软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2021,32(9):2837−2848 [doi: 10.13328/j.cnki.jos.005972]
©中国科学院软件研究所版权所有.
E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

概率积分及其在 PUFFIN 算法中的应用*

尚方舟, 孙 兵, 刘国强, 李 超

(国防科技大学 文理学院,湖南 长沙 410073)

通讯作者: 李超, E-mail: academic_lc@163.com

摘 要: 积分分析是一种针对分组密码十分有效的分析方法,其通常利用密文某些位置的零和性质构造积分区分器.基于高阶差分理论,可通过研究密文与明文之间多项式的代数次数来确定密文某些位置是否平衡.从传统的积分分析出发,首次考虑常数对多项式首项系数的影响,提出了概率积分分析方法,并将其应用于 PUFFIN 算法的安全性分析.针对 PUFFIN 算法,构造了 7 轮概率积分区分器,比已有最好的积分区分器轮数长 1 轮.进一步,利用构造的概率积分区分器,对 9 轮 PUFFIN 算法进行密钥恢复攻击.该攻击可恢复 92 比特轮密钥,攻击的数据复杂度为 $2^{24.8}$ 个选择明文,时间复杂度为 $2^{35.48}$ 次 9 轮算法加密,存储复杂度为 2^{20} 个存储单元.

关键词: 积分分析;高阶差分分析;概率积分;PUFFIN 算法

中图法分类号: TP309

中文引用格式: 尚方舟,孙兵,刘国强,李超.概率积分及其在 PUFFIN 算法中的应用.软件学报,2021,32(9):2837−2848. http://www.jos.org.cn/1000-9825/5972.htm

英文引用格式: Shang FZ, Sun B, Liu GQ, Li C. Probability integral cryptanalysis and its application on PUFFIN. Ruan Jian Xue Bao/Journal of Software, 2021,32(9):2837−2848 (in Chinese). http://www.jos.org.cn/1000-9825/5972.htm

Probability Integral Cryptanalysis and Its Application on PUFFIN

SHANG Fang-Zhou, SUN Bing, LIU Guo-Qiang, LI Chao

(College of Liberal Arts and Sciences, National University of Defense Technology, Changsha 410073, China)

Abstract: Integral cryptanalysis is an effective method of block cipher analysis, and the integral distinguisher is usually constructed using a zero-sum property of some positions in the ciphertext. Based on the theorem of higher-order differential attack, the order of plaintexts can be exploited, to determine if some positions of the ciphertext are balanced. Inspired by the conventional integral cryptanalysis, the influence of constant on the leading-coefficient of polynomial is considered and the construction of probability integral distinguisher as well as the attack method are proposed in this study. When applied to PUFFIN, a 7-round probability integral distinguisher is constructed and used to mount a 9-round attack, and this attack can recover 92-bit round key. The data/time complexity is $2^{24.8}$ chosen plaintexts, and $2^{35.48}$ 9 round encryptions, and the space complexity is 2^{20}.

Key words: integral cryptanalysis; higher-order differential attack; probability integral cryptanalysis; PUFFIN

积分分析[1]是针对分组密码十分有效的分析方法之一,它是由 Square 攻击[2]、Multiset 攻击[3]和 Saturation 攻击[4]发展而来的一种分析方法.

积分分析分为两个阶段:第 1 阶段为构造尽可能长的积分区分器,第 2 阶段为利用构造的积分区分器进行密钥恢复攻击.其中,第 1 阶段是积分分析的关键所在,主要利用具有特殊结构的明文加密后的密文在某些字节或比特上异或和为常数,通常来说该常数为 0;第 2 阶段进行密钥恢复攻击,若猜测的密钥为正确密钥,则密文反解后对应字节或比特异或和一定为常数;若猜测的密钥为错误密钥,则对应异或和随机分布.

* 基金项目: 国家自然科学基金(61672530, 61702537, 61772545)
Foundation item: National Natural Science Foundation of China (61672530, 61702537, 61772545)
收稿时间: 2018-12-27; 修改时间: 2019-05-29; 采用时间: 2019-10-13

© 中国科学院软件研究所 http://www.jos.org.cn

积分分析是一种选择明文攻击,早期主要应用于基于字节设计的密码算法.2008 年,Z'aba 等人对早期的积分分析进行扩展,首次提出了基于比特的积分分析[5],并应用于 Noekeon[6],Serpent[7]和 PRESENT[8]算法.为更加充分地利用非线性组件信息,Todo 在 EUROCRYPT 2015 上提出了可分性[9]理论,该理论利用计算机搜索更长轮数的积分区分器.接着,Todo 等人[10]在 FSE 2016 上将可分性概念应用到基于比特的分组密码,并对 SIMON 算法积分区分器进行搜索.进一步,在 ASIACRYPT 2016 上,向泽军等人[11]首次利用混合整数线性规划(mixed-integer linear programming,简称 MILP)模型对可分性进行刻画,针对 6 种轻量级分组密码积分区分器进行搜索,得到了更优的结果.

积分分析也可以看成是差分分析的扩展,其与高阶差分分析在建立区分器上具有某些联系.如果在高阶差分中考虑异或差分,那么建立具有零和性质的 d 阶积分区分器与密文某些比特代数次数最大为 d–1 是等价的.2013 年,吴生宝等人[12]首次利用高阶差分分析理论,结合 S 盒具体的代数性质,构造算法新的积分区分器.针对 PRESENT 算法,他们利用 S 盒及线性层的性质,改进了原有的积分攻击结果.

在传统的积分分析中,利用一组选择明文来构造积分区分器时,区分器末端得到的密文异或和为零都是以概率为 1 成立的.那么,能否构造概率不为 1 的有效积分区分器呢?在文献[13]中,Knudsen 等人指出:"与差分类似,积分也可以是有概率的."这揭示了概率积分存在的可能性.本文对该问题进行深入研究,考虑常数对布尔函数首项系数的影响,阐述了构造概率积分区分器的具体方法,并给出了利用概率积分区分器恢复密钥的理论模型.为验证概率积分分析方法的有效性,本文以 PUFFIN 算法为例进行分析,构造了 7 轮概率积分区分器,并利用构造的概率积分区分器,对 9 轮 PUFFIN 算法进行密钥恢复攻击.

1 概率积分分析方法

本节从传统的积分分析出发,提出概率积分区分器的构造以及概率积分分析方法.

1.1 积分分析

令 f 为 $\mathbb{F}_2^n$ 到 $\mathbb{F}_2$ 的 n 元布尔函数,其代数正规型可以表示为

$$f(x_1,x_2,...,x_n)=\sum_{I\in P(N)}a_I\left(\prod_{i\in I}x_i\right)=\sum_{I\in P(N)}a_Ix^I.$$

这里,$N=\{1,2,\ldots,n\}$,$P(N)$表示 N 的幂集,即 N 的所有子集构成的集合.加法为 $\mathbb{F}_2$ 中的加法运算,即模 2 加运算.

非零布尔函数 f 的代数正规型中,系数非零项所含有最多变元的个数称为 f 的代数次数,记为 $\deg(f)$,即:

$$\deg(f)=\max\{|I||a_I\neq 0,I\in P(N)\}.$$

规定:零函数的代数次数为 0.

在积分分析中,通过判定某个位置的平衡性来构造积分区分器.基于高阶差分思想,可以通过有限域上多项式函数的首项系数取值来判定某个位置的平衡性.

引理 1[14]. 设 f 为 n 元布尔函数,其代数正规型为

$$f(x_1,x_2,...,x_n)=\sum_{I\in P(N)}a_Ix^I,a_I\in\mathbb{F}_2,$$

则对任意 $I\in P(N)$,均有:

$$a_I=\sum_{x\in\mathbb{F}_2^n,supp(x)\subseteq I}f(x),$$

其中,$supp(x)=\{1\leqslant i\leqslant n|x_i=1\}$.

引理 1 说明:要确定某个位置密文是否平衡,可通过研究该位置密文与明文之间多项式函数的首项系数来判断.设算法分组长度为 n,其中的 s 个比特遍历$\{0,1\}^s$,记作 $x_0,x_1,\ldots,x_{s-1}$;其余比特为常数,记作 $c_0,c_1,\ldots,c_{n-s-1}$,$0<s\leqslant n$.密文每个比特的值都是关于 $x_0,x_1,\ldots,x_{s-1},c_0,c_1,\ldots,c_{n-s-1}$ 的多项式函数,不妨记作 $f(x_0,x_1,\ldots,x_{s-1},c_0,c_1,\ldots,c_{n-s-1})$,易知,$f$ 是从 $\mathbb{F}_2^s$ 到 $\mathbb{F}_2$ 的一个映射.若密文某个比特位置的表达式 f 对任意的 $c_0,c_1,\ldots,c_{n-s-1}$ 都满足:

© 中国科学院软件研究所 http://www.jos.org.cn

$$\deg(f)\leqslant s-1,$$

则对此位置上出现的所有 2^s 个元素求和为 0,此时认为该比特是平衡的.

1.2 概率积分区分器的构造

在传统的积分分析中,一般只考虑 $\deg(f)\leqslant s-1$ 的比特位置,将 $\deg(f)=s$ 的比特位置视为无用信息.而概率积分的思想是考虑多项式代数次数达到最大时,常数 $c_0,c_1,\ldots,c_{n-s-1}$ 对首项系数的影响.

令 $p(c_i=1)=p_i,0\leqslant i\leqslant n-s-1$,由此可计算得到首项系数为 0 的概率,即平衡概率.若平衡概率与 50%存在偏差,那么就认为此情况能够与随机情况区分,就说构造了一个概率积分区分器.

例 1:若某一比特位置的多项式函数可写作:

$$y=f(x_0,x_1,c_0,c_1)=c_0c_1x_0x_1+c_0x_0+c_1x_1+c_0c_1,$$

则当且仅当 $c_0=c_1=1$ 时,该比特位置 $\deg(f)=2$.根据引理 1,该比特位置不平衡.因此,不平衡概率为 p_0p_1,平衡概率为 $1-p_0p_1$.

由此,我们引入概率布尔多项式的概念.

定义 1. 设 f 为 n 变元布尔函数,其对应的概率布尔多项式 $\hat{f}$ 可表示为

$$\hat{f}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{a}_I x^I,$$

其中,$\hat{a}_I=p(a_I=1),0\leqslant\hat{a}_I\leqslant 1$.

在概率积分区分器中,常数 $c_0,c_1,\ldots,c_{n-s-1}$ 的概率 $p_0,p_1,\ldots,p_{n-s-1}$ 可以通过选择明文进行控制.本文为体现概率积分区分器存在的普适性,考虑最一般的情况,即令所有涉及到的常数随机生成,故其取 0 或 1 的概率相同,均为 0.5.在例 1 中,令 $p_0=p_1=0.5$,则其概率布尔多项式表示为

$$\hat{y}=\hat{f}(x_0,x_1)=0.25x_0x_1+0.5x_0+0.5x_1+0.25.$$

并由此得到概率布尔多项式首项系数与平衡概率之间的关系:

推论 1. 设 n 元概率布尔多项式 $\hat{f}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{a}_I x^I,0\leqslant a_I\leqslant 1$,则:

$$p\left(\sum_{x\in\mathbb{F}_2^n,\sup p(x)\subseteq I}\hat{f}(x)=0\right)=1-\hat{a}_I,$$

其中,$supp(x)=\{1\leqslant i\leqslant n|x_i=1\}$.

推论 1 表明:由某一比特位置的概率布尔多项式,可以求得该比特位置的平衡概率.

下面在布尔多项式系数彼此独立的条件下,给出概率布尔多项式的运算法则.

性质 1(加法法则). 设 n 元概率布尔多项式:

$$\hat{f}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{a}_I x^I,\hat{g}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{b}_I x^I,$$

则:

$$\hat{f}(x_1,x_2,...,x_n)+\hat{g}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}(\hat{a}_I+\hat{b}_I-2\hat{a}_I\hat{b}_I)x^I.$$

证明:由概率布尔多项式的定义可知:$f(x_1,x_2,\ldots,x_n)$中,单项 x^I 系数为 1 的概率为 $\hat{a}_I$,系数为 0 的概率为 $1-\hat{a}_I$.同理,多项式 $g(x_1,x_2,\ldots,x_n)$中,单项 x^I 系数为 1 的概率为 $\hat{b}_I$,系数为 0 的概率为 $1-\hat{b}_I$.因此,若要多项式 $f(x_1,x_2,\ldots,x_n)+g(x_1,x_2,\ldots,x_n)$中单项 x_I 系数为 1,则多项式 f,g 中有且仅有一个包含 x_I,x_I 系数为 1 的概率为

$$\hat{a}_I(1-\hat{b}_I)+(1-\hat{a}_I)\hat{b}_I=\hat{a}_I+\hat{b}_I-2\hat{a}_I\hat{b}_I.$$

故而:

$$\hat{f}(x_1,x_2,...,x_n)+\hat{g}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}(\hat{a}_I+\hat{b}_I-2\hat{a}_I\hat{b}_I)x^I.$$ □

下面通过一个例子,直观说明概率布尔多项式的乘法法则.

© 中国科学院软件研究所 http://www.jos.org.cn

例 2:设 $\hat{f}(x_1,x_2,x_3)=0.3x_1x_2+x_3,\hat{g}(x_1,x_2,x_3)=0.5x_1x_2x_3+0.2x_2x_3$,计算 $\hat{f}(x_1,x_2,x_3)\cdot\hat{g}(x_1,x_2,x_3)$.

若不考虑系数为 0 的单项,则 f,g 所有可能的情况如下:

$$\begin{array}{llll} f_1=x_1x_2+x_3 & p(f_1)=0.3\times1=0.3 & g_1=x_1x_2x_3+x_2x_3 & p(g_1)=0.5\times0.2=0.1\\ f_2=x_1x_2 & p(f_2)=0.3\times0=0 & g_2=x_1x_2x_3 & p(g_2)=0.5\times0.8=0.4\\ f_3=x_3 & p(f_3)=0.7\times1=0.7 & g_3=x_2x_3 & p(g_3)=0.5\times0.2=0.1\\ f_4=0 & p(f_4)=0.7\times0=0 & g_4=0 & p(g_4)=0.5\times0.8=0.4 \end{array}$$

将 f_i,g_i(i=1,2,3,4)逐个相乘(其中,f_2,f_4 概率为 0,此处省略):

$$\begin{array}{ll} f_1\cdot g_1=x_1x_2x_3+x_2x_3 & p(f_1\cdot g_1)=0.3\times0.1=0.03\\ f_1\cdot g_2=0 & p(f_1\cdot g_2)=0.3\times0.4=0.12\\ f_1\cdot g_3=x_1x_2x_3+x_2x_3 & p(f_1\cdot g_3)=0.3\times0.1=0.03\\ f_1\cdot g_4=0 & p(f_1\cdot g_4)=0.3\times0.4=0.12\\ f_3\cdot g_1=x_1x_2x_3+x_2x_3 & p(f_3\cdot g_1)=0.7\times0.1=0.07\\ f_3\cdot g_2=x_1x_2x_3 & p(f_3\cdot g_2)=0.7\times0.4=0.28\\ f_3\cdot g_3=x_2x_3 & p(f_3\cdot g_3)=0.7\times0.1=0.07\\ f_3\cdot g_4=0 & p(f_3\cdot g_4)=0.7\times0.4=0.28 \end{array}$$

统计以上结果:

$$\begin{array}{l} p(f\cdot g=x_1x_2x_3+x_2x_3)=p(f_1\cdot g_1)+p(f_1\cdot g_3)+p(f_3\cdot g_1)=0.03+0.03+0.07=0.13\\ p(f\cdot g=x_1x_2x_3)=p(f_3\cdot g_2)=0.28\\ p(f\cdot g=x_2x_3)=p(f_3\cdot g_3)=0.07\\ p(f\cdot g=0)=p(f_1\cdot g_2)+p(f_1\cdot g_4)+p(f_3\cdot g_4)=0.12+0.12+0.28=0.52 \end{array}$$

由此得到 $\hat{f}(x_1,x_2,x_3)\cdot\hat{g}(x_1,x_2,x_3)$ 的 4 种情形以及相应的概率.

性质 2(乘法法则). 若考虑系数为 0 的单项,则 n 元布尔函数中的单项总数为 2^n.由于每一单项的系数可能为 1 可能为 0,故 n 元布尔函数可能出现的全部情形有 2^{2^n} 种,每种情形的概率为其概率布尔多项式中各单项系数的乘积.算法 1 说明概率布尔多项式具体的乘法运算法则,该算法复杂度最大为 $2^{2^{n+1}+1}$ 次乘法运算.

算法 1. 概率布尔多项式乘法运算法则.

输入: $\hat{f}(x_1,x_2,...,x_n),\hat{g}(x_1,x_2,...,x_n)$;

输出:乘积结果 $\hat{h}(x_1,x_2,...,x_n)$.

1: 令 $\hat{f}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{a}_Ix^I,\hat{g}(x_1,x_2,...,x_n)=\sum_{I\in P(N)}\hat{b}_Ix^I$

2: 列举 $\hat{f}$ 所有可能的情况 f_i 以及概率 $p_i,i=1,2,...,2^{2^n}$,p_i 为 f_i 中各系数之积;

3: 列举 $\hat{g}$ 所有可能的情况 g_j 以及概率 $q_j,j=1,2,...,2^{2^n}$,q_j 为 g_j 中各系数之积;

4: **for** i=1 to 2^{2^n} **do**

5: **for** j=1 to 2^{2^n} **do**

6: $h_{(i-1)\cdot2^{2^n}+j}=f_i\cdot g_j$

7: $P_{(i-1)\cdot2^{2^n}+j}=p_i\cdot q_j$ //P_k 为 h_k 出现的概率, $k=1,2,...,2^{2^{n+1}}$

8: **end for**

9: **end for**

10: **return** $h_k,P_k(k=1,2,...,2^{2^{n+1}})$

由概率布尔多项式及其运算法则,可以通过如下方式构造概率积分区分器.

性质 3(概率积分区分器构造方法). 针对分组长度为 n 的分组密码算法,令活跃比特数为 s,记密文某一比特布尔多项式 f 中,首项 $x_0x_1\ldots x_{s-1}$ 的系数 a_I 为关于 $c_0,c_1,\ldots,c_{n-s-1}$ 的多项式,其中,I={0,1,…,s−1}.取 $p_0=p_1=\ldots=p_{n-s-1}=$

© 中国科学院软件研究所 http://www.jos.org.cn

0.5,则可得到 $\hat{a}_I$,由推论 1 得到该比特平衡概率为 $1-\hat{a}_I$.若 $1-\hat{a}_I \neq 0.5$,则可构造一个概率积分区分器.

1.3 概率积分分析模型

传统积分攻击模型与概率积分攻击模型具有一定的相似之处,但在具体的复杂度计算上存在差异.先给出需要用到的符号定义.

- b:平衡/概率平衡比特数;
- k:需要猜测的密钥比特数;
- p:明文集合平衡概率;
- N:实验次数.

在传统积分分析中,对于每次实验,若利用 m 组明文集合(根据区分器,每组明文均为部分位置遍历,其余位置固定为常数)进行攻击,当剩余密钥量的期望值 $(2^k-1)(2^{-b})^m<1$ 时,则此次攻击成功.而在概率积分分析中,m 组明文集合全平衡的概率为 p^m,也就是在 $\lceil p^{-m} \rceil$ 次实验中,至少有一次 m 组明文集合全平衡.而只有当每次实验的 m 组明文全平衡时,此次实验才能成功猜测正确密钥.为刻画正确密钥与错误密钥之间的区分性,本文引入显著度的概念,记为 C.若进行 $N=\lceil C\cdot p^{-m} \rceil$ 次实验,则 m 组明文集合全平衡的次数至少为 C,这使得正确密钥与错误密钥之间的区分更加显著.由此,概率积分分析需满足如下定理.

定理 1. 在概率积分分析中,若进行 N 次实验,每次实验利用 m 组明文集合进行攻击,当 $N\cdot 2^k\cdot(2^{-b})^m<1$ 时,此次攻击成功.

证明:不妨设 N 次实验中,明文集合全部平衡的实验次数为 N_1,不全平衡的实验次数为 N_2,$N_1+N_2=N$.

在 N_1 次实验中,与传统积分分析相同,最终剩余的正确密钥量为 1,错误密钥量为 $(2^k-1)(2^{-b})^m$;在 N_2 次实验中,与随机情况相同,最终剩余的密钥量均为 $2^k\cdot(2^{-b})^m$.因此,最终剩余的总密钥量 K 为

$$K=N_1\cdot[1+(2^k-1)\cdot(2^{-b})^m]+N_2\cdot 2^k\cdot(2^{-b})^m=N_1+N_1\cdot(2^k-1)\cdot(2^{-b})^m+N_2\cdot 2^k\cdot(2^{-b})^m\approx N_1+N\cdot 2^k\cdot(2^{-b})^m.$$

故而,当除正确密钥出现次数 N_1 外的候选密钥个数 $N\cdot 2^k\cdot(2^{-b})^m$ 小于 1 时,此次攻击成功.证毕. □

2 运用概率积分攻击 PUFFIN 算法

本节利用概率积分的思想,对 PUFFIN 算法抵抗积分攻击的安全性进行分析.首先,针对 PUFFIN 算法构造 6 轮概率积分区分器;接着,利用高阶积分理论,将 6 轮区分器扩展为 7 轮概率积分区分器,并对 9 轮 PUFFIN 算法进行攻击.

2.1 PUFFIN算法描述

PUFFIN[15]算法是一种硬件实现良好的轻量级 SPN 结构算法,分组规模为 64 比特,密钥规模为 128 比特,轮数为 32 轮.算法的 64 比特明文(中间状态、轮密钥及密文)排列为 4 行 16 列的二维数组形式,即 $(p_0,p_1,\ldots,p_{63})$ 可表示成 $V_0,V_1,\ldots,V_{15}$ 共 16 个向量,其中,$V_i=(p_{4i},p_{4i+1},p_{4i+2},p_{4i+3})^T$,$0\leqslant i\leqslant 15$,如图 1 所示.

V_0	V_1	V_2	V_3	V_4	V_5	V_6	V_7	V_8	V_9	V_{10}	V_{11}	V_{12}	V_{13}	V_{14}	V_{15}
p_0	p_4	p_8	p_{12}	p_{16}	p_{20}	p_{24}	p_{28}	p_{32}	p_{36}	p_{40}	p_{44}	p_{48}	p_{52}	p_{56}	p_{60}
p_1	p_5	p_9	p_{13}	p_{17}	p_{21}	p_{25}	p_{29}	p_{33}	p_{37}	p_{41}	p_{45}	p_{49}	p_{53}	p_{57}	p_{61}
p_2	p_6	p_{10}	p_{14}	p_{18}	p_{22}	p_{26}	p_{30}	p_{34}	p_{38}	p_{42}	p_{46}	p_{50}	p_{54}	p_{58}	p_{62}
p_3	p_7	p_{11}	p_{15}	p_{19}	p_{23}	p_{27}	p_{31}	p_{35}	p_{39}	p_{43}	p_{47}	p_{51}	p_{55}	p_{59}	p_{63}

Fig.1 Block state of PUFFIN

图 1 PUFFIN 算法分组比特顺序

PUFFIN 算法加密过程可表示为

© 中国科学院软件研究所 http://www.jos.org.cn

$$\prod_{r=1}^{32}(P64\circ\sigma_{k_r}\circ\gamma)\circ P64\circ\sigma_{k_0}.$$

首先进行密钥白化以及 $P64$ 线性变换,然后进行 32 轮轮函数迭代.PUFFIN 算法的轮函数包含以下 3 个变换:非线性层γ,密钥加σ和线性变换 $P64$.

(1) 非线性层γ:由 16 个相同的 4×4 S 盒并置组成,每列(V_i)通过一个 S 盒.S 盒映射见表 1;

(2) 密钥加σ:64 比特轮密钥与 64 比特中间状态进行异或运算;

(3) 线性层 $P64$:64 比特置换,其映射见表 2.

Table 1 S box map (in hexadecimal)

表 1 PUFFIN 算法 S 盒映射(16 进制表示)

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

Table 2 $P64$ map

表 2 $P64$ 映射

	0	1	2	3	4	5	6	7
0	13	2	60	50	51	27	10	36
1	25	7	32	61	1	49	47	19
2	34	53	16	22	57	20	48	41
3	9	52	6	31	62	30	28	11
4	37	17	58	8	33	44	46	59
5	24	55	63	38	56	39	15	23
6	14	4	5	26	18	54	42	45
7	21	35	40	3	12	29	43	64

值得注意的是,PUFFIN 算法中 S 盒及 $P64$ 均为对合变换,即 $S(S(x))=x$,$P64(P64(y))=y$,其中, $x\in\mathbb{F}_2^4$, $y\in\mathbb{F}_2^{64}$.

目前对 PUFFIN 的安全性分析主要有差分分析[16]、线性分析[17]和积分分析[18].文献[18]给出了对 PUFFIN 算法的首个积分攻击结果;文献[19,20]给出了 PUFFIN 算法 6 轮积分区分器,并对 8 轮 PUFFIN 算法进行攻击;文献[21]利用混合整数线性规划搜索得到 PUFFIN 算法 8 轮积分区分器,并对 10 轮 PUFFIN 算法进行攻击.

2.2 PUFFIN算法6/7轮区分器

定理 2. 设明文 $P=(p_0,p_1,\ldots,p_{63})$,当$(p_6,p_{24},p_{31},p_{60})$遍历$\{0,1\}^4$时,6 轮加密后密文中高概率平衡比特位置如图 2 所示(单位:%).

					90.7					85.3	85.6		90.2	85.0	
				88.4	87.3			87.1		84.6			89.6		
89.7			98.4		98.6			89.9	99.6						
			89.8	82.3					90.5		87.3		98.6		

Fig.2 High probability balance position of ciphertext after 6 rounds

图 2 6 轮加密后密文高概率平衡位置

证明:当输入明文的活跃位置为 p_6,p_{24},p_{31},p_{60} 时,状态可用下图表示(方格内数字表示比特顺序,每一轮开始时重新编号).

0	4	8	12	16	20	24	28	32	36	40	44	48	52	56	60
1	5	9	13	17	21	25	29	33	37	41	45	49	53	57	61
2	6	10	14	18	22	26	30	34	38	42	46	50	54	58	62
3	7	11	15	19	23	27	31	35	39	43	47	51	55	59	63

© 中国科学院软件研究所 http://www.jos.org.cn

经过 *P*64 后,活跃位置将位于同一列:

12	50	24	0	33	56	8	61	36	32	23	55	13	17	20	11
1	26	6	48	52	19	51	29	16	43	54	38	3	53	34	28
59	9	31	46	15	47	5	27	57	45	62	14	4	41	39	42
49	35	60	18	21	40	30	10	7	58	37	22	25	44	2	63

再经过第 1 轮的非线性层γ后,状态为:

0	4	y_0	12	16	20	24	28	32	36	40	44	48	52	56	60
1	5	y_1	13	17	21	25	29	33	37	41	45	49	53	57	61
2	6	y_2	14	18	22	26	30	34	38	42	46	50	54	58	62
3	7	y_3	15	19	23	27	31	35	39	43	47	51	55	59	63

记这 4 个位置的变量分别为 y_0,y_1,y_2,y_3,再利用算法 2 运算得到经过前 5 轮加密以及第 6 轮非线性层和密钥加运算后每一比特位置概率布尔多项式具体表达式.

算法 2. 概率布尔多项式求解过程.

输入:*state*[8]=y_0,*state*[9]=y_1,*state*[10]=y_2,*state*[11]=y_3,其余 *state*[*i*]=0.5;

输出:*state*[*j*](*j*=0,1,…,63).

1: **for** *i*=1 to 5 **do**

2: *P*(*state*); //对 *state* 进行线性变换

3: *Gamma*(*state*); //对 *state* 每一列通过 S 盒变换

4: *Add-key*(*state*); //对 *state* 每一比特进行密钥异或运算

5: **end for**

算法 2 中,*state* 的每一个比特位置均用概率布尔多项式表示,初始状态的第 8 个~第 11 个位置设为变量,其余位置为常数 0.5.依次经过第 1 轮线性变换、第 2 轮~第 5 轮轮函数以及第 6 轮非线性层和密钥加运算.其中,S 盒具体表示如图 3 所示.运算均遵循概率布尔多项式运算法则,密钥异或操作是将概率布尔多项式中常数项重置为 0.5.由于在加密过程中将所有轮密钥视作随机的,取值为 0 或为 1 的概率相同,均为 0.5,因此对于密钥加过程置常数为 0.5 这一操作,使得该方法适用于所有密钥.为验证该步骤的正确性,基于密钥扩展算法,我们进行了 10 000 000 次实验,每次实验随机选定种子密钥,计算所有轮密钥每一比特位置为 1 的概率,并将该概率代入算法 2 中进行实验,得到的结果与"常数重置"方法基本完全一致,误差小于 0.1%.这是由于该方法主要利用概率布尔多项式的首项系数,因此末尾常数项对于首项系数影响很小.为简化算法,故而密钥异或操作是将概率布尔多项式中的常数项重置为 0.5.

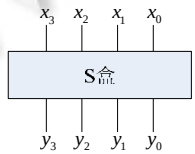

Fig.3 Input and output of S box

图 3 S 盒的输入输出

该算法的计算复杂度主要集中在 Gamma 函数中概率布尔多项式之间的乘法,由于 PUFFIN 算法采用 4 比特 S 盒,由算法 1 可知,每次乘法的计算复杂度至多为 $2^{2^{4+1}+1}=2^{33}$.该 S 盒共包含 29 次乘法运算,故求解 PUFFIN 算法输出每一比特概率布尔多项式表达式的计算复杂度至多为 $5\times16\times29\times2^{33}\approx2^{44.18}$.本文在具体实现过程中,通过设置阈值的方式降低复杂度,即概率低于 0.000 5 时,将此概率置为 0.阈值越小,算法误差越小,但复杂度越大.

© 中国科学院软件研究所 http://www.jos.org.cn

由于该方法计算复杂度较高,对于基于变元个数多于 4 的 S 盒设计的算法,求解效果不佳.图 4 表示经过算法 2 后可求得的概率布尔多项式首项系数(单位:10^{-2}).这里,首项统一视为 $y_0y_1y_2y_3$:

$$y_0 = x_0x_1x_2 + x_0x_1x_3 + x_0x_1 + x_0x_2 + x_0x_3 + x_1x_2 + x_2x_3 + 1$$
$$y_1 = x_0x_1x_3 + x_0x_1 + x_0 + x_1x_2 + x_1 + x_3$$
$$y_2 = x_0x_1x_2 + x_0x_2x_3 + x_1x_2x_3 + x_1x_3 + x_1 + x_2x_3 + x_2 + 1$$
$$y_3 = x_0x_1x_3 + x_0x_1 + x_0x_2 + x_0 + x_1x_2x_3 + x_1x_2 + x_1 + x_2x_3 + 1$$

				12.9	15.0						1.4		11.6	9.3	
				9.8	17.7						1.4		10.4	10.1	
				10.2	12.7						1.6		15.4	9.5	
				12.7	14.7						1.4		14.4	10.3	

Fig.4 Highest order coefficient of probability boolean polynomial

图 4 概率布尔多项式首项系数

再经过第 6 轮的非线性层 P64,由推论 1 可得到此 20 个比特位置平衡的概率(单位:%).

					90.7					85.3	85.6		90.2	85.0	
				88.4	87.3			87.1		84.6			89.6		
89.7			98.4		98.6			89.9	99.6						
			89.8	82.3					90.5		87.3		98.6		

综上讨论,当(p_6,p_{24},p_{31},p_{60})遍历$\{0,1\}^4$时,得到 6 轮加密后密文 20 个位置的平衡概率.证毕. □

为验证该 6 轮积分区分器的准确性,我们考虑布尔函数 $y=f(k,m_1,m_2)$,当 m_1 遍历$\{0,1\}^4$且 m_2 在$\{0,1\}^{60}$中等概率取值时,对固定的密钥 k 是否存在高概率零和区分器.根据定义,其相对于密钥 k 是 0 和区分器的概率是:

$$p_k = \frac{1}{2^{60}}\#\left\{m_2 \in \{0,1\}^{60} : \bigoplus_{m_1 \in \{0,1\}^4} f(k,m_1,m_2) = 0\right\}.$$

这样,就可以通过将 f 看作黑盒,对于 M 个随机选择的密钥 k:k_1,k_2,…,k_M,随机选定 N 个 m_2: $m_2^{(1)}, m_2^{(2)}, ..., m_2^{(N)}$,统计:

$$p_{k_i} = 1 - \frac{1}{N}\sum_{i=1}^{N}\left[\bigoplus_{m_1 \in \{0,1\}^4} f(k_i, m_1, m_2^{(i)})\right],$$

并核实 $p_{k_1}, p_{k_2}, ..., p_{k_M}$ 是否集中在一个聚集在 1 附近的小区域之内.

这样,就可通过简单的实验验证该结果(感谢金晨辉教授对本实验方法的指导).

取 M=100,N=8000,得到论文中 20 个比特位置的平衡概率,如图 5 所示(单位:%).

					93.0					89.4	89.4		92.0	88.7	
				89.7	89.7			90.1		88.3			92.2		
92.4			99.4		99.3			92.9	99.7						
			92.7	87.2					93.4		88.7		99.5		

Fig.5 Experiment to verify the 7-round integral distinguisher

图 5 实验验证 6 轮积分区分器

进一步,根据高阶积分的思想,在 6 轮概率积分区分器前加一轮,可将其扩展为 7 轮的概率积分区分器.

定理 3. 当明文的 16 个比特{p_5,p_8,p_9,p_{10},p_{11},p_{26},p_{27},p_{28},p_{29},p_{30},p_{35},p_{42},p_{50},p_{51},p_{61},p_{63}}遍历$\{0,1\}^{16}$时,7 轮 PUFFIN 算法加密后的密文平衡位置与定理 2 所述的 6 轮概率积分区分器输出概率平衡位置相同.

© 中国科学院软件研究所 http://www.jos.org.cn

定理 3 说明:6 轮概率积分区分器中涉及的 20 个高概率平衡位置,在扩展的 7 轮概率积分区分器中仍然高概率平衡.文献[15]可证明这一命题.算法加密 7 轮后,由于遍历位置增多,扩展后的 7 轮概率积分区分器与 6 轮概率积分区分器相比,各个位置的平衡概率甚至要更高.该结果与图 5 基本一致.由于求解概率积分区分器过程中,通过设置阈值的方式来简化算法,因此为获得更准确的攻击结果,在后续攻击过程中,选用图 6 所示算法加密得到的平衡概率(单位:%).

					93.4					88.2	88.4		92.0	87.5	
				88.9	89.4			89.8		87.5			91.3		
92.6			99.6		99.6			93.1	99.8						
			92.7	85.5					93.6		88.0		99.6		

Fig.6 Experiment to verify the ciphertext balance probability after 7 rounds

图 6 实验验证 7 轮加密后密文平衡概率

2.3 攻击步骤

利用 7 轮概率积分区分器,可以对 9 轮 PUFFIN 算法进行概率积分攻击,从而获取部分轮密钥信息.攻击过程如图 7 所示,其中,首个状态图中,灰色底纹表示算法 7 轮加密后高概率平衡位置;箭头所指表示以 20~22 这 3 个高概率平衡位置为例描述攻击步骤;后续状态图中,斜纹表示高概率平衡位置经过非线性变换,灰色底纹表示高概率平衡位置经过线性变换.

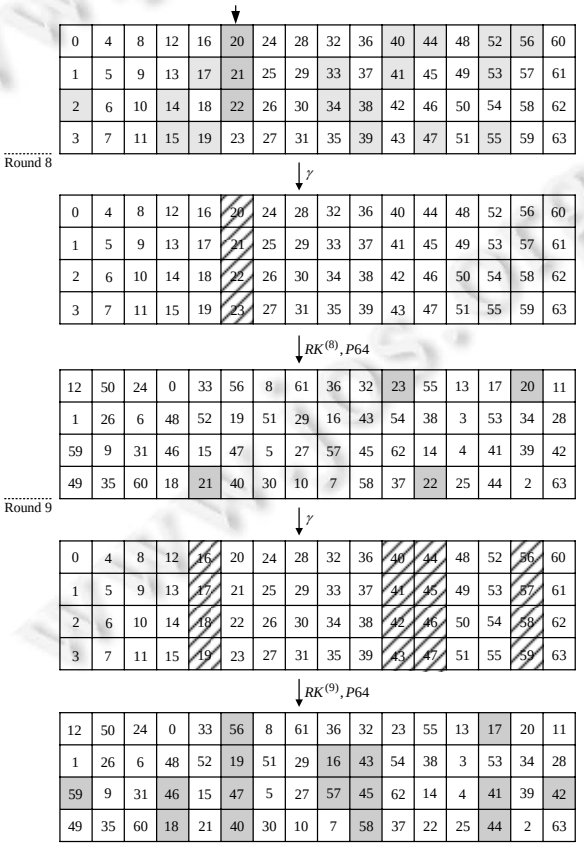

Fig.7 Probability integral cryptanalysis on 9-round PUFFIN

图 7 9 轮 PUFFIN 算法的概率积分攻击

© 中国科学院软件研究所 http://www.jos.org.cn

攻击的主要思想是:通过猜测第 9 轮轮密钥 $RK^{(9)}$及第 8 轮轮密钥 $RK^{(8)}$的部分比特,对密文进行部分解密后,观察第 7 轮输出的对应位置是否概率平衡来得到正确密钥.

以 20~22 这 3 个高概率平衡位置为例,进行 N 次实验(其中,$N=\lceil C\cdot p^{-m}\rceil$),每次实验的攻击步骤如下.

- 第 1 步,选取 m 组明文,对每组明文进行如下操作:
  (1) 对每组明文(其中,$\{p_5,p_8,p_9,p_{10},p_{11},p_{26},p_{27},p_{28},p_{29},p_{30},p_{35},p_{42},p_{50},p_{51},p_{61},p_{63}\}$遍历$\{0,1\}^{16}$,其余位置随机选定为常数,故一组明文包含 2^{16} 个明文)进行 9 轮加密,密文记为 $C_0,C_1,...,C_{2^{16}-1}$;
  (2) 猜测 $RK^{(9)}$的 4 个密钥字(共 16 比特) $RK_4^{(9)},RK_{10}^{(9)},RK_{11}^{(9)},RK_{14}^{(9)}$,计算 $Q_j^{(i)}=\gamma^{-1}(P64^{-1}(C_i)\oplus RK_j^{(9)})$,$j\in\{4,10,11,14\}$;
  (3) 计算 $T^i=P64^{-1}(Q^{(i)})$,猜测 $RK^{(8)}$的一个密钥字 $RK_5^{(8)}$,计算 $R^i=S^{-1}(T_5^i\oplus RK_5^8)$;
  (4) 判断 $t_j=\oplus_{i=0}^{2^{16}-1}R_j^i$ 是否为 0,$j\in\{20,21,22\}$,若 t_{20},t_{21},t_{22} 均为 0,则给该组密钥($RK_4^{(9)},RK_{10}^{(9)},RK_{11}^{(9)},RK_{14}^{(9)},RK_5^{(8)}$,共 20 比特)计数器+1;
- 第 2 步,对 m 组明文重复上述过程,在 2^{20} 个计数器中,若某组密钥计数值与 m 相同,则将其视为该次实验的候选密钥.

统计 N 次实验候选密钥总数,其中正确密钥出现次数约为 C,与错误密钥相比具有一定的显著性,故正确密钥可唯一恢复.

2.4 密钥恢复实验结果

在 PC 机上,利用 Visual Studio 2010 编程模拟了密钥筛选过程.以 20~22 这 3 个高概率平衡位置为例,分析组数 m、显著度 C 与复杂度的关系.每种情形分别进行了 100 次实验.

表 3 表明:随着组数的增加,错误密钥量越来越少,但数据复杂度越来越大.正确密钥量与显著度 C 基本一致.

Table 3 Relationship among group number m, key amount and complexity when C=2

表 3 固定 C=2,分析组数 m、密钥量以及复杂度之间的关系

	正确密钥量/个	错误密钥量/个	数据复杂度
m=7	2.57	15.1	$8\times7\times2^{16}\approx2^{21.81}$
m=8	2.23	3.79	$9\times8\times2^{16}\approx2^{22.17}$
m=9	2.26	1.46	$11\times9\times2^{16}\approx2^{22.63}$
m=10	2.22	1.11	$13\times10\times2^{16}\approx2^{23.02}$

表 4 表明:随着显著度 C 的增加,正确密钥与错误密钥相比越来越显著,但错误密钥整体个数也会相应有所增长,数据复杂度也随之增大.

Table 4 Relationship among saliency C, key amount, and complexity when m=7

表 4 固定 m=8,分析显著度 C、密钥量以及复杂度之间的关系

	正确密钥量/个	错误密钥量/个	数据复杂度
C=2	2.23	3.79	$9\times8\times2^{16}\approx2^{22.17}$
C=3	3.54	5.62	$14\times8\times2^{16}\approx2^{22.81}$
C=4	5.78	9.92	$23\times8\times2^{16}\approx2^{23.52}$

以上两个表格中的实验结果,验证了通过控制组数 m 与显著度 C,概率积分分析方法可唯一恢复正确密钥.

2.5 复杂度分析

实际攻击时可根据表 5 从上到下(利用的平衡位置由多到少排序)对密钥字进行猜测,即攻击共需进行 10 轮实验.表 5 中:$RK^{(9)}$对应列粗体标注表示此密钥字经过前几轮猜测已经唯一确定;第 6 列给出了当使用 m 组明文进行攻击时,各次实验中除正确密钥外候选密钥量.经过 10 轮实验,可将表 5 中涉及的 $RK^{(8)}$中 10 个密钥字和 $RK^{(9)}$中 13 个密钥字共 92 比特密钥信息唯一确定.由于经过加密的明密对在 10 轮实验中可重复使用,故攻击的数据复杂度选取最大组数,为 $11\times41\times2^{16}\approx2^{42.17}$.以表 5 中第 1 轮实验为例,其所需的时间复杂度为 $9\times8\times2^{16}\times2^{20}\approx$

© 中国科学院软件研究所 http://www.jos.org.cn

$2^{42.17}$ 次查表,10 轮实验的时间复杂度之和约为 $2^{42.65}$ 次查表,这相当于 $2^{42.65}/(9\times16)\approx2^{35.48}$ 次 9 轮加密.另外,为存储密钥,攻击需要对猜测的密钥字进行存储,存储复杂度为$(2^{20}+2^{16}+2\times2^{12}+2\times2^{8}+4\times2^{4})\approx2^{20}$ 个存储单元.

Table 5 Correspondence between probability balance positions and guess key words (C=2)

表 5 概率平衡位置与猜测密钥字的对应关系(令 C=2)

序号	概率平衡位置	$RK^{(8)}$	$RK^{(9)}$	全平衡概率	组数 m	除正确密钥外候选密钥量
1	20,21,22	5	4,10,11,14	83	8	$9\times2^{20}\times(2^{-3})^{8}<1$
2	52,53,55	13	**4**,**10**,**11**,13	83	4	$5\times2^{8}\times(2^{-3})^{4}<1$
3	14,15	3	0,**4**,**11**,12	92	8	$4\times2^{12}\times(2^{-2})^{8}<1$
4	17,19	4	3,5,8,**13**	76	11	$41\times2^{16}\times(2^{-2})^{11}<1$
5	33,34	8	1,**4**,9,**14**	84	8	$9\times2^{12}\times(2^{-2})^{8}<1$
6	38,39	9	**8**,**10**,**11**,**14**	93	3	$3\times2^{4}\times(2^{-2})^{3}<1$
7	40,41	10	**5**,**9**,**13**,15	77	6	$10\times2^{8}\times(2^{-2})^{6}<1$
8	44,47	11	**3**,**5**,**9**,**13**	78	4	$6\times2^{4}\times(2^{-2})^{4}<1$
9	2	0	**0**,**3**,**12**,**14**	93	6	$3\times2^{4}\times(2^{-1})^{6}<1$
10	56	14	**0**,**5**,**8**,**9**	87	7	$6\times2^{4}\times(2^{-1})^{7}<1$

3 总结与展望

本文从传统的积分分析出发,考虑常数对多项式首项系数的影响,将“概率”这一思想运用到积分分析中,首次提出了概率积分区分器的构造方法以及概率积分攻击模型.针对 PUFFIN 算法,构造了 7 轮概率积分区分器,并由此对 9 轮 PUFFIN 算法进行攻击.该攻击可恢复 92 比特轮密钥,数据复杂度为 $2^{24.8}$ 个选择明文,时间复杂度为 $2^{35.48}$ 次 9 轮加密,存储复杂度为 2^{20} 个存储单元.这是目前针对 PUFFIN 算法最好的实际积分分析结果.概率积分分析方法的提出,完善了积分区分器的理论和实际应用,对于积分分析的进一步研究和改进,有学术价值和重要意义.这一方法也可以应用于其他轻量级分组密码的分析中.

References:

[1] Sun B, Zhang P, Li C. Impossible differential and integral cryptanalysis of Zodiac. Ruan Jian Xue Bao/Journal of Software, 2011,22(8):1911−1917 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3875.htm [doi: 10.3724/SP.J.1001.2011.03875]

[2] Daemen J, Knudsen L, Rijmen V. The block cipher square In: Proc. of the Int'l Workshop on Fast Software Encryption. 1997. 149−165. [doi: 10.1007/BFb0052343]

[3] Biryukov A, Shamir A. Structural cryptanalysis of SASAS. Journal of Cryptology, 2010,23(4):505−518. [doi: 10.1007/s00145-010-9062-1]

[4] Lucks S. The saturation attack—A bait for Twofish. In: Proc. of the Revised Papers from the Int'l Workshop on Fast Software Encryption. Springer-Verlag, 2001. 1−15. [doi: 10.1007/3-540-45473-X_1]

[5] Z'Aba MR, Raddum H, Henricksen M, *et al*. Bit-pattern based integral attack. In: Proc. of the Fast Software Encryption. 2008. 363−381. [doi: 10.1007/978-3-540-71039-4_23]

[6] Daemen J, Peeters M, Van Assche G, Rijmen V. Nessie proposal: NOEKEON. In: Proc. of the 1st Open NESSIE Workshop. 2000. http://gro.noekeon.org/

[7] Anderson R, Biham E, Knudsen L. Serpent: A proposal for the advanced encryption standard. In: Proc. of the NIST AES Proposal. 1998. http://www.cl.cam.ac.uk/rja14/serpent.html [doi: 10.1007/3-540-69710-1_15]

[8] Bogdanov A, Knudsen LR, Leander G, *et al*. PRESENT: An ultra-lightweight block cipher. In: Proc. of the Int'l Workshop on Cryptographic Hardware and Embedded Systems. Berlin, Heidelberg: Springer-Verlag, 2007. 450−466. [doi: 10.1007/978-3-540-74735-2_31]

[9] Todo Y. Structural evaluation by generalized integral property. In: Proc. of the Advances in Cryptology (EUROCRYPT 2015). Berlin Heidelberg: Springer-Verlag, 2015. 287−314. [doi: 10.1007/978-3-662-46800-5_12]

[10] Todo Y, Morii M. Bit-based division property and application to Simon family. In: Proc. of the Fast Software Encryption. Berlin Heidelberg: Springer-Verlag, 2016. 357−377. [doi: 10.1007/978-3-662-52993-5_18]

© 中国科学院软件研究所 http://www.jos.org.cn

[11] Xiang Z, Zhang W, Bao Z, *et al*. Applying MILP method to searching integral distinguishers based on division property for 6 lightweight block ciphers. In: Proc. of the Int'l Conf. on the Theory & Application of Cryptology & Information Security. Berlin, Heidelberg: Springer-Verlag, 2016. 648−678. [doi: 10.1007/978-3-662-53887-6_24]

[12] Wu S, Wang M. Integral attacks on reduced-round PRESENT. In: Proc. of the Information and Communications Security. Springer Int'l Publishing, 2013. 331−345. [doi: 10.1007/978-3-319-02726-5_24]

[13] Knudsen L, Wagner D. Integral cryptanalysis. In: Proc. of the Revised Papers from the Int'l Workshop on Fast Software Encryption. 2002. 112−127. [doi: 10.1007/3-540-45661-9_9]

[14] Li C, Qu LJ, Zhou Y. Analysis of the security index of cryptographic functions. Beijing: Science Press, 2011 (in Chinese).

[15] Cheng H, Heys HM, Wang C. Puffin: A novel compact block cipher targeted to embedded digital systems. In: Proc. of the 11th EUROMICRO Conf. on Digital System Design Architectures, Methods and Tools (DSD 2008). 2008. 383−390.

[16] Blondeau C, Gérard B. Differential cryptanalysis of PUFFIN and PUFFIN2. In: Proc. of the ECRYPT Workshop on Lightweight Cryptography (LC 2011). 2011.

[17] Leander G. On linear hulls, statistical saturation attacks, PRESENT and a cryptanalysis of PUFFIN. In: Proc. of the Annual Int'l Conf. on the Theory and Applications of Cryptographic Techniques. Berlin, Heidelberg: Springer-Verlag, 2011. 303−322. [doi: 10.1007/ 978-3-642-20465-4_18]

[18] Wei YC, Sun B, Li C. An integral attack on PUFFIN and PUFFIN-like SPN cipher. Journal of National University of Defense Technology, 2010,32(3):139−143 (in Chinese with English abstract). [doi: 10.3969/j.issn.1001-2486.2010.03.026]

[19] Zhao GY, Cheng L, Li RL, Li C, Sun B. Integral cryptanalysis on reduced-round PUFFIN. Journal of National University of Defense Technology, 2015,37(6):129−134 (in Chinese with English abstract). [doi: 10.11887/j.cn.201506024]

[20] Zhao GY. Security analysis of lightweight block cipher algorithm [Ph.D. Thesis]. Changsha: National University of Defense Technology, 2015 (in Chinese with English abstract).

[21] Shang FZ, Shen X, Liu GQ, Li C. Integral cryptanalysis on PUFFIN based on MILP. Journal of Cryptologic Research, 2019,6(5): 627−638 (in Chinese with English abstract). [doi: cnki:sun:mmxb.0.2019-05-008]

附中文参考文献:

[1] 孙兵,张鹏,李超.Zodiac 算法的不可能差分和积分攻击.软件学报,2011,22(8):1911−1917. http://www.jos.org.cn/1000-9825/3875.htm [doi: 10.3724/SP.J.1001.2011.03875]

[14] 李超,屈龙江,周悦.密码函数的安全性指标分析.北京:科学出版社,2011.

[18] 魏悦川,孙兵,李超.一种 PUFFIN 类 SPN 型分组密码的积分攻击.国防科技大学学报,2010,32(3):139−143. [doi: 10.3969/j.issn.1001-2486.2010.03.026]

[19] 赵光耀,成磊,李瑞林,李超,孙兵.低轮 PUFFIN 算法的积分攻击.国防科技大学学报,2015,37(6):129−134. [doi: 10.11887/j.cn.201506024]

[20] 赵光耀.轻量级分组密码算法的安全性分析[博士学位论文].长沙:国防科技大学,2015.

[21] 尚方舟,沈璇,刘国强,李超.基于 MILP 搜索的 PUFFIN 算法积分分析.密码学报,2019,6(5):627−638.[doi: cnki:sun:mmxb.0.2019-05-008]

尚方舟(1995－),女,硕士生,主要研究领域为编码密码理论及其应用.

刘国强(1986－),男,博士,讲师,主要研究领域为编码密码理论及其应用.

孙兵(1981－),男,博士,副教授,主要研究领域为编码密码理论及其应用.

李超(1966－),男,博士,教授,主要研究领域为编码密码理论及其应用.

© 中国科学院软件研究所 http://www.jos.org.cn