud–assisted Internet of Things. IEEE Transactions on Information Forensics and Security, 2023, 18: 2961–2976. [doi: 10.1109/TIFS.2023.3272772]

[3] Omolara AE, Alabdulatif A, Abiodun OI, Alawida M, Alabdulatif A, Alshoura WH, Arshad H. The Internet of Things security: A survey encompassing unexplored areas and new insights. Computers & Security, 2022, 112: 102494. [doi: 10.1016/j.cose.2021.102494]

[4] Wang CY, Xie L, Zhao YC, Zhang DQ, Ye BL, Lu SL. Survey on RFID−based battery−less sensing. Journal of Software, 2022, 33(1): 297–323 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.006344]

[5] Li WT, Wang D, Wang P. Insider attacks against multi−factor authentication protocols for wireless sensor networks. Journal of Software, 2019, 30(8): 2375–2391 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.005766]

[6] Nagarajan SM, Deverajan GG, Kumaran U, Thirunavukkarasan M, Alshehri MD, Alkhalaf S. Secure data transmission in Internet of Medical Things using RES–256 algorithm. IEEE Transactions on Industrial Informatics, 2022, 18(12): 8876–8884. [doi: 10.1109/TII.2021.3126119]

[7] Fan Q, Chen JH, Shojafar M, Kumari S, He DB. SAKE*: A symmetric authenticated key exchange protocol with perfect forward secrecy for industrial Internet of Things. IEEE Transactions on Industrial Informatics, 2022, 18(9): 6424–6434. [doi: 10.1109/TII.2022.3145584]

[8] Saqib M, Moon AH. A systematic security assessment and review of Internet of Things in the context of authentication. Computers & Security, 2023, 125: 103053. [doi: 10.1016/j.cose.2022.103053]

[9] Song C, Zhang L, Wu WL. General subspace trail cryptanalysis of SPN ciphers. Journal of Software, 2023, 34(12): 5807–5821 (in Chinese with English abstract). http://www.jos.org.cn/201000-9825/6761.htm [ [doi: 10.13328/j.cnki.jos.006761]

[10] Kang BR, Zhang L, Zhang R, Meng XY, Chen T. Cryptographic algorithms against backdoored pseudorandom number generator. Journal of Software, 2021, 32(9): 2887–2900 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.005976]

[11] Yang F, Zhang QY, Shi ZP, Guan Y. Survey on software side–channel attacks in trusted execution environment. Journal of Software, 2023, 34(1): 381–403 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.006501]

[12] Wu WB, Liu Z, Yang H, Zhang JP. Survey of side–channel attacks and countermeasures on post–quantum cryptography. Journal of Software, 2021, 32(4): 1165–1185 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.006165]

[13] Tang BX, Wang LN, Wang R, Zhao L, Chen QS. General side channel defense schema of motion sensor based on Laplace mechanism. Journal of Software, 2019, 30(8): 2392–2414 (in Chinese with English abstract). [doi: 10.13328/j.cnki.jos.005760]

[14] Boneh D, DeMillo RA, Lipton RJ. On the importance of checking cryptographic protocols for faults. In: Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Konstanz: Springer, 1997. 37–51. [doi: 10.1007/3-540-69053-0_4]

[15] Ramzanipour H, Vafaei N, Bagheri N. Practical differential fault analysis on CRAFT, a lightweight block cipher. The ISC International Journal of Information Security, 2022, 14(3): 21–31. [doi: 10.22042/isecure.2022.14.3.3]

[16] Rivain M. Differential fault analysis on DES middle rounds. In: 11th Int'l Workshop on Cryptographic Hardware and Embedded Systems. Lausanne: Springer, 2009. 457–469. [doi: 10.1007/978-3-642-04138-9_32]

[17] Derbez P, Fouque PA, Leresteux D. Meet–in–the–middle and impossible differential fault analysis on AES. In: 13th Int'l Workshop on Cryptographic Hardware and Embedded Systems. Nara: Springer, 2011. 274–291. [doi: 10.1007/978-3-642-23951-9_19]

[18] Fuhr T, Jaulmes E, Lomné V, Thillard A. Fault attacks on AES with faulty ciphertexts only. In: 2013 Workshop on Fault Diagnosis and Tolerance in Cryptography. Los Alamitos: IEEE, 2013. 108–118. [doi: 10.1109/FDTC.2013.18]

[19] Li W, Li JY, Gu DW, Wang ML, Cai TP. Statistical fault analysis of the Piccolo lightweight cryptosystem. Chinese Journal of

Computers, 2021, 44(10): 2104–2121 (in Chinese with English abstract). [doi: 10.11897/SP.J.1016.2021.02104]

[20] Zhao XJ, Guo SZ, Wang T, Zhang F, Liu HY, Huang J, Wang P. Research of algebraic fault analysis on Piccolo. Chinese Journal of Computers, 2013, 36(4): 882–894 (in Chinese with English abstract). [doi: 10.3724/SP.J.1016.2013.00882]

[21] Chen H, Wang T, Zhang F, Zhao XJ, He W, Xu LM, Ma YF. Stealthy hardware trojan based algebraic fault analysis of HIGHT block cipher. Security and Communication Networks, 2017, 2017: 8051728. [doi: 10.1155/2017/8051728]

[22] Le DP, Yeo SL, Khoo K. Algebraic differential fault analysis on SIMON block cipher. IEEE Transactions on Computers, 2019, 68(11): 1561–1572. [doi: 10.1109/TC.2019.2926081]

[23] Li W, Liu C, Gu DW, Gao JN, Sun WQ. Statistical differential fault analysis of the Saturnin lightweight cryptosystem in the mobile wireless sensor networks. IEEE Transactions on Information Forensics and Security, 2023, 18: 1487–1496. [doi: 10.1109/TIFS.2023.3244083]

[24] Li W, Zhang YX, Gu DW, Zhang JY, Zhu XM, Liu C, Cai TP, Li JY. Ciphertext–only fault analysis on the MANTIS lightweight cipher. Acta Electonica Sinica, 2022, 50(4): 967–976 (in Chinese with English abstract). [doi: 10.12263/DZXB.20211026]

[25] Bagheri N, Sadeghi S, Ravi P, Bhasin S, Soleimany H. SIPFA: Statistical ineffective persistent faults analysis on Feistel ciphers. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2022, 2022(3): 367–390. [doi: 10.46586/tches.v2022.i3.367-390]

[26] Baksi A, Bhasin S, Breier J, Khairallah M, Peyrin T, Sarkar S, Sim SM. DEFAULT: Cipher level resistance against differential fault attack. IACR Cryptology ePrint Archive, 2021.

[27] Baksi A, Bhasin S, Breier J, Khairallah M, Peyrin T, Sarkar S, Sim SM. DEFAULT: Cipher level resistance against differential fault attack. In: 27th Int'l Conf. on the Theory and Application of Cryptology and Information Security. Singapore: Springer, 2021. 124–156. [doi: 10.1007/978-3-030-92075-3_5]

[28] Dey C, Pandey SK, Roy T, Sarkar S. Differential fault attack on DEFAULT. IACR Cryptology ePrint Archive, 2021.

[29] Nageler M, Dobraunig C, Eichlseder M. Information–combining differential fault attacks on DEFAULT. In: 41st Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Trondheim: Springer, 2022. 168–191. [doi: 10.1007/978-3-031-07082-2_7]

[30] Jang K, Baksi A, Breier J, Seo H, Chattopadhyay A. Quantum implementation and analysis of DEFAULT. IACR Cryptology ePrint Archive, 2022.

[31] Courtois NT, Meier W. Algebraic attacks on stream ciphers with linear feedback. In: Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Warsaw: Springer, 2003. 345–359. [doi: 10.1007/3-540-39200-9_21]

[32] Courtois NT, Bard GV. Algebraic cryptanalysis of the data encryption standard. In: 11th IMA Int'l Conf. on Cryptography and Coding. Cirencester: Springer, 2007. 152–169. [doi: 10.1007/978-3-540-77272-9_10]

[33] Courtois NT, O'Neil S, Quisquater JJ. Practical algebraic attacks on the Hitag2 stream cipher. In: Proc. of the 12th Int'l Conf. on Information Security. Pisa: Springer, 2009. 167–176. [doi: 10.1007/978-3-642-04474-8_14]

[34] Zhang F, Zhao XJ, Guo SZ, Wang T, Shi ZJ. Improved algebraic fault analysis: A case study on Piccolo and applications to other lightweight block ciphers. In: 4th Int'l Workshop on Constructive Side–Channel Analysis and Secure Design. Paris: Springer, 2013. 62–79. [doi: 10.1007/978-3-642-40026-1_5]

[35] Zhao XJ, Guo SJ, Zhang F, Wang T, Shi ZJ, Ma CJ, Gu DW. Algebraic fault analysis on GOST for key recovery and reverse engineering. In: 2014 Workshop on Fault Diagnosis and Tolerance in Cryptography. Busan: IEEE, 2014. 29–39. [doi: 10.1109/FDTC.2014.13]

[36] Gruber M, Karl P, Sigl G. Algebraic fault analysis of Subterranean 2.0. In: Workshop on Fault Detection and Tolerance in Cryptography. Milan: IEEE, 2021. 45–55. [doi: 10.1109/FDTC53659.2021.00016]

[37] Fang X, Zhang HX, Wang DZ, Yan H, Fan F, Shu L. Algebraic persistent fault analysis of SKINNY_64 based on S_box decomposition. Entropy, 2022, 24(11): 1508. [doi: 10.3390/e24111508]

[38] Fang X, Zhang HX, Cui XT, Wang YZ, Ding LX. Efficient attack scheme against SKINNY–64 based on algebraic fault analysis. Entropy, 2023, 25(6): 908. [doi: 10.3390/e25060908]

[39] Qiu Z, Zhang F, Feng TX, Gong X. RAFA: Redundancies–assisted algebraic fault analysis and its implementation on SPN block ciphers. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2023, 2023(3): 570–596. [doi: 10.46586/tches.v2023.i3.570-596]

[40] Nozaki Y, Yoshikawa M. Statistical fault analysis for a lightweight cipher Midori. In: 2017 IEEE Int'l Conf. on Information and Automation. Macau, China: IEEE, 2017. 236–241. [doi: 10.1109/ICInfA.2017.8078912]

[41] Ramezanpour K, Ampadu P, Diehl W. A statistical fault analysis methodology for the ASCON authenticated cipher. In: 2019 IEEE Int'l Symp. on Hardware Oriented Security and Trust. McLean: IEEE, 2019. 41–50. [doi: 10.1109/HST.2019.8741029]

[42] Reed I. A class of multiple-error-correcting codes and the decoding scheme. Transactions of the IRE Professional Group on Information Theory, 1954, 4(4): 38–49. [doi: 10.1109/TIT.1954.1057465]

[43] Anderson TW, Darling DA. A test of goodness of fit. Journal of the American Statistical Association, 1954, 49(268): 765–769. [doi: 10.1080/01621459.1954.10501232]

附中文参考文献:

[4] 王楚豫, 谢磊, 赵彦超, 张大庆, 叶保留, 陆桑璐. 基于 RFID 的无源感知机制研究综述. 软件学报, 2022, 33(1): 297–323. [doi: 10.13328/j.cnki.jos.006344]

[5] 李文婷, 汪定, 王平. 无线传感器网络下多因素身份认证协议的内部人员攻击. 软件学报, 2019, 30(8): 2375–2391. [doi: 10.13328/j.cnki.jos.005766]

[9] 宋蝉, 张蕾, 吴文玲. SPN 型密码的通用子空间迹分析. 软件学报, 2023, 34(12): 5807–5821. [doi: 10.13328/j.cnki.jos.006761]

[10] 康步荣, 张磊, 张蕊, 孟欣宇, 陈桐. 抗随机数后门攻击的密码算法. 软件学报, 2021, 32(9): 2887–2900. [doi: 10.13328/j.cnki.jos.005976]

[11] 杨帆, 张倩颖, 施智平, 关永. 可信执行环境软件侧信道攻击研究综述. 软件学报, 2023, 34(1): 381–403. [doi: 10.13328/j.cnki.jos.006501]

[12] 吴伟彬, 刘哲, 杨昊, 张吉鹏. 后量子密码算法的侧信道攻击与防御综述. 软件学报, 2021, 32(4): 1165–1185. [doi: 10.13328/j.cnki.jos.006165]

[13] 唐奔宵, 王丽娜, 汪润, 赵磊, 陈青松. 基于 Laplace 机制的普适运动传感器侧信道防御方案. 软件学报, 2019, 30(8): 2392–2414. [doi: 10.13328/j.cnki.jos.005760]

[19] 李玮, 李嘉耀, 谷大武, 汪梦林, 蔡天培. 轻量级密码算法 Piccolo 的统计故障分析. 计算机学报, 2021, 44(10): 2104–2121. [doi: 10.11897/SP.J.1016.2021.02104]

[20] 赵新杰, 郭世泽, 王韬, 张帆, 刘会英, 黄静, 王平. Piccolo 密码代数故障分析研究. 计算机学报, 2013, 36(4): 882–894. [doi: 10.3724/SP.J.1016.2013.00882]

[24] 李玮, 张雨希, 谷大武, 张金煜, 朱晓铭, 刘春, 蔡天培, 李嘉耀. 轻量级密码 MANTIS 的唯密文故障分析. 电子学报, 2022, 50(4): 967–976. [doi: 10.12263/DZXB.20211026]

## 附录 A. 实验数据

明文: 随机生成

密钥: 0x0123456789abcdef0123456789abcdef

表 A1 各区分器恢复 DEFAULT 密码 24 比特等价密钥的成功率 (%)

| 故障数 | SEI | AD–SEI | MLE | AD–MLE | HW | AD–HW |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 0 | 0 | 0 | 0 | 0 | 0 |
| 2 | 0 | 0 | 0 | 0 | 0 | 0 |
| 3 | 0 | 0 | 0 | 0 | 0 | 0 |
| 4 | 0 | 0 | 2 | 1 | 1 | 2 |
| 5 | 0 | 2 | 4 | 5 | 3 | 10 |
| 6 | 0 | 4 | 4 | 8 | 5 | 13 |
| 7 | 0 | 8 | 9 | 8 | 7 | 20 |
| 8 | 0 | 6 | 16 | 17 | 14 | 26 |
| 9 | 0 | 7 | 17 | 21 | 28 | 29 |
| 10 | 0 | 10 | 27 | 27 | 23 | 35 |
| 11 | 0 | 9 | 31 | 32 | 26 | 37 |
| 12 | 0 | 18 | 37 | 41 | 37 | 43 |
| 13 | 0 | 18 | 40 | 45 | 48 | 56 |
| 14 | 0 | 10 | 48 | 50 | 46 | 57 |
| 15 | 0 | 30 | 54 | 58 | 55 | 61 |
| 16 | 0 | 24 | 61 | 64 | 62 | 64 |
| 17 | 0 | 30 | 53 | 68 | 58 | 70 |

表 A1　各区分器恢复 DEFAULT 密码 24 比特等价密钥的成功率 (%)(续)

| 故障数 | SEI | AD–SEI | MLE | AD–MLE | HW | AD–HW |
|---|---|---|---|---|---|---|
| 18 | 0 | 30 | 58 | 65 | 69 | 73 |
| 19 | 0 | 44 | 60 | 62 | 68 | 76 |
| 20 | 0 | 37 | 65 | 77 | 75 | 78 |
| 21 | 0 | 47 | 63 | 81 | 77 | 77 |
| 22 | 1 | 45 | 63 | 80 | 74 | 80 |
| 23 | 2 | 50 | 67 | 83 | 77 | 80 |
| 24 | 0 | 52 | 63 | 84 | 81 | 85 |
| 25 | 2 | 53 | 58 | 87 | 83 | 87 |
| 26 | 3 | 50 | 66 | 82 | 84 | 87 |
| 27 | 2 | 60 | 70 | 83 | 89 | 90 |
| 28 | 0 | 51 | 68 | 85 | 87 | 93 |
| 29 | 3 | 64 | 69 | 89 | 88 | 90 |
| 30 | 1 | 63 | 71 | 91 | 89 | 92 |
| 31 | 2 | 65 | 79 | 89 | 90 | 91 |
| 32 | 4 | 81 | 77 | 88 | 92 | 94 |
| 33 | 2 | 65 | 83 | 87 | 91 | 98 |
| 34 | 0 | 79 | 72 | 90 | 93 | 97 |
| 35 | 0 | 67 | 73 | 96 | 96 | 97 |
| 36 | 1 | 70 | 73 | 95 | 95 | 96 |
| 37 | 2 | 72 | 74 | 92 | 96 | 98 |
| 38 | 0 | 81 | 78 | 97 | 97 | 98 |
| 39 | 2 | 66 | 77 | 97 | 97 | 96 |
| 40 | 1 | 74 | 79 | 94 | 98 | 98 |
| 41 | 1 | 77 | 74 | 97 | 97 | 97 |
| 42 | 0 | 81 | 83 | 98 | 96 | 99 |
| 43 | 0 | 77 | 77 | 98 | 98 | – |
| 44 | 1 | 75 | 72 | 99 | 98 | – |
| 45 | 2 | 80 | 77 | – | 97 | – |
| 46 | 1 | 86 | 73 | – | 99 | – |
| 47 | 1 | 70 | 81 | – | – | – |
| 48 | 3 | 87 | 78 | – | – | – |
| 49 | 0 | 90 | 75 | – | – | – |
| 50 | 4 | 85 | 79 | – | – | – |
| 51 | 2 | 83 | 81 | – | – | – |
| 52 | 5 | 84 | 74 | – | – | – |
| 53 | 2 | 84 | 74 | – | – | – |
| 54 | 4 | 83 | 82 | – | – | – |
| 55 | 3 | 82 | 74 | – | – | – |
| 56 | 0 | 87 | 76 | – | – | – |
| 57 | 6 | 85 | 70 | – | – | – |
| 58 | 3 | 86 | 74 | – | – | – |
| 59 | 4 | 92 | 74 | – | – | – |
| 60 | 1 | 96 | 84 | – | – | – |
| 61 | 2 | 94 | 78 | – | – | – |
| 62 | 5 | 96 | 78 | – | – | – |
| 63 | 3 | 97 | 76 | – | – | – |
| 64 | 4 | 99 | 80 | – | – | – |
| 65 | 2 | – | 84 | – | – | – |

表 A2　各区分器恢复 DEFAULT 密码 24 比特等价密钥的耗时 (秒)

| 故障数 | SEI | AD–SEI | MLE | AD–MLE | HW | AD–HW |
|---|---|---|---|---|---|---|
| 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| 1 | 6.17 | 2.52 | 8.07 | 9.75 | 4.75 | 4.32 |
| 2 | 6.97 | 3.96 | 7.63 | 9.01 | 6.44 | 6.84 |
| 3 | 7.66 | 5.34 | 10.15 | 10.88 | 8.39 | 9.02 |
| 4 | 8.61 | 6.84 | 12.33 | 13.12 | 10.44 | 11.73 |
| 5 | 9.10 | 8.35 | 14.31 | 15.25 | 12.51 | 13.91 |
| 6 | 10.27 | 9.87 | 16.59 | 17.02 | 14.59 | 16.96 |
| 7 | 12.01 | 11.38 | 18.62 | 19.13 | 16.67 | 19.76 |
| 8 | 13.60 | 12.90 | 20.78 | 21.24 | 18.71 | 20.21 |
| 9 | 14.81 | 14.46 | 23.00 | 23.50 | 20.95 | 22.34 |
| 10 | 16.36 | 15.97 | 24.72 | 26.21 | 23.01 | 24.46 |
| 11 | 17.93 | 17.49 | 26.83 | 28.41 | 25.08 | 26.86 |
| 12 | 19.78 | 19.01 | 29.10 | 30.66 | 27.23 | 29.00 |
| 13 | 21.09 | 20.56 | 30.62 | 32.17 | 29.47 | 31.42 |
| 14 | 22.53 | 22.11 | 33.20 | 34.26 | 31.25 | 33.56 |
| 15 | 24.10 | 23.68 | 35.34 | 36.82 | 33.30 | 35.72 |
| 16 | 25.80 | 25.22 | 36.87 | 39.34 | 35.31 | 38.11 |
| 17 | 27.19 | 26.73 | 39.44 | 41.41 | 37.46 | 40.34 |
| 18 | 28.73 | 28.25 | 40.30 | 43.74 | 39.65 | 42.71 |
| 19 | 30.19 | 29.79 | 42.10 | 46.11 | 41.53 | 44.88 |
| 20 | 31.79 | 31.30 | 44.17 | 48.30 | 43.90 | 47.17 |
| 21 | 33.32 | 32.75 | 46.28 | 50.35 | 47.12 | 49.45 |
| 22 | 34.86 | 34.31 | 48.72 | 52.87 | 51.50 | 51.70 |
| 23 | 36.39 | 35.84 | 51.02 | 53.73 | 53.88 | 53.98 |
| 24 | 38.00 | 37.41 | 52.47 | 56.32 | 55.32 | 56.53 |
| 25 | 39.46 | 38.89 | 54.49 | 59.03 | 55.09 | 59.12 |
| 26 | 41.09 | 40.45 | 56.58 | 63.31 | 57.27 | 61.09 |
| 27 | 42.64 | 42.03 | 58.85 | 63.29 | 60.60 | 63.78 |
| 28 | 44.29 | 43.67 | 62.07 | 66.15 | 61.65 | 66.26 |
| 29 | 45.87 | 45.27 | 65.39 | 65.40 | 63.55 | 69.66 |
| 30 | 47.52 | 46.77 | 65.95 | 69.44 | 65.54 | 71.51 |
| 31 | 48.93 | 48.30 | 65.08 | 69.81 | 67.69 | 74.25 |
| 32 | 50.69 | 49.83 | 69.15 | 73.76 | 69.80 | 75.63 |
| 33 | 51.95 | 51.41 | 76.82 | 74.06 | 72.04 | 77.23 |
| 34 | 53.58 | 53.00 | 78.44 | 77.41 | 74.21 | 79.60 |
| 35 | 55.07 | 54.52 | 76.02 | 79.72 | 76.15 | 81.96 |
| 36 | 56.66 | 56.08 | 77.37 | 82.18 | 77.90 | 84.31 |
| 37 | 58.09 | 57.60 | 79.25 | 84.28 | 82.76 | 86.58 |
| 38 | 59.72 | 59.17 | 81.12 | 85.33 | 85.50 | 88.85 |
| 39 | 61.29 | 60.71 | 83.09 | 87.37 | 87.52 | 92.11 |
| 40 | 63.02 | 62.29 | 85.56 | 89.16 | 86.98 | 93.63 |
| 41 | 64.76 | 63.81 | 90.79 | 91.73 | 88.92 | 97.21 |
| 42 | 65.06 | 65.51 | 90.39 | 93.96 | 91.23 | 99.78 |
| 43 | 67.16 | 67.01 | 97.41 | 96.23 | 93.89 | – |
| 44 | 69.69 | 68.52 | 102.10 | 98.38 | 96.60 | – |
| 45 | 72.06 | 70.07 | 99.05 | – | 99.02 | – |
| 46 | 73.52 | 71.63 | 97.91 | – | 101.19 | – |
| 47 | 75.68 | 73.05 | 99.59 | – | – | – |
| 48 | 77.49 | 74.63 | 101.80 | – | – | – |
| 49 | 79.39 | 76.15 | 106.00 | – | – | – |

表 A2　各区分器恢复 DEFAULT 密码 24 比特等价密钥的耗时 (秒)(续)

| 故障数 | SEI | AD–SEI | MLE | AD–MLE | HW | AD–HW |
|---|---|---|---|---|---|---|
| 50 | 81.62 | 77.72 | 109.06 | – | – | – |
| 51 | 83.79 | 78.11 | 111.79 | – | – | – |
| 52 | 85.89 | 79.53 | 114.88 | – | – | – |
| 53 | 87.62 | 80.99 | 116.51 | – | – | – |
| 54 | 89.81 | 82.57 | 120.04 | – | – | – |
| 55 | 90.04 | 84.03 | 123.24 | – | – | – |
| 56 | 92.36 | 85.60 | 125.87 | – | – | – |
| 57 | 94.07 | 87.22 | 128.66 | – | – | – |
| 58 | 96.24 | 88.61 | 131.04 | – | – | – |
| 59 | 98.27 | 90.08 | 133.74 | – | – | – |
| 60 | 100.41 | 91.65 | 136.50 | – | – | – |
| 61 | 102.17 | 93.22 | 138.79 | – | – | – |
| 62 | 104.21 | 94.70 | 141.45 | – | – | – |
| 63 | 106.33 | 96.19 | 144.09 | – | – | – |
| 64 | 108.08 | 97.80 | 145.54 | – | – | – |
| 65 | 110.16 | – | 149.16 | – | – | – |

**李玮**(1980－), 女, 博士, 教授, 博士生导师, CCF 会员, 主要研究领域为对称密码算法的设计与分析.

**连晟**(2002－), 女, 硕士生, 主要研究领域为分组密码的故障分析.

**秦梦洋**(2000－), 男, 硕士生, 主要研究领域为分组密码的唯密文故障分析.

**温云华**(1990－), 女, 博士, 硕士生导师, 主要研究领域为密码学.

**谷大武**(1970－), 男, 博士, 教授, 博士生导师, 主要研究领域为密码学与计算机安全.