密码学报 ISSN 2095-7025 CN 10-1195/TN
Journal of Cryptologic Research, 2019, 6(1): 18–26

E-mail: jcr@cacrnet.org.cn
http://www.jcr.cacrnet.org.cn
Tel/Fax: +86-10-82789618

对轻量级分组密码算法 LBlock 的差分故障攻击*

王 涛, 王永娟, 高 杨, 张诗怡

信息工程大学, 郑州 450001
通信作者: 王涛, E-mail: wt107263@163.com

摘 要: 本文首先分析差分故障攻击的故障模型与原理, 利用 S 盒的差分不均匀性, 通过建立输入差分、输出差分和可能输入值之间的对应关系, 给出差分故障分析的优化方案, 实现快速归约, 提高差分故障攻击的效率. 本文通过对 LBlock 算法建立对应关系, 可以快速直观缩小输入值取值空间, 进而快速确定对应扩展密钥. 对于不同故障值 (输入差分), 对应的输出差分和可能输入值均不相同, 可以得到二元关系集合. 由于轻量级分组密码 S 盒多为 4×4 S 盒, 该集合中元素较少, 注入少量不同故障值, 通过查表, 对可能输入值取交集即可快速确定唯一可能输入值. 将优化方案应用于 LBlock 轻量级分组密码算法, 在最后一轮输入处注入 2 次宽度为 16 bit 的故障可恢复最后一轮轮密钥, 然后将状态回推一轮, 在倒数第二轮输入处注入 2 次宽度为 16 bit 的故障可恢复倒数第二轮密钥. 根据密钥扩展方案, 恢复两轮轮密钥后将恢复主密钥的计算复杂度降为 2^{19}.

关键词: 轻量级分组密码; LBlock 算法; 差分故障攻击

中图分类号: TP309.7 **文献标识码:** A DOI: 10.13868/j.cnki.jcr.000279

中文引用格式: 王涛, 王永娟, 高杨, 张诗怡. 对轻量级分组密码算法 LBlock 的差分故障攻击[J]. 密码学报, 2019, 6(1): 18–26.

英文引用格式: WANG T, WANG Y J, GAO Y, ZHANG S Y. Differential fault attack on lightweight block cipher LBlock[J]. Journal of Cryptologic Research, 2019, 6(1): 18–26.

Differential Fault Attack on Lightweight Block Cipher LBlock

WANG Tao, WANG Yong-Juan, GAO Yang, ZHANG Shi-Yi

Information Engineering University, Zhengzhou 450001, China
Corresponding author: WANG Tao, E-mail: wt107263@163.com

Abstract: Firstly, this paper analyzes the fault model and principle of differential fault attack. By using the differential inhomogeneity of S-boxes, this paper gives an optimization of differential fault analysis by establishing the corresponding relationship between input differentials, output differentials, and possible input values to improve the efficiency of differential fault attack. In this paper, the corresponding relationship for LBlock algorithm is established, which can be used to effectively reduce the value space of input values, and then quickly determine the corresponding extended key. For different fault values (input differentials), the corresponding output differences, and possible input values are not the same, there exists a set of binary relationships. Since the lightweight S-boxes are mostly 4×4 S boxes, there are fewer elements in the set and a small number of different false values

* 基金项目: 国家自然科学基金 (61872381)
Foundation: National Natural Science Foundation of China (61872381)
收稿日期: 2017-11-13 定稿日期: 2018-05-12

are injected. By looking up the table, the only possible input value can be quickly identified by taking the intersection of possible input values. The optimization scheme is applied to the LBlock lightweight block cipher algorithm. In the last round of input, two 16-bit wide faults are recoverable to the last round key, and then the state is pushed one round back. In the second last round, by injecting 2 faults in 16-bit width, the second last round key can be recovered. According to the key expansion scheme, the recovery of two-round key reduces the computational complexity of recovering master key to 2^{19}.
Key words: lightweight block cipher; LBlock algorithm; differential fault attack

1 引言

低功耗无线移动设备的发展给人们的生活带来了很大方便, 其中的安全问题越来越受到重视. 因为终端资源非常有限, 所以轻量级密码算法 [1–4] 和协议 [5–7] 成了密码学术界关注的焦点. 吴文玲等 [1] 于 2011 年在 ACNS 上提出的轻量级密码算法 LBlock 就是受到较多关注的热点算法之一.

1996 年, Kocher 等人发现密码运行时执行时间的差异会泄露密钥信息, 并成功分析了 RSA 等非对称密码算法 [8]. 随后, 运行故障 [9]、功率消耗 [10]、电磁辐射 [11] 等各种类型的泄露被发现并成功应用于密码分析. 利用算法运行泄露信息结合传统攻击的方法被称为旁路攻击. 差分故障攻击利用干扰情况下的运算错误来恢复密钥, 是一种旁路攻击的方法. 1996 年, Boneh 等人利用智能卡上的故障输出成功恢复了 RSA 算法的密钥 [9]. 这是密码故障首次应用于密码分析. 1997 年, Biham 和 Shamir 首次系统提出差分故障攻击, 成功分析了 DES 算法 [12], 为后来的故障攻击提供思路. 后来, 差分故障分析被应用于 ECC [13] 等公钥密码算法, AES [14]、Camellia [15]、SMS4、ARIA、PRESENT、MIBS、GOST 等分组密码算法, RC4 [16]、Trivium [17]、Grain-128、Rabbit 等序列密码算法.

本文对差分故障攻击进行了改进, 并将其应用于 LBlock 算法.

2 LBlock 算法简介

LBlock [1] 算法分组长 64 位, 初始密钥 80 位, 迭代 32 轮. LBlock 采用 Feistel 结构的一个变种, 加密流程如图1. 令 $M = X_1||X_0$ 表示 64 位的明文, 则 $X_i = F(X_{i-1}, K_{i-1}) \oplus (X_{i-2} <<< 8), i = 2, 3, \cdots, 33$. $C = X_{32}||X_{33}$ 作为 64 位密文输出.

在每轮中用到的构件定义如下. 轮函数 F 定义如图2, 其中 S 和 P 在下文中有定义. 混淆层 S 是 F 的非线性组件, 包含 8 个 4 进 4 出的 S 盒, S 盒见表1.

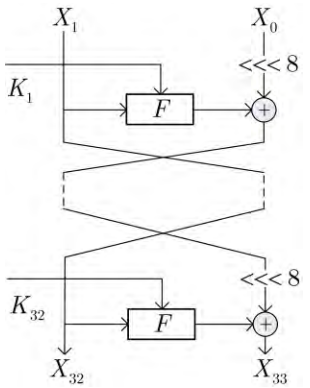


图 1 LBlock 加密流程
Figure 1 LBlock encryption process

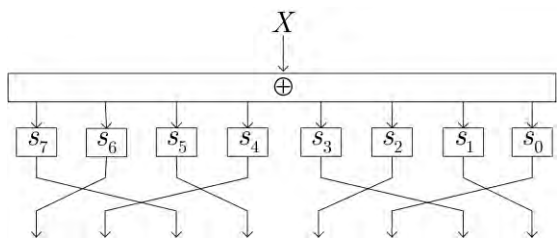


图 2 LBlock 算法轮函数
Figure 2 LBlock algorithm round function

$$S : \{0,1\}^{32} \to \{0,1\}^{32}$$

$$Y = Y_7||Y_6||Y_5||Y_4||Y_3||Y_2||Y_1||Y_0 \to Z = Z_7||Z_6||Z_5||Z_4||Z_3||Z_2||Z_1||Z_0$$

$$Z_7 = s_7(Y_7), Z_6 = s_6(Y_6), Z_5 = s_5(Y_5), Z_4 = s_4(Y_4),$$
$$Z_3 = s_3(Y_3), Z_2 = s_2(Y_2), Z_1 = s_1(Y_1), Z_0 = s_0(Y_0)$$

表 1 LBlock 算法 S 盒
Table 1 S box of LBlock algorithm

s_0	14, 9, 15, 0, 13, 4, 10, 11, 1, 2, 8, 3, 7, 6, 12, 5	s_5	2, 13, 11, 12, 15, 14, 0, 9, 7, 10, 6, 3, 1, 8, 4, 5
s_1	4, 11, 14, 9, 15, 13, 0, 10, 7, 12, 5, 6, 2, 8, 1, 3	s_6	11, 9, 4, 14, 0, 15, 10, 13, 6, 12, 5, 7, 3, 8, 1, 2
s_2	1, 14, 7, 12, 15, 13, 0, 6, 11, 5, 9, 3, 2, 4, 8, 10	s_7	13, 10, 15, 0, 14, 4, 9, 11, 2, 1, 8, 3, 7, 5, 12, 6
s_3	7, 6, 8, 11, 0, 15, 3, 14, 9, 10, 12, 13, 5, 2, 4, 1	s_8	8, 7, 14, 5, 15, 13, 0, 6, 11, 12, 9, 10, 2, 4, 1, 3
s_4	14, 5, 15, 0, 7, 2, 12, 13, 1, 8, 4, 9, 11, 10, 6, 3	s_9	11, 5, 15, 0, 7, 2, 9, 13, 4, 8, 1, 12, 14, 10, 3, 6

扩散层 P 作用在 8 个 nibble 上, 表示如下.

$$P : \{0,1\}^{32} \to \{0,1\}^{32}$$
$$Y = Y_7||Y_6||Y_5||Y_4||Y_3||Y_2||Y_1||Y_0 \to Z = Z_7||Z_6||Z_5||Z_4||Z_3||Z_2||Z_1||Z_0$$
$$U_7 = Z_6, U_6 = Z_4, U_5 = Z_7, U_4 = Z_5, U_3 = Z_2, U_2 = Z_0, U_1 = Z_3, U_0 = Z_1$$

LBlock 初始密钥 80 位, 记为 $K = k_{79}k_{78}\cdots k_0$. 将初始密钥的左 32 位作为第一轮轮密钥 K_1, 然后运算如算法1.

算法 1 LBlock 算法密钥扩展方案

Input: 80 bit 初始密钥
Output: 轮密钥

$S^0 = K$
for $r = 1$ to 31 **do**
  $S^i = S^i <<< 29$
  $[k_{79}k_{78}k_{77}k_{76}] = s_9([k_{79}k_{78}k_{77}k_{76}])$
  $[k_{75}k_{74}k_{73}k_{72}] = s_8([k_{75}k_{74}k_{73}k_{72}])$
  $[k_{50}k_{49}k_{48}k_{47}k_{46}] \oplus [i]_2$
  $\mathrm{rk}^i = [k_{79}k_{78}\cdots k_{48}]$
end

3 LBlock 算法结构性质

3.1 LBlock 算法差分扩散规律

LBlock 算法的基本运算单位是 nibble, 将第 r 轮的输入记为 $x_0^r, x_1^r, \cdots, x_{15}^r$, 在最后一轮 $x_0^{32}, x_1^{32}, \cdots, x_7^{32}$ 处注入随机故障 f. 传播过程如图3所示. Δx^{32} 可由密文异或得到. 输入差分对应的输出差分由扩散层决定, 见表2.

表 2 输入差分与输出差分对应关系
Table 2 Input differential and output differential correspondence

输入差分	0	1	2	3	4	5	6	7
输出差分	10	8	11	9	14	12	15	13

3.2 LBlock 算法 S 盒差分分布情况

LBlock 使用 S 盒增加非线性度. 使用在进行 S 盒查表时, 对于未知的输入值 a, 导入随机的故障值 f, 可以得到输出差分 f'. 它们之间的关系满足 $S[a] \oplus S[a \oplus f] = f'$. a 有 2 或 4 个可能值. 由于 LBlock 算法混淆层使用 8 个不同的 S 盒, 篇幅所限, 这里只列出 s_0 可能输入值的表格, 见表3.

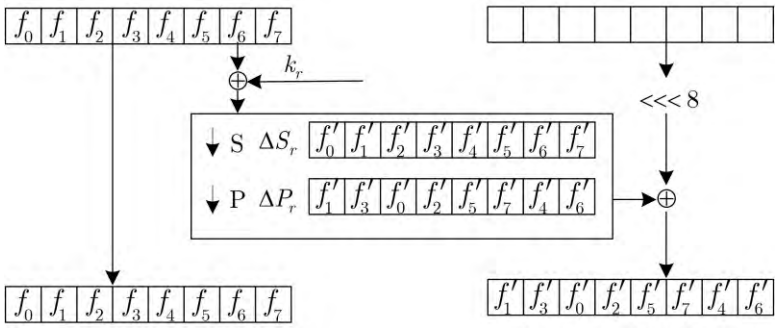


图 3　LBlock 算法轮函数
Figure 3　LBlock algorithm round function

表 3　LBlock 算法 s_0 在不同输入差分下输出差分与可能输入值对应关系
Table 3　Input differential and output differential correspondence

输入差分		在输入差分一定情况下输出差分与输入对应关系							
1	输出差分	1	3	7	9	B	F		
	输入值	6,7,C,D	8,9	0,1	4,5,E,F	A,B	2,3		
2	输出差分	1	3	7	9	B	F		
	输入值	0,2,9,B	D,F	4,6	1,3,8,A	C,E	5,7		
3	输出差分	2	6	A	E				
	输入值	8,B,C,F	1,2,4,7	9,A,D,E	0,3,5,6				
4	输出差分	3	4	5	6	B	D		
	输入值	0,4	9,A,D,E	2,6	8,B,C,F	3,7	1,5		
5	输出差分	4	5	7	A	D	F		
	输入值	1,2,4,7	9,C	8,D	0,3,5,6	A,F	B,E		
6	输出差分	2	4	5	7	D	F		
	输入值	1,2,4,7	0,3,5,6	B,D	9,F	8,E	A,C		
7	输出差分	3	4	5	B	D	E		
	输入值	1,6	8,B,C,F	0,7	2,5	3,4	9,A,D,E		
8	输出差分	2	3	6	7	A	B	E	F
	输入值	5,D	3,B	6,E	2,A	4,C	1,9	7,F	0,8
9	输出差分	3	7	8	b	c	F		
	输入值	5,C	7,E	1,3,8,A	4,D	0,2,9,B	6,F		
A	输出差分	1	2	6	A	D	E		
	输入值	4,5,E,F	3,9	0,A	1,B	6,7,C,D	2,8		
B	输出差分	1	8	C	D				
	输入值	1,3,8,A	4,5,E,F	6,7,C,D	0,2,9,B				
C	输出差分	2	3	5	6	8	9	C	F
	输入值	6,A	2,E	3,F	5,9	7,B	0,C	4,8	1,D
D	输出差分	3	5	8	9	A	C	E	F
	输入值	7,A	5,8	0,D	6,B	2,F	3,E	1,C	4,9
E	输出差分	2	5	6	7	8	9	B	C
	输入值	0,E	4,A	3,D	5,B	2,C	7,9	6,8	1,F
F	输出差分	5	7	8	9	A	B	C	E
	输入值	1,E	3,C	6,9	2,D	7,8	0,F	5,A	4,B

4 差分故障攻击改进及应用

4.1 差分故障攻击

4.1.1 基本攻击模型

分组密码常用 S 盒作为基本构件, 使用 S 盒可以增加算法的非线性度, 增强抵抗线性攻击和差分攻击等密码分析方法的能力. 但 S 盒具有的差分分布特性并不完美, 不仅受到差分攻击的威胁, 也容易遭到差分故障攻击. 在进行 S 盒代换时, 对于未知 S 盒输入 a, 导入随机故障 f, f 的宽度与 a 相同, 可以得到输出差分 f', 且满足: $S[a] \oplus S[a \oplus f] = f'$.

对 Feistel 结构而言, f 是已知的. 根据这个等式, 通过两到三次注入故障, 可以确定输入值 a. 而 a 同扩展密钥相关性很大, 对本文中的 LBlock 算法而言, a 是某个已知状态和轮密钥的异或值. 得到 a 后, 可以直接推出轮密钥. 一般在最后一轮输入位置多次注入故障, 得到最后一轮轮密钥. 之后可以解密这一轮, 将故障位置提高一轮, 获得上一轮的轮密钥. 足量的轮密钥可以推得初始密钥. 以 LBlock 算法 s_0 为例, 其差分 S 盒如表4所示.

表 4 LBlock 算法 s_0 差分 S 盒
Table 4 LBlock algorithm differential S box

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

4.1.2 差分故障攻击的改进

差分 S 盒体现 S 盒的差分分布全局特性, 但对于具体参数未有直观刻画. 本文在此基础上, 分析分组密码算法 S 盒的差分传播特性, 发现当 S 盒输入差分 f 一定的情况下, 对于每一个可能的输出差分 f', 其对应的输入 a 可看成一个集合 $\{a_1, a_2, \cdots, a_n\}$. 记为 $\{f, f'\} = \{a | S[a] \oplus S[a \oplus f] = f', a \in F_2^n\}$.

本文通过对 LBlock 算法建立 a, f, f' 的对应关系 (见表 4), 可以快速直观缩小输入值取值空间, 进而快速确定对应扩展密钥. 对于不同故障值 (输入差分), 对应的输出差分和可能输入值均不相同, 根据表 4 可以得到二元关系集合, 根据表 4 对应关系即得 $\#(1,7) = \{0,1\}, \#(4,3) = \{0,4\}$. 如果分别注入故障值 1 和 4, 则通过查表, 对可能输入值取交集即可快速确定唯一可能输入值为 0, $\{0,1\} \cap \{0,4\} = \{0\}$. 同时对于任意两个输入差分, 其对应的输出差分不完全相同, 这样通过观察输出差分可以很大概率确定输入差分的值. 攻击流程如图4.

由 LBlock 算法的差分 S 盒可知, 在理想情况下, 只需在算法最后一轮注入两次不同故障, 即可快速获得 S 盒输入 a, 从而高效准确的恢复出轮密钥, 最后通过密钥扩展算法恢复主密钥.

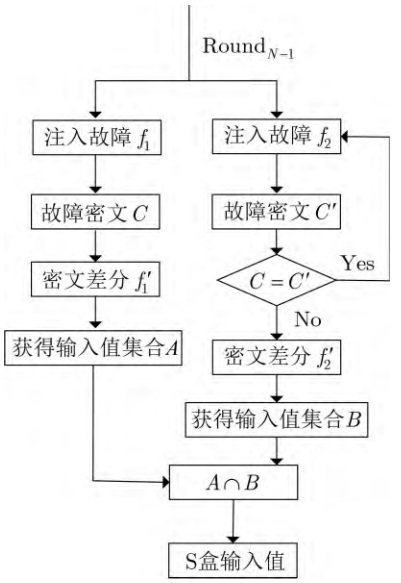


图 4 优化的差分故障攻击流程

Figure 4 Optimized differential fault attack process

4.2 攻击应用

4.2.1 攻击流程

(1) 用密钥 K(80 bit) 加密任意明文 P, 获取正确密文 C.

(2) 在 $x_0^{32}, x_1^{32}, \cdots, x_7^{32}$ 注入随机的故障值, 获取故障密文 C^*.

(3) 将密文异或得到 Δx^{32}. 输入差分与输出差分对应关系见表3和表4, 可确定每个 nibble 2 个或 4 个可能输入值.

(4) 重复 (2)(3) 确定 S 盒输入.

(5) 由 S 盒输入与正确密文的 L_{32} 得到 rk_{31}.

(6) 获取 rk_{31} 后, 可解密最后一轮, 得到正确加密过程的 L^{31} 与 R^{31}. 其中 $L^{31}=L^{32}$, $R^{31}=(R^{32}\oplus P(S(L^{31}\oplus \mathrm{rk}_{31})))>>>8$.

(7) 重复 (2)–(5), 将 (2) 中的故障注入改为第 31 轮处, 可获得 rk_{30}.

(8) 获取 rk_{31} 与 rk_{30} 后, 可将初始密钥搜索空间降至 2^{19}.

4.2.2 密钥推断过程

(1) 根据密钥扩展方案, rk_{30} 是 S^{30} 的左 32 位, $S^{30}=\mathrm{rk}_{30}[0:31]||S^{30}[47:0]$.

(2) 移位之后, $S^{30}=\mathrm{rk}_{30}[29:31]||S^{30}[47:0]||\mathrm{rk}_{30}[0:28]$.

(3) 过 S 盒之后, $S^{30}=s_9(\mathrm{rk}_{30}[29:31]||S^{30}[47])||s_8(S^{30}[46:43])||S^{30}[42:0]||\mathrm{rk}_{30}[0:28]$.

(4) 加轮常量之后, $S^{31}=s_9(\mathrm{rk}_{30}[29:31]||S^{30}[47])||s_8(S^{30}[46:43])||S^{30}[42:22]||S^{30}[21:17]\oplus 31||S^{30}[16:0]||\mathrm{rk}_{30}[0:28]$.

(5) rk_{31} 是 S^{31} 的左 32 位, 可以看出 S^{31} 已知 $32+29=61$ 位, 剩余 19 位未知, 可以通过穷举的方式获得初始密钥.

4.2.3 仿真实验结果与复杂度分析

在普通笔记本电脑 (CPU 为 Core(TM) i5-5200U 2.20 GHz, 内存 12 G) 上使用 C 语言 (Visual Studio 2010) 编程实现了本文的攻击方法, 计算机模拟故障注入过程. 进行 8 次攻击实验, 其结果如表5所示. 从实验结果可以看出, 恢复出 80 bit 的 LBlock 密钥实际需要约 5 个错误密文和一个正确密文.

本节对 LBlock 算法进行了攻击, 注入 2 次宽度为 16 bit 的故障可恢复一轮轮密钥. 恢复 rk_{31} 与 rk_{30} 最少需 4 次宽度为 16 bit 的故障. 之后求解初始密钥需经过 2^{19} 次加密运算, 计算复杂度为 2^{19}.

表 5 模拟实验结果
Table 5 Simulation results

序号	攻击第 32 轮所用故障数	攻击第 31 轮所用故障数	故障总数
1	3	2	5
2	3	3	6
3	2	2	4
4	4	2	6
5	3	2	5
6	3	3	6
7	2	3	5
8	3	2	5

4.3 结果对比

Zhao 等 [18] 最早提出对 LBlock 的差分故障攻击, 在 25–31 轮以 bit 为单位注入故障, 成功恢复密钥. Jeong 等人 [19] 在 29 轮以 nibble 为单位注入五次故障, 经 2^{25} 次加密运算后恢复主密钥; 在 29 轮以 nibble 为单位注入七次故障, 经 2^{30} 次加密运算后恢复主密钥. Wei 等 [20] 在 27 到 29 轮以 nibble 为单位注入故障, 平均 4.3 个故障可恢复一轮子密钥, 13.3 个故障可恢复主密钥. 潘晓中等 [21] 在算法的最后 3 轮以 nibble 为单位注入故障, 恢复最后一轮子密钥全部信息需要的故障数目为 16.62; 为得到主密钥全部信息, 平均需要故障的数目为 46.8. 本文的创新点在于以改进的差分故障攻击模型来恢复密钥, 平均 5 次宽度为 16 bit 的故障可恢复最后两轮轮密钥, 将恢复主密钥的计算复杂度降为 2^{19}.

5 结束语

本文给出差分故障分析的优化方案, 提高差分故障攻击的效率. 进而将优化方案应用于 LBlock 轻量级分组密码算法, 注入 4 次宽度为 16 bit 的故障可恢复最后两轮轮密钥, 将计算复杂度降为 2^{19}. 并且对提出的攻击方法做了仿真实验, 实验结果表明差分故障攻击效果良好.

未来可以研究 SPN 结构的分组密码算法, 将改进的差分故障攻击应用其上.

References

[1] WU W L, ZHANG L. LBlock: A lightweight block cipher[C]. In: Applied Cryptography and Network Security—ACNS 2011. Springer Berlin Heidelberg, 2011: 327–344. [DOI: 10.1007/978-3-642-21554-4_19]

[2] ZCZECHOWIAK P, COLLIER M. TinyIBE: Identity-based encryption for heterogeneous sensor networks. In: Proceedings of International Conference on Intelligent Sensors, Sensor Networks and Information Processing 2009 (ISSNIP 2009). Melbourne, VIC, Australia, 2009: 319–354. [DOI: 10.1109/ISSNIP.2009.5416743]

[3] SINGH S, SHARMA P K, MOON S Y, et al. Advanced lightweight encryption algorithms for IoT devices: Survey, challenges and solutions[J]. Journal of Ambient Intelligence and Humanized Computing, 2017: 1–13. [DOI: 10.1007/s12652-017-0494-4]

[4] WANG B C, HU Y P. Public key cryptosystem based on two cryptographic assumptions[J]. IEE Proceedings-Communications, 2005, 152(6): 861– 865. [DOI: 10.1049/ip-com:20045278]

[5] GUERMAZI A, ABID M. An efficient key distribution scheme to secure data-centric routing protocols in hierarchical wireless sensor networks[J]. Procedia Computer Science, 2011, 5: 208–215. [DOI: 10.1016/j.procs.2011.07.028]

[6] CAO X , KOU W , DU X. A pairing-free identity-based authenticated key agreement protocol with minimal message exchanges[J]. Information Sciences, 2010, 180(15): 2895–2903. [DOI：10.1016/j.ins.2010.04.002]

[7] TENG J K, WU C K. An identity-based group key agreement protocol for low power mobile devices[J]. Chinese Journal of Electronics, 2016, 25(4): 726–733. [DOI: 10.1049/cje.2016.06.038]

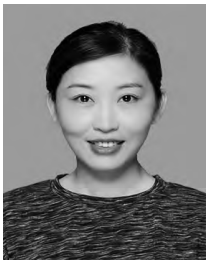
[8] KOCHER F. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[C]. In: Advances in Cryptology—CRYPTO 1996. Springer Berlin Heidelberg, 1996: 104–113. [DOI:10.1007/3-540-68697-5_9]

[9] BONEH D, DEMILLO R A, LIPTON R J. On the importance of checking cryptographic protocals for faults[C]. In: Advances in Cryptology—EUROCRYPT 1997. Springer Berlin Heidelberg, 1997: 37–51. [DOI:10.1007/3-540-

69053-0_4]
[10] KOCHER P, JAFFE J, JUN B. Differential power analysis[C]. In: Advances in Cryptology—CRYPTO 1999. Springer Berlin Heidelberg, 1999: 388–397. [DOI: 10.1007/3-540-48405-1_25]
[11] QUISQUATER J J, SAMYDE D. A new tool for non-intrusive analysis of smart cards based on electro-magnetic emissions: The SEMA and DEMA methods[EB/OL].
http://www.iacr.org/conferences/eurocrypt2000/posterrump.html, 2000.
[12] BIHAM E, SHAMIR A. Differential fault analysis of secret key cryptosystems[C]. In: Advances in Cryptology—CRYPTO 1997. Springer Berlin Heidelberg, 1997: 513–525. [DOI: 10.1007/bfb0052259]
[13] BIEHL I, MEYER B, MÜLLER V. Differential fault analysis on elliptic curve cryptosystems[C]. In: Advances in Cryptology—CRYPTO 2000. Springer Berlin Heidelberg, 2000: 131–146. [DOI: 10.1007/3-540-44598-6_8]
[14] DUSART P, LETOURNEUX G, VIVOLO O. Differential fault analysis on A.E.S[C]. In: Applied Cryptography and Network Security—ACNS 2003. Springer Berlin Heidelberg, 2003: 293–306. [DOI: 10.1007/978-3-540-45203-4_23]
[15] ZHOU Y B. WU W L, XU N N, et al. Differential fault attack on Camellia[J]. Chinese Journal of Electronics, 2009, 18(1): 13–19. [DOI: 10.3724/sp.j.1016.2011.00613]
[16] BIHAM E, GRANBOULAN L, NGUYN P Q. Impossible fault analysis of RC4 and differential fault analysis of RC4[C]. In: Fast Software Encryption—FSE 2005. Springer Berlin Heidelberg, 2005: 359–367. [DOI: 10.1007/11502760_24]
[17] HOJSIK M, RUDOLF B. Differential fault analysis of Trivium[C]. In: Fast Software Encryption—FSE 2008. Springer Berlin Heidelberg, 2005: 158–172. [DOI: 10.1007/978-3-540-71039-4_10]
[18] ZHAO L, NISHIDE T, SAKURAI K. Differential fault analysis of full LBlock[C]. In: Constructive Side-Channel Analysis and Secure Design—COSADE 2012. Springer Berlin Heidelberg, 2012: 135–150. [DOI: 10.1007/978-3-642-29912-4_11]
[19] JEONG K, LEE C, LIM J I. Improved differential fault analysis on lightweight block cipher LBlock for wireless sensor networks[J]. Eurasip Journal on Wireless Communications & Networking, 2013, 2013(1): 151. [DOI: 10.1186/1687-1499-2013-151]
[20] WEI Y C, RONG Y S, WANG X A. New differential fault attack on lightweight cipher LBlock[C]. In: Proceedings of 2016 International Conference on Intelligent Networking and Collaborative Systems (INCoS). Ostrawva, Czech Republic, 2016: 285–288. [DOI: 10.1109/incos.2016.32]
[21] PAN X Z, CHENG L. Differential fault analysis on cipher LBlock based one-round diffusion[J]. Journal of Engineering University of PAP, 2016, 32(6): 43–46.
潘晓中, 程璐. 针对 LBlock 密码单轮扩散的差分故障分析. 武警工程大学学报, 2016, 32(6): 43–46.

作者信息

王涛 (1995–), 山东临沂人, 硕士生在读. 主要研究领域为侧信道攻击.
wt107263@163.com



王永娟 (1972–), 河南开封人, 副教授. 主要研究方向为网络空间安全、密码算法分析.
pinkywyj@163.com

高杨 (1994–), 河南洛阳人, 硕士生在读. 主要研究领域为故障攻击.
gaoyang_1279@126.com

张诗怡 (1993–), 四川峨眉人, 实习研究员. 主要研究领域为密码学中的置换函数.
syzhang1352@163.com

6 附录

LBlock 算法一次攻击数据:

选取明文: 01 23 45 67 89 ab cd ef 密钥: 01 23 45 67 89 ab cd ef fe dc

(1) 首先进行一次正确加密, 得密文: 4b 71 79 d8 eb ee 0c 26.

(2) 在最后一轮引入随机差分故障, 得到故障密文: 5a 60 68 c9 de 5b 2b 14.

(3) 将故障密文同正确密文异或得到: 11 11 11 11 35 b5 27 32, 其中 35 b5 27 32 经过后得到 b3 55 32 27 为 S 盒输出差分, 11 11 11 11 就是引入的故障差分, 即输入差分.

(4) 引入不同的随机故障, 得到另一组故障密文: e1 db d3 72 fa 7a 76 38. 可以得到差分密文: aa aa aa aa 11 94 7a 1e. 其中 11 94 7a 1e 经过 P^{-1} 后得到 91 41 17 ea 为 S 盒输出差分, aa aa aa aa 就是引入的故障差分, 即输入差分.

(5) 结合表4确定出 S 盒未加故障时的输入: 每一个未知的输入候选值均只有一个交集, 这样就唯一确定出 S 盒的输入为: be a4 2f e1. 将其与 4b 71 79 d8 异或得第 32 轮轮密钥 f5 d5 56 39.

(6) 解密最后一轮后, 重复 (2)–(5), 在第 31 轮引入故障, 可得第 31 轮轮密钥 6c d2 cb 29.

(7) 根据密钥扩展方案, 穷举未知的 16 bit, 可得初始密钥 01 23 45 67 89 ab cd ef fe dc.