密码学报 ISSN 2095-7025 CN 10-1195/TN
Journal of Cryptologic Research,2015,2(1):54–65

E-mail: jcr@cacrnet.org.cn
http://www.jcr.cacrnet.org.cn
Tel/Fax: +86-10-81033101

19 轮 RECTANGLE-80 的相关密钥差分分析*

单进勇[1,2], 胡 磊[1,2], 宋 凌[1,2], 孙思维[1,2], 马小双[1,2]

1. 中国科学院信息工程研究所 信息安全国家重点实验室, 北京 100093
2. 中国科学院数据与通信保护研究教育中心, 北京 100093

通讯作者: 胡磊, E-mail: hu@is.ac.cn

摘 要: RECTANGLE 是最近提出来基于 bit-slice 技术的可在多个平台快速实现的轻量级分组密码. 它采用的是 SPN 结构, 分组长度为 64 比特, 密钥长度为 80 或 128 比特, 迭代轮数为 25 轮. 到目前为此, 针对 RECTANGLE 算法的分析很少, 其中包括算法设计者给出的 18 轮差分攻击. 对于特定的输入、输出和轮子密钥差分, 本文找出了所有活跃 S 盒个数为 26-30 的 15 轮相关密钥差分特征, 总的差分概率为 $2^{-60.5}$. 利用这些差分特征, 我们将相应的差分区分器分别向前和向后扩展两轮, 提出了 19 轮的相关密钥差分攻击, 其中数据复杂度为 2^{62}, 时间复杂度为 2^{70}, 内存复杂度为 2^{72}. 数据和时间复杂度都低于设计者给出的 18 轮攻击.

关键词: RECTANLE 分组密码; 混合整数规划; 相关密钥差分攻击
中图法分类号: TP309.7 **文献标识码**: A DOI: 10.13868/j.cnki.jcr.000060

中文引用格式: 单进勇, 胡磊, 宋凌, 孙思维, 马小双. 19 轮 RECTANGLE 的相关密钥差分攻击[J]. 密码学报, 2015, 2(1): 54–65.
英文引用格式: Shan J Y, Hu L, Song L, Sun S W, Ma X S. Related-key differential attack on 19-round reduced RECTANGLE-80[J]. Journal of Cryptologic Research, 2015, 2(1): 54–65.

Related-Key Differential Attack on 19-Round Reduced RECTANGLE-80

SHAN Jin-Yong[1,2], HU Lei[1,2], SONG Ling[1,2], SUN Si-Wei[1,2], MA Xiao-Shuang[1,2]

1. State Key Laboratory of Information Security, Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093, China
2. Data Assurance and Communication Security Research Center, Chinese Academy of Sciences, Beijing 100093, China

Corresponding author: HU Lei, E-mail: hu@is.ac.cn

Abstract: RECTANGLE is a newly proposed lightweight block cipher which allows fast implementations for multiple platforms by using bit-slice techniques. It is an iterative 25-round SPN structured block cipher with a 64-bit block size and a 80-bit or 128-bit key size. So far, there are few results about the analysis of the cipher, including an attack proposed by the designers themselves on the 18-round reduced version. In this paper, we found all the 15-round differential characteristics with 26-30 active S-boxes for some specific input, output and round subkey differences, the overall probability of differential is $2^{-60.5}$. Based on these differential characteristics, we extend the corresponding differential distinguisher to 2 rounds backward and forward respectively, and propose an attack on the 19-round reduced RECTANGLE-80 with data complexity of 2^{62} plaintexts, time

* 基金项目: 国家重点基础研究发展项目(973 计划)(2013CB834203); 国家自然科学基金项目(61472417, 61402469, 61472415)
收稿日期: 2014-12-05 定稿日期: 2015-01-29

complexity of about 2^{70} encryptions and memory complexity of 2^{72}. These data and time complexities are lower than those of the designers for the 18-round reduced RECTANGLE-80.
Key words: RECTANGLE block cipher; mixed-integer linear programming; related-key differential attack

1 引言

轻量级分组密码在诸如无线射频识别、无线传感器等资源受限的设备上有着广泛的应用. 近年来, 人们设计了大量的轻量级分组密码, 如 PRESENT[1]、LED[2]、LBlock[3]、PRINCE[4]和 Zorro[5]等. 此外, 还有美国国家安全局公布的两个算法 SIMON 和 SPECK[6]. 最近, 张文涛等人提出了一个基于 bit-slice 技术的轻量级分组密码 RECTANGLE[7]. 它采用的是 SPN 结构, 分组长度为 64 比特, 密钥长度为 80 或 128 比特, 迭代轮数为 25 轮, 可以在多种平台快速实现. 它的硬件实现在 0.13μm 工艺下具有相当高的吞吐量, 见下一节的表 1.

安全性对密码算法至关重要, 一个新的算法必须抵抗所有已知的攻击方法. 差分[8]和线性[9]分析是两个最基本、最有效的分析方法. 差分分析方法有很多变种, 如相关密钥差分分析[10]、截断差分攻击[11]和不可能差分攻击[12]等. 本质上讲, 这些差分攻击都是考察被攻击算法的差分特性.

一个高概率的差分特征可以用来构造区分器或恢复部分密钥. 对于 SPN 结构的分组密码, 一条差分特征的概率可以用该条特征的活跃 S 盒的个数来刻画. 因此, 计算活跃 S 盒的个数是一个有意义的问题. 一方面, 如果活跃 S 盒个数的极小值足够大, 那么我们可以证明这个算法能抵抗差分攻击; 另一方面, 如果能够找到一条活跃 S 盒个数很少的差分特征, 那么这条特征就可以用来构造区分器.

对面向字节的分组密码, Mouha 等人把计算活跃 S 盒个数的问题转变成一个混合整数规划的求解问题[13]. 他们用一系列整数环上的等式和不等式来刻画特征为 2 的有限域上的运算, 如异或运算和线性变换. 用这样的方法, 他们证明了 Enocoro-128v2 可以抵抗差分和线性攻击. 后来, 孙思维等人找到了在整数环上刻画 S 盒的方法, 并把 Mouha 等人的方法推广到面向比特的分组密码[14,15]. 在文献[15]中, 他们证明了 PRESENT-80 能够抵抗标准相关密钥差分攻击, 同时也证明了全轮 LBlock 的相关密钥差分特征的概率最大值小于等于 2^{-56}. 最近, 利用 Mouha 和孙思维等人的方法, 乔珂欣等人证明了 FOX 可以抵抗单密钥差分攻击[16], 马小双等人证明了 18 轮的 MIBS 可以抵抗单密钥差分攻击, 39 轮可以抵抗相关密钥差分攻击[17].

在 RECTANGLE 的设计文档中, 设计者给出了一些分析结果[7]. 他们详细地描述了怎样通过两条具有相同输入和输出差分的 14 轮差分特征来攻击 18 轮的 RECTANGLE, 其时间、数据和内存复杂度分别为 $2^{78.69}$, 2^{64} 和 2^{72}. 对于其他攻击方法, 设计者们指出线性特征的相关系数偏差最大为 2^{-37}, 同时宣称 RECTANGLE 可以抵抗统计饱和攻击[18]、不可能差分攻击[12]、积分攻击[19]和密钥编排攻击[10,20]. 最近, Selvam 等人还给出了在无防护状态下的 RECTANGLE 的差分能量攻击[21].

本文中, 我们利用 Mouha 和孙思维等人的方法找到了一条活跃 S 盒个数为 26 的 15 轮相关密钥差分特征, 并且它的输入、输出和轮密钥差分的汉明重量都很小. 固定这些输入、输出和轮子密钥差分, 我们找出活跃 S 盒个数为 26-30 的所有 15 轮差分特征, 一共有 1254 条, 它们的总概率为 $2^{-60.5}$. 利用这些差分特征, 我们将相应的差分区分器分别向前和向后扩展两轮, 给出了 19 轮 RECTANGLE 的攻击, 其中数据复杂度为 2^{62}, 时间复杂度为 2^{70}, 内存复杂度为 2^{72}. 这个攻击方法的困难之处在于, 前后扩展的 2 轮中有很多活跃的 S 盒. 如果使用一般的方法, 那么计数器的个数就会大于 2^{80}. 我们采取的方法是, 尽可能地猜测第 0 轮的子密钥, 使得第 1 轮、第 18 轮和第 19 轮的子密钥可以通过已猜测的子密钥得到. 从而, 计数器的个数减少到 2^{72} 个.

本文组织如下: 第 2 节简要描述 RECTANGLE 分组密码; 第 3 节介绍 Mouha 和孙思维等人的方法; 第 4 节给出 19 轮 RECTANGLE-80 的相关密钥差分攻击; 最后一节为总结.

2 RECTANGLE 简介

在本节中, 我们简要地描述 RECTANGLE 分组密码, 具体内容请读者参阅文献[7].

RECTANGLE 分组密码是一个基于 bit-slice 技术的具有 SPN 结构的轻量级分组密码. 它的分组长度为 64 比特, 密钥长度为 80 或 128 比特. 这里我们仅考虑 80 比特的情形. 为了表述的方便, 我们引入一些记号. 设 $B=\left(b_{4l-1},\cdots,b_{3l},\cdots,b_{l-1},b_0\right)$ 是一个 $4l$ 比特的分块, 我们总是把它看成一个 $4\times l$ 的矩形

$$\begin{bmatrix} b_{l-1} & b_{l-2} & \cdots & b_2 & b_1 & b_0 \\ b_{2l-1} & b_{2l-2} & \cdots & b_{l+2} & b_{l+1} & b_l \\ b_{3l-1} & b_{3l-2} & \cdots & b_{2l+2} & b_{2l+1} & b_{2l} \\ b_{4l-1} & b_{4l-2} & \cdots & b_{3l+2} & b_{3l+1} & b_{3l} \end{bmatrix}$$

B 的第 j 列就是这个矩形的第 j 列, 记为 $B^{(j)}=\left(b_j,b_{j+l},b_{j+2l},b_{j+3l}\right)$, 其中 $0\leqslant j\leqslant l-1$. 用 $B^{(i,j)}$ 表示矩形的第 i 行第 j 列, 即 $B^{(i,j)}=b_{j+il}$. 例如, 一个 64 比特的明文或中间状态 $\left(w_{63},w_{62},\cdots,w_0\right)$ 均可以看成 4×16 的矩形

$$\begin{bmatrix} w_{15} & w_{14} & \cdots & w_2 & w_1 & w_0 \\ w_{31} & w_{30} & \cdots & w_{18} & w_{17} & w_{16} \\ w_{47} & w_{46} & \cdots & w_{34} & w_{33} & w_{32} \\ w_{63} & w_{62} & \cdots & w_{50} & w_{49} & w_{48} \end{bmatrix}$$

同样地, 80 比特的主密钥或者轮子密钥 $\left(k_{79},k_{78},\cdots,k_0\right)$ 可以看成 4×20 的矩形

$$\begin{bmatrix} k_{19} & k_{18} & \cdots & k_2 & k_1 & k_0 \\ k_{39} & k_{38} & \cdots & k_{22} & k_{21} & k_{20} \\ k_{59} & k_{58} & \cdots & k_{42} & k_{41} & k_{40} \\ k_{79} & k_{78} & \cdots & k_{62} & k_{61} & k_{60} \end{bmatrix}$$

RECTANGLE 分组密码包含了 25 轮的迭代, 每次迭代都包含三个步骤: 异或轮密钥 AddRoundkey、S 盒变换 SubColumn 和行移位 ShiftRow, 见图 1.

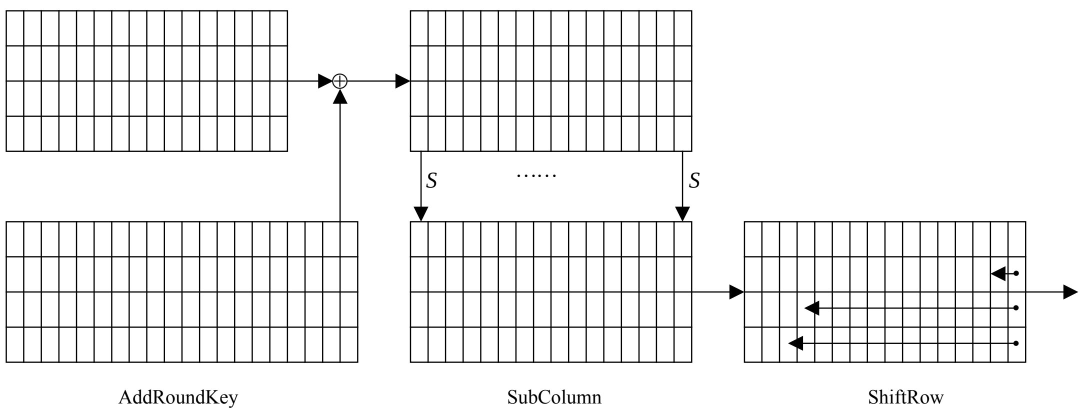


图 1 RECTANGLE 的轮函数
Figure 1 Round transformation of RECTANGLE

在第一步 AddRoundKey 中, 用轮子密钥的最右边 64 比特和明文或中间状态异或. 第二步 SubColumn

中用到的 S 盒是 4 比特到 4 比特的置换, 具体描述如下(十六进制):

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

第三步 ShiftRow 中, 第 0 行不移位, 第 1 行循环左移 1 位, 第 2 行循环左移 12 位, 第 3 行循环左移 13 位. 经过 25 轮迭代后, 再异或一个轮子密钥.

密钥的编排也包含三个步骤, 见图 2. 其中, 用到的 S 盒和轮函数中的一样, 第 0、1、2 和 3 行分别循环左移 7、9、11 和 13 位. 最后一步异或的轮常数 $\mathrm{RC}[i]$ ($0\leqslant i\leqslant 24$)是由初始状态为 $\mathrm{RC}[0]=(0,0,0,0,1)$ 的 5 级线性反馈移位寄存器生成的, 第 i 轮的轮常数 $\mathrm{RC}[i]=(r_{i,4},\cdots,r_{i,1},r_{i,0})=(r_{i-1,3},\cdots,r_{i-1,0},r_{i-1,4}\oplus r_{i-1,2})$.

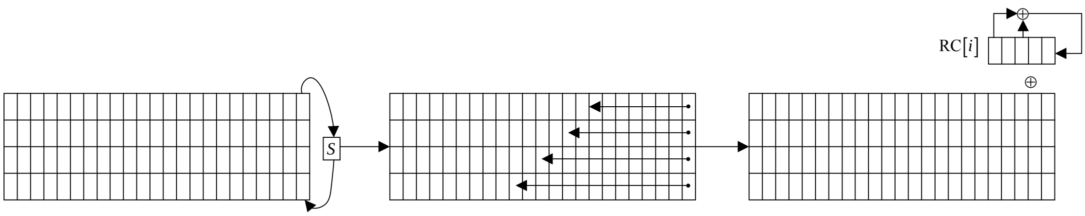


图 2 RECTANGLE 的密钥编排
Figure 2 Key schedule of RECTANGLE

在 RECTANGLE 密码的设计文档中, 设计者们给出了该算法的硬件实现, 也给出了和其他算法的比较, 见表 1. 通过表 1 可以发现, 在 0.13μm 工艺下, RECTANGLE 具有相当高的吞吐量.

表 1 几种轻量级分组密码硬件实现的比较
Table 1 Comparison with some light weight block cipher implementations

	密钥长度	分组长度	吞吐量(Kbps)	工艺(μm)	面积(GE)
RECTANGLE-80	80	64	246	0.13	1467
RECTANGLE-128	128	64	246	0.13	1787
PRESENT[22]	80	64	200	0.18	1570
AES-128[23]	128	128	12.4	0.35	3400
DESXL[24]	184	64	44.4	0.18	2168

在 RECTANGLE 密码的设计文档中, 设计者们给出了一些安全性的分析结果[7]. 除此之外, Selvam 等人也给出了对 RECTANGLE 的差分能量攻击[21]. 在本文中, 我们将给出 19 轮 RECTANGlE-80 的相关密钥差分分析.

3 基于 MILP 的方法简介

本节中, 我们介绍由 Mouha 和孙思维等人提出的基于 MILP 的方法. 所谓混合整数规划(MILP)问题, 就是部分或全部变量为整数的线性规划问题, 具体描述如下:

MILP: 设目标函数为 $c_1x_1+c_2x_2+\cdots+c_nx_n$, 约束条件为 $Ax\leqslant b$, 其中 $(c_1,c_2,\cdots,c_n)\in R^n$, $A\in R^{m\times n}$, $b\in R^m$. MILP 问题是指: 寻找向量 $x\in Z^k\times R^{n-k}\subset R^n$ 满足约束条件, 且使目标函数为最小值或最大值.

Mouha 等人把计算活跃 S 盒个数的最小值问题转化为 MILP 问题[13]. 为此, 他们用很多整数环上的线

性约束条件来刻画特征为 2 的有限域上的运算. Mouha 等人考虑了下面的两种运算.

(1) 异或运算 $\oplus$: $F_2^w \times F_2^w \to F_2^w$. 设 $a=(a_0,\cdots,a_{w-1})$, $b=(b_0,\cdots,b_{w-1})$ 和 $c=(c_0,\cdots,c_{w-1})$ 为异或运算输入和输出差分, 即 $a_i \oplus b_i = c_i$ 对所有的 $0\leqslant i\leqslant w-1$. 那么这个特征 2 上的运算可以用下面整数环上的不等式表示

$$\begin{cases} a_i + b_i + c_i \geqslant 2d_i \\ d_i \geqslant a_i \\ d_i \geqslant b_i \\ d_i \geqslant c_i \end{cases} \tag{1}$$

其中 $d_i \in F_2$ 是额外增加的新变量, $0\leqslant i\leqslant w-1$.

值得注意的是不等式(1)对异或运算的刻画并不完备. 例如, 对于 $(a_i,b_i,c_i,d_i)=(1,1,1,1)$, 很容易验证它是满足不等式(1)的, 但是它不满足异或运算, 因为 $1\oplus 1=0$. 不等式(1)和 $a_i+b_i+c_i\leqslant 2$ 相结合可以等价地刻画异或运算.

(2) 线性变换 L: $F_{2^w}^m \to F_{2^w}^m$, 其中 m 是字的大小. L 的分支数 B_L 定义为

$$B_L = \min_{x\neq 0}\left\{\mathrm{wt}(x)+\mathrm{wt}(Lx) : x\in F_{2^w}^m\right\}$$

其中, 汉明重量 $\mathrm{wt}(x)$ 表示向量 x 分量不为零的个数. 令 $\Delta x=(\Delta x_0,\Delta x_1,\cdots,\Delta x_{m-1})$ 为线性变换 L 的输入差分的特征向量, $\Delta y=(\Delta y_0,\Delta y_1,\cdots,\Delta y_{m-1})$ 为线性变换 L 的输出差分的特征向量, 即如果输入(输出)差分的第 i 个分量为零, 那么 $\Delta x_i=0$ ($\Delta y_i=0$), 否则 $\Delta x_i=1$ ($\Delta y_i=1$). 线性变换 L 在整数环上的刻画为

$$\begin{cases} \sum_{i=0}^{m-1}(\Delta x_i+\Delta y_i)\geqslant B_L d_L \\ d_L\geqslant \Delta x_i,\ 0\leqslant i\leqslant m-1 \\ d_L\geqslant \Delta y_i,\ 0\leqslant i\leqslant m-1 \end{cases}$$

其中 $d_L \in F_2$ 是额外增加的新变量.

随后, 孙思维等人通过逻辑条件模型和计算几何给出了 S 盒的两种刻画方法[14,15].

(3) S 盒 S: $F_2^w \to F_2^w$. 设 $\Delta\alpha=(\Delta\alpha_0,\Delta\alpha_1,\cdots,\Delta\alpha_{w-1})$ 和 $\Delta\beta=(\Delta\beta_0,\Delta\beta_1,\cdots,\Delta\beta_{w-1})$ 分别为 S 盒 S 的输入和输出差分. 用 δ 表示 S 盒 S 是否活跃, 即

$$\delta=\begin{cases}0, & (\Delta\alpha_0,\Delta\alpha_1,\cdots,\Delta\alpha_{w-1})=0\\ 1, & \text{其他}\end{cases}$$

在整数环上, δ 和 $\Delta\alpha$ 的关系有一种等价的刻画

$$\begin{cases}\delta-\Delta\alpha_i\geqslant 0,\ 0\leqslant i\leqslant w-1\\ \Delta\alpha_0+\Delta\alpha_1+\cdots+\Delta\alpha_{w-1}-\delta\geqslant 0\end{cases} \tag{2}$$

此外, 对于可逆的 S 盒, 它的输入差分非零当且仅当它的输出差分非零, 这同样可以通过整数环上的线性不等式表示

$$\begin{cases}(\Delta\alpha_0+\Delta\alpha_1+\cdots+\Delta\alpha_{w-1})w-(\Delta\beta_0+\Delta\beta_1+\cdots+\Delta\beta_{w-1})\geqslant 0\\ (\Delta\beta_0+\Delta\beta_1+\cdots+\Delta\beta_{w-1})w-(\Delta\alpha_0+\Delta\alpha_1+\cdots+\Delta\alpha_{w-1})\geqslant 0\end{cases} \tag{3}$$

最后, 类似于线性变换的刻画方法, 设 S 盒 S 的分支数 B_S 定义为

$$B_S = \min_{\Delta\alpha \neq 0} \left\{ \mathrm{wt}(\Delta\alpha) + \mathrm{wt}(\Delta\beta) \middle| \#(\Delta\alpha, \Delta\beta) > 0 \right\}$$

其中, $\#(\Delta\alpha,\Delta\beta) = \#\left\{ x \in F_2^w \middle| S(x) \oplus S(x \oplus \Delta\alpha) = \Delta\beta \right\}$. 那么 S 在整数环上的线性刻画可以用下面的不等式组表示

$$\begin{cases} \sum_{i=0}^{w-1} (\Delta\alpha_i + \Delta\beta_i) \geqslant B_S d_S \\ d_S \geqslant \Delta\alpha_i, 0 \leqslant i \leqslant w-1 \\ d_S \geqslant \Delta\beta_i, 0 \leqslant i \leqslant w-1 \end{cases} \tag{4}$$

其中, $d_S \in F_2$ 是额外增加的新变量.

一个 S 盒通过不等式(2)、(3)和(4)来刻画也是不完备的. 即存在一对向量同时满足不等式(2)、(3)和(4), 但它是 S 盒的不可能差分特征. 正是由于这个原因, 孙思维等人提出了用基于计算几何的新方法来获得 S 盒的线性约束条件[15].

凸包的 H-表示. 设 X 是 n 维线性空间 R^n 上的离散点集. 集合 X 的凸包是指包含 X 的最小凸集. 凸包可以看成是线性等式和不等式构成的可行解. 有很多的算法可以计算一个有限点集的凸包. 对于 $n \times n$ 的 S 盒 S, 它的所有差分特征都可以看成线性空间 R^{2n} 上的离散点集. 利用现成的计算代数软件, 如 SAGE, 可以求出 S 的凸包.

一般地, 随着 n 的增大, 集合 $X \subset R^n$ 的凸包中等式和不等式的数量会快速增大. 例如, 一个 4×4 的 S 盒, 它的凸包需要上百个不等式表示. 这样生成的一个 MILP 问题, 在有限的时间和资源里就变得不可解了. 因此, 只能从上百个不等式中选取尽可能少的不等式. 孙思维等人利用贪婪算法来选取一些“好”的不等式[15].

对于如何刻画特征 2 上的运算和怎样生成 MILP 问题, 请读者参阅文献[13–15].

4 RECTANGLE-80 的相关密钥差分攻击

我们先介绍一些下面会频繁用到的符号. 令 P (P')、C (C')、ΔP 和 ΔC 分别表示明文、密文以及明文和密文的差分. 令 K_i (K_i')、I_i (I_i')和 O_i (O_i')分别表示第 i 轮的轮子密钥, 以及 SubColumn 运算的输入和输出. 同样地, 令 ΔK_i、ΔI_i 和 ΔO_i 分别表示第 i 轮的轮子密钥差分, 以及 SubColumn 运算的输入和输出差分.

有了上面的准备, 我们将会给出 19 轮 RECTANGLE-80 的相关密钥差分攻击. 首先, 我们找到了大量的输入、输出和轮子密钥差分相同的 15 轮差分特征. 利用这些差分特征, 我们分别向前和向后扩展两轮, 得到了 19 轮的相关密钥差分攻击.

4.1 RECTANGLE-80的差分特征

在本小节中, 我们利用 Mouha 和孙思维等人的方法得到了大量的差分特征. 事实上, 在 RECTANGLE 分组密码中, 只有异或运算和 S 盒需要转化成整数环上的线性约束条件. 根据第 3 节的描述, 我们可以得到关于异或运算和 S 盒的所有线性约束条件. 此外, 由于是相关密钥差分攻击, 所以密钥的差分不能为零, 即需要另外一个约束条件 $\Delta k_0 + \Delta k_1 + \cdots + \Delta k_{79} \geqslant 1$. 目标函数为 $\min \sum \delta$, 其中 δ 表示对应的 S 盒是否活跃, 具体定义见第 3 节的不等式(2). 上述过程可以产生一个完整的 MILP 问题, 即包含约束条件和目标函数. 利用 Gurobi 求解器可以求解相应的 MILP 问题.

表 2 是 RECTANGLE 中用到的 S 盒的差分分布. 根据表 2, 我们可以计算具体差分特征的概率. 对 $7 \leqslant n \leqslant 15$, 表 3 列举了含较少活跃 S 盒个数的 n 轮差分特征, 其中差分概率对应着某条含相应活跃 S 盒个数

的差分特征. 当 $n = 7$、8、9 时(标记为“*”), 表格中活跃 S 盒个数对应着最小值, 也就是说所有可能的 n 轮差分特征的活跃 S 盒个数都会大于等于表格中的数据. 当 $n = 10$、11、12、13、14、15 时, 表格中活跃 S 盒个数对应着较小值, 也就是说可能存在某条 n 轮差分特征, 它的活跃 S 盒的个数小于表格中所给的数据

表 2 RECTANGLE 中 S 盒的差分分布

Table 2 Differential distributions of the S-box in RECTANGLE

	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
1	0	0	0	0	0	2	2	0	2	2	0	4	0	2	0	2
2	0	0	0	0	0	0	4	4	0	0	0	0	0	0	4	4
3	0	2	0	2	4	0	0	0	2	0	0	2	0	0	2	2
4	0	0	0	0	0	4	2	2	0	0	0	0	4	0	2	2
5	0	0	0	0	0	2	0	2	2	2	4	0	0	2	2	0
6	0	4	2	2	0	0	0	0	0	4	2	2	0	0	0	0
7	0	2	2	0	4	0	0	0	2	0	2	0	0	0	2	2
8	0	0	2	2	0	4	0	0	0	0	2	2	4	0	0	0
9	0	0	0	4	0	2	0	2	2	2	0	0	0	2	2	0
A	0	0	2	2	0	0	0	0	0	0	2	2	4	4	0	0
B	0	2	0	2	4	0	2	2	2	0	0	2	0	0	0	0
C	0	0	2	2	0	0	2	2	0	0	2	2	0	0	2	2
D	0	0	4	0	0	2	2	0	2	2	0	0	0	2	0	2
E	0	4	0	0	0	0	0	0	0	4	0	0	4	4	0	0
F	0	2	2	0	4	0	2	2	2	0	2	0	0	0	0	0

表 3 含较少活跃 S 盒的个数的 n 轮差分特征及对应的差分概率

Table 3 *n*- round differential characteristics with fewer active S-boxes

轮数	活跃 S 盒的个数	差分概率	轮数	活跃 S 盒的个数	差分概率
7*	7	2^{-18}	12	20	2^{-51}
8*	10	2^{-25}	13	23	2^{-56}
9*	12	2^{-32}	14	24	2^{-59}
10	16	2^{-41}	15	26	2^{-64}
11	19	2^{-44}			

我们找到了一条差分概率为 2^{-64} 的 15 轮差分特征. 为了说明的方便, 我们把它看成是 RECTANGLE 中从第 2 轮到第 16 轮的差分特征. 那么它在第 2 轮 SubColumn 运算的输入差分和在第 16 轮 SubColumn 运算的输出差分分别为

$$\Delta I_2 = \begin{matrix} 0000010000001000 \\ 0000010000000000 \\ 0000000000000000 \\ 0000000000000000 \end{matrix} \text{和} \ \Delta O_{16} = \begin{matrix} 0000001000000000 \\ 0000000000000000 \\ 0000000000000000 \\ 0000000000000000 \end{matrix}$$

另外, 这条差分特征在第 2 轮和第 16 轮的轮子密钥差分

$$\Delta K_2 = \begin{matrix} 00000000000000000000 \\ 00000000000000000000 \\ 00001000000000000000 \\ 00000000000000000000 \end{matrix} \text{和} \ \Delta K_{16} = \begin{matrix} 00000000000000000000 \\ 00000000000000000000 \\ 00000000001000000000 \\ 00000000000000000000 \end{matrix}$$

固定 ΔI_2 、 ΔO_{16} 和 ΔK_2 , 根据密钥编排算法可知 ΔK_i ($i = 2,3,\cdots,15,16$)均被固定下来. 借助于计算机, 我们找出了所有活跃 S 盒个数为 26–30 的 15 轮差分特征, 见表 4. 这些差分特征总的概率为 $2^{-60.5}$, 可以用于构造区分器并恢复部分密钥. 我们将在 4.2 节给出详细的攻击过程.

表 4 活跃 S 盒个数为 26-30 的所有 15 轮差分特征

Table 4 All 15-round differential characteristics with 26-30 active S-boxes

活跃 S 盒的个数	差分特征的条数	总的差分概率
26	4	2^{-62}
27	30	2^{-62}
28	119	$2^{-62.82}$
29	324	$2^{-64.31}$
30	777	$2^{-65.97}$

4.2 RECTANGLE-80的相关密钥攻击

本小节中, 我们将利用 4.1 节得到的差分特征向前和向后分别扩展 2 轮来攻击 19 轮的 RECTANGLE. 这里假设 19 轮的 RECTANGLE 指的是经过 19 轮的迭代和最后一次异或轮密钥, 从第 0 轮开始标记, 即第 0 轮到第 18 轮. 第 0 轮的子密钥就是主密钥, 最后异或的轮密钥记为 K_{19}. 下面将详细地描述我们的攻击过程.

首先, 我们着重说明是怎样向前扩展 2 轮的(向后扩展 2 轮是类似的). 由 ΔI_2 和 ΔK_2 可以得到第 1 轮 SubColumn 运算的输出差分为

$$\Delta O_1 = \begin{matrix} 0000010000001000 \\ 0000001000000000 \\ 0000000000001000 \\ 0000000000000000 \end{matrix} \tag{5}$$

考虑 ΔO_1 的第 10 列, 根据表 2 中 S 盒的差分分布, 只有当 S 盒的输入差分为 1100、0110、1110、1101、0111 或 1111 时, S 盒的输出差分才能是 1000. 对于 ΔO_1 的第 3 列和第 9 列也会有类似的发现. 因此, 第 1 轮 SubColumn 运算的输入差分必须具备下面的形式

$$\Delta I_1 = \begin{matrix} 00000??00000?000 \\ 000001?000000000 \\ 00000??00000?000 \\ 00000??00000?000 \end{matrix}$$

其中“?”表示 0 或 1 中不确定的值. 而且 ΔI_1 只有 $6 \times 7 \times 6 = 2^{7.98}$ 种可能. 对于符合条件的 ΔI_1, 使得差分恰好为(5)式的 ΔO_1 的概率为

$$\frac{1}{6} \times \frac{1}{7} \times \frac{1}{6} = 2^{-7.98}$$

根据密钥编排, 第 1 轮的子密钥差分为

$$\Delta K_1 = \begin{matrix} 00000000000000000000 \\ 00000000000000000000 \\ 00000000000000010000 \\ 00000000000000000000 \end{matrix}$$

类似于第 1 轮的分析, 我们得到第 0 轮 SubColumn 运算的输出和输入差分分别为

$$\Delta O_0 = \begin{matrix} 00000??00000?000 \\ 0000001?00000000 \\ 0??00001?0000000 \\ 00??00000?000000 \end{matrix} \text{和}\ \Delta I_0 = \begin{matrix} 0???0?????00?000 \\ 0???0?????00?000 \\ 0???0?????00?000 \\ 0???0?????00?000 \end{matrix}$$

这里 ΔO_0 有 $2^{7.98}$ 情况, 而 ΔI_0 有 2^{36} 种情况. 因此, 对于一个随机选择的 ΔI_0 、 ΔO_0 属于这 $2^{7.98}$ 种情况的概率为 $2^{-28.02}$. 此外, 第 0 轮的子密钥和明文差分分别为

$$\Delta K_0 = \begin{matrix} 00000000000000000000 \\ 00000000000000000000 \\ 00000010000000000000 \\ 00000000000000000000 \end{matrix} \text{和}\ \Delta P = \begin{matrix} 0???0?????00?000 \\ 0???0?????00?000 \\ 0???0?????00?000 \\ 0???0?????00?000 \end{matrix}$$

同样地, 我们也可以向后扩展两轮. 由于第 17 轮的子密钥差分为

$$\Delta K_{17}=\begin{array}{c}00000000000000000000\\00000000000000000000\\00000000000000000001\\00000000000000000000\end{array}$$

所以第 17 轮 SubColumn 运算的输入和输出差分分别为

$$\Delta I_{17}=\begin{array}{c}0000001000000000\\0000000000000000\\0000000000000001\\0000000000000000\end{array} \text{和}\ \Delta O_{17}=\begin{array}{c}000000?00000000?\\000000?00000000?\\000000?000000001\\000000?00000000?\end{array}$$

根据表 2 中 S 盒的差分分布, 当 S 盒的输入差分为 $\Delta I_S=0010$ 时, 它的输出差分 ΔO_S 只可以属于集合 $\{1010,0110,1110,0011,0111,1111\}$. 因此, 第 18 轮的子密钥差分为

$$\Delta K_{18}=\begin{array}{c}000000000000\heartsuit0000000\\0000000000\clubsuit000000000\\00000000100000000000\\000000\diamondsuit0000000000000\end{array}$$

其中 $\heartsuit\clubsuit1\diamondsuit\in\{1010,0110,1110,0011,0111,1111\}$. 在攻击过程中, 我们总是假设 ΔK_{18} 是确定的, 因为可以对每一个属于集合 $\{1010,0110,1110,0011,0111,1111\}$ 的元素使用一次我们的攻击方法, 即我们总是假设 $\heartsuit$ 、$\clubsuit$ 和 $\diamondsuit$ 是确定的值. 那么第 18 轮 SubColumn 运算的输入和输出差分分别为

$$\Delta I_{18}=\begin{array}{c}000000?0\heartsuit000000?\\00000?\clubsuit0000000?0\\0001100000?00000\\00?000000?000000\end{array} \text{和}\ \Delta O_{18}=\begin{array}{c}00?????0???000??\\00?????0???000??\\00?11??0???000??\\00?????0???000??\end{array}$$

其中, "$\heartsuit$"和"$\clubsuit$"表示 0 和 1 中某个确定的值, 而"?"表示 0 和 1 中某个不确定的值. ΔO_{18} 最多有 $2^{26.54}$ 种可能.由于最后的子密钥的差分

$$\Delta K_{19}=\begin{array}{c}00000\heartsuit00000000000000\\0\clubsuit000000000000000000\\00000000000000000100\\0000000000000\diamondsuit000000\end{array}$$

是确定的, 密文差分

$$\Delta C=\begin{array}{c}0\heartsuit?????0???000??\\0?????0???000??0\\00??00?11??0???0\\0??00?????0???00\end{array}$$

最多也只有 $2^{26.54}$ 种可能.

数据收集过程. 选择 2^x 个结构体. 在每个结构体中, 第 0、1、2、4、5、11 和 15 列的值都相同, 第 3、6、7、8、9、10、12、13 和 14 列取遍所有的可能值, 这样每个结构体中就包含 2^{36} 组明文. 这 2^{36} 组明文又可以组成约 2^{72} 个有序对. 由第 0 轮和第 1 轮给出的概率知, 结构体中的任意一对明文在正确密钥下产生差分 ΔI_2 的概率为 2^{-36} . 因此满足 ΔI_2 和 ΔO_{16} 的明文对平均有 $2^{x+72-36-60.5}=2^{x-24.5}$ 对.

密钥恢复过程. 根据密文差分 ΔC , 每个结构体中期望产生的有序对为 $2^{72-37.46}=2^{34.54}$ 对. 因此, 通过密文差分筛选后, 剩下的明文对平均有 $2^{x+34.54}$.

步骤 1. 猜测子密钥 K_0 的部分比特:

(a) 猜测第 0 轮子密钥 K_0 的第 3 列 $K_0^{(3)}$, 计算剩下明文对的第 3 列经过 S 盒变换的差分, 即

$$S\left(P^{(3)}\oplus K_0^{(3)}\right)\oplus S\left(P'^{(3)}\oplus K_0^{(3)}\oplus\Delta K_0^{(3)}\right)$$

如果差分不具有?000 这样的形式, 那么删除这对明文对. 那么期望剩下的明文对数为 $2^{x+31.54}$.

(b) 重复步骤 1(a), 分别猜测 $K_0^{(6)}$ 、$K_0^{(7)}$ 、$K_0^{(8)}$ 、$K_0^{(9)}$ 、$K_0^{(10)}$ 、$K_0^{(12)}$ 、$K_0^{(13)}$ 和 $K_0^{(14)}$, 最后期望剩下的正确明文对数为 $2^{x+8.54}$.

步骤 2. 通过猜测 K_0 或 K_1 的部分比特, 来获得 K_1 的部分比特:

(a) 由于子密钥 K_1 的很多比特都可以通过移位和异或轮常数直接从 K_0 中得到, 因此很多已经猜测过的比特不需要重复猜测, 这大大地降低了计数器的数目. 比如, 对于子密钥 K_1 的第 3 列, 由密钥编排可知

$$\left(K_1^{(0,3)},K_1^{(1,3)},K_1^{(2,3)},K_1^{(3,3)}\right)=\left(K_0^{(0,16)},K_0^{(1,14)},K_0^{(2,12)},K_0^{(3,10)}\right)$$

这里只需要猜测 $K_0^{(0,16)}=K_1^{(0,3)}$, 由于其他 3 比特已经在步骤 1 中猜测过. 那么剩下的明文对还有 $2^{x+4.54}$ 对.

(b) 猜测比特 $K_0^{(1,1)}$ 、 $K_0^{(2,19)}$ 和 $K_0^{(3,17)}$, 检验是否有下面的等式成立

$$S\left(I_1^{(10)}\oplus K_1^{(10)}\right)\oplus S\left(I_{1'}^{(10)}\oplus K_1^{(10)}\oplus \Delta K_1^{(10)}\right)=1000$$

这是由于 $\left(K_1^{(0,10)},K_1^{(1,10)},K_1^{(2,10)},K_1^{(3,10)}\right)=\left(K_0^{(0,3)},K_0^{(1,1)},K_0^{(2,19)},K_0^{(3,17)}\right)$. 此时, 剩下的明文对有 $2^{x+0.54}$ 对.

(c) 类似于步骤 2(b), 猜测比特 $K_0^{(0,2)}$ 、 $K_1^{(1,9)}$ 、 $K_0^{(2,18)}$ 和 $K_0^{(3,16)}$, 那么剩下的明文对有 $2^{x-3.46}$ 对.

步骤 3. 猜测子密钥 K_{19} 的部分比特, 其中有很多比特可以通过猜测 K_0 的部分比特获得:

(a) 考虑 O_{18} 的第 11 列, 涉及到的第 19 轮子密钥比特为 $K_{19}^{(0,11)}$ 、 $K_{19}^{(1,12)}$ 、 $K_{19}^{(2,7)}$ 和 $K_{19}^{(3,8)}$. 猜测第 0 轮子密钥的一些比特 $K_0^{(0,18)}$ 、 $K_0^{(3,2)}$ 、 $K_0^{(0,19)}$ 、 $K_0^{(1,2)}$ 和 $K_0^{(3,1)}$, 那么根据密钥编排和步骤 1 和 2 已经猜测的比特, 第 19 轮子密钥的一些比特 $K_{19}^{(0,11)}$ 、 $K_{19}^{(1,12)}$ 、 $K_{19}^{(2,7)}$ 和 $K_{19}^{(3,8)}$ 被确定下来. 类似步骤 1(a), 剩下的明文对为 $2^{x-7.46}$. 进一步地, 第 19 轮的另外一些比特 $K_{19}^{(0,12)}$ 、 $K_{19}^{(1,13)}$ 、 $K_{19}^{(2,8)}$ 和 $K_{19}^{(3,9)}$ 也被确定下来, 这些比特和 O_{18} 和第 12 列有关, 这样剩下来的明文对平均有 $2^{x-11.46}$ 对.

(b) 猜测 $K_0^{(1,16)}$ 、$K_0^{(2,4)}$ 和 $K_0^{(1,11)}$, 再加上之前步骤 1 到步骤 3(a)已经猜测过的比特, 第 19 轮子密钥的比特 $K_{19}^{(0,1)}$ 、 $K_{19}^{(1,2)}$ 、 $K_{19}^{(2,13)}$ 和 $K_{19}^{(3,14)}$ 已经被确定下来. 而这些比特又和 O_{18} 的第 1 列相关, 这样期望剩下的明文对就只有 $2^{x-14.46}$ 对.

(c) 类似于步骤 3(a)和 3(b), 对于 O_{18} 的第 6 列, 我们猜测 $K_0^{(0,1)}$ 、 $K_0^{(3,19)}$ 和 $K_0^{(0,4)}$; 对于第 7 列, 猜测第 19 轮的子密钥比特 $K_{19}^{(1,8)}$ 和第 0 轮的子密钥比特 $K_0^{(1,18)}$ 、 $K_0^{(2,2)}$; 对于第 0 列, 猜测 $K_{19}^{(2,12)}$; 对于第 9 列, 猜测 $K_{19}^{(0,9)}$ 、 $K_{19}^{(2,5)}$ 、 $K_0^{(0,17)}$ 、 $K_0^{(1,19)}$ 和 $K_0^{(2,1)}$; 对于第 10 列, 猜测 $K_{19}^{(2,6)}$ 和 $K_{19}^{(3,7)}$; 对于第 5 列, 猜测 $K_{19}^{(0,5)}$ 、 $K_{19}^{(1,6)}$ 和 $K_{19}^{(3,2)}$; 对于第 13 列, 猜测 $K_{19}^{(0,13)}$ 、 $K_{19}^{(1,14)}$ 和 $K_{19}^{(2,9)}$. 此时, 期望剩下的明文对只有 $2^{x-36.46}$ 对.

步骤 4. 对于 O_{17}, 涉及到的第 18 轮的子密钥比特都已经在步骤 1-3 中猜测过, 因此在这一步中我们不需要猜测其他的比特. 平均可以剩下 $2^{x-44.46}$ 对明文对. 在对应的计数器上记上最终剩下的明文对数.

步骤 5. 如果计数器大于 1, 我们就把对应的密钥比特看成候选的正确密钥比特. 对每个候选的部分密钥, 对剩余的密钥比特进行穷搜, 直到找到正确的密钥.

复杂度分析. 由于经过 ΔI_2 和 ΔO_{16} 筛选之后, 期望剩下的明文对数为 $2^{x+72-36-60.5}=2^{x-24.5}$. 我们取 $x=26$, 对正确密钥而言, 期望剩下的明文对数为 3. 因此, 数据复杂度为 2^{62}.

为了得到时间复杂度, 我们先分析每一步的时间复杂度. 在加密过程中, 由于数据复杂度为 2^{62} , 所以时间需要 2^{63} 次的 19 轮加密. 在步骤 1(a)中, 需要

$$2\times 2^{x+34.54}\times 2^4\times\frac{1}{16}\times\frac{1}{19}\approx 2^{x+29.54}$$

次的 19 轮加密. 在步骤 1(b)中, 时间复杂度为

$$2\times\left(2^{x+39.54}+2^{x+40.54}+2^{x+41.54}+2^{x+42.54}+2^{x+43.54}+2^{x+44.54}+2^{x+45.54}+2^{x+46.54}\right)\times\frac{1}{16}\times\frac{1}{19}\approx 2^{x+40.54}$$

步骤 2 的时间复杂度为

$$2\times\left(2^{x+45.54}+2^{x+44.54}+2^{x+44.54}\right)\times\frac{1}{16}\times\frac{1}{19}\approx 2^{x+39.54}$$

步骤 3 的时间复杂度为

$$2\times\left(2^{x+45.54}+2^{x+41.54}+2^{x+40.54}+2^{x+40.54}+2^{x+40.54}+2^{x+37.54}+2^{x+39.54}+2^{x+38.54}+2^{x+38.54}+2^{x+38.54}\right)\times\frac{1}{16}\times\frac{1}{19}\approx 2^{x+38.54}$$

步骤 4 的时间复杂度为 $2^{x+28.54}$. 因此, 对于一个给定的 ΔK_{18}, 步骤 1 到步骤 5 的时间复杂度为 $2^{67.42}$. 由于 ΔK_{18} 有 6 种取值情况, 所以总的时间复杂度为 2^{70}. 内存复杂度为 2^{72} 个计数器.

5 总结

在本文中, 我们利用 Mouha 和孙思维等人的方法找到了一条含 26 个活跃 S 盒的 15 轮差分特征, 它的输入、输出和轮子密钥差分的汉明重量都很小. 固定这些差分, 我们找到了所有活跃 S 盒数为 26-30 的 15 轮差分特征. 这些差分特征总的概率为 $2^{-60.5}$. 基于这些差分特征, 我们将相应的差分区分器分别向前和向后扩展了两轮, 得到了一个 19 轮的相关密钥差分攻击, 其中数据复杂度为 2^{62}, 时间复杂度为 2^{70}, 内存复杂度为 2^{72}.

为了能攻击更多的轮数, 需要找到更多的高轮差分特征. 同时, 确定某些轮数的活跃 S 盒的下界也很有意义, 这可以说明 RECTANGLE 分组密码是否可以抵抗标准的相关密钥差分攻击.

References

[1] Bogdanov A, Knudsen L, Leander G, et al. PRESENT: an ultra-lightweight block cipher[C]. In: Cryptographic Hardware and Embedded Systems—CHES 2007. Springer Berlin Heidelberg, 2007: 450–466.

[2] Guo J, Peyrin T, Poschmann A, et al. The LED block cipher[C]. In: Cryptographic Hardware and Embedded Systems—CHES 2011. Springer Berlin Heidelberg, 2011: 326–341.

[3] Wu W, Zhang L. LBlock: a lightweight block cipher[C]. In: Applied Cryptography and Network Security. Springer Berlin Heidelberg, 2011: 327–344.

[4] Borghoff J, Canteaut A, Güneysu T, et al. PRINCE: a low-latency block cipher for pervasive computing applications[C]. In: Advances in Cryptology—ASIACRYPT 2012. Springer Berlin Heidelberg, 2012: 208–225.

[5] Gérard B, Grosso V, Naya-Plasencia M, et al. Block ciphers that are easier to mask: how far can we go?[C]. In: Cryptographic Hardware and Embedded Systems—CHES 2013. Springer Berlin Heidelberg, 2013: 383–399.

[6] Beaulieu R, Shors D, Smith J, et al. The SIMON and SPECK families of lightweight block ciphers[EB/OL]. http://eprint.iacr.org/2013/404.pdf

[7] Zhang W, Bao Z, Lin D, et al. RECTANGLE: A bit-slice ultra-lightweight block cipher suitable for multiple platforms[EB/OL]. http://eprint.iacr.org/2014/084.pdf

[8] Biham E, Shamir A. Differential cryptanalysis of DES-like cryptosystems[J]. Journal of Cryptology, 1991, 4(1): 3–72.

[9] Matsui M. Linear cryptanalysis method for DES cipher[C]. In: Advances in Cryptology—EUROCRYPT '93. Springer Berlin Heidelberg, 1994: 386–397.

[10] Biham E. New types of cryptanalytic attacks using related keys[J]. Journal of Cryptology, 1994, 7(4): 229–246.

[11] Knudsen L R. Truncated and higher order differentials[C]. In: Fast Software Encryption—FSE '95. Springer Berlin Heidelberg, 1995: 196–211.

[12] Biham E, Biryukov A, Shamir A. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials[C]. In: Advances in Cryptology—EUROCRYPT '99. Springer Berlin Heidelberg, 1999: 12–23.

[13] Mouha N, Wang Q, Gu D, et al. Differential and linear cryptanalysis using mixed-integer linear programming[C]. In: Information Security and Cryptology. Springer Berlin Heidelberg, 2012: 57–76.

[14] Sun S, Hu L, Song L, et al. Automatic security evaluation of block ciphers with S-bP structures against related-key differential attack[C]. In: Information Security and Cryptology. Springer Berlin Heidelberg, 2013: 39–51.

[15] Sun S, 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[C]. In: Advances in Cryptology—ASIACRYPT 2014. Springer Berlin Heidelberg, 2014: 158–178.

[16] Qiao K, Hu L, Sun S, et al. Improved MILP modeling for automatic security evaluation and application to FOX[J]. IEICE Transactions on Fundamentals of Electronics Communications and Computer Sciences, to appear.

[17] Ma X, Hu L, Sun S, et al. Tighter security bound of MIBS block cipher against differential attack[C]. In: Network and System Security. Springer International Publishing, 2014: 518–525.

[18] Collard B, Standaert F X. A statistical saturation attack against the block cipher PRESENT[C]. In: Topics in Cryptology—CT-RSA 2009. Springer Berlin Heidelberg, 2009: 195–210.

[19] Knudsen L, Wagner D. Integral cryptanalysis[C]. In: Fast Software Encryption—FSE 2002. Springer Berlin Heidelberg, 2002: 112–127.

[20] Biryukov A, Wagner D. Slide attacks[C]. In: Fast Software Encryption—FSE '99. Springer Berlin Heidelberg, 1999: 245–259.

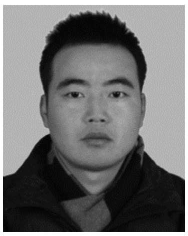
[21] Selvam R, Shanmugam D, Annadurai S. Side channel attacks: vulnerability analysis of PRINCE and RECTANGLE using DPA[EB/OL]. https://eprint.iacr.org/2014/644.pdf

[22] Rolfes C, Poschmann A, Leander G, et al. Ultra-lightweight implementations for smart devices–security for 1000 gate equivalents[C]. In: Smart Card Research and Advanced Applications. Springer Berlin Heidelberg, 2008: 89–103.

[23] Feldhofer M, Dominikus S, Wolkerstorfer J. Strong authentication for RFID systems using the AES algorithm[C]. In: Cryptographic Hardware and Embedded Systems—CHES 2004. Springer Berlin Heidelberg, 2004: 357–370.

[24] Poschmann A, Leander G, Schramm K, et al. A family of light-weight block ciphers based on DES suited for RFID applications[C]. In: Workshop on RFID Security, Graz, Austria, July 2006.

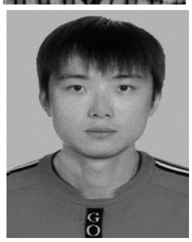
作者信息



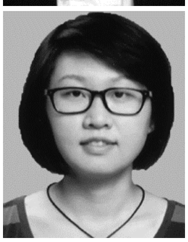
单进勇(1987–), 江苏大丰人, 博士研究生. 主要研究领域为对称密码学.
E-mail: jyshan12@is.ac.cn

胡磊(1967–), 博士, 研究员. 主要研究领域为密码学与信息安全.
E-mail: hu@is.ac.cn

宋凌(1987–), 湖南岳阳人, 博士研究生. 主要研究领域为对称密码学.
E-mail: lsong@is.ac.cn



孙思维(1985–), 博士, 助理研究员. 研究方向为密码学.
E-mail: swsun@is.ac.cn



马小双(1991–), 吉林长春人, 博士研究生. 主要研究领域为对称密码学.
E-mail: xshma13@is.ac.cn