PRIDE 轻量级密码的不可能统计故障分析

李玮[1,2,3,4]，孙文倩[1]，谷大武[2]，张爱琳[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. 上海市可扩展计算与系统重点实验室，上海 200240；4. 上海市信息安全综合管理技术研究重点实验室，上海 200240）

摘 要：针对 2014 年美密会上提出的 PRIDE 轻量级密码的实现安全，提出了面向唯密文攻击假设的新型不可能统计故障分析方法，设计了卡方拟合优度−汉明重量区分器、卡方拟合优度−极大似然估计区分器等新型区分器。所提方法基于随机半字节故障模型，结合统计分布状态和不可能关系分析，围绕导入故障前后中间状态的变化，最少仅需 432 个故障即可恢复出 PRIDE 算法的 128 bit 原始密钥，且成功率达 99%及以上。实验分析表明，所提方法不仅能减少故障数和耗时，而且进一步提升了准确率。该结果对轻量级密码的实现安全性提供了重要参考。

关键词：侧信道分析；不可能统计故障分析；轻量级密码；PRIDE；智能无人系统

中图分类号：TP309.7

文献标志码：A

DOI: 10.11959/j.issn.1000−436x.2024019

Impossible statistical fault analysis of the PRIDE lightweight cryptosystem

LI Wei[1,2,3,4], SUN Wenqian[1], GU Dawu[2], ZHANG Ailin[1], WEN Yunhua[1]

1. School of Computer Science and Technology, Donghua University, Shanghai 201620, China
2. Department of Computer Science and Engineering, Shanghai Jiao Tong University, Shanghai 200240, China
3. Shanghai Key Laboratory of Scalable Computing and System, Shanghai 200240, China
4. Shanghai Key Laboratory of Integrate Administration Technologies for Information Security, Shanghai 200240, China

Abstract: To analyze the implementation security of the PRIDE lightweight cryptosystem proposed at CRYPTO in 2014, a novel method of impossible statistical fault analysis on the ciphertext-only attack assumption was proposed. Furthermore, new distinguishers were designed, such as the Chi-square goodness-of-fit test-Hamming weight, and Chi-square goodness-of-fit test-maximum likelihood estimation. The proposed method had a random nibble-oriented fault model, and combined the statistical distribution states with the impossible relationship. On the difference among the intermediate states before and after the fault injections, at least 432 faults were required to recover the 128 bit secret key of PRIDE with a reliability of at least 99%. The experimental analysis demonstrates that the proposed method can not only reduce injected faults and latency, but also increase the accuracy. The results provide a vital reference for exploring the implementation security of lightweight cryptosystems.

Keywords: side-channel analysis, impossible statistical fault analysis, lightweight cryptosystem, PRIDE, intelligent unmanned system

---

收稿日期：2023−08−29；**修回日期：**2023−12−08

基金项目：国家自然科学基金资助项目（No.61772129，No.62172395，No.62102077）；国家密码发展基金资助项目（No.MMJJ20180101）；信息安全国家重点实验室开放课题基金资助项目（No.2021-MS-05）；上海市扬帆计划基金资助项目（No.21YF1401200, No.23YF1401000）；中央高校基本科研业务费专项资金资助项目（No.223202D-25）

Foundation Items: The National Natural Science Foundation of China (No.61772129, No.62172395, No.62102077), The National Cryptography Development Fund (No.MMJJ20180101), State Key Laboratory of Information Security (No.2021-MS-05), Shanghai Sailing Plan (No.21YF1401200, No.23YF1401000), The Fundamental Research Funds for the Central Universities (No.223202D-25)

0 引言

近年来，无人机、无人车和工业机器人等新型智能无人系统已被广泛应用于交通运输、公共安全和工业互联网等领域，如图 1 所示[1-4]。然而，由于系统的开放性以及自主性，这些智能无人系统面临侧信道攻击、网络窃听以及拒绝服务攻击等多种安全威胁。传统的密码算法在进行信息加解密时，需要消耗较多的资源，无法满足这些新型设备高安全性、高效率和低功耗的需求。因此，轻量级密码算法应运而生，旨在保证信息的保密性、完整性和可认证性，保护智能无人系统的通信安全。随着国际轻量级密码标准的相关工作开展和遴选，轻量级密码已成为国际密码学领域的研究热点[5-8]。

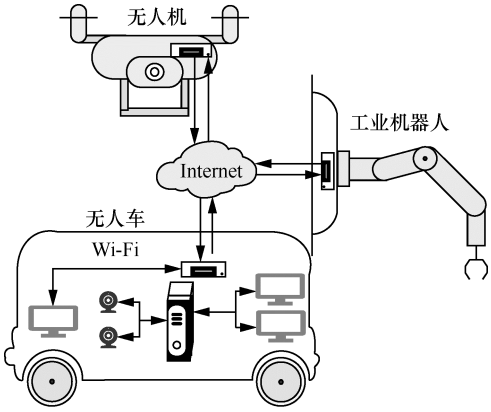


图 1 智能无人系统

2014 年，Albrecht 等[9]在美密会上提出了 PRIDE 轻量级密码算法，旨在面向 8 bit 微处理器的轻量化实现，并具有高效率、低功耗和快速响应等特点。该算法整体框架采用 FX 结构，密钥长度为 128 bit。目前，PRIDE 算法可以抵抗线性分析、差分分析、基于深度学习差分分析和高阶差分分析等多种经典密码分析[10-13]，最高攻击轮数为 19 轮。表 1 总结了 PRIDE 算法的安全性分析。

故障分析作为一种侧信道攻击方法，已成为评估密码算法实现安全性的重要指标之一[14]。分析测评人员利用特定的干扰手段，引起密码算法运行产生故障，得到正确/错误密文对，进而通过数学分析逐步推断出密钥，从而破译密码算法。常见的故障分析包括有差分故障分析、不可能差分故障分析、统计故障分析、无效故障分析、代数故障分析、中间相遇故障分析和持久故障分析等[15-23]。其中，常见的差分类故障分析通过比较密码算法产生的正确密文与错误密文之间的不同来破译密码[11]；统计故障分析是结合随机错误密文，并利用中间状态的概率统计，从而获取密钥[15]。攻击者根据自身的能力强弱选择合适的故障分析方法来实现分析。

基本假设作为密码分析的重要前提，用于表明攻击者能力的强弱以及实现的难易程度。目前，常见的基本假设包括已知明文攻击、选择明文攻击、已知密文攻击以及唯密文攻击等。例如，传统的线性分析、差分分析以及大多数故障分析的基本假设均为已知明文攻击或选择明文攻击。攻击者能够已知或选择明密文对[10-13]，而统计故障分析基于唯密文攻击，对攻击者的能力要求最弱，仅需截获随机密文，在面向诸如智能无人系统时更易实现[15]。

目前，国内外尚未有 PRIDE 轻量级密码抵御唯密文攻击的公开结果。本文基于唯密文攻击的基本假设，面向随机半字节故障模型，在统计故障分析的基础上，提出了新型的不可能统计故障分析方法，并使用汉明重量（HW, Hamming weight）、极大似然估计（MLE, maximum likelihood estimate）等经典

表 1 PRIDE 算法的安全性分析

分析类型	基本假设	最高攻击轮数/轮	文献
线性分析	已知明文攻击	18	文献[10]
差分分析	选择明文攻击	18	文献[11]
基于深度学习差分分析	选择明文攻击	18	文献[12]
高阶差分分析	选择明文攻击	19	文献[13]
统计故障分析	唯密文攻击	20	本文
不可能统计故障分析	唯密文攻击	20	本文

区分器，以及卡方拟合优度−汉明重量（CS-HW, Chi-square goodness-of-fit test-Hamming weight）和卡方拟合优度−极大似然估计（CS-MLE, Chi-square goodness-of-fit test-maximum likelihood estimate）等新型区分器，恢复 PRIDE 密码的原始密钥。本文提出的新型不可能统计故障分析方法以及区分器在故障数、耗时、精确度和复杂度等方面均具有较大优势，且成功率为 99%及以上。

1　PRIDE 算法简介

1.1　符号说明

设 $\boldsymbol{Z}_2^e$ 为 e bit 的二进制向量集；$\boldsymbol{M}\in(\boldsymbol{Z}_2^4)^{16}$ 和 $\boldsymbol{C}\in(\boldsymbol{Z}_2^4)^{16}$ 分别为明文和密文，$\hat{\boldsymbol{C}}\in(\boldsymbol{Z}_2^4)^{16}$ 为错误密文；$\boldsymbol{K}=\boldsymbol{K}_0\|\boldsymbol{K}_1\in(\boldsymbol{Z}_2^4)^{32}$ 为原始密钥，$\boldsymbol{K}_0\in(\boldsymbol{Z}_2^4)^{16}$ 为白化密钥，$\boldsymbol{K}_1\in(\boldsymbol{Z}_2^4)^{16}$ 为轮密钥；$\mathbf{rk}_i\in(\boldsymbol{Z}_2^4)^{16}$ 为第 i 轮的轮子密钥，其中，$1\leqslant i\leqslant 20$；Sub、Pub 和 Lin 分别为非线性替换、比特置换和线性变换，Sub^{-1} 和 Pub^{-1} 为 Sub 和 Pub 的逆运算；$\boldsymbol{T}\in(\boldsymbol{Z}_2^4)^{16}$ 为中间状态值，$\boldsymbol{L}_0$、$\boldsymbol{L}_1$、$\boldsymbol{L}_2$ 和 $\boldsymbol{L}_3$ 为线性变换中的不同 16 阶方阵；⊕、‖、mod 和 ⫲ 分别为按比特异或、连接、模运算和取绝对值。

1.2　PRIDE 算法

PRIDE 算法于 2014 年由 Albrecht 等[9]在美密会上提出，整体框架采用 FX 结构，如图 2 所示。该算法的设计采用了 64 bit 的分组长度和 128 bit 的密钥长度。算法迭代轮数为 20 轮，如图 3 所示。前 19 轮为 R 变换，每轮包括轮密钥加、非线性替换、比特置换、线性变换和比特逆置换；最后 1 轮为 R' 变换，由轮密钥加和非线性替换组成，具体介绍如下。

1) 轮密钥加（AddK, AddRoundKey）：64 bit 轮密钥按比特异或到中间状态值中。

2) 非线性替换（Sub, SubCells）：共有 16 个 S 盒，每个 S 盒完成 4 bit 替换。

3) 比特置换（Pub, PermutationBits）：按比特在 0~63 进行位置移动。

4) 线性变换（Lin, LinearLayer）：共有 4 个 16 bit 向量，分别与 16 阶方阵 $\boldsymbol{L}_0$、$\boldsymbol{L}_1$、$\boldsymbol{L}_2$ 和 $\boldsymbol{L}_3$ 相乘。

5) 比特逆置换（Pub^{-1}, InversePermutation-Bits）：比特置换的逆运算。

1.2.1　加解密过程

PRIDE 密码的加密过程如算法 1 所示，其中，KeySchedule(K) 表示密钥编排方案。解密过程是加密的逆过程。

算法 1　PRIDE 密码的加密过程

输入　$\boldsymbol{M},\boldsymbol{K}$

输出　$\boldsymbol{C}$

1) $\boldsymbol{K}_0\|\boldsymbol{K}_1=\boldsymbol{K}$;

2) $\boldsymbol{T}=\text{Pub}^{-1}(\boldsymbol{M})\oplus\boldsymbol{K}_0$;

3) $(\mathbf{rk}_1,\mathbf{rk}_2,\cdots,\mathbf{rk}_{19},\mathbf{rk}_{20})=\text{KeySchedule}(\boldsymbol{K})$;

4) for $i=1$ to 20 do

5)　if $i\leqslant 19$ do

6)　　$\boldsymbol{T}=\text{Pub}^{-1}(\text{Lin}(\text{Pub}(\text{Sub}(\boldsymbol{T}\oplus\text{Pub}^{-1}(\mathbf{rk}_i)))))$;

7)　else

8)　　$\boldsymbol{T}=\text{Sub}(\boldsymbol{T}\oplus\text{Pub}^{-1}(\mathbf{rk}_{20}))$;

9)　end if

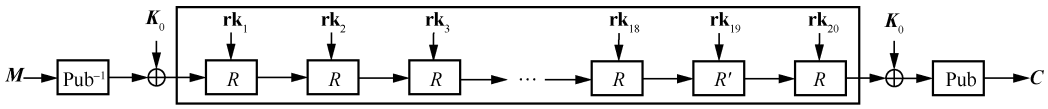


图 2　PRIDE 算法的整体框架

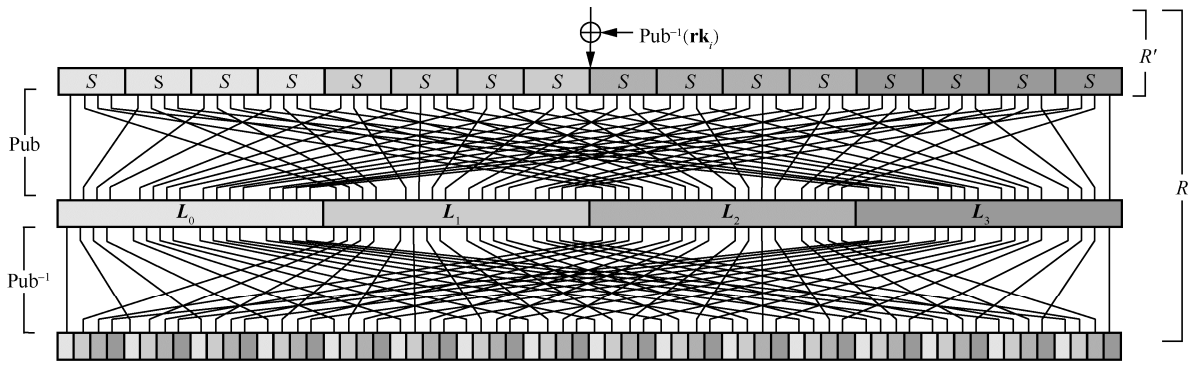


图 3　PRIDE 算法的轮变换

10) end for

11) $\boldsymbol{C} = \mathrm{Pub}(\boldsymbol{T} \oplus \boldsymbol{K}_0)$;

12) return $\boldsymbol{C}$

1.2.2 密钥编排方案

原始密钥 $\boldsymbol{K}$ 均分为白化密钥 $\boldsymbol{K}_0$ 和轮密钥 $\boldsymbol{K}_1$，具体表示为

$$\boldsymbol{K} = \boldsymbol{K}_0 \| \boldsymbol{K}_1 = (\boldsymbol{k}_0 \| \boldsymbol{k}_1 \| \cdots \| \boldsymbol{k}_7) \| (\boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \cdots \| \boldsymbol{k}_{15}) \tag{1}$$

如算法 2 所示，轮子密钥 rk_i 可由原始密钥 $\boldsymbol{K}$ 计算获得，其中，Q 表示已知数组，$1 \leqslant i \leqslant 20$。

算法 2 密钥编排方案

输入 $\boldsymbol{K}$

输出 $\mathbf{rk}_1, \mathbf{rk}_2, \cdots, \mathbf{rk}_{19}, \mathbf{rk}_{20}$

1) $\boldsymbol{K}_0 \| \boldsymbol{K}_1 = \boldsymbol{K}$;

2) $\boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \boldsymbol{k}_{10} \| \boldsymbol{k}_{11} \| \boldsymbol{k}_{12} \| \boldsymbol{k}_{13} \| \boldsymbol{k}_{14} \| \boldsymbol{k}_{15} = \boldsymbol{K}_1$;

3) $Q = [193,165,81,197]$;

4) for $i = 1$ to 20 do

5) for $t = 0$ to 3 do

6) $\boldsymbol{k}_{2t+9} = (\boldsymbol{k}_{2t+9} + Q[t]i) \bmod 256$;

7) end for

8) $\mathbf{rk}_i = \boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \boldsymbol{k}_{10} \| \boldsymbol{k}_{11} \| \boldsymbol{k}_{12} \| \boldsymbol{k}_{13} \| \boldsymbol{k}_{14} \| \boldsymbol{k}_{15}$;

9) end for

10) return $\mathbf{rk}_1, \mathbf{rk}_2, \cdots, \mathbf{rk}_{19}, \mathbf{rk}_{20}$

2 PRIDE 密码的故障分析

2.1 基本假设

统计故障分析和不可能统计故障分析的基本假设均为唯密文攻击，即攻击者采用同一原始密钥对随机明文进行加密，并获得密文。唯密文攻击仅能获得随机密文，对攻击者能力要求低于常见的已知明文攻击和选择明文攻击，属于密码分析要求中最弱的攻击，因此在轻量级环境中更易实现。

2.2 统计故障分析简述

2.2.1 故障模型

统计故障分析采用随机半字节故障模型，即攻击者以按比特与的方式将随机半字节故障注入加密过程的某一轮中，密文输出会呈现出非均匀的概率分布。受半字节故障影响的中间状态值分布如图 4 所示。攻击者使用区分器对候选密钥进行筛选，发现算法中的漏洞，从而可以快速恢复出轮密钥和原始密钥。

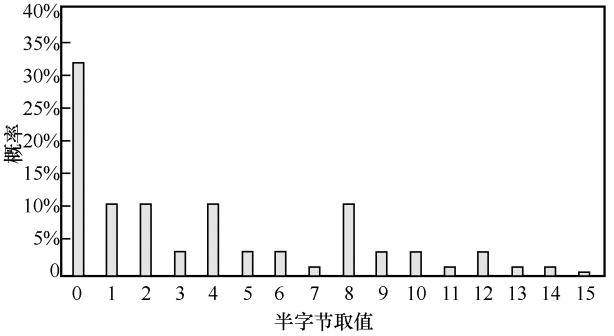


图 4 受半字节故障影响的中间状态值分布

2.2.2 攻击过程简述

统计故障分析是第一种使用唯密文攻击作为基本假设的故障分析方法，需要大量的故障注入和有效的统计分析方法来破译密码。它最早由 Fuhr 等[15]提出，并应用于 AES 密码算法的分析中，后来应用于 LED 和 Simeck 等算法中[24-25]。攻击过程包括以下 4 步。

步骤 1 随机故障注入。攻击者通过控制算法加密过程中的参数或环境变量，如干扰电压、时钟、温度、电磁辐射或修改代码等软硬件攻击手段注入随机故障，导致算法运行过程出现故障，从而获得随机错误密文。

步骤 2 统计关系分析。鉴于每个候选密钥均与中间状态值相对应，攻击者需结合错误密文，利用统计分析，穷举所有可能的候选密钥，收集并推导注入故障位置的中间状态值。

步骤 3 区分器选择。攻击者选取合适的区分器，将中间状态值代入区分器中进行计算，并获得区分器最值对应的候选轮密钥。

步骤 4 原始密钥破译。攻击者不断缩小候选轮密钥集合范围，并利用密钥编排方案求出原始密钥。

2.3 不可能统计故障分析

2.3.1 不可能关系分析

在密码分析中，国内外研究者经常使用不可能关系发现密码的脆弱处。1999 年，Biham 等[26-27]首次提出了不可能差分分析，并通过实验证明该方法在恢复 IDEA 密钥中具有很大优势且该方法恢复 31 轮 Skipjack 密钥的速度比穷举搜索更快。此后，不可能差分分析方法被广泛应用于各种轻量级密码的研究中，例如 XTEA、TEA、AES、Camellia、CLEFIA、LBlock、Simon、QARMA、PRINCEcore 等[28-34]。同时，不可能关系还被扩展应用到故障分析中。2005 年，Biham 等[35]使用不可能故障分析方

法以更低的复杂度破译了流密码 RC4。2011 年，Derbez 等[18]对 AES 密码进行了研究，采用不可能故障分析方法，在倒数第 4 轮注入故障并以更少的故障数破译了密码。2018 年，Li 等[36]将不可能关系与中间相遇相结合对 LED 密码进行了分析，在倒数第 5 轮注入故障，恢复了 LED-64 和 LED-128 的密钥。

2.3.2　故障模型

本文将统计故障分析技术与不可能关系分析技术相结合，提出一种新型的不可能统计故障分析方法。该分析方法采用随机半字节故障模型，即攻击者以按比特与的方式将随机半字节故障注入加密过程的某一轮中，同样密文输出会呈现出非均匀的概率分布。在故障注入之后，根据与运算特点可知，任何中间状态和 $0b1111$ 进行按比特与运算都不会改变其值，相当于无效故障，因此半字节中间状态概率满足

$$\frac{3^{n-\mathrm{hw}(\beta)}-1}{2^n(2^n-1)} \tag{2}$$

其中，n 为注入故障的比特数，β 为中间状态所有可能取值，$\mathrm{hw}(\beta)$ 为 β 的汉明重量，$0\leqslant\beta\leqslant15$，$0\leqslant\mathrm{hw}(\beta)\leqslant4$，$n=4$。

2.3.3　攻击过程

不可能统计故障分析过程具体包括以下 4 个步骤。

步骤 1　注入随机故障。攻击者使用原始密钥对随机明文进行加密，在倒数第 2 轮中导入随机半字节故障，生成多个故障密文。图 5 展示了导入故障后的扩散路径，其中灰色阴影为受故障影响的单元。

步骤 2　建立不可能统计关系。攻击者利用故障密文集合可以推导出倒数第 2 轮非线性替换层后的中间状态值为

$$\boldsymbol{T}=\mathrm{Sub}^{-1}(\mathrm{Pub}^{-1}(\hat{\boldsymbol{C}})\oplus\boldsymbol{K}_0)\oplus\mathrm{Pub}^{-1}(\mathbf{rk}_{20}) \tag{3}$$

其中，$\boldsymbol{T}$、$\hat{\boldsymbol{C}}$、$\boldsymbol{K}_0$ 和 $\mathbf{rk}_{20}$ 分别为中间状态值、错误密文、白化密钥和轮子密钥。结合图 5 分析可知，利用白化密钥 $\boldsymbol{K}_0$ 的 4 bit 和轮子密钥 $\mathbf{rk}_{20}$ 的 4 bit，可推导出故障注入后的中间状态 4 bit，即

$$\begin{aligned}&\boldsymbol{T}^{60}\|\boldsymbol{T}^{61}\|\boldsymbol{T}^{62}\|\boldsymbol{T}^{63}=\\&\mathrm{Sub}^{-1}(\mathrm{Pub}^{-1}(\hat{\boldsymbol{C}}^{60}\|\hat{\boldsymbol{C}}^{61}\|\hat{\boldsymbol{C}}^{62}\|\hat{\boldsymbol{C}}^{63})\oplus(\boldsymbol{K}_0^{60}\|\boldsymbol{K}_0^{61}\|\\&\boldsymbol{K}_0^{62}\|\boldsymbol{K}_0^{63}))\oplus\mathrm{Pub}^{-1}(\mathbf{rk}_{20}^{60}\|\mathbf{rk}_{20}^{61}\|\mathbf{rk}_{20}^{62}\|\mathbf{rk}_{20}^{63})\end{aligned} \tag{4}$$

其中，$\boldsymbol{T}^j$、$\hat{\boldsymbol{C}}^j$、$\boldsymbol{K}_0^j$ 和 $\mathbf{rk}_{20}^j$ 分别为中间状态值、错误密文、白化密钥和轮子密钥的第 j bit，

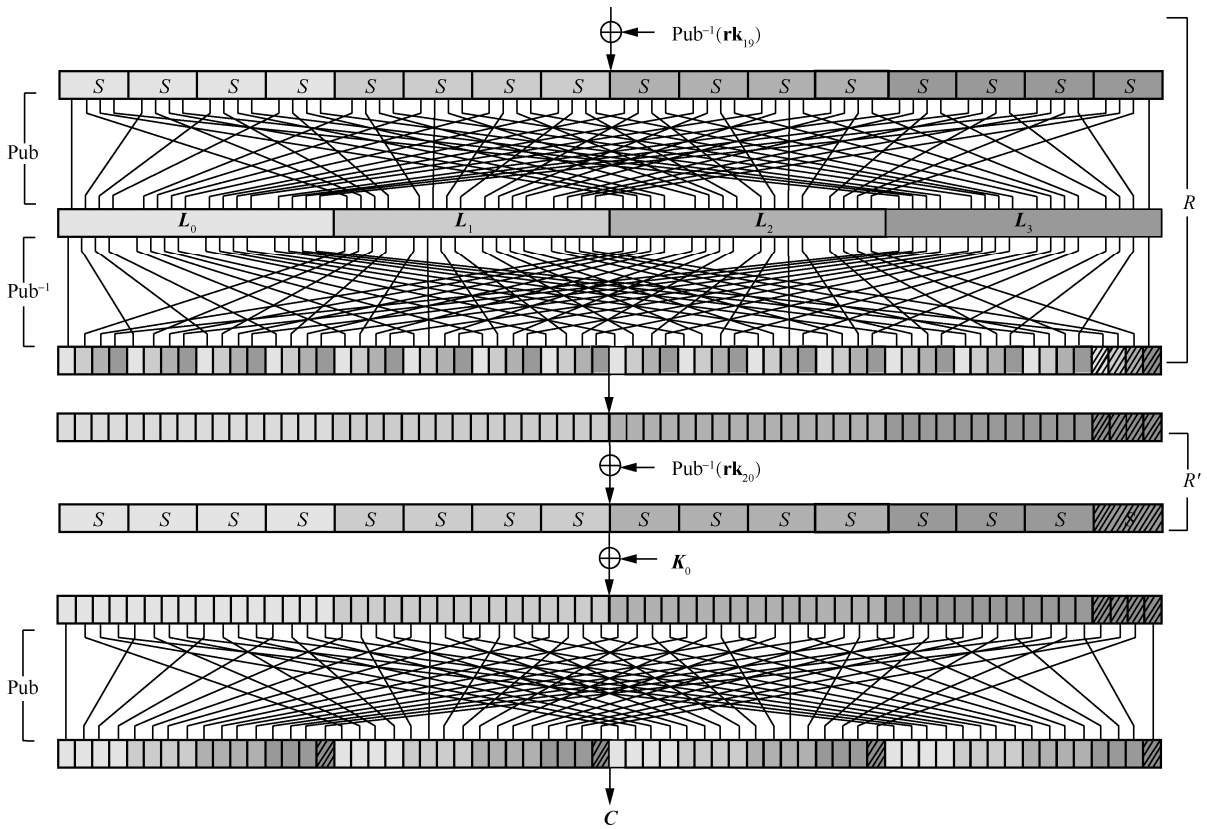


图 5　PRIDE 算法加密倒数第 2 轮导入随机半字节故障后的扩散路径

$0 \leqslant j \leqslant 63$。结合故障模型分析可知，中间状态值为零，即

$$\boldsymbol{T}^{j} \| \boldsymbol{T}^{j+1} \| \boldsymbol{T}^{j+2} \| \boldsymbol{T}^{j+3} \oplus 0b1111 = 0 \tag{5}$$

该中间状态值对应的候选密钥不可能为正确密钥。因此，通过排除不可能为正确密钥的部分候选密钥，进而缩小需要使用区分器进行筛选的候选密钥范围。

步骤 3 选择合适的区分器。在步骤 2 排除部分密钥后，剩余每个候选密钥都对应着中间状态值。攻击者可以选取 2.4 节中合适的区分器，获得剩余候选密钥对应的中间状态值的区分器的所有可能值。基于区分器的不同特性，攻击者选择最大值或最小值，从而计算获得正确的 8 bit 密钥，包括白化密钥 $\boldsymbol{K}_0$ 的 4 bit 和轮子密钥 $\mathbf{rk}_{20}$ 的 4 bit，其注入故障位置与可恢复密钥位置对应关系如图 6 所示。

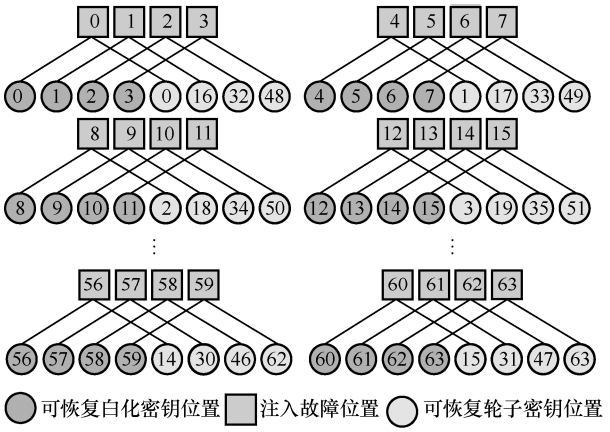


图 6 注入故障位置与可恢复密钥位置对应关系

步骤 4 恢复原始密钥。通过多次重复步骤 1~步骤 3，直到恢复出白化密钥 $\boldsymbol{K}_0$ 和轮子密钥 $\mathbf{rk}_{20}$ 的全部 64 bit。基于密钥编排方案和算法 3，求出原始密钥为

$$\begin{aligned}&\boldsymbol{K} = \boldsymbol{K}_0 \| \boldsymbol{K}_1 = \\ &\boldsymbol{K}_0 \| (\boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \boldsymbol{k}_{10} \| \boldsymbol{k}_{11} \| \boldsymbol{k}_{12} \| \boldsymbol{k}_{13} \| \boldsymbol{k}_{14} \| \boldsymbol{k}_{15})\end{aligned} \tag{6}$$

算法 3 恢复原始密钥计算过程

输入 $\boldsymbol{K}_0, \mathbf{rk}_{20}$

输出 $\boldsymbol{K}$

1) $\boldsymbol{K}_1 = \mathbf{rk}_{20}$;

2) $\boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \boldsymbol{k}_{10} \| \boldsymbol{k}_{11} \| \boldsymbol{k}_{12} \| \boldsymbol{k}_{13} \| \boldsymbol{k}_{14} \| \boldsymbol{k}_{15} = \boldsymbol{K}_1$;

3) $Q = [193,165,81,197]$;

4) for $t = 0$ to 3 do

5) $\boldsymbol{k}_{2t+9} = \boldsymbol{k}_{2t+9} - (Q[t]20) \bmod 256$;

6) end for

7) $\boldsymbol{K}_1 = \boldsymbol{k}_8 \| \boldsymbol{k}_9 \| \boldsymbol{k}_{10} \| \boldsymbol{k}_{11} \| \boldsymbol{k}_{12} \| \boldsymbol{k}_{13} \| \boldsymbol{k}_{14} \| \boldsymbol{k}_{15}$;

8) $\boldsymbol{K} = \boldsymbol{K}_0 \| \boldsymbol{K}_1$;

9) return $\boldsymbol{K}$

2.4 区分器

2013 年，Fuhr 等[15]使用汉明重量、极大似然估计等区分器对 AES 密码及其工作模式进行统计故障分析。本文提出了新型的卡方拟合优度−汉明重量区分器和卡方拟合优度−极大似然估计区分器对 PRIDE 密码进行分析。

2.4.1 经典区分器

1) 汉明重量区分器

1954 年，Reed[37]提出了一种计算二进制字符串的汉明重量的概念。Fuhr 等[15]首次将汉明重量区分器应用于故障分析中，通过计算中间状态值的二进制中非零的个数，选择最小个数对应的统计样本。汉明重量区分器表达式为

$$\mathrm{HW} = \sum_{a=1}^{m} \mathrm{hw}(\lambda_a) \tag{7}$$

其中，m 表示导入故障数，λ_a 表示第 a 个故障推导出来的中间状态值，$\mathrm{hw}(\lambda_a)$ 表示 λ_a 的汉明重量。由于导入随机故障后，会破坏中间状态值 0 和 1 的平衡，受故障影响后的半字节会呈现 0 比 1 多的情况，因此计算得到汉明重量值最小的候选密钥，即正确密钥。

2) 极大似然估计区分器

Wilks[38]于 1938 年提出基于参数估计的极大似然估计方法。2013 年，Fuhr 等[15]使用极大似然估计区分器应用于统计故障分析 AES 密码算法，通过似然函数来计算每一组样本值理论应该出现的概率，并选出最大概率值对应的 AES 候选密钥。极大似然估计区分器表达式为

$$\mathrm{MLE} = \prod_{a=1}^{m} p(\lambda_a) \tag{8}$$

其中，m 表示注入故障数量，λ_a 表示第 a 个故障推导出来的中间状态值，$p(\lambda_a)$ 表示中间状态值为 λ_a 的理论概率。当极大似然估计区分器取最大值时，对应的候选密钥即正确密钥。

2.4.2 新型区分器

1900 年，Pearson[39]首次提出卡方检验，广泛

应用于分类变量的独立性检验和比较检验中，是统计学中常用的假设检验方法之一。卡方拟合优度在卡方检验的基础上进行了优化，可以用来计算观察到的结果与理论结果之间的差异大小。卡方拟合优度检验区分器表达式为

$$\chi^2=\sum_{a=1}^{\varepsilon}\frac{(\varphi_a-m\psi_a)^2}{m\psi_a} \tag{9}$$

其中，m 为注入的故障数，ε 为注入的半字节故障所有的可能取值个数，φ_a 为每一个半字节中间状态值在导入 m 个故障数时出现的频次，ψ_a 为每一个半字节中间状态值理论上的频率。卡方拟合优度值越小，则观察到的结果与理论结果之间差异越小。

1) 卡方拟合优度–汉明重量区分器

CS-HW 区分器将卡方拟合优度检验与汉明重量相结合。攻击者先使用汉明重量区分器对候选密钥进行初步筛选，排除汉明重量值较大的候选密钥，再使用卡方拟合优度区分器对剩余的候选密钥进行筛选，卡方拟合优度值最小的候选密钥即正确密钥。

2) 卡方拟合优度–极大似然估计区分器

CS-MLE 区分器将卡方拟合优度检验与极大似然估计相结合。攻击者先使用极大似然估计区分器对候选密钥进行初步筛选，保留较大的极大似然估计值对应的候选密钥，排除较小的，再使用卡方拟合优度区分器对剩余的候选密钥进行筛选，保留区分器值最小的候选密钥，即正确密钥。表 2 给出了本文使用的区分器取值情况以及筛选过程说明。

表 2　本文使用的区分器取值情况以及筛选过程说明

区分器	取值范围	筛选过程
HW	最小值	评估中间状态的汉明重量，选择值最小的统计样本
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
CS-HW	HW 最小值、CS 最小值	先使用 HW 区分器选择值最小的统计样本，再使用 CS 区分器选择值最小的统计样本
CS-MLE	MLE 最大值、CS 最小值	先使用 MLE 区分器选择出概率最大的统计样本，再使用 CS 区分器选择值最小的统计样本

3　实验分析

本节实验使用的 PC 端设备配置为 Intel(R) Core(TM) i9-13900H CPU，使用 Python 语言编程实现 PRIDE 算法的不可能统计故障分析过程，导入故障的动作由计算机软件模拟实现。本文共进行了 10 000 次实验，每次实验包括故障注入以及恢复原始密钥的过程。

3.1　故障数

故障数是衡量所有故障分析方法优劣的重要指标之一，指攻击者在故障分析过程中实际导入算法中的故障总数。若注入的故障数越少，则表明分析代价越小，在实际应用中越易实现。表 3 列出了使用不同故障分析方法以及区分器在成功率为 99%及以上时恢复原始密钥 128 bit 的故障数。使用不可能统计故障分析方法恢复出 128 bit 原始密钥，并选择 HW 区分器、MLE 区分器、CS-HW 区分器和 CS-MLE 区分器，分别需要 592、560、448 和 432 个故障，均小于经典统计故障分析方法的 656、624、608 和 560 个故障。并且，2 个新型复合区分器仅需 448 和 432 个故障，均小于经典区分器所需故障数。

表 3　使用不同故障分析方法以及区分器在成功率为 99%及以上时恢复原始密钥 128 bit 的故障数

区分器	统计故障分析		不可能统计故障分析	
	故障数/个	成功率	故障数/个	成功率
HW	656	99%	592	99%
MLE	624	99%	560	99%
CS-HW	608	99%	448	99%
CS-MLE	560	99%	432	99%

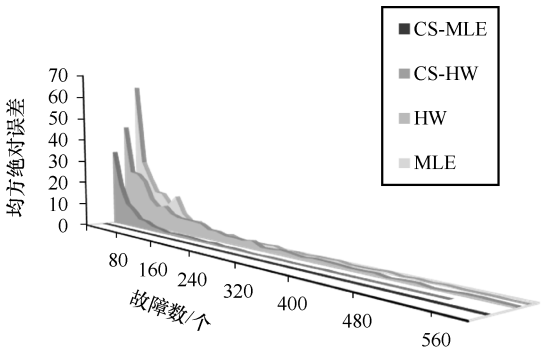
3.2　精确度

精确度表示统计结果与真实结果之间的接近程度。精确度越高，表示候选密钥越接近真实密钥。本文采用平均绝对误差来判断精确度大小，平均绝对误差值越小，表示精确度越高。平均绝对误差的表达式为

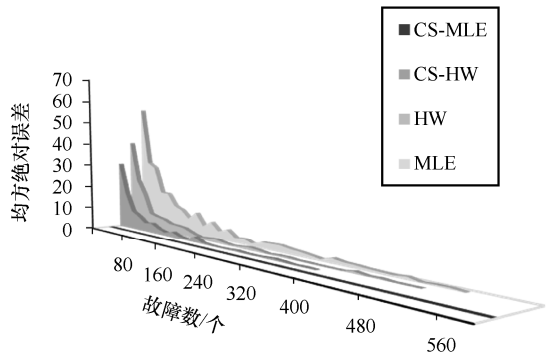
$$\mathrm{MAE}=\frac{1}{u}\sum_{a=1}^{u}|\phi(a)-1| \tag{10}$$

其中，u 表示实验次数，$\phi(a)$ 表示第 a 次实验时获取的候选密钥个数。实验筛选出的候选密钥集合中密钥数量越少，则候选密钥越接近真实密钥，此时，平均绝对误差越趋于 0，区分器性能越优。图 7 是统计故障分析方法和不可能统计故障分析方法分别使用 4 个区分器时，实验筛选出候选密钥的平均绝对误差随故障数变化的情况。

对比其他区分器，新型区分器 CS-MLE 在导入较少故障数时也能够更精确地筛选出候选密钥集合，即平均绝对误差值在导入故障数很少时就趋于 0。新型区分器 CS-MLE 的特点在于其使用的卡方拟合优度取最小值，极大似然估计取最大值。这种筛选密钥候选集的方式可以在最大限度上将非正确密钥排除，使筛选出的候选密钥集合精确度更高。与经典区分器相比，本文提出的 2 个新型区分器具有更高的精确度，其中 CS-MLE 区分器表现更优。



(a) 统计故障分析方法使用各区分器恢复原始密钥精确度



(b) 不可能统计故障分析方法使用各区分器恢复原始密钥精确度

图 7　统计故障分析方法和不可能统计故障分析方法使用各区分器恢复原始密钥的精确度

3.3　耗时

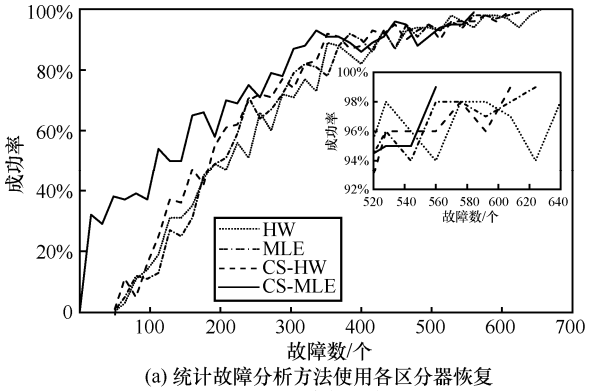
耗时是指从故障注入到恢复原始密钥所消耗的时间。表 4 给出了统计故障分析和不可能统计故障分析使用不同区分器以 99%以上成功率恢复 128 bit 原始密钥的耗时。表 4 表明，使用不同区分器恢复 128 bit 密钥时，不可能统计故障分析的耗时分别为 1.90 s、1.81 s、1.43 s 和 1.40 s，均少于统计故障分析方法的 2.07 s、1.99 s、1.92 s 和 1.80 s。新型复合区分器的耗时仅为 1.43 s 和 1.40 s，均少于经典区分器的耗时。

表 4　使用不同故障分析方法以及区分器在成功率为 99%及以上时恢复原始密钥 128 bit 的耗时

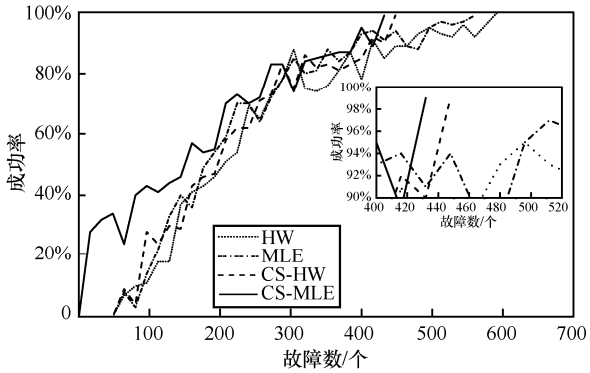
区分器	统计故障分析		不可能统计故障分析	
	耗时/s	成功率	耗时/s	成功率
HW	2.07	99%	1.90	99%
MLE	1.99	99%	1.81	99%
CS-HW	1.92	99%	1.43	99%
CS-MLE	1.80	99%	1.40	99%

3.4　成功率

成功率指破译密钥的成功概率。图 8 表示统计故障分析方法和不可能统计故障分析方法使用各区分器恢复 128 bit 原始密钥的成功率，横轴表示注入的故障数，纵轴表示恢复 128 bit 原始密钥的成功率。实验结果表明，所有方法和区分器均能达到 99%及以上的成功率。本文提出的新型方法和区分器能以较小的故障数，先达到 99%及以上的成功率，表现更佳。



(a) 统计故障分析方法使用各区分器恢复 128 bit原始密钥的成功率



(b) 不可能统计故障分析方法使用各区分器恢复 128 bit原始密钥的成功率

图 8　统计故障分析方法和不可能统计故障分析方法使用各区分器恢复 128 bit 原始密钥的成功率

3.5　复杂度

时间复杂度、数据复杂度和空间复杂度可用于衡量破译密码时所需的时间、数据量和空间量，计算式分别为

$$m2^n w \tag{11}$$

$$m2^n \tag{12}$$

$$vm2^n \tag{13}$$

其中，m 为总故障数，n 为枚举密钥的比特数，2^n 为枚举密钥的个数，w 为所选区分器本身的复杂度，v 为存储的密文比特数。汉明重量、极大似然估计和卡方拟合优度区分器本身的复杂度均为 $2^{4.09}$，复合区分器的复杂度为 2 个单区分器叠加之和。表 5 分别给出了不同故障分析方法使用所有区分器恢复 128 bit 原始密钥所需的时间、数据和空间复杂度。对比可知，不可能统计故障分析方法所需的时间、数据和空间复杂度均比统计故障分析方法低。

因此，结合故障数、精确度、耗时、成功率以及复杂度等指标，不可能统计故障分析在破译 PRIDE 密码的 128 bit 原始密钥表现较佳。

表 5　区分器恢复 128 bit 原始密钥的复杂度分析

区分器	统计故障分析			不可能统计故障分析		
	时间复杂度	数据复杂度	空间复杂度	时间复杂度	数据复杂度	空间复杂度
HW	$2^{21.45}$	$2^{17.36}$	$2^{23.36}$	$2^{21.30}$	$2^{17.21}$	$2^{23.21}$
MLE	$2^{21.37}$	$2^{17.29}$	$2^{23.29}$	$2^{21.22}$	$2^{17.13}$	$2^{23.13}$
CS-HW	$2^{22.34}$	$2^{17.25}$	$2^{23.25}$	$2^{21.89}$	$2^{16.81}$	$2^{22.81}$
CS-MLE	$2^{22.22}$	$2^{17.13}$	$2^{23.13}$	$2^{21.84}$	$2^{16.75}$	$2^{22.75}$

4　结束语

本文针对 PRIDE 算法抵抗不可能统计故障分析的安全性进行了研究。分析表明，新型不可能统计故障分析方法不仅破译密码的成功率能够达到 99%及以上，还能在更短的时间内以更少的故障数和更低的复杂度破译 PRIDE 算法。因此，PRIDE 算法易受不可能统计故障分析的威胁。在实际智能无人系统中使用该算法时，建议采取必要的有效措施抵御故障分析的攻击。下一步工作将结合 PRIDE 算法的内部更深轮进行安全性分析。

参考文献：

[1] SINGH R, BHUSHAN B. Evolving intelligent system for trajectory tracking of unmanned aerial vehicles[J]. IEEE Transactions on Automation Science and Engineering, 2022, 19(3): 1971-1984.

[2] ALRAWI O, LEVER C, ANTONAKAKIS M, et al. SoK: security evaluation of home-based IoT deployments[C]//Proceedings of IEEE Symposium on Security and Privacy. Piscataway: IEEE Press, 2019: 1362-1380.

[3] 吴武飞, 李仁发, 曾刚, 等. 智能网联车网络安全研究综述[J]. 通信学报, 2020, 41(6): 161-174.
WU W F, LI R F, ZENG G, et al. Survey of the intelligent and connected vehicle cybersecurity[J]. Journal on Communications, 2020, 41(6): 161-174.

[4] 王圣宝, 周鑫, 文康, 等. 适用于智能电网的三方认证密钥交换协议[J]. 通信学报, 2023, 44(02): 210-218.
WANG S B, ZHOU X, WEN K, et al. Tripartite authenticated key exchange protocol for smart grid[J]. Journal on Communications, 2023, 44(2): 210-218.

[5] NAITO Y, SASAKI Y, SUGAWARA T. Lightweight authenticated encryption mode suitable for threshold implementation[C]//Proceedings of Annual International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 2020: 705-735.

[6] CHENG J, GUO S, HE J. An extended type-1 generalized Feistel networks: lightweight block cipher for IoT[J]. IEEE Internet of Things Journal, 2022, 9(13): 11408-11421.

[7] CHENG H, GROSSSCHÄDL J, MARSHALL B, et al. RISC-V instruction set extensions for lightweight symmetric cryptography[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2022(1): 193-237.

[8] ZHU D, ZHANG R, OU L, et al. Low-latency design and implementation of the squaring in class groups for verifiable delay function using redundant representation[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2022(1): 438-462.

[9] ALBRECHT M R, DRIESSEN B, KAVUN E B, et al. Block ciphers – focus on the linear layer (feat. PRIDE)[C]//Proceedings of Advances in Cryptology. Berlin: Springer, 2014: 57-76.

[10] 伊文坛, 田亚, 陈少真. 减缩轮 PRIDE 算法的线性分析[J]. 电子学报, 2017, 45(2): 468-476.
YIN W T, TIAN Y, CHEN S Z. Linear cryptanalysis of reduced-round PRIDE block cipher[J]. Chinese Journal of Electronics, 2017, 45(2): 468-276.

[11] LALLEMAND V, RASOOLZADEH S. Differential cryptanalysis of 18-round PRIDE[C]//Proceedings of International Conference on Cryptology in India. Berlin: Springer, 2017: 126-146.

[12] PAL D, MANDAL U, DAS A, et al. Deep learning based differential classifier of PRIDE and RC5[C]//Proceedings of International Conference on Applicat ions and Techniques in Information Security. Berlin: Springer, 2023: 46-58.

[13] YANG Q, HU L, SUN S, et al. Improved differential analysis of block cipher PRIDE[C]//Proceedings of International Conference on Information Security Practice and Experience. Berlin: Springer, 2015: 209-219.

[14] BONEH D, DEMILLO R A, LIPTON R J. On the importance of checking cryptographic protocols for faults[C]//Proceedings of International Conference on the Theory and Applications of Cryptographic Techniques. Berlin: Springer, 1997: 37-51.

[15] FUHR T, JAULMES E, LOMNE V, et al. Fault attacks on AES with faulty ciphertexts only[C]//Proceedings of Workshop on Fault Diagnosis and Toler ance in Cryptography. Piscataway: IEEE Press, 2013: 108-118.

[16] CLAVIER C. Secret external encodings do not prevent transient fault analysis[C]//Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2007: 181-194.

[17] BIHAM E, SHAMIR A. Differential fault analysis of secret key cryptosystems[C]//Proceedings of Annual International Cryptology Conference. Berlin: Springer, 1997: 513-525.

[18] DERBEZ P, FOUQUE P A, LERESTEUX D. Meet-in-the-middle and impossible differential fault analysis on AES[C]//Proceedings of International Workshop on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2011: 274-291.

[19] ZHANG F, LOU X, ZHAO X, et al. Persistent fault analysis on block ciphers[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2018(3): 150-172.

[20] JANA A, PAUL G. Differential fault attack on PHOTON-Beetle[C]// Proceedings of Workshop on Attacks and Solutions in Hardware Security. New York: ACM Press, 2022: 25-34.

[21] ZHANG F, FENG T, LI Z, et al. Free fault leakages for deep exploitation: algebraic persistent fault analysis on lightweight block ciphers[J]. IACR Transactions on Cryptographic Hardware and Embedded Systems, 2022(2): 289-311.

[22] 王永娟, 樊昊鹏, 代政一, 等. 侧信道攻击与防御技术研究进展[J]. 计算机学报, 2023, 46(1): 202-228.
WANG Y J, FAN H P, DAI Z Y, et al. Advances in side channel attacks and countermeasures[J]. Chinese Journal of Computers, 2023, 46(1): 202-228.

[23] 李玮, 刘春, 谷大武, 等. Saturnin-Short 轻量级认证加密算法的统计无效故障分析[J]. 通信学报, 2023, 44(4): 167-175.
LI W, LIU C, GU D W, et al. Statistical ineffective fault analysis of the lightweight authenticated cipher algorithm Saturnin-Short[J]. Journal on Communications, 2023, 44(4): 167-175.

[24] LI W, LIAO L, GU D, et al. Ciphertext-only fault analysis on the LED lightweight cryptosystem in the Internet of things[J]. IEEE Transactions on Dependable and Secure Computing, 2019, 16(3): 454-461.

[25] LI W, LI J, GU D, et al. Statistical fault analysis of the Simeck lightweight cipher in the ubiquitous sensor networks[J]. IEEE Transactions on Information Forensics and Security, 2021, 16: 4224-4233.

[26] BIHAM E, BIRYUKOV A, SHAMIR A. Miss in the middle attacks on IDEA and Khufu[C]//Proceedings of International Workshop on Fast Software Encryption. Berlin: Springer, 1999: 124-138.

[27] BIHAM E, BIRYUKOV A, SHAMIR A. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials[J]. Journal of Cryptology, 2005, 18(4): 291-311.

[28] MOON D, HWANG K, LEE W, et al. Impossible differential cryptanalysis of reduced round XTEA and TEA[C]//Proceedings of International Workshop on Fast Software Encryption. Berlin: Springer, 2002: 49-60.

[29] CHEN J, HU Y, ZHANG Y. Impossible differential cryptanalysis of advanced encryption standard[J]. Science China Information Sciences, 2007, 50(3): 342-350.

[30] WU W, ZHANG L, ZHANG W. Improved impossible differential cryptanalysis of reduced-round Camellia[C]//Proceedings of International Workshop on Selected Areas in Cryptography. Berlin: Springer, 2009: 442-456.

[31] BOURA C, NAYA-PLASENCIA M, SUDER V. Scrutinizing and improving impossible differential attacks: applications to CLEFIA, Camellia, LBlock and Simon[C]//Proceedings of International Conference on the Theory and Application of Cryptology and Information Security. Berlin: Springer, 2014: 179-199.

[32] LIU Y, SHI Y, GU D, et al. Improved impossible differential cryptanalysis of large-block Rijndael[J]. Science China Information Sciences, 2019, 62(3): 1-14.

[33] DU J, WANG W, LI M, et al. Related-tweakey impossible differential attack on QARMA-128[J]. Science China Information Sciences, 2019, 62(3): 15-26.

[34] ZHANG L, WU W, MAO Y. Impossible differential cryptanalysis on reduced-round PRINCEcore[C]//Proceedings of International Conference on Inform ation Security and Cryptology. Berlin: Springer, 2023: 61-77.

[35] BIHAM E, GRANBOULAN L, NGUYỄN P Q. Impossible fault analysis of RC4 and differential fault analysis of RC4[C]//Proceedings of Internation al Workshop on Fast Software Encryption. Berlin: Springer, 2005: 359-367.

[36] LI W, RIJMEN V, TAO Z, et al. Impossible meet-in-the-middle fault analysis on the LED lightweight cipher in VANETs[J]. Science China Information Sciences, 2018, 61(3): 032110.

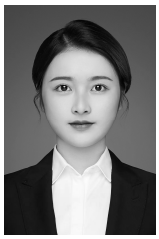
[37] REED I. A class of multiple-error-correcting codes and the decoding scheme[J]. Transactions of the IRE Professional Group on Information Theory, 1954, 4(4): 38-49.

[38] WILKS S S. The large-sample distribution of the likelihood ratio for testing composite hypotheses[J]. The Annals of Mathematical Statistics, 1938, 9(1): 60-62.

[39] PEARSON K X. On the criterion that a given system of deviations from the probable in the case of a correlated system of variables is such that it can be reasonably supposed to have arisen from random sampling[J]. The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science, 1900, 50(302): 157-175.

[作者简介]

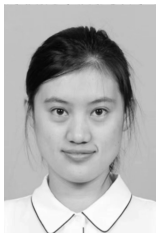
李玮（1980− ），女，安徽寿县人，博士，东华大学教授、博士生导师，主要研究方向为对称密码的设计与分析。



孙文倩（2000− ），女，安徽铜陵人，东华大学硕士生，主要研究方向为对称密码的故障分析。

谷大武（1970− ），男，河南漯河人，博士，上海交通大学教授、博士生导师，主要研究方向为密码学和计算机安全。

张爱琳（2001− ），女，吉林四平人，东华大学硕士生，主要研究方向为轻量级分组密码的故障分析。



温云华（1990− ），女，山东临清人，博士，东华大学讲师、硕士生导师，主要研究方向为密码学。