

LiCi 密码的差分故障攻击

陈伟建¹, 罗皓翔²

(1. 电子科技大学信息与通信工程学院, 四川 成都 611731;

2. 电子科技大学格拉斯哥学院, 四川 成都 611731)

摘要: LiCi 轻量级分组密码算法是 2017 年提出的一种新型密码算法, 其具有结构微小、消耗能量少等优点, 适用于物联网等资源受限的环境。在 LiCi 的设计文档中, 对该算法抵御差分攻击和线性攻击的能力进行了分析, 但 LiCi 算法对于差分故障攻击的抵抗能力尚未得到讨论。针对 LiCi 算法每轮迭代的移位规律, 在第 31 轮迭代时的左半侧多次注入单比特故障, 结合其差分性质, 可以恢复 32 bit 长度的轮密钥。根据 LiCi 算法的密钥编排方案, 再对第 30、29、28、27、26 轮迭代进行同样的差分故障攻击, 最终可以恢复全部原始密钥。该攻击共需要 48 个单比特故障, 计算复杂度为 2^{32} , 说明 LiCi 算法难以抵抗差分故障攻击。

关键词: LiCi 密码; 轻量级分组密码; 差分故障攻击; 故障模型

中图分类号: TP309

文献标识码: A

doi: 10.11959/j.issn.2096-109x.2021033

Differential fault attack on LiCi cipher

CHEN Weijian¹, LUO Haoxiang²

1. School of Information and Communication Engineering, University of Electronic Science and Technology of China, Chengdu 611731, China

2. Glasgow College, University of Electronic Science and Technology of China, Chengdu 611731, China

Abstract: LiCi lightweight block cipher is a new algorithm proposed in 2017. With advantages of small structure and low energy consumption, LiCi is more suitable for resource-constrained environments such as the internet of things (IoT). In the design document of LiCi, the ability of LiCi algorithm to resist differential attack and linear attack is analyzed, but the resistance of LiCi algorithm to differential fault attack has not been discussed. According to the permutation law of each round iteration of LiCi algorithm, 32-bit key can be recovered by injecting a single bit fault into the left half of the 31st round iteration combined with its differential property. According to the key choreography scheme of the LiCi algorithm, the same differential fault attack was performed on iterations 30th, 29th, 28th, 27th and 26th round to recover all the original keys. The attack requires a total of 48-bit faults, and the computational complexity is 2^{32} , which indicates the LiCi algorithm is difficult to resist differential fault attacks.

Keywords: LiCi cipher, lightweight block cipher, differential fault attack, fault model

收稿日期: 2020-08-04; 修回日期: 2020-12-07

通信作者: 罗皓翔, lhx991115@163.com

基金项目: 电子科技大学创新创业院长基金 (2019007)

Foundation Item: Dean's Fund of Innovation and Entrepreneurship, UESTC (2019007)

论文引用格式: 陈伟建, 罗皓翔. LiCi 密码的差分故障攻击[J]. 网络与信息安全学报, 2021, 7(2): 104-109.

CHEN W J, LUO H X. Differential fault attack on LiCi cipher[J]. Chinese Journal of Network and Information Security, 2021, 7(2): 104-109.

1 引言

轻量级分组密码算法因其结构较为简单、加解密速度快、消耗计算资源较少且利于软硬件实现等特点,被广泛应用在物联网终端中,如 RFID、无线传感器网络(WSN)、智能卡等计算资源受限的终端设备。针对这些设备信息安全保护、传输的需求,众多学者提出了许多低成本、低能耗的轻量级密码算法,如 HIGHT^[1]、PRESENT^[2]、GIFT^[3]、SPECK^[4]、SMS4^[5]等,这些算法具有扩散快速、加解密一致、易于实现等特点。

2017年,Patil等^[6]提出了一种新的轻量级分组密码算法 LiCi。该算法采用了平衡 Feistel 结构,在 LiCi 算法的单轮加密结构中, S 盒可以影响到左右两支,相比于普通的 Feistel 结构分组密码算法具有更快的扩散性,能够在较少的轮运算下产生数目最多的活跃 S 盒。该算法具有 64 bit 的分组长度和 128 bit 的密钥长度,具有结构紧凑、能耗低、占用面积小等特点。另外, LiCi 算法可以很好地抵抗差分攻击和线性攻击。此后,许多学者对该算法的密码安全性进行了大量的评估,韦永壮等^[7]提出了关于 LiCi 的 16 轮不可能差分分析,数据复杂度为 $2^{59.76}$; 信文倩等^[8]利用 12 轮积分器对 LiCi 进行 13 轮攻击,得到恢复 17 bit 轮密钥的数据复杂度为 2^{63} ; 王红艳等^[9]针对 LiCi 进行了积分区分器的搜索,发现 LiCi 存在 12 轮积分区分器,所需数据复杂度为 2^{61} 。但该算法能否较好地抵抗差分故障攻击仍有待进一步分析。

差分故障攻击^[10]是 Biham 和 Shamir 在 1997 年结合数学研究方法和物理方法的基础上,提出的一种新的密码分析方式。该方法被应用在许多经典分组密码上,如 GIFT^[11]、SMS4^[12]、Klein^[13]、MIBS^[14]、PRESENT^[15]、TWINE^[16]等。

本文基于 LiCi 算法本身的移位特性,结合 S 盒的差分性质,对该算法进行了差分故障攻击。当 LiCi 第 31 轮左半侧注入单比特故障后,经过 S 盒非线性变换、与右半侧及密钥进行异或操作后,向左移位 3 bit。此时故障会扩散到新的左半侧的两个半字节,因此对一轮迭代中进行 8 次不同位置的单比特故障注入,可以恢复该轮密钥,具体的故障注入位置在下文进行分析。此外,根

据 LiCi 密钥编排方案,需要连续 6 轮轮密钥,才能推出完整的 128 bit 原始密钥。因此,以同样的方式攻击第 30、29、28、27、26 轮迭代,共计 48 个单比特故障。结果证实 LiCi 算法在面对差分故障攻击时并不安全。

2 LiCi 密码算法

2.1 符号与术语约定

为更好地理解推导过程,下面对本文常用符号进行约定和说明。

$RK1_i$: 第 i 轮轮密钥的右半侧 32 bit。

$RK2_i$: 第 i 轮轮密钥的左半侧 32 bit。

X_i : 第 i 轮左半侧 32 bit 输入。

$x_{j,i}$: 第 i 轮位置为 j 的 S 盒 4 bit 输入。

$\Delta x_{j,i}$: 第 i 轮位置为 j 的 S 盒 4 bit 输入差分。

Y_i : 第 i 轮右半侧 32 bit 输入。

Cl_i : 第 i 轮的右半侧正确密文。

Cl_i^* : 第 i 轮的右半侧错误密文。

ΔCl_i^* : 第 i 轮正确密文与错误密文的右半侧差分。

$\Delta cl_{j,i}^*$: 右半侧第 i 轮位置为 j 的正确密文与错误密文的 4 bit 差分。

$C2_i$: 第 i 轮的左半侧正确密文。

$C2_i^*$: 第 i 轮的左半侧错误密文。

$\Delta C2_i^*$: 第 i 轮正确密文与错误密文的左半侧差分。

$\Delta c2_{j,i}^*$: 左半侧第 i 轮位置为 j 的正确密文与错误密文的 4 bit 差分。

S : S 盒运算符号。

S^{-1} : S 盒逆运算符号。

ΔW_i : 第 i 轮移位前的输出差分。

$\Delta w_{j,i}$: 第 i 轮位置为 j 的移位前 4 bit 差分。

$a \leftarrow b$: b 的值赋给 a 。

$a \parallel b$: 数据 a 和数据 b 的级联。

2.2 LiCi 加密流程

LiCi 算法具有 64 bit 的分组长度和 128 bit 的密钥长度,迭代轮数为 31 轮。该算法的单轮加密流程如图 1 所示,包含 S 盒字节替换、密钥异或

运算、循环移位等。其中,字节替换部分是唯一的非线性运算,由 8 个并列 4 乘 4 的 S 盒实现。

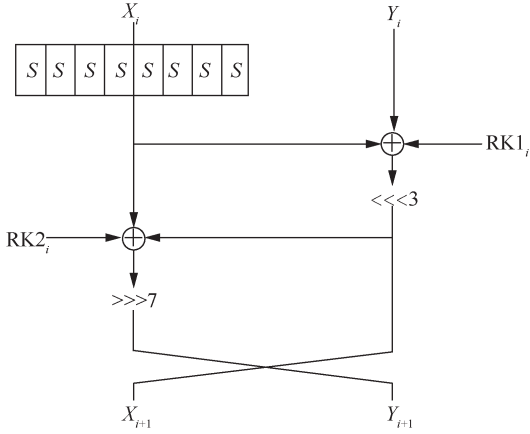


图 1 LiCi 加密流程
Figure 1 LiCi encryption processing

其加密过程可由下式表示。

$$\begin{aligned} X_{i+1} &= [S[X_i] \oplus Y_i \oplus RK1_i] \lll 3 \\ Y_{i+1} &= [S[X_i] \oplus X_{i+1} \oplus RK2_i] \ggg 7 \end{aligned} \quad (1)$$

S 盒变换对应值如表 1 所示。

表 1 LiCi 的 S 盒
Table 1 S-box value of LiCi

W	0	1	2	3	4	5	6	7
S(W)	3	F	E	1	0	A	5	8
W	8	9	A	B	C	D	E	F
S(W)	C	4	B	2	9	7	6	D

2.3 LiCi 密钥编排方案

LiCi 算法的密钥长度为 128 bit, 写作 $K = k_{127}k_{126} \cdots k_1k_0$ 。每轮加密时取右端的 64 bit 作为轮密钥, 写作 $RK_i = RK2_i \parallel RK1_i$, 其中 $RK1_i$ 用于 LiCi 左半侧的加密, $RK2_i$ 用于 LiCi 右半侧的加密。密钥编排方案如下。

$$K = k_{127}k_{126} \cdots k_1k_0 \quad (2)$$

$$RK1_i = k_{31}k_{30} \cdots k_1k_0 \quad (3)$$

$$\begin{aligned} RK2_i &= k_{63}k_{62} \cdots k_{33}k_{32} \\ K &\leftarrow K \lll 13 \end{aligned} \quad (4)$$

$$\begin{aligned} &= k_{114}k_{113} \cdots k_1k_0k_{127}k_{126} \cdots k_{115} \\ k_3k_2k_1k_0 &\leftarrow S[k_3k_2k_1k_0] \\ k_7k_6k_5k_4 &\leftarrow S[k_7k_6k_5k_4] \\ k_{63}k_{62}k_{61}k_{60} &\leftarrow [k_{63}k_{62}k_{61}k_{60}] \oplus i \end{aligned} \quad (5)$$

根据文献[8]的理论,6 轮轮密钥即可恢复 128 bit 原始密钥的全部信息。因此,本文需对连续 6 轮迭代进行差分故障攻击。

3 LiCi 算法的性质

3.1 S 盒差分性质

与文献[7]类似,本文对 LiCi 的 S 盒差分分布表进行分析。当给定输出差分经过 S 盒逆运算后,其对应的输入差分存在对应的概率。若输出差分为 0001,则输入差分为 **11,其输入差分可能为 0011、0111、1011、1111 这 4 种情况,且每种情况对应的概率均为 $P(0011)=P(0111)=P(1011)=P(1111)=4/16=1/4$ 。当输出差分为其他值时,概率计算的方法同理,如表 2 所示(表中仅列出概率非 1/16 的部分)。那么在进行密钥恢复时,利用 S 盒特殊的输入输出差分中存在的概率,可以有效地降低密钥猜测的复杂度,有助于密钥的恢复。

表 2 LiCi 的 S 盒输入/输出差分对应概率
Table 2 S-box input/output differential probability of LiCi

输入差分	**11	**1*	***1	0001
输出差分	0001	0010	0100	1***
对应概率	1/4	1/8	1/8	1/8

3.2 密钥编排性质

本文主要针对参与左半侧加密的轮密钥进行密码学分析,通过差分故障的方法对左半侧轮密钥进行恢复。而对于右半侧轮密钥,需要结合密码编排规律进行穷举猜测。对右半侧 32 bit 轮密钥进行穷举的计算复杂度较高(为 2^{32}),因此需要分析每两轮轮密钥之间的关系。本文发现 LiCi 算法连续两轮之间右半侧轮密钥存在 11 bit 的重复区间,可以有效减少对右半侧轮密钥的穷举复杂度,证明过程如下。

由式(2)~式(5)可知更新后的 128 bit 密钥为

$$\begin{aligned} K_{\text{new}} &= k_{114}k_{123} \cdots k'_{63}k'_{62}k'_{61}k'_{60} \cdots k'_7k'_6k'_5k'_4 \\ &\parallel k'_3k'_2k'_1k'_0k_{127}k_{126} \cdots k_{115} \end{aligned} \quad (6)$$

其中, k'_j 代表 k_j 更新后的值。此时第 $i+1$ 轮的轮密钥右半侧可以表示为 $RK1_{i+1} = k_{18}k_{17} \cdots k'_8k'_7k'_6 \cdots k'_1k'_0k_{127}k_{126} \cdots k_{115}$, 与 $RK1_i = k_{31}k_{30} \cdots k_0$ 进行对比,

可以发现两者都具有 $k_{18}k_{17} \cdots k_8$ 的部分。因此,两者存在 11 bit 的重复区间。

3.3 故障扩散规律

LiCi 的移位操作可以让导入的故障进行扩散,为了提高差分故障攻击分析的效率,找到合适的故障导入位置,需要对其故障扩散规律进行研究。首先考虑左半侧输出,根据图 1,左半侧 32 bit 输入在依次经过 S 盒变换、与右半侧轮密钥加密运算、移位操作后,会作为输出的左半侧 32 bit 输出。此外,会与左半侧轮密钥进行加密,并经过移位操作后,作为输出的右半侧输出。该侧的移位长度为左移 3 bit,因此若将故障注入某个半字节中,在移位操作后故障会扩散到左侧相邻的半字节中,如图 2 所示。

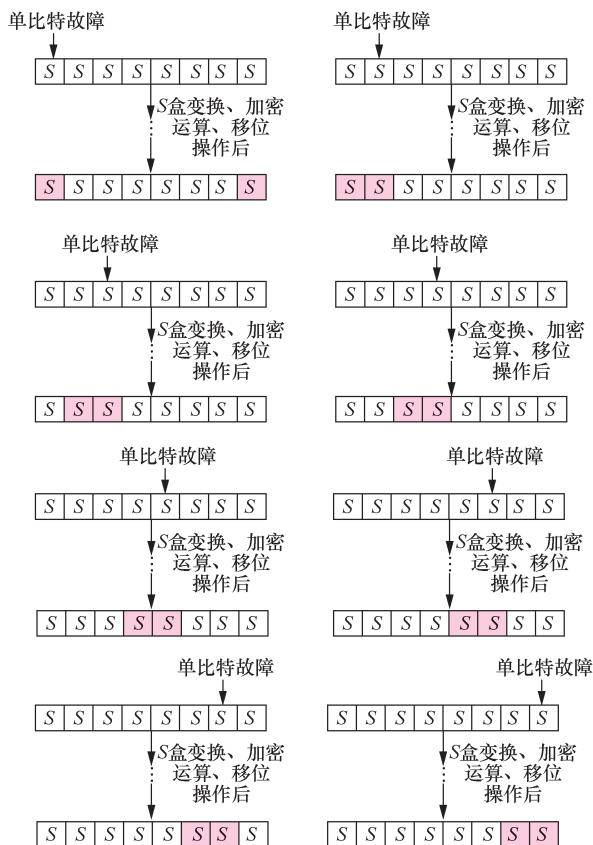


图 2 LiCi 左半侧故障扩散规律
Figure 2 Fault-diffusion law of LiCi on the left side

其次,考虑右半侧输出,因该侧的移位长度为先向左移 3 bit,再向右移 7 bit。此外,该侧加密时会与导入故障的 S 盒直接进行异或计算,因此,若将故障注入某个半字节中,在移位操作后故障会扩散到右侧相邻的半字节中,如图 3 所示。

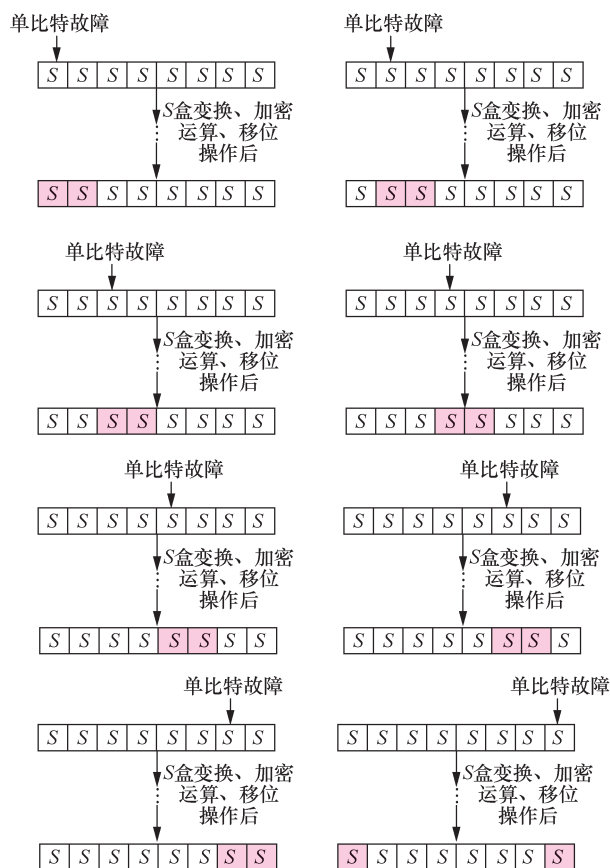


图 3 LiCi 右半侧故障扩散规律
Figure 3 Fault-diffusion law of LiCi on the right side

图 2 和图 3 中的紫色部分,为故障扩散的位置。借助这一扩散规律,可以通过正确密文与错误密文的差分推出对应该位置的密钥值。因 S 盒是对 4 bit 长度的半字节值进行代换,因此对一个 S 盒中故障的导入,1 bit、2 bit、3 bit 和 4 bit 故障产生的效果是相同的。考虑到故障长度越长,在实际操作中越难实现,本文采用单比特故障进行导入。

4 LiCi 算法的差分故障攻击

4.1 攻击基本假设

为了更好地进行分析,在此提出两点基本假设。

(1)攻击者可以在任意迭代轮次对指定的半字节中导入单比特故障,并得到正确密文 C 和错误密文 C^* ,但具体的故障值未知。

(2)对于相同的明文和轮密钥而言,攻击者可以在相同轮数和相同的半字节导入随机的单比特故障,并得到对应的错误密文 C^* 。

4.2 攻击模型和原理

为便于计算,本文首先选择在第 31 轮迭代时 S 盒操作前导入随机故障。

随机生成一组明文和密钥,通过 LiCi 算法的加密流程得到正确的密文。之后在加密过程的第 31 轮其中一个半字节导入单比特故障,结合差分故障攻击特点,获取加密过程中密文的部分特性。进一步通过加密算法的逆过程,推导出该轮部分轮密钥。在同样的密文和密钥的条件下,多次重复上述过程,直到恢复的部分轮密钥为唯一值。以相同的方式攻击第 31 轮的其他 7 个半字节,直到恢复参与加密的左半侧 32 bit 轮密钥,接着对右半侧 32 bit 轮密钥进行穷举猜测。根据 LiCi 的密钥更新方案,完整的 128 bit 原始密钥可以由 6 个连续的轮密钥推出。通过相同的步骤,分别获取第 30、29、28、27 轮的轮密钥,从而得到 LiCi 的整个原始密钥。

4.3 攻击具体步骤

攻击步骤如下。

(1) 随机生成一组明文和密钥,经过 31 轮的 LiCi 密码加密,得到正确的密文。

(2) 使用原文 B 和密钥 K 进行加密。在 LiCi 第 31 轮迭代时,在任一半字节处诱导一个随机的单比特故障,获取故障密文,并与正确的密文进行差分运算,得到两侧的差分故障密文。

$$\begin{aligned}\Delta C1_{31}^* &= C1_{31} \oplus C1_{31}^* \\ \Delta C2_{31}^* &= C2_{31} \oplus C2_{31}^*\end{aligned}\quad (7)$$

以左侧的第一个半字节为例,导入一个单比特故障,右侧会以 $(\Delta c1_{7,31}^*, 0, 0, 0, 0, 0, 0, \Delta c1_{0,31}^*)$ 的形式输出差分结果,左侧会以 $(\Delta c2_{7,31}^*, \Delta c2_{6,31}^*, 0, 0, 0, 0, 0, 0)$ 的形式输出差分结果。

(3) 通过正确密文和故障密文,找到导入故障的 S 盒和它的差分输出 ΔW_{31} 。

(4) 因只对第一个 S 盒导入故障,故 ΔW_{31} 只有一个非零半字节 $\Delta w_{7,31}$, 列出其差分方程。

$$\begin{aligned}\Delta w_{7,31} &= S(x_{7,31}) \oplus S(x_{7,31} \oplus \Delta x_{7,31}) \\ \Delta x_{7,31} &\text{有 } 2^4=16 \text{ 个可能的值,在这 16 个值中,} \\ &\text{求出满足式(8)的 } x_{7,31} \text{ 的值,再将计算出的 } x_{7,31} \\ &\text{存储在一个集合 } M \text{ 中。}\end{aligned}\quad (8)$$

(5) 根据 LiCi 的 S 盒特性,可能会有多个匹配结果,所以攻击步骤 (2) ~ 步骤 (4) 会

不断重复,直到集合 M 中只保留唯一半字节,这就是正确的 $x_{7,31}$ 。

(6) 重复步骤 (2) ~ 步骤 (5),以相同的方式攻击其他 7 个半字节,直到恢复到 32 bit 左半侧输入 X_{31} 为止。

(7) 通过下式,得到左半侧轮密钥 $RK2_{31}$ 的值。

$$RK2_{31} = S(X_{31}) \oplus (C1_{31} \lll 7) \oplus C2_{31} \quad (9)$$

(8) 此时对右半侧轮密钥 $RK1_{31}$ 进行穷举,共有 2^{32} 种可能。

(9) 同理,在 LiCi 第 30、29、28、27、26 轮进行差分故障攻击,重复步骤 (2) ~ 步骤 (8) 可以恢复左侧轮密钥 $RK2_{30}$ 、 $RK2_{29}$ 、 $RK2_{28}$ 、 $RK2_{27}$ 和 $RK2_{26}$ 。此外,因右侧连续两轮轮密钥存在 11 bit 重复区间,故对于这 5 轮右侧轮密钥的穷举复杂度均为 $2^{32-11} = 2^{21}$ 。

(10) 在获取连续 6 轮轮密钥后,根据 LiCi 密钥编排方案,推出 128 bit 原始密钥。

5 复杂度分析

本文根据故障传播特性建立的攻击方式,相较于对 LiCi 算法的密钥进行穷举的方式有一定的优化。传统的穷举密钥方法,恢复出原始密钥的复杂度为 2^{128} 。本文首先对输出差分的非零半字节所在位置的可能值进行穷举,再对右半侧轮密钥进行穷举,可以有效降低攻击所需的复杂度。

5.1 数据复杂度

一个单比特故障能恢复 4 bit 密钥,那么恢复 LiCi 一轮左半侧 32 bit 密钥最少需要 8 个单比特故障。为了恢复全部 128 bit 原始密钥,需要连续 6 轮的轮密钥,因此共需要 48 个单比特故障。

5.2 计算复杂度

首先对式(8)中非零字节 $\Delta x_{7,31}$ 进行穷举,复杂度为 2^4 。恢复一轮左半侧密钥需要穷举 8 次这样的非零字节,复杂度为 $8 \times 2^4 = 2^7$ 。此外,对第 31 轮右半侧轮密钥 $RK1_{31}$ 进行穷举,复杂度为 2^{32} 。因此,恢复第 31 轮轮密钥的计算复杂度为 $2^7 + 2^{32} \approx 2^{32}$ 。恢复第 30、29、28、27、26 轮轮密钥的复杂度均为 $2^7 + 2^{21} \approx 2^{21}$,故恢复全部 128 bit 原始密钥复杂度为 $2^{32} + 2^{21} \times 5 \approx 2^{32}$ 。

6 结束语

本文提出了一种对 LiCi 密码算法的单比特差分故障攻击方法,通过分析移位操作的规律,并结合其差分特性,将单比特故障分别导入加密流程的第 26、27、28、29、30 和 31 轮迭代,可恢复这 6 轮右侧轮密钥,再对右侧轮密钥进行穷举。理论上,完全恢复原始的 128 bit 密钥需要 48 个单比特故障,计算复杂度为 2^{32} 。结论表明, LiCi 密码算法不能很好地抵抗本文所提出的差分故障攻击。

参考文献:

- [1] HONG D, SUNG J, HONG S, et al. HIGHT: a new block cipher suitable for low-resource device[C]//CHES 2006: 46-59.
- [2] BOGDANOV A, KNUDSEN L R, LEANDER G, et al. PRESENT: an ultra-lightweight block cipher[C]//Proceeding of 9th International Workshop on Cryptographic Hardware and Embedded System. 2007: 450-466.
- [3] BANIK S, PANDEY S K, PEYRIN T, et al. Gif: a small present[C]//Proceedings of the 19th International Conference on Cryptographic Hardware and Embedded System. 2017: 321-345.
- [4] BEAULIEU R, SHORS D, SMITH J, et al. The SIMON and SPECK lightweight block ciphers[C]//Proceedings of the 52nd Annual Design Automation Conference. 2015: 175:1-175:6.
- [5] 国家商用密码管理办公室. 无线局域网产品使用的 SMS4 密码算法[EB].
Office of State Commercial Cipher Administration. Block cipher for WLAN products-SMS4[EB].
- [6] PATIL J, BANSOD G, KANT K S. LiCi: a new ultra-lightweight block cipher[C]//International Conference on Emerging Trends & Innovation in Ict. 2017: 40-45.
- [7] 韦永壮, 史佳利, 李灵琛. LiCi 分组密码算法的不可能差分分析[J]. 电子与信息学报, 2019, 41(7): 1610-1617.
WEI Y Z, SHI J L, LI L C. Impossible differential cryptanalysis of LiCi block cipher[J]. Journal of Electronics & Information Technology, 2019, 41(7):1610-1617.
- [8] 信文倩, 孙兵, 李超. LiCi 算法的基于比特积分攻击[J]. 计算机工程, 2020, 46(7).
XIN W Q, SUN B, LI C. Bit-based integral attack on LiCi[J]. Computer Engineering, 2020, 46(7).
- [9] 王红艳, 韦永壮, 刘文芬. ANU, ANU-II 和 LiCi 算法的积分区分器搜索[J]. 小型微型计算机系统, 2020, 41(7): 1470-1475.
WANG H Y, WEI Y Z, LIU W F. Integral distinguisher search of ANU, ANU-II and LiCi block ciphers[J]. Journal of Chinese Computer Systems, 2020, 41(7): 1470-1475.
- [10] BIHAM E, SHAMIR A. Differential fault analysis of secret key cryptosystems[C]//1997 Annual International Cryptology Conference. 1997: 513-525.
- [11] LUO H X, CHEN W J, MING X Y, WU Y F. General differential fault attack on PRESENT and GIFT cipher with nibble[J]. IEEE Access, 2021.
- [12] 张蕾, 吴文玲. SMS4 密码算法的差分故障攻击[J]. 计算机学报, 2006(9): 1596-1602.
ZHANG L, WU W L. Differential fault analysis on SMS4[J]. Chinese Journal of Computers, 2006(9): 1596-1602.
- [13] 王永娟, 任泉宇, 张诗怡. 轻量级分组密码 Klein 的差分故障攻击[J]. 通信学报, 2016, 37(S1): 111-115.
WANG Y J, REN Q Y, ZHANG S Y. Differential fault attack on lightweight block cipher Klein[J]. Journal on Communications, 2016, 37(S1): 111-115.
- [14] 王永娟, 张诗怡, 王涛, 等. 对 MIBS 分组密码的差分故障攻击[J]. 电子科技大学学报, 2018, 47(4): 601-605.
WANG Y J, ZHANG S Y, WANG T, et al. Differential fault attack on block cipher MIBS[J]. Journal of University of Electronic Science and Technology of China, 2018, 47(4): 601-605.
- [15] 陈伟建, 赵思宇, 邹瑞杰, 等. PRESENT 密码的差分故障攻击[J]. 电子科技大学学报, 2019, 48(6): 865-869.
CHEN W J, ZHAO S Y, ZOU R J, et al. The differential fault attack of PRESENT cipher[J]. Journal of University of Electronic Science and Technology of China, 2019, 48(6): 865-869.
- [16] LUO H X, WU Y F, CHEN W J. Differential fault attack on TWINE block cipher with nibble[C]//2020 IEEE 20th International Conference on Communication Technology (ICCT). 2020: 1151-1155.

[作者简介]



陈伟建 (1956-), 男, 浙江杭州人, 电子科技大学教授, 主要研究方向为无线与移动通信、物联网信息安全、密码学理论与技术。



罗皓翔 (1999-), 男, 四川成都人, 主要研究方向为密码学理论、区块链技术。