轻量级分组密码 LED 的相关密钥差分分析

樊 婷[1,3], 韦永壮[1,2], 武小年[2,3], 张润莲[2,3]

[1]桂林电子科技大学广西无线宽带通信与信号处理重点实验室 桂林 中国 541004
[2]桂林电子科技大学广西密码学与信息安全重点实验室 桂林 中国 541004
[3]广西高校云计算与复杂系统重点实验, 桂林 中国 541004

摘要 在 CHES2011 国际会议上, 轻量级分组密码算法 LED 被郭等人提出, 该密码算法具有硬件实现规模小, 加解密速度快等优点, 因而备受业界关注。目前设计者给出了单密钥攻击模型下 LED 算法活跃 S 盒个数的下界, 以评估其抵御经典差分密码分析的能力。然而, 相关密钥攻击模型下 LED 算法抵御差分密码分析的能力仍有待进一步解决。本文基于 LED 密码算法的结构及密钥编排特点, 结合面向字节的自动化搜索方法, 构建了适用于相关密钥差分分析的混合整形规划(MILP) 搜索模型。研究结果表明: 全轮 LED-64 至少存在 100 个活跃 S 盒, 全轮 LED-128 至少存在 150 个活跃 S 盒; 15 轮简化 LED 算法足以抵抗相关密钥差分分析。此外, 针对多种变体的 LED 密钥编排方法进行了测试, 找到了一些新的密钥编排方案, 并使 LED 算法具有最佳能力抵御相关密钥差分分析。

关键词 混合整数规划(MILP); 活跃 S 盒; 相关密钥差分分析; LED 算法
中图法分类号 TP309.7 **DOI 号** 10.19363/J.cnki.cn10-1380/tn.2020.01.04

Related-key Differential Attack on lightweight block cipher LED

FAN Ting[1,3], WEI Yongzhuang[1,2], WU Xiaonian[2,3], ZHANG Runlian[2,3]

[1] Guangxi Key Laboratory of Wireless Wideband Communication and Signal Processing, Guilin University of Electronic Technology, Guilin 541004, China
[2] Guangxi Key Laboratory of Cryptography and Information Security, Guilin University of Electronic Technology, Guilin 541004, China
[3]Guangxi Colleges and Universities Key Laboratory of cloud computing and complex systems, Guilin University of Electronic Technology, Guilin 541004, China

Abstract Lightweight block cipher LED was designed by Guo Jian et al. at CHES 2011. LED has attracted extensive attention because of its small hardware implementation, fast encryption and decryption. Currently, the lower bound of the active S-box of LED under the single key model was given by the designers. This result is helpful for evaluating the ability of resisting differential attack. However, the resistance of LED against differential attack under the related-key model appears to be an unsolved problem. In this paper, the MILP search model of related-key differential attack by basing on the structure and key schedule of LED and combining with the byte-oriented automatic search method is constructed. It is shown that there are at least 100 active S-boxes in full round LED-64, and at least 150 active S-boxes in full round LED-128. It also illustrates that the reduced 15-round of LED can resist to the related-key differential attack. In addition, a variety of variant LED key schedule are checked. Some new key schedule for LED against the related-key differential attack are investigated.

Key words mixed-integer linear programming; active S-box; related-key differential attack; LED algorithm

1 引言

随着 5G 通信的快速发展, 物存联网终端设备受到存储资源、运算资源、能耗资源等限制[1]。传统加密算法面临资源消耗快, 高实现成本等诸多问题, 轻量级密码算法应运而生。轻量级密码软硬件实现

通讯作者: 韦永壮, 博士, 教授, Email: walker_wyz@guet.edu.cn。
本课题得到国家自然科学基金(No.61572148, No.61872103); 广西重点研发计划(桂科 No.AB18281019); 广西自然科学基金(No.2018GXNSFAA294036); 广西研究生教育创新计划资助项目(No.YCBZ2018051); 桂林电子科技大学研究生科研创新项目(No.2018YJCX45)资助。
收稿日期: 2019-08-07; 修改日期: 2019-11-11; 定稿日期: 2019-12-10

效率快, 运用范围广, 在保证设备安全性的同时, 缩减了所占内存。目前, 国内外陆续提了许多轻量级分组密码算法, 如采用 SPN 结构的轻量级分组密码算法 PRESENT[2]、GIFT[3]、LED[4]; Feitesl 结构的 Lbock[5]、LiCi[6]、ESF[7]; 硬件和软件实现速度较快的 SIMON 和 SPECK[8]; 基于 bit-slice 技术且同时兼具很好的软件和硬件性能的 RECTANGLE 算法[9]。其中 LED 算法[4]是郭建等人于 2011 年提出的一种类 AES 算法[10]结构的轻量级分组密码, 其占用面积小, 节省了资源消耗, 不仅有紧凑的硬件实现, 同时保持了软件的一些友好特性, 成为当前最具实现效率的算法之一。

差分密码分析[11]和线性密码分析[12]是分析密码算法两个最基本的方法。2011 年, Mouha 等人[13]首次利用混合整数线性规划 (Mixed-Integer Linear Programming, MILP) 求得流密码算法 Enocoro128v2 和分组密码 AES 的活跃 S 盒下界。2014 年, 孙思维等人[14]将这种技术扩展到计算基于比特的密码算法活跃 S 盒下界。目前, MILP 技术在评估密码算法安全性方面取得了许多结果, 如 SM4[15]、Simeck[16]以及对现有基于 MILP 模型分析方法的改进[17]等。目前, LED 设计者分别给出了单密钥和相关密钥攻击模型下活跃 S 盒个数的下界值。2018 年, 刘波涛等人[18]求解单密钥 MILP 模型下 LED 算法的活跃 S 盒下界。然而, 相关密钥攻击模型下 LED 算法抵御差分密码分析的能力仍有待进一步解决。

本文基于 LED 密码算法的结构及密钥编排特点, 结合面向字节的自动化搜索方法, 针对不同情况利用 MILP 方法搜索其差分活跃 S 盒下界, 并对算法进行安全性评估, 得到全轮 LED-64 和 LED-128 算法活跃 S 盒的下界。结果表明: 15 轮 LED 可以抵抗相关密钥差分攻击。最后, 对算法密钥编排进行变体, 通过测试不同密钥编排情况下算法活跃 S 盒数目, 提出了新型的密钥编排方案, 使算法抵抗相关密钥差分分析的能力达到最优。

文章组织如下, 第二节给出符号说明, 并介绍 LED 算法结构及密钥编排; 第三节阐述基于 MILP 面向字节相关密钥差分分析模型; 第四节对 LED 算法进行实例应用; 第五节提出新密钥编排; 第六节进行总结。

2 预备知识

2.1 符号说明

P: 表示明文;

K_1: 表示 LED-64 算法密钥;

$K_1 \| K_2$: 表示 LED-128 算法密钥;

B_d: 表示算法的差分分支数;

R: 加密轮数;

C: 表示正确密文。

2.2 LED 算法简介

LED 算法是基于 SPN 结构的轻量级分组密码算法, 分组长度为 64 比特, 由 16 个 4 比特的半字节组成。密钥长度为 64 或 128 比特, 该算法的两个版本分别记为 LED-64 与 LED-128, 对应迭代轮数 R 为 32 轮和 48 轮, 详细内容请见参考文献[4]。

$$P=\begin{pmatrix} p_0 & p_1 & p_2 & p_3 \\ p_4 & p_5 & p_6 & p_7 \\ p_8 & p_9 & p_{10} & p_{11} \\ p_{12} & p_{13} & p_{14} & p_{15} \end{pmatrix} \tag{1}$$

算法在加密时, 经过轮密钥加(AddRoundkey)和轮函数(RoundFunction)等变换, 其中轮函数使用与 Midori-64[19]同样状态的 SPN 结构, 初始状态分布在 4×4 的矩阵中。

轮函数由轮常数加(AddConstants)、S 盒代换(SubCell)、行移位(ShiftRows)和列混淆(MixColumn)四个操作组成。连续四轮的迭代操作称为一个“步骤(step)”。假设输入明文为 P, 输出密文为 C, 经过 LED 算法加密的过程如图 1 所示。

(1) 轮密钥更新

LED 算法轮密钥不进行密钥更新, 即对于 LED-64, 初始密钥等于轮密钥; 对于 LED-128, 当轮密钥为 K_1 时, K_1 相当于初始密钥前 64 比特; 当轮密钥为 K_2 时, K_2 相当于初始密钥剩余 64 比特。

(2) 轮常数加

轮常数加是指中间状态矩阵的前两列与固定常数进行异或操作, 轮常数更新见文献[4]。

(3) S 盒代换

LED 算法使用 PRESENT 算法的 S 盒, 状态矩阵中的每个半字节经过非线性部件 S 盒后都被新的半字节替换, S 盒替换表如表 1 所示。

(4) 行移位

行移位是基半字节的移位操作, 第一行不动第二行循环左移 1 位, 第三行循环左移 2 位, 第四行循环左移 3 位, 如图 1 所示。

(5) 列混淆

列混淆是对状态矩阵的每一列进行更新, 即每一列状态矩阵与固定矩阵相乘, 更新后的状态矩阵作为下一轮的输入值。

表 1 LED 算法 S 盒

Table 1 S-box of LED algorithm

x	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
$S[x]$	C	5	6	B	9	0	A	D	3	E	F	8	4	7	1	2

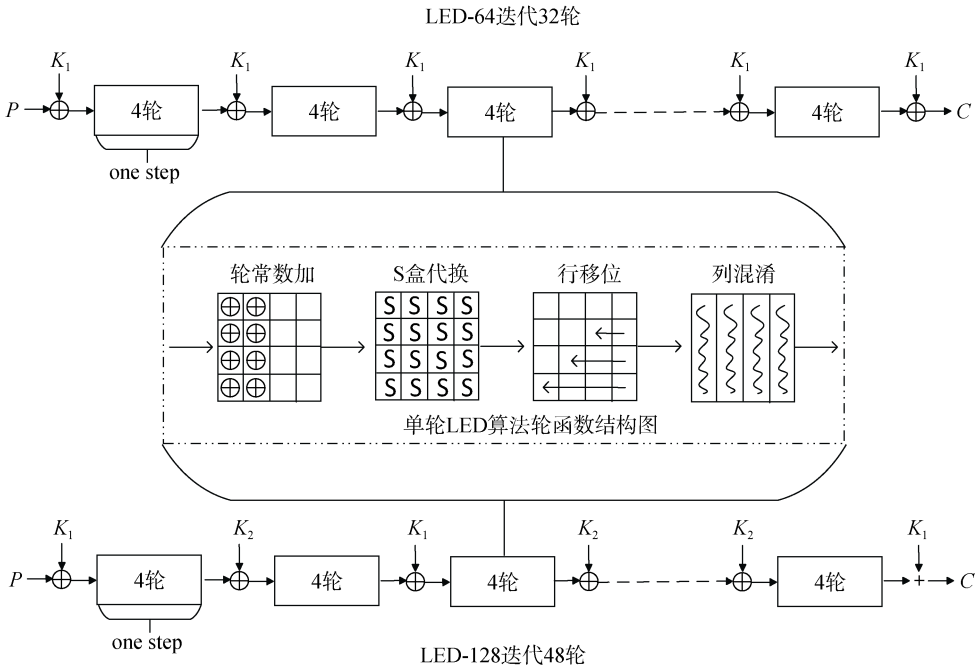

图 1 LED 算法加密的过程

Figure 1 LED algorithm encryption process

3 面向字节的 MILP 模型

线性规划(LP): 优化(最小化或最大化)线性目标函数, 受决策变量, 线性不等式的影响。当全部或部分决策变量是整数时, LP 问题称为混合整数线性规划(MILP)。下面介绍 Mouha 等人[13]2011 年提出的基于 MILP 差分分析模型。

定义 1. 假设由 n 个字节组成的串记为 $\Delta=(\Delta_0,\Delta_1,\cdots,\Delta_{n-1})$。然后, 定义 Δ 对应的输入差分向量为 $x=(x_0,x_1,\cdots,x_{n-1})$, 则:

$$x_i=\begin{cases}1, & \Delta i=0\\ 0, & \text{其他}\end{cases} \tag{2}$$

$x_i=0$, 即输入差分为 0 时, 该字节不活跃; $x_i\neq 0$, 表示输入差分非 0, 则该字节活跃。

(1) XOR 运算约束:

x_1 与 x_2 进行异或, x_3 为异或操作的结果, 将该异或操作用如下不等式来表示:

$$\begin{cases} x_1+x_2+x_3\geqslant 2d\\ d\geqslant x_1, d\geqslant x_2, d\geqslant x_3\end{cases} \tag{3}$$

其中 d 是一个虚拟变量, 取值为{0, 1}。

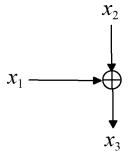

图 2 异或示意图

Figure 2 XOR diagram

(2) 线性变换约束:

这里假设线性变换 L 的输入差分向量为 $(x_0^L,x_1^L,\cdots,x_m^L)$, 其输出差分向量为 $(y_0^L,y_1^L,\cdots,y_m^L)$, 则线性变换操作 L 由如下不等式表示:

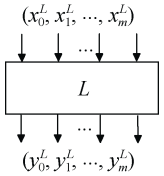

图 3 线性变换示意图

Figure 3 Schematic diagram of linear transformation

$$\begin{cases} \sum_{i}^{m-1} x_i^L + \sum_{i}^{m-1} y_i^L \geqslant B_d d' \\ d' \geqslant x_i^L, d' \geqslant y_i^L, i \in \{0,\cdots,m-1\} \end{cases} \tag{4}$$

其中 d' 是一个虚拟变量, 取值为{0, 1}; B_d 为差分分支数, 即对于非零输入差分, 在输入和输出差分中总共至少存在 B_d 个非零字。线性变换 L 的差分分支数定义为:

$$B_d(\theta) = \min_{x, x \neq x^*} \{\omega_b(x \oplus x^*) + \omega_b(\theta(x) \oplus \theta(x^*))\} \tag{5}$$

(3) 目标函数:

$$f = \min \sum x_i \tag{6}$$

4 LED 算法相关密钥 MILP 模型

相关密钥攻击是通过分析密钥扩展算法对分组密码安全性影响的分析方法, 它使用密钥调度的一些弱点, 通过研究不同密钥之间的关系对加密的影响来得到密钥信息[20]。相关密钥攻击与差分攻击结合称为相关密钥差分攻击。本节中, 我们使用前面介绍的面向字节的自动化搜索方法, 对 LED 算法建立相关密钥差分的 MILP 模型。

LED 算法共包含 5 个操作: 轮密钥加、轮常数加、S 盒代换、行移位和列混淆。其中 S 盒代换达到混淆的效果, 行移位和列混淆操作都属于线性操作, 具有扩散作用。由于 S 盒代换操作不影响活跃 S 盒位置, 所以这里只需要对轮密钥加、行移位的列混淆三个操作进行建模。LED-64 算法和 LED-128 算法使用不同长度的密钥, 下面将分别进行介绍。

4.1 LED-64 相关密钥 MILP 模型

LED-64 算法中只用到 64 比特的密钥, 因此只需考虑在密钥部分存在输入差分。下列所涉及的所有变量 x_i 都表示状态差分。

(1) 轮密钥加

轮密钥加是将输入明文的初始状态与轮密钥 K_1 进行异或操作, 得到输入值的一个中间状态, 从而为轮函数操作做准备。为方便起见, 仅使用 x 作为向量表示。假设算法一共迭代 R ($1 < R \leqslant 32$)轮, 初始密钥 $K_1 = (x_0, \cdots, x_{15})$, 明文为 $P_n = (x_{16}, \cdots, x_{31})$, 经过轮密钥加操作后的中间状态变为 $P_m = (x_{32}, \cdots, x_{47})$, 第一轮加密过程完成后的输出状态为 $P_f = (x_{48}, \cdots, x_{63})$, 并且每个 K_i, P_n, P_m, P_f 都是半字节, 其中需要注明的是 $i \in 0, \cdots, 15$, $n \in 16, \cdots, 31$, $m \in 32, \cdots, 47$, $f \in 48, \cdots, 63$。

使用刻画异或操作的不等式来描述轮密钥加操作得到如下不等式:

$$\begin{pmatrix} x_0 & x_1 & x_2 & x_3 \\ x_4 & x_5 & x_6 & x_7 \\ x_8 & x_9 & x_{10} & x_{11} \\ x_{12} & x_{13} & x_{14} & x_{15} \end{pmatrix} \oplus \begin{pmatrix} x_{16} & x_{17} & x_{18} & x_{19} \\ x_{20} & x_{21} & x_{22} & x_{23} \\ x_{24} & x_{25} & x_{26} & x_{27} \\ x_{28} & x_{29} & x_{30} & x_{31} \end{pmatrix} \tag{7}$$

$$\downarrow$$

$$\begin{pmatrix} x_{32} & x_{33} & x_{34} & x_{35} \\ x_{36} & x_{37} & x_{38} & x_{39} \\ x_{40} & x_{41} & x_{42} & x_{43} \\ x_{44} & x_{45} & x_{46} & x_{47} \end{pmatrix}$$

$$\begin{gathered} x_{16} + x_0 + x_{32} - 2d_0 \geqslant 0 \\ d_0 - x_{16} \geqslant 0 \\ d_0 - x_0 \geqslant 0 \\ d_0 - x_{32} \geqslant 0 \\ \cdots \\ x_{31} + x_{15} + x_{47} - 2d_{15} \geqslant 0 \\ d_{15} - x_{31} \geqslant 0 \\ d_{15} - x_{15} \geqslant 0 \\ d_{15} - x_{47} \geqslant 0 \end{gathered} \tag{8}$$

(2) 行移位操作

行移位是以半字节作为单位进行位置的变换, 只会改变差分变量的位置, 即活跃 S 盒的位置, 并不会产生新的差分变量, 可用如下等式表示:

$$\begin{pmatrix} x_{32} & x_{33} & x_{34} & x_{35} \\ x_{36} & x_{37} & x_{38} & x_{39} \\ x_{40} & x_{41} & x_{42} & x_{43} \\ x_{44} & x_{45} & x_{46} & x_{47} \end{pmatrix} \xrightarrow{SR} \begin{pmatrix} x_{32} & x_{33} & x_{34} & x_{35} \\ x_{37} & x_{38} & x_{39} & x_{36} \\ x_{42} & x_{43} & x_{40} & x_{41} \\ x_{47} & x_{44} & x_{45} & x_{46} \end{pmatrix} \tag{9}$$

$$\begin{gathered} x_{32} = x_{32} \\ x_{36} = x_{37} \\ x_{40} = x_{42} \\ x_{44} = x_{47} \\ \cdots \\ x_{43} = x_{41} \\ x_{47} = x_{46} \end{gathered} \tag{10}$$

(3) 列混淆

列混淆操作是有限域 $GF(2^4)$ 上矩阵的相乘。状态矩阵的列向量左乘一个 4 阶 MDS 矩阵, 状态矩阵中的元素以列为单位进行更新, 在更新的同时, 差分变量的值也会发生改变, 产生新的差分向量。其中, MDS 矩阵相当于算法的扩散层, 分支数是衡量扩散层安全性的重要指标, 它反映了扩散层扩散性的好坏, 分支数越大, 扩散层的扩散效果越好。因此, 在设计

扩散层时要求分支数尽可能大。对于 n 阶矩阵, 分支数最大为 $n+1$。利用分支数可以给出分组密码活跃 S 盒数目的界, 进一步量化密码算法对差分密码分析和线性密码分析的抵抗力。值得注意的是, 对于任何线性变换, 差分分支数和分支数相等, 但线性分支数和分支数不一定相等[21], LED 算法的差分分支数为 5。在行移位操作的基础上, 使用刻画线性变换操作的不等式约束条件来描述列混淆操作, 得到如下不等式:

$$\begin{pmatrix} x_{32} & x_{33} & x_{34} & x_{35} \\ x_{37} & x_{38} & x_{39} & x_{36} \\ x_{42} & x_{43} & x_{40} & x_{41} \\ x_{47} & x_{44} & x_{45} & x_{46} \end{pmatrix} \xrightarrow{MC} \begin{pmatrix} x_{48} & x_{52} & x_{56} & x_{60} \\ x_{49} & x_{53} & x_{57} & x_{61} \\ x_{50} & x_{54} & x_{58} & x_{62} \\ x_{51} & x_{55} & x_{59} & x_{63} \end{pmatrix} \tag{11}$$

$$\begin{gathered} x_{32}+x_{37}+x_{42}+x_{47}+x_{48}+x_{49}+x_{50}+x_{51}-5d_{16}\geqslant 0 \\ d_{16}-x_{32}\geqslant 0 \\ \cdots \\ d_{16}-x_{51}\geqslant 0 \\ \cdots \end{gathered} \tag{12}$$

$$\begin{gathered} x_{35}+x_{36}+x_{41}+x_{46}+x_{60}+x_{61}+x_{62}+x_{63}-5d_{19}\geqslant 0 \\ d_{19}-x_{35}\geqslant 0 \\ \cdots \\ d_{19}-x_{63}\geqslant 0 \end{gathered}$$

值得注意的是, LED-64 与 AES 算法不同, 在只加密一轮的情况下, 依然进行列混淆操作, 然后再次异或轮密钥。因此, 如果算法在只加密一轮, 依旧需要添加列混淆不等式约束条件, 最后再添加轮密钥异或操作的不等式约束。上述步骤建立的是相关密钥差分 MILP 模型, 需向不等式系统中添加一个额外的条件, 即 $x_0+x_1+\cdots+x_{15}\geqslant 1$, 确保在输入的初始密钥 K_1 部分至少有一个半字节存在输入差分。最后设定目标函数 $f=\min\sum x_i$, i 为经过非线性部件 S 盒的半字节状态。将产生的不等式系统放到 Gurobi 8.1.0 软件[22]中求解, 得到全轮最小差分活跃 S 盒数目, 具体结果如表 2 所示。

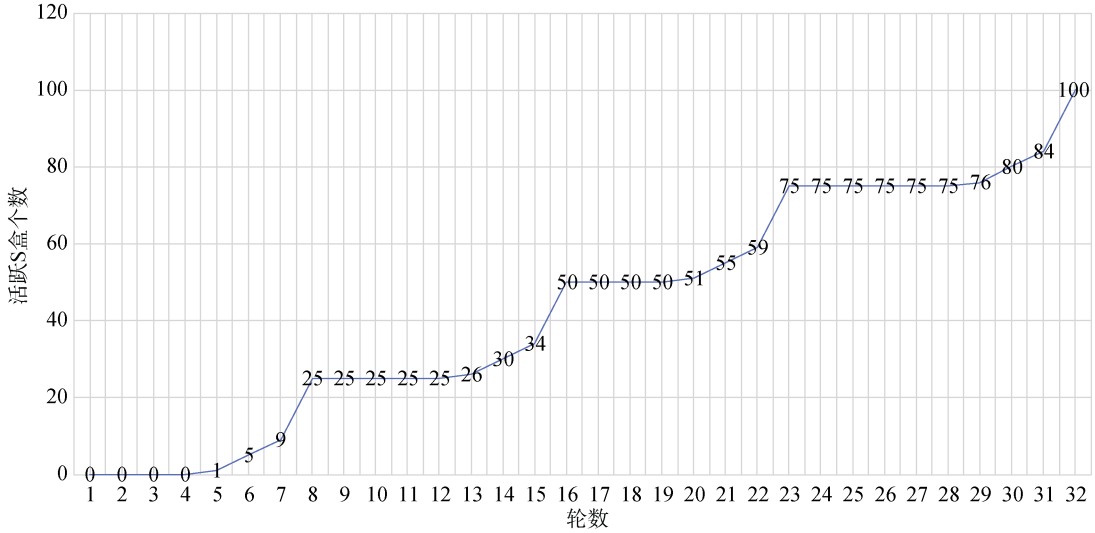

图 2　LED-64 相关密钥模型的结果

Figure 2　Results of the LED-64 related-key model

在文献[9]中 Kanda 等人, 给出了具有 SPN 结构密码算法差分攻击中最小活跃 S 盒的理论计算公式, $p_d\leqslant(2^{-2})^N<2^{-64}$。其中, p_d 是 LED 算法的最大差分概率, $N=32$ 为活跃 S 盒个数。从表 2 可以看出, 相关密钥模型中 LED-64 全轮至少有 100 个活跃 S 盒。因此, 全轮 LED-64 的最大差分概率的上限是 $(2^{-2})^{100}=2^{-200}<2^{-64}$, 可以得出结论, 全轮的 LED-64 可以抵抗相关密钥差分攻击。其 15 轮至少有 34 个活跃 S 盒, $34>N=32$, 显然对于 LED-64 来说, 15 轮迭代足以抵抗相关密钥差分攻击。

4.2　LED-128 相关密钥 MILP 模型

LED-128 算法与 LED-64 算法相比, 前者使用的密钥长度是后者的两倍, 即 128 比特的密钥 $K_1\|K_2$。LED-128 算法在输入明文后、最后一轮迭代完成之后都要添加轮密钥 K_1, 与 4.1 部分类似, 要运用同样的方法对轮密钥加、行移位和列混淆三个操作构建不等式约束条件。唯一不同的是每隔四轮要与轮密钥 K_1, K_2 交替进行异或操作。假设 $K_1=(x_0,\cdots,x_{15})$, $K_2=(x_{16},\cdots,x_{31})$, 需要分两种情况进行讨论。第一,

在密钥 K_1 部分或者 K_2 部分存在输入差分; 第二, 密钥 K_1 和 K_2 部分中都存在输入差分。

(1) 密钥 K_1 部分或 K_2 部分存在输入差分

在密钥 K_1 部分或 K_2 部分中存在输入差分, 又分为两种情况。一是密钥 K_1 中存在输入差分; 二是密钥 K_2 中存在输入差分。

①密钥 K_1 部分存在输入差分

若密钥 K_1 部分存在输入差分, 在构建轮密钥加、行移位和列混淆三个操作不等式约束条件完成后, 需向不等式系统添加一个额外条件, 即 $x_0 + x_1 + \cdots + x_{15} \geqslant 1$, 确保在输入的初始密钥 K_1 部分至少有一个半字节存在输入差分。

②密钥 K_2 部分存在输入差分

若密钥 K_2 部分存在输入差分, 同样按照①中步骤在构建三个操作的不等式约束条件, 最后添加一个额外条件, 此时额外条件应确保在输入的初始密钥 K_2 部分至少有一个半字节存在输入差分, 即 $x_{16} + x_{17} + \cdots + x_{31} \geqslant 1$。

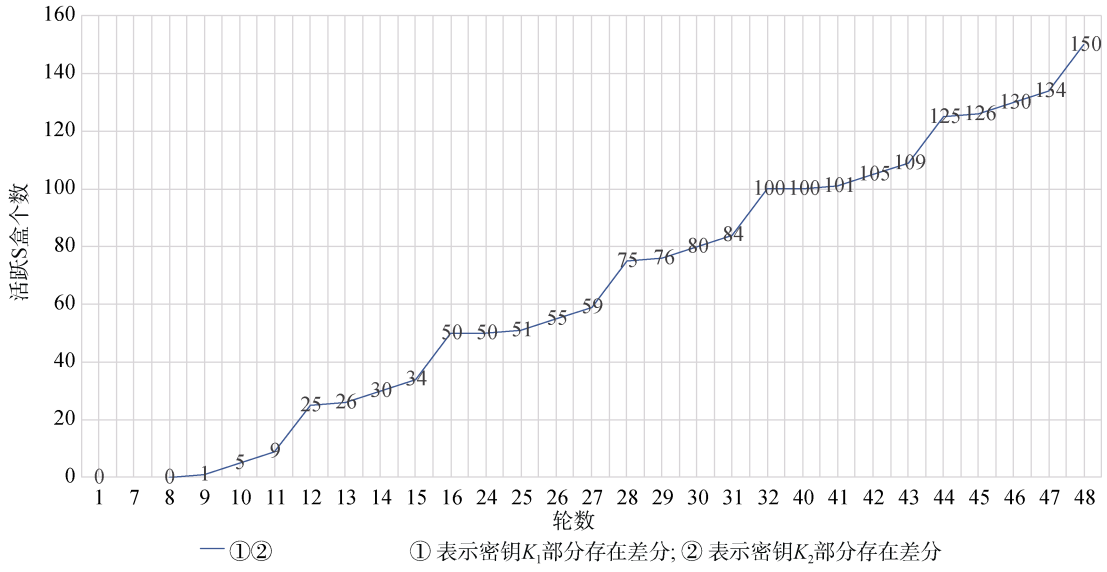

图 3 LED-128 相关密钥模型的结果

Figure 3 Results of the LED-128 related-key model

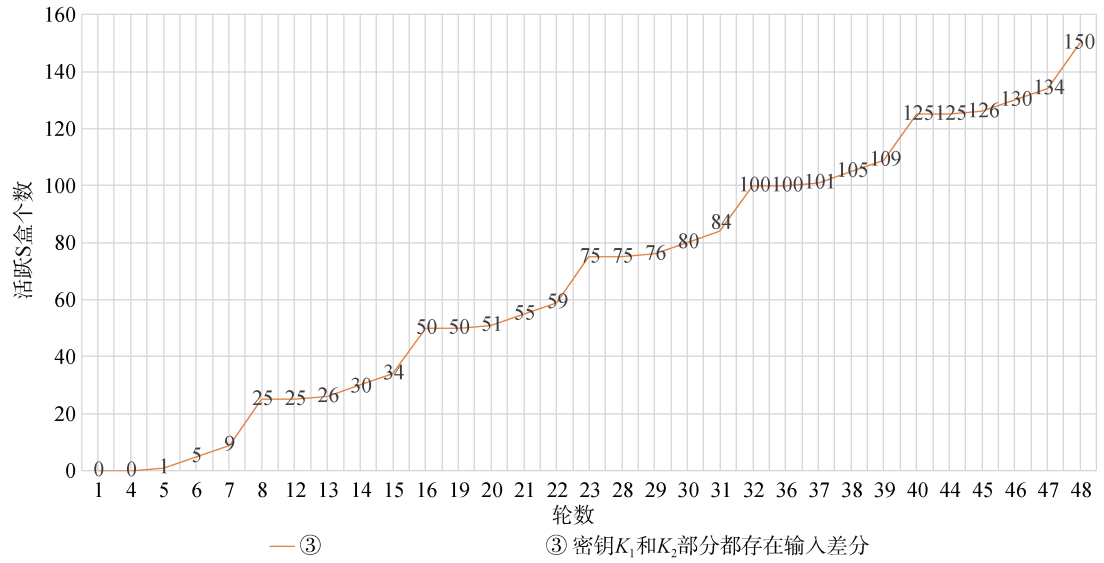

表 4 LED-128 相关密钥模型的结果

Figure 4 Results of the LED-128 related-key model

(2) 密钥 K_1 和 K_2 部分都存在输入差分

若密钥 K_1 和 K_2 部分都存在输入差分, 按照①中步骤在构建三个操作的不等式约束后, 需向不等式系统添加两个额外条件 $x_0+x_1+...+x_{15}\geqslant 1$ 和 $x_{16}+x_{17}+...+x_{31}\geqslant 1$, 保在输入的初始密钥 K_1 和 K_2 部分都至少有一个半字节存在输入差分。

值得注意的是, LED-128 算法只加密一轮的情况与 LED-64 类似, 同样需要添加列混淆操作, 最后再次与 K_1 进行异或操作; 加密四轮时, 没有与 K_2 进行异或操作, 也是与 K_1 再次进行异或操作完成加密过程。因此, 若 LED-128 只加密一轮, 不等式约束条件与 LED-64 相同; 若算法加密四轮, 添加两次与 K_1 进行异或操作的不等式约束即可, 不产生与 K_2 进行异或操作的不等式约束条件。这里讨论的三种情况, 实际是添加额外约束条件的不同。由于经过非线性部件 S 盒的半字节状态是确定的, 所以可设定同一个目标函数 $f=\min\sum x_i$, i 为经过非线性部件 S 盒的半字节状态, 求解结果即为相关密钥模型下 LED-128 算法最少差分活跃 S 盒的数目。

在相关密钥模型下获得三种情况全轮 LED-128 最少差分活跃 S 盒数目, 结果见表 3 和表 4。前两种情况每轮所测结果都一致, 而三种情况得到全轮差分活跃 S 盒的下界均为 150。因此, 全轮 LED-128 的最大差分概率的上限是 $(2^{-2})^{150}=2^{-300}<2^{-64}$。可以出, 全轮的 LED-128 对于相关密钥差分攻击是足够安全的。结果表明: 其 15 轮至少有 34 个活跃 S 盒, $34>N=32$。所以, 在这三种情况下, 15 轮 LED-128 足以抵抗相关密钥差分攻击。

5 新型密钥编排

5.1 结果对比

在相关密钥模型下进行安全性分析主要是考虑密钥编排对安全性的影响。本节通过对已有密钥编排作一些变体, 提出一种新型的密钥编排方案。表 5 列出了 7 种变体密钥编排情况的测试结果。其中, 定义如下:

(1) $K_1(4)\to K_1,K_2(4)\to K_2(4)$: 表示每隔 4 轮交替异或 K_1、K_1,K_2、K_2;

(2) $K_1(4)\to K_2(4)\to K_1,K_2(4)$: 表示每隔 4 轮交替异或 K_1、K_2、K_1,K_2;

(3) $K_1(4)\to K_2(4)\to K_2,K_1(4)$: 表示每隔 4 轮交替异或 K_1、K_2、K_2,K_1;

(4) $K_1,K_2(4)\to K_1(4)\to K_2(4)$: 表示每隔 4 轮交替异或 K_1,K_2、K_1、K_2;

(5) $K_1\to K_3(4)\to K_4(4)\to K_2$: 明文异或白化密钥 K_1, 每隔 4 轮交替异或 K_3,K_4, 最后异或白化密钥 K_2;

(6) $K_3\to K_4(4)\to K_3(4)\to K_1$: 明文异或白化密钥 K_3, 每隔 4 轮交替异或 K_4,K_3, 最后异或白化密钥 K_1;

(7) $K_3\to K_4(4)\to K_3(4)\to K_2$: 明文异或白化密钥 K_3, 每隔 4 轮交替异或 K_4,K_3, 最后异或白化密钥 K_2;

若初始密钥 K_1, K_2 为:

$$K_1:\begin{bmatrix}k_0&k_1&k_2&k_3\\k_4&k_5&k_6&k_7\\k_8&k_9&k_{10}&k_{11}\\k_{12}&k_{13}&k_{14}&k_{15}\end{bmatrix}\tag{13}$$

$$K_2:\begin{bmatrix}k_{16}&k_{17}&k_{18}&k_{19}\\k_{20}&k_{21}&k_{22}&k_{23}\\k_{24}&k_{25}&k_{26}&k_{27}\\k_{28}&k_{29}&k_{30}&k_{31}\end{bmatrix}\tag{14}$$

这里假设 $K_3=(k_0,k_1,\cdots,k_7,k_{16},k_{17},\cdots,k_{23})$, $K_4=(k_8,k_9,\cdots,k_{15},k_{24},k_{25},\cdots,k_{31})$, 即:

$$K_3:\begin{bmatrix}k_0&k_1&k_2&k_3\\k_4&k_5&k_6&k_7\\k_{16}&k_{17}&k_{18}&k_{19}\\k_{20}&k_{21}&k_{22}&k_{23}\end{bmatrix}\tag{15}$$

$$K_4:\begin{bmatrix}k_8&k_9&k_{10}&k_{11}\\k_{12}&k_{13}&k_{14}&k_{15}\\k_{24}&k_{25}&k_{26}&k_{27}\\k_{28}&k_{29}&k_{30}&k_{31}\end{bmatrix}\tag{16}$$

5.2 新型密钥编排结构

从表 5 可知对于任何变体密钥编排, 当 K_1 和 K_2 部分都存在差分时, 活跃 S 盒的数目大于等于差分存在于 K_1 或 K_2 部分的情况。变体(2)与(3)表明当异或密钥 K_1,K_2 时, 异或 K_1 和 K_2 的次序会影响活跃 S 盒下界; (1)(2)(4)表明异或 K_1、K_2、K_1,K_2 的次序同样影响活跃 S 盒下界。与(6)和(7)相比, (1)(5)两种密钥变体测得的活跃 S 盒相对较少, 表明这两种结构不能更好的保证算法安全性; 对于 LED 算法来说最好的变体结构是当密钥编排为 $K_3\to K_4(4)\to K_3(4)$

$\rightarrow K_1$(或K_2), 如图 4 所示。此时, 三种情况下测得的结果可同时达到 150。这种变体密钥结构与设计者给出的密钥编排相比, 二者抵抗相关密钥差分分析的能力都达到最佳。

表 5 LED-128 变体密钥结构的活跃 S 盒数量

Table 5 Number of active S-box for LED-128 variant key structure

LED-128	存在差分的位置		
变体密钥编排	K_1	K_2	K_1, K_2
$K_1(4) \rightarrow K_1, K_2(4) \rightarrow K_2(4)$	58	58	58
$K_1(4) \rightarrow K_2(4) \rightarrow K_1, K_2(4)$	58	83	83
$K_1(4) \rightarrow K_2(4) \rightarrow K_2, K_1(4)$	54	79	83
$K_1, K_2(4) \rightarrow K_1(4) \rightarrow K_2(4)$	83	58	83
$K_1 \rightarrow K_3(4) \rightarrow K_4(4) \rightarrow K_2$	104	104	104
$K_3 \rightarrow K_4(4) \rightarrow K_3(4) \rightarrow K_1$	150	150	150
$K_3 \rightarrow K_4(4) \rightarrow K_3(4) \rightarrow K_2$	150	150	150

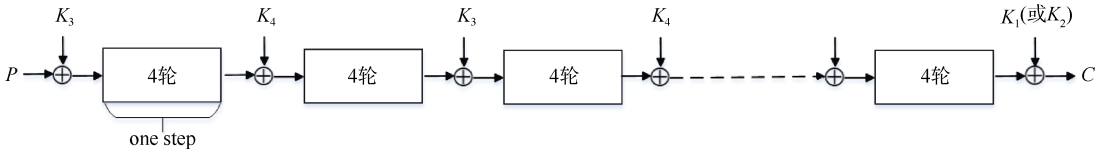

图 4 LED-128 变体结构

Figure 4 LED-128 variant structure

6 总结

本文基于LED密码算法的结构及密钥编排特点, 结合面向字节的自动化搜索方法, 针对 LED 提出一种在相关密钥模型下基于字节搜索活跃 S 盒下界的方法。运用该模型, 首先搜索到 LED-64 算法全轮最少有 100 个活跃 S 盒。其次, 对 LED-128 算法的三种情况进行搜索, 得到全轮最少有 150 个活跃 S 盒。这个结果与 LED 算法设计者给出的活跃 S 盒下界理论值一致, 从而验证了本文提出的面向字节相关密钥MILP模型是正确的, 该模型对任何基于字节且密钥值不进行固定的密码算法进行安全性评估都是有效的。在此基础上, 证明了 15 轮 LED 算法可以抵抗相关密钥差分攻击。最后, 针对不同种类的变体密钥编排对算法进行测试, 提出新的密钥编排方案, 与设计者给出的密钥编排相比, 两者抵抗相关密钥差分分析的能力都达到最优。下一步工作中, 将使用自动化工具搜索高概率的差分特征, 提出一种通用的面向字节搜索差分特征的MILP模型, 应用于更多的密码算法分析中。

参考文献

[1] Wu C K. An Overview on the Security Techniques and Challenges of the Internet of Things[J]. *Journal of Cryptologic Research*, 2015, 2(1): 40-53.
(武传坤. 物联网安全关键技术与挑战[J]. 密码学报, 2015, 2(1): 40-53.)

[2] Bogdanov A, Knudsen L R, Leander G, et al. PRESENT: An Ultra-Lightweight Block Cipher[C]. *International workshop on cryptographic hardware and embedded systems(CHES),* 2007: 450-466.

[3] Banik S, Pandey S K, Peyrin T, et al. GIFT: A Small Present[M]. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2017: 321-345.

[4] Guo J, Peyrin T, Poschmann A, et al. The LED Block Cipher[M]. Cryptographic Hardware and Embedded Systems – CHES 2011. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 326-341.

[5] Wu W L, Zhang L. LBlock: A Lightweight Block Cipher[M]. Applied Cryptography and Network Security. Berlin, Heidelberg: Springer Berlin Heidelberg, 2011: 327-344.

[6] Patil J, Bansod G, Kant K S. LiCi: A New Ultra-lightweight Block Cipher[C]. *2017 International Conference on Emerging Trends & Innovation in ICT (ICEI)*, February 3-5, 2017. Pune, India. Piscataway, NJ: IEEE, 2017: 40-45.

[7] Liu X, Zhang W Y, Liu X Z, et al. Eight-sided Fortress: A Lightweight Block Cipher[J]. *The Journal of China Universities of Posts and Telecommunications*, 2014, 21(1): 104-128.

[8] Beaulieu R, Treatman-Clark S, Shors D, et al. The SIMON and

SPECK lightweight block ciphers[C]. *52nd ACM/EDAC/IEEE Design Automation Conference (DAC)*, 2015: 1-6.

[9] Zhang W T, Bao Z Z, Lin D D, et al. RECTANGLE: A Bit-slice Lightweight Block Cipher Suitable for Multiple Platforms[J]. *Science China Information Sciences*, 2015, 58(12): 1-15.

[10] Daemen J, Rijmen V. The Design of Rijndael: AES-the Advanced Encryption Standard[M]. Springer Science & Business Media, 2013.

[11] Biham E, Shamir A. Differential Cryptanalysis of DES-like Cryptosystems[J]. *Journal of Cryptology*, 1991, 4(1): 3-72.

[12] Matsui M. Linear Cryptanalysis Method for DES Cipher[M]. Advances in Cryptology — EUROCRYPT '93. Berlin, Heidelberg: Springer Berlin Heidelberg, 1994: 386-397.

[13] Mouha N, Wang Q J, Gu D W, et al. Differential and Linear Cryptanalysis Using Mixed-Integer Linear Programming[M]. Information Security and Cryptology. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012: 57-76.

[14] Sun S W, Hu L, Wang P, et al. Automatic Security Evaluation and (Related-key) Differential Characteristic Search: Application to SIMON, PRESENT, LBlock, DES(L) and other Bit-Oriented Block Ciphers[M]. Lecture Notes in Computer Science. Berlin, Heidelberg: Springer Berlin Heidelberg, 2014: 158-178.

[15] Zhang J, Wu W L, Zheng Y F. Security of SM4 Against (Related-Key) Differential Cryptanalysis[M]. Information Security Practice and Experience. Cham: Springer International Publishing, 2016: 65-78.

[16] Wang S H. Related-key Differential Analysis of Round-reduced Simeck[C]. *2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC)*, February 10-11, 2017. Palladam, Tamilnadu, India. Piscataway, NJ: IEEE, 2017: 834-838.

[17] Zhou C, Zhang W, Ding T, et al. Improving the MILP-based Security Evaluation Algorithms against Differential Cryptanalysis Using Divide-and-Conquer Approach. *IACR Cryptology ePrint Archive*, 2019.

[18] B.T. Liu, C.G. Peng, R.X. Wu. Based on MILP method for security analysis of LED[J]. *Application Research of Computers*, 2018, 37(2). (刘波涛, 彭长根, 吴睿雪等, "基于 MILP 方法的 LED 密码安全性分析", *计算机应用研究*, 2018, 37(2).)

[19] Banik S, Bogdanov A, Isobe T, et al. Midori: A Block Cipher for Low Energy[M]. Advances in Cryptology – ASIACRYPT 2015. Berlin, Heidelberg: Springer Berlin Heidelberg, 2015: 411-436.

[20] Biham E. New Types of Cryptanalytic Attacks Using Related Keys[J]. *Journal of Cryptology*, 1994, 7(4): 229-246.

[21] W.L. Wu, D.G. Feng, W.T. Zhang. Design and Analysis of Block Ciphers (2nd Edition)[M]. Tsinghua University Press, 2009. (吴文玲, 冯登国, 张文涛. 分组密码的设计与分析(第 2 版)[M]. 清华大学出版社, 2009.)

[22] Optimization-Gurobi:Gurobi optimizer reference manual. http://www.gurobi.cn/, 2018.

樊婷 于 2017 年在太原科技大学计算机科学与技术专业获得工学学士学位。现在桂林电子科技大学计算机科学与技术专业攻读硕士学位。研究领域为信息安全。研究兴趣包括: 分组密码算法。Email: 1294563809@qq.com

韦永壮 于 2009 年在西安电子科技大学密码学专业获得军事学博士学位。现任桂林电子科技大学教授。研究领域为信息安全。研究兴趣包括加密芯片侧信道攻击防御、密码算法设计与分析、网络安全协议分析等。Email: walker_wyz@guet.edu.cn

武小年 于 2004 年在国防科学技术大学计算机科学与技术专业获得工学硕士学位。现任桂林电子科技大学副教授。研究领域为信息安全、分布式计算。Email: xnwu@guet.edu.cn

张润莲 于 2010 年在西安交通大学计算机科学与技术专业获得工学博士学位。现任桂林电子科技大学副教授。研究领域为信息安全、分布式计算。Email: zhangrl@guet.edu.cn