软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(8):2349−2361 [doi: 10.13328/j.cnki.jos.005762]
©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

Piccolo 算法的相关密钥-不可能差分攻击*

徐林宏, 郭建胜, 崔竞一, 李明明

(信息工程大学,河南 郑州 450001)

通讯作者: 郭建胜, E-mail: tsg_31@126.com

摘 要: 现有的对于Piccolo算法的安全性分析结果中,除Biclique分析外,以低于穷举搜索的复杂度最长仅攻击至14轮Piccolo-80和18轮Piccolo-128算法.通过分析Piccolo算法密钥扩展的信息泄漏规律,结合算法等效结构,利用相关密钥-不可能差分分析方法,基于分割攻击思想,分别给出了15轮Piccolo-80和21轮Piccolo-128含前向白化密钥的攻击结果.当选择相关密钥量为2^8时,攻击所需的数据复杂度分别为$2^{58.6}$和$2^{62.3}$,存储复杂度分别为$2^{60.6}$和$2^{64.3}$,计算复杂度分别为2^{78}和$2^{82.5}$;在选择相关密钥量为2^4时,攻击所需的数据复杂度均为$2^{62.6}$和$2^{62.3}$,存储复杂度分别为$2^{64.6}$和$2^{64.3}$,计算复杂度分别为$2^{77.93}$和$2^{124.45}$.分析结果表明,仅含前向白化密钥的15轮Piccolo- 80算法和21轮Piccolo-128算法在相关密钥-不可能差分攻击下是不安全的.

关键词: 轻量级分组密码;Piccolo;相关密钥-不可能差分;密码分析

中图法分类号: TP309

中文引用格式: 徐林宏,郭建胜,崔竞一,李明明.Piccolo 算法的相关密钥-不可能差分攻击.软件学报,2019,30(8):2349−2361. http://www.jos.org.cn/1000-9825/5762.htm

英文引用格式: Xu LH, Guo JS, Cui JY, Li MM. Related-key impossible differential attack on Piccolo. Ruan Jian Xue Bao/ Journal of Software, 2019,30(8):2349−2361 (in Chinese). http://www.jos.org.cn/1000-9825/5762.htm

Related-key Impossible Differential Attack on Piccolo

XU Lin-Hong, GUO Jian-Sheng, CUI Jing-Yi, LI Ming-Ming

(Information Engineering University, Zhengzhou 450001, China)

Abstract: The existing security analysis results for Piccolo attack only up to 14-round Piccolo-80 and 18-round Piccolo-128 with lower complexity than exhaustive analysis, except for biclique analysis. By analyzing the information disclosure law of the key-schedule algorithm of Piccolo, the related-key impossible differential cryptanalysis method is used to give the attack results on 15-round Piccolo-80 and 21-round Piccolo-128 with pre-whitening keys respectively. When 2^8 related-keys are used, the data complexity of the attack is $2^{58.6}$ and $2^{62.3}$, the memory complexity is $2^{60.6}$ and $2^{64.3}$, and the computational complexity is 2^{78} and $2^{82.5}$ respectively. When 2^4 related-keys are used, the data complexity, memory complexity, and computational complexity of attack are $2^{62.6}$, $2^{62.3}$; $2^{64.6}$, $2^{64.3}$; $2^{77.93}$, $2^{124.45}$ respectively. The analysis shows that the 15-round Piccolo-80 and 21-round Piccolo-128 with pre-whitening keys are insecure under the related-key impossible differential attack.

Key words: lightweight block cipher; Piccolo; related-key impossible differential; cryptanalysis

随着物联网技术的不断发展,人们对其安全性的要求也越来越高,适用于这些资源受限环境的密码算法即轻量级密码算法成为密码学研究的一个热点,大量轻量级密码算法被设计出来,如 PRESENT[1],LBlock[2],

* 基金项目: 信息保障技术重点实验室开放基金(KJ-17-003)
Foundation item: Open Foundation of Science and Technology on Information Assurance Laboratory (KJ-17-003)
本文由“面向自主安全可控的可信计算”专题特约编辑林璟锵教授推荐.
收稿时间: 2018-05-22; 修改时间: 2018-09-21; 采用时间: 2018-12-13; jos 在线出版时间: 2019-03-28
CNKI 网络优先出版: 2019-03-29 09:16:32, http://kns.cnki.net/kcms/detail/11.2560.TP.20190329.0915.013.html

© 中国科学院软件研究所 http://www.jos.org.cn

PRINCE[3],SIMON[4]等算法.

Piccolo 算法是由日本密码学家 Shibutani 等人[5]于 CHES 2011 上提出的一种适用于资源受限环境的轻量级分组密码算法,其算法结构为广义 Feistel 结构,硬件实现效率较高.该算法一经提出,就受到学界的广泛关注.

不可能差分分析是由 Knudsen[6]和 Biham[7]两人独自提出的单密钥条件下的密码分析方法,Knudsen 在分析高级加密标准的候选算法 DEAL 的安全性时,提出了轮函数为双射的 Feistel 结构存在 5 轮的不可能差分路径;Biham 等人在 FSE'99 上介绍了基于中间相错的思想构造不可能差分的方法.此后,不可能差分在研究很多标准密码算法的安全性时得到了广泛应用,如 AES[8],CLEFIA[9]等.

相关密钥攻击同样是由 Biham[10]和 Knudsen[11]各自独立提出的,其主要思想是:基于密码算法密钥扩展方式对算法安全性的影响,利用密钥扩展算法的一些弱点,通过分析相关密钥与原有密钥之间的关系对加密结果造成的影响恢复出原有密钥.该攻击方法对于具有简单密钥扩展方式的密码算法尤其有效.当前,主流应用是在此攻击方法假设条件下,与其他密码分析方法相结合,如差分分析、不可能差分分析、矩阵攻击等,对算法进行安全性分析.因此,本文结合上述两种攻击方法,对 Piccolo 算法进行了相关密钥-不可能差分分析.

- 相关工作

现有的对于 Piccolo 算法的安全性分析结果较多,这里主要介绍差分类分析结果.其中,利用 Biclique 分析方法可得到对于全轮 Piccolo-80 和 Piccolo-128 算法低于穷举复杂度的攻击结果[12−14].在单密钥条件下,Azimi 等人[15]于 2014 年利用不可能差分分析方法,对含前向白化密钥 12 轮 Piccolo-80 算法、不含白化密钥的 13 轮 Piccolo-80 算法和含后向白化密钥的 15 轮 Piccolo-128 算法进行了安全性分析;2015 年,Tolba 等人[16]给出了 14 轮 Piccolo-80 不含白化密钥和 17 轮 Piccolo-128 含后向白化密钥的中间相遇攻击结果;2017 年,Liu 等人[17]在文献[16]基础上进行改进,利用中间相遇攻击方法,给出了不含白化密钥的 14 轮 Piccolo-80 和含后向白化密钥的的 18 轮 Piccolo-128 的安全性分析结果.在相关密钥条件下,Minier[18]在 2013 年结合相关密钥攻击和不可能差分分析,实现了对均不含白化密钥的 14 轮 Piccolo-80 和 21 轮 Piccolo-128 的攻击,但攻击所需的数据量超过算法的分组规模 2^{64}.2016 年,Zheng 等人[19]利用 U-method 分别给出了 Piccolo-80 和 Piccolo-128 算法 11 轮和 17 轮的相关密钥-不可能差分区分器,但未能给出具体的安全性分析结果.

- 本文工作

本文主要使用相关密钥-不可能差分分析方法评估了 Piccolo 算法的安全性:首先,基于中间相错思想和密钥扩展的性质,寻找到所有可能的最长相关密钥-不可能差分区分器;而后,利用轮函数等效结构减少攻击所需选择明(密)文量,给出了 15 轮 Piccolo-80 和 21 轮 Piccolo-128 算法的攻击结果,所需数据复杂度和计算复杂度均低于穷举分析.在不考虑 Biclique 分析结果的情况下,均优于现有分析结果.本文攻击结果见表 1.

Table 1 Comparison of attack results on Piccolo

表 1 Piccolo 算法的攻击结果对比

算法	攻击轮数	密钥条件	攻击方法	所含白化密钥	选择密钥量	数据复杂度	计算复杂度	存储复杂度	来源文献
Piccolo-80	12	Single-key	ID	Pre-whitening	–	$2^{36.34}$CC	$2^{55.18}$	2^{63}	文献[15]
	13	Single-key	ID	None	–	$2^{43.25}$CP	$2^{69.7}$	2^{62}	文献[15]
	14	Single-key	MM	None	–	2^{48}CP	$2^{75.39}$	$2^{73.49}$	文献[16]
	14	Single-key	MM	None	–	2^{52}CP	$2^{67.44}$	$2^{64.91}$	文献[17]
	14	Related-key	ID	None	2	2^{64}CP†	$2^{68.19}$	–	文献[18]
	15	Related-key	ID	Pre-whitening	2^{8}	$2^{58.6}$CC	2^{78}	$2^{60.6}$	Section 2
	15	Related-key	ID	Pre-whitening	2^{4}	$2^{62.6}$CC	$2^{77.93}$	$2^{64.6}$	Section 2
Piccolo-128	15	Single-key	ID	Post-whitening	–	$2^{58.7}$CP	$2^{125.4}$	2^{61}	文献[15]
	17	Single-key	MM	Post-whitening	–	2^{48}CP	$2^{126.87}$	$2^{125.99}$	文献[16]
	18	Single-key	MM	Post-whitening	–	2^{52}CP	$2^{126.63}$	$2^{125.29}$	文献[17]
	21	Related-key	ID	None	2	2^{64}CP†	$2^{117.77}$	–	文献[18]
	21	Related-key	ID	Pre-whitening	2^{8}	$2^{62.3}$CP	$2^{82.5}$	$2^{64.3}$	Section 3
	21	Related-key	ID	Pre-whitening	2^{4}	$2^{62.3}$CP	$2^{124.45}$	$2^{64.3}$	Section 3

注:ID 和 MM 分别表示不可能差分分析和中间相遇攻击;
CC,CP 和†分别表示选择密文、选择明文和数据复杂度超过分组规模

© 中国科学院软件研究所 http://www.jos.org.cn

- 文章结构安排

本文首先介绍研究背景.第 1 节主要介绍 Piccolo 算法的相关知识.第 2 节给出 Piccolo-80 的相关密钥-不可能差分分析结果.Piccolo-128 算法的相关密钥-不可能差分分析主要在第 3 节中给出.最后对全文进行总结.

1 相关知识

本节中,首先给出下文中一些常用符号说明,而后给出 Piccolo 算法具体描述.

1.1 符号说明

- (P,P'):输入明文状态.
- (C,C'):输出密文状态.
- X_i^r:第 r 轮算法的第 i 个字节,$i\in\{0,1,\ldots,7\}$.
- Y_i^r:第 r 轮算法在字节置换变换前的第 i 个字节,$i\in\{0,1,\ldots,7\}$.
- K_i^L:第 i 个主密钥的左半部分.
- K_i^R:第 i 个主密钥的右半部分.
- WK_t:第 t 个白化密钥,$t\in\{0,1,2,3\}$.
- (rk_{2r},rk_{2r+1}):第 r 轮轮密钥.

1.2 Piccolo算法简介

Piccolo 算法的分组规模为 64bits,根据密钥规模不同,可分为 Piccolo-80 和 Piccolo-128 两种,对应轮数分别为 25(0~24)轮和 31(0~30)轮.其加密环节主要由密钥白化、F 函数和字节置换操作组成.密钥白化操作仅在第 1 轮和最后一轮存在,保证加脱密结构的一致性.轮变换包括 F 函数和字节置换,具体的 Piccolo 算法轮函数各环节的结构示意图详见文献[5]的图 1~图 4.轮函数各个环节和密钥扩展算法的介绍如下给出.

- F 函数

每一个 F 函数包含了 8 个相同的 4bits 双射 S 盒和一个列混合矩阵 M,通过对 16bits 数据进行作用,实现算法的混乱和扩散.即有$(x_{0(4)},x_{1(4)},x_{2(4)},x_{3(4)})^t=M\cdot(x_{0(4)},x_{1(4)},x_{2(4)},x_{3(4)})^t$,其中,列混合矩阵 M 的运算定义在由不可约多项式 x^4+x+1 生成的 $GF(2^4)$上.

- 字节置换 RP

该环节主要实现各字节的移位替换,即

$$(x_{0(8)},x_{1(8)},x_{2(8)},x_{3(8)},x_{4(8)},x_{5(8)},x_{6(8)},x_{7(8)})\xrightarrow{RP}(x_{2(8)},x_{7(8)},x_{4(8)},x_{1(8)},x_{6(8)},x_{3(8)},x_{0(8)},x_{5(8)}).$$

- 密钥扩展算法

Piccolo 算法的密钥扩展有 80bits 和 128bits 密钥两种.其中,对于 Piccolo-80,将 80bits 主密钥划分为 5 个部分,每个部分包括 2 字节,即 $K=K_0\|K_1\|K_2\|K_3\|K_4$,其中, $K_i=(K_i^L\|K_i^R),i\in\{0,1,2,3,4\}$.根据 F 函数中采用的是 4bits 的 S 盒,可对每一字节密钥进行再划分,即 $K_i=(K_i^{L_0}\|K_i^{L_1}\|K_i^{R_1}\|K_i^{R_0})$.

白化密钥 $WK_j(j\in\{0,1,2,3\})$由主密钥直接生成,其中,

$$\text{Piccolo-80:}\quad WK_0=(K_0^L\|K_1^R),WK_1=(K_1^L\|K_0^R),WK_2=(K_4^L\|K_3^R),WK_3=(K_3^L\|K_4^R).$$

轮密钥(rk_{2r},rk_{2r+1})由主密钥与固定常数异或所得,$r\in\{0,\ldots,24\}$.在 Piccolo-128 中,增加了 6 个字节的主密钥,有 $K=K_0\|K_1\|K_2\|K_3\|K_4\|K_5\|K_6\|K_7$,相应的白化密钥也有所变化,即

$$\text{Piccolo-128:}\quad WK_0=(K_0^L\|K_1^R),WK_1=(K_1^L\|K_0^R),WK_2=(K_4^L\|K_7^R),WK_3=(K_7^L\|K_4^R).$$

表 2 和表 3 分别给出了两个版本 Piccolo 算法轮子密钥和主密钥的对应关系.

© 中国科学院软件研究所 http://www.jos.org.cn

Table 2 Correspondence between the master key and the round keys of Piccolo-80

表 2 Piccolo-80 算法的主密钥与轮密钥之间的对应关系

轮数 r	轮密钥	轮数 r	轮密钥
初始白化	$K_0^L \parallel K_1^R, K_1^L \parallel K_0^R$	13	K_4,K_4
0	K_2,K_3	14	K_0,K_1
1	K_0,K_1	15	K_2,K_3
2	K_2,K_3	16	K_0,K_1
3	K_4,K_4	17	K_2,K_3
4	K_0,K_1	18	K_4,K_4
5	K_2,K_3	19	K_0,K_1
6	K_0,K_1	20	K_2,K_3
7	K_2,K_3	21	K_0,K_1
8	K_4,K_4	22	K_2,K_3
9	K_0,K_1	23	K_4,K_4
10	K_2,K_3	24	K_0,K_1
11	K_0,K_1	最终白化	$K_4^L \parallel K_3^R, K_3^L \parallel K_4^R$
12	K_2,K_3	–	–

Table 3 Correspondence between the master key and the round keys of Piccolo-128

表 3 Piccolo-128 算法的主密钥与轮密钥之间的对应关系

轮数 r	轮密钥	轮数 r	轮密钥
初始白化	$K_0^L \parallel K_1^R, K_1^L \parallel K_0^R$	16	K_2,K_7
0	K_2,K_3	17	K_4,K_3
1	K_4,K_5	18	K_6,K_5
2	K_6,K_7	19	K_2,K_1
3	K_2,K_1	20	K_6,K_5
4	K_6,K_7	21	K_0,K_7
5	K_0,K_3	22	K_4,K_3
6	K_4,K_5	23	K_6,K_1
7	K_6,K_1	24	K_4,K_3
8	K_4,K_5	25	K_2,K_5
9	K_2,K_7	26	K_0,K_7
10	K_0,K_3	27	K_4,K_1
11	K_4,K_1	28	K_0,K_7
12	K_0,K_3	29	K_6,K_3
13	K_6,K_5	30	K_2,K_5
14	K_2,K_7	最终白化	$K_4^L \parallel K_7^R, K_7^L \parallel K_4^R$
15	K_0,K_1	–	–

2 15 轮 Piccolo-80 算法的相关密钥-不可能差分攻击

本节中首先介绍了 Piccolo 算法具有的 3 点性质,而后基于中间相错思想,分析了 Piccolo-80 算法在状态值和主密钥值均仅有单比特块差分活动的情况下,其能够具有的所有最长 11 轮相关密钥-不可能差分区分器.通过选择合适的区分器,向上向下各拓展两轮,得到了 15 轮 Piccolo-80 含前向白化密钥的安全性分析结果.

2.1 Piccolo算法性质

性质 1(密钥扩展算法性质). 主要有以下 3 个方面.

(1) 对于 Piccolo 两种密钥规模的算法,除白化密钥外,其每一主密钥字 K_i 在各轮密钥中分布的位置均与其第 1 次出现的位置相同.具体来说,若主密钥 K_i 第 1 次出现在轮密钥的左(右)半部分,则后续的均出现在轮密钥的左(右)半部分.

(2) 在不考虑轮常数的影响时,对于 Piccolo-80,其轮密钥(除白化密钥外)是由主密钥构成的一个周期为 5 的循环.

(3) 对于 Piccolo-128 算法,在不考虑轮常数的影响下,自第 0 轮开始,其轮密钥左半部分是按($K_2,K_4,K_6,K_2,K_6,K_0,K_4,K_6,K_4,K_2,K_0,K_4,K_0,K_6,K_2,K_0$)排列的一个 16 轮循环,右半部分是按($K_3,K_5,K_7,K_1,K_7,K_3,K_5,K_1,K_5,$

© 中国科学院软件研究所 http://www.jos.org.cn

K_7,K_3,K_1)排列的一个 12 轮循环.

证明:分别观察表 2、表 3 中 Piccolo-80 和 Piccolo-128 主密钥与轮密钥间的对应关系,可得上述结论. □

性质 2. 根据算法结构中的 F 函数采用的是 4bits 的 S 盒,且在字节置换操作中,实现的是对字节整体的移位操作,未改变各个字节内部每一比特的状态,因此在引入明文差分时,可在与密钥进行差分抵消的位置选取 4bits 或 8bits 差分,相应地选择 4bits 或 8bits 的相关密钥差分实现差分抵消.当选取明文差分为 4bits 时,攻击所需的相关密钥量约为 2^4,较差分选为 8bits 时有所减少,具体应用见第 2.3 节和第 3.2 节.

性质 3(轮函数等效结构)[12]. 改变 Piccolo 算法中轮密钥加的顺序,不影响算法加解密效果,且根据轮密钥加变换顺序的不同有两种算法轮函数的等效结构,具体结构如图 1 所示.

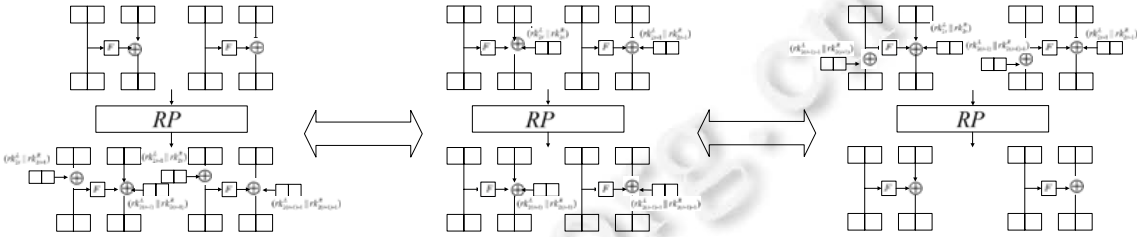

Fig.1 Equivalent structure of round function

图 1 轮函数等效结构

在下文给出的攻击过程中,利用性质 3,通过在数据收集阶段采用算法等效结构,可以有助于减少攻击过程中的选择明(密)文量,降低攻击所需数据复杂度.这里仅以 Piccolo-80 算法为例进行具体说明,对于 Piccolo-128 算法具有同样的效果.

例 1:对于 Piccolo-80 算法,由图 2 的攻击路径可得:在数据收集阶段使用算法等效结构,使得第 14 轮不含密钥加运算的作用,进而可由预计算直接得到满足第 14 轮的输入状态差分为(α,0,0,0,0,β,γ,0)的密文对.也就是说,对于 2^n 个明文结构,所需的选择密文量为 2^{n+24}.而若未使用算法等效结构,则根据密文差分中有 6 个活动差分字节,攻击所需的选择密文量 2^{n+48}.进一步的,在攻击计算复杂度相同的条件下,显然,未使用等效结构所需的数据复杂度更高.

2.2 11轮相关密钥-不可能差分区分器

根据算法轮函数结构,在不含相关密钥差分,输入状态仅存在 1 个差分活动比特块时,算法达到完全性有两种情况.

(1) 活动比特块位于 F 函数的输入,经过 3 轮变换后,算法达到完全.

(2) 活动比特块位于密钥加操作的输入,经过 4 轮变换后,算法达到完全.

引入相关密钥差分,一般用于实现与输入状态差分的相互抵消,得到更长的区分器,提升密钥恢复攻击时的轮数,所以说,密钥差分选取的位置也十分关键.一般情况下,针对 Piccolo 算法,所能构造区分器的最长轮数由同一主密钥字在轮密钥中连续出现 5 次的相距轮数决定.但并不是说,所能构造的区分器轮数就一定达到相距轮数,还需受到算法达到完全性的轮数限制.此外,直观地可以发现:若引入的密钥差分和输入状态差分活动比特块越多,算法达到完全所需的轮数越少.

综合上述考虑,利用中间相错思想,在主密钥差分和输入状态差分均仅有 1 个活动比特块时构造区分器.特别地,当轮密钥为(K_4,K_4)时,若在 K_4 中引入一个差分活动块,则为了实现差分抵消,需要在状态差分中有 2 个差分活动比特块,此种情况的区分器也在本节考虑范围内.由此,利用性质 1,尤其是 Piccolo-80 密钥扩展中轮密钥具有的 5 轮循环性,能够构造得到的 Piccolo-80 的最长 11 轮相关密钥-不可能差分区分器主要有 3 种情形,见表 4.其中,密钥差分选取在 K_0 中与选取在 K_1 中,所得的区分器情形基本一致,且根据差分选取的左右位置不同,共有 12 种情况;密钥差分选取在 K_2 中与选取在 K_3 中,所得的区分器情形一致,且与情形 1 类似,也有 12 种情况;密钥

© 中国科学院软件研究所 http://www.jos.org.cn

差分选取在 K_4 中时,根据差分选取的左右位置不同,共有 6 种情况.综上,能够找到 30 条 11 轮 Piccolo-80 的相关密钥-不可能差分区分器.附录 A、附录 B 给出了 3 种情形区分器的示例.

Table 4 11-round related-key impossible differential distinguishers of Piccolo-80

表 4 11 轮 Piccolo-80 算法的相关密钥-不可能差分区分器

	区分器位置	密钥差分	区分器具体表示	类似区分器总数
情形 1	第 1 轮~第 11 轮 第 6 轮~第 16 轮 第 11 轮~第 21 轮	$\Delta K_1^R=\gamma$	$(0,0,0,0,0,0,0,\gamma)\xrightarrow[(\Delta K_0,\Delta K_1)=[0,(\theta\|\gamma)]]{}(0,\gamma,0,0,0,0,0,0)$	12
情形 2	第 2 轮~第 12 轮 第 7 轮~第 17 轮 第 12 轮~第 22 轮	$\Delta K_2^L=\gamma$	$(0,0,\gamma,0,0,0,0,0)\xrightarrow[(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]]{}(\gamma,0,0,0,0,0,0,0)$	12
情形 3	第 3 轮~第 13 轮 第 8 轮~第 18 轮 第 13 轮~第 23 轮	$\Delta K_4^L=\gamma$	$(0,0,\gamma,0,0,0,\gamma,0)\xrightarrow[(\Delta K_4,\Delta K_4)=[(\gamma\|0),(\gamma\|0)]]{}(\gamma,0,0,0,\gamma,0,0,0)$	6

由于本文攻击中考虑前向白化密钥对算法安全性的影响,因此密钥差分选取在 K_0 和 K_1 中的区分器不适合用于攻击环节.而为了使攻击轮数尽可能长,且攻击复杂度尽可能低,考虑选用输入差分活动块较少的区分器,故在下文的攻击中,选用的区分器为情形 2 中的区分器,且区分器的位置取为第 2 轮~第 12 轮,具体路径见附录 A 中表 A 的左半部分.特别指出:文献[19]给出的 Piccolo-80 区分器也是情形 2 中的一种,且情形 2 中的位于第 12 轮~第 22 轮的区分器也可用来分析 15 轮 Piccolo-80 算法仅含后向白化密钥的安全性,具体攻击过程和复杂度估计与下文给出的攻击算法 1、算法 2 类似.

2.3 15轮密钥恢复攻击

根据上述构造所得区分器,向上解密 2 轮,向下加密 2 轮,基于分割攻击思想,可对 15 轮 Piccolo-80 算法进行密钥恢复,攻击总共分为两个阶段:一为数据收集阶段,二为密钥猜测阶段.本小节中根据选取密钥差分的比特块规模不同,给出了两种攻击算法,可以恢复出 Piccolo-80 算法的全部 80bits 主密钥,具体的攻击步骤基本类似,但在所需复杂度方面有一定差异,下面具体介绍这两种攻击算法.具体攻击路径如图 2 所示,其中,黑色表示差分活动比特块,灰色标识差分值为γ的比特块,差分不活动比特块用白色标识.

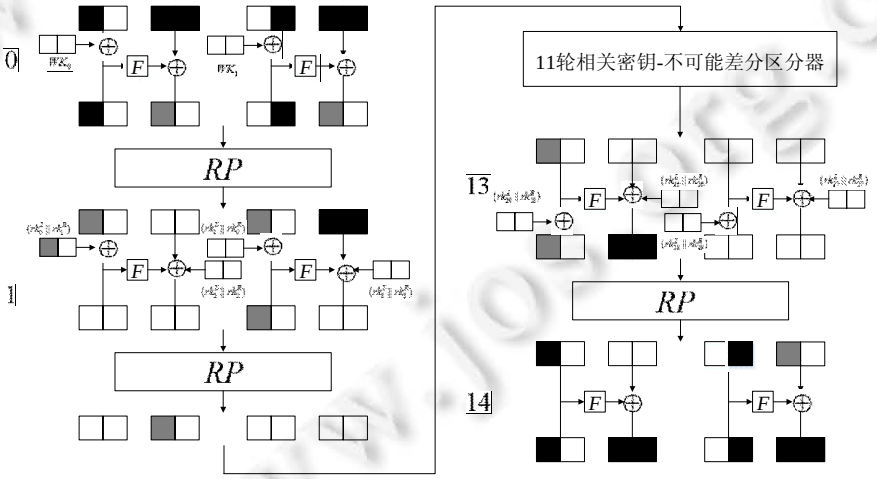

Fig.2 15-round related-key impossible differential cryptanalysis of Piccolo-80

图 2 15 轮 Piccolo-80 算法相关密钥-不可能差分分析

2.3.1 攻击算法 1

- Step 1. 数据收集.

© 中国科学院软件研究所 http://www.jos.org.cn

基于性质 3,对于密文(C,C'),选择其在第 14 轮的输入差分为$(\alpha,0,0,0,0,\beta,\gamma,0)$,其中,$\alpha,\beta,\gamma$表示差分活动比特块,且每一比特块表示一个字节.则显然:对于一个结构,需要选择 2^{24} 个选择密文,则得到 2^{47} 个密文对.当选定γ的一个非零取值时,因为有 2^8-1 种可能,所以对应的密文对数量 2^{39} 个.此时选择相应的密钥差分 $\Delta K_2^L=\gamma$,筛选明文差分为(*,0,*,*,0,*,*,*)的密文对,予以保留,则平均剩余的密文对个数为 $2^{39}\times2^{-16}=2^{23}$.构造 2^n 个结构,当γ选定一个非零值时,可得到 2^{n+23} 个平均剩余密文对.

- Step 2. 密钥猜测.

第 2 步中具体介绍密钥猜测阶段的攻击步骤,其中,Step 2.1~Step 2.4 是基于γ取定的情况,$\gamma\neq0$.

- Step 2.1. 对于选定的一个γ,根据已得到的密文对,猜测密钥(rk_{29}^L,rk_{28}^R),筛选使得 $\Delta X_{2,3}^{13}=0$ 的密文对,予以保留,则平均剩余密文对个数为 $2^{n+23}\times2^{-16}=2^{n+7}$.
- Step 2.2. 已知密钥(rk_{29}^L,rk_{28}^R),根据密钥扩展算法,可计算得到对应的主密钥(K_1^L,K_0^R),也就是 WK_1.再通过已知密文对可得相应的明文对,进行第 0 轮的右半部分加密.筛选 $\Delta Y_6^0=\gamma,\Delta Y_7^0=0$ 的密文对,予以保留,则平均剩余密文对的个数为 $2^{n+7}\times2^{-16}=2^{n-9}$.
- Step 2.3. 猜测密钥 WK_0,也就是$(K_0^L\|K_1^R)$,进行第 0 轮的左半部分加密,筛选使得 $\Delta Y_2^0=\gamma,\Delta Y_3^0=0$ 的密文对,予以保留,则平均剩余密文对的个数为 $2^{n-9}\times2^{-16}=2^{n-25}$.
- Step 2.4. 猜测密钥(rk_1^L,rk_0^R),根据上述步骤剩余密文对,计算并验证 $\Delta Y_{6,7}^1=0$ 是否成立:若成立,即此时仍有密文对通过验证,则说明此时的密钥为错误密钥,将密钥筛去,存入错误密钥候选密钥集;否则,将猜测密钥作为正确密钥候选密钥予以保留.
- Step 2.5. 遍历所有可能的γ值,重复进行上述步骤 2.1~步骤 2.4,将均不满足第 2.4 步验证条件的密钥可作为最终正确密钥的候选密钥.对于任一γ值,存在满足验证条件的密钥作为错误密钥筛除.上述攻击步骤已对 48bits 密钥$(rk_{29}^L,rk_{28}^R,K_0^L,K_1^R,rk_1^L,rk_0^R)$进行了猜测,将筛选后得到的候选密钥与未猜测 32bits 密钥进行穷举,得到最终正确密钥.

攻击所需的各项复杂度由定理 1 给出.

定理 1. 根据攻击算法 1,可恢复出 Piccolo-80 算法的全部主密钥,攻击所需数据复杂度为 $2^{58.6}$,计算复杂度为 2^{78},存储复杂度为 $2^{60.6}$.

证明:攻击所需数据复杂度主要由选择密文量决定.由攻击步骤 1 可知,对于每一个结构,选择密文的数目为 2^{24},选取 2^n 个这样的结构,得到攻击所需的选择密文量为 2^{n+24},也就是数据复杂度为 2^{n+24} 个选择密文.本文中所有攻击所需计算复杂度均采用的是文献[20]中的方法进行估计,即总的计算复杂度由构造明文结构、密钥猜测筛选阶段以及穷举剩余密钥这 3 部分的计算复杂度组成.由于 Piccolo 算法中 F 函数采用的是 SDS 结构,在一轮运算中,与计算其他环节操作相比,计算 F 函数所需的复杂度要高出很多,因此攻击的计算复杂度可依所需计算的 F 函数的个数进行近似估计,下面进行具体分析.

1. 由于攻击所需选择密文量为 2^{n+24},因此构造明密文对所需的计算复杂度为 $T_N=2^{n+24}\times2=2^{n+25}$.
2. Step 2.1 中,选定γ的一个值,猜测密钥(rk_{29}^L,rk_{28}^R)共有 2^{16} 种可能,根据已得的密文对,进行 $\frac{1}{2}$ 轮 Piccolo-80 算法解密操作,筛选 $\Delta X_{2,3}^{13}=0$ 的密文对,因此,攻击所需的计算复杂度为 $\frac{1}{2}\times(2^{n+23}\times2^{16})\times2=2^{n+39}$.
3. Step 2.2 中,根据步骤 2.1 中得到的密钥(K_1^L,K_0^R)和剩余 2^{n+7} 个密文对,进行 $\frac{1}{2}$ 轮加密操作,筛选 $\Delta Y_6^0=\gamma,\Delta Y_7^0=0$ 的密文对,因此,该步骤所需的计算复杂度为 $\frac{1}{2}\times(2^{n+7}\times2^{16})\times2=2^{n+23}$.
4. Step 2.3 中,与前两步类似,猜测密钥 WK_0,有 2^{16} 种取值,筛选符合条件的密文对,则该步骤的计算复杂度为 $\frac{1}{2}\times(2^{n-9}\times2^{32})\times2=2^{n+23}$.

© 中国科学院软件研究所 http://www.jos.org.cn

5. Step 2.4 中,猜测密钥 (rk_1^L, rk_0^R),验证 $\Delta Y_{6,7}^1 = 0$ 是否成立,则该步骤的计算复杂度为

$$\frac{1}{2}\times(2^{n-25}\times 2^{48})\times 2 = 2^{n+23}.$$

6. Step 2.5 中,对γ所有可能的 2^8-1 值进行遍历,进行密钥筛选.也就是重复上述攻击步骤 2.1~步骤 2.4,所以密钥猜测阶段总的计算复杂度体现在该步骤中,也就是需要再进行约 2^8-2 次步骤 2.1~步骤 2.4,因此密钥猜测阶段总的计算复杂度为

$$T_G = (2^8-1)\times\frac{1}{15}\times(2^{n+39}+2^{n+23}+2^{n+23}+2^{n+23}) = 2^{n+43.09}.$$

在上述攻击中,错误密钥不通过验证的概率为 $P=((1-2^{-16})^{2^{n-25}})^{2^8-1}\approx(1-2^{-16})^{2^{n-17}}$,总计对 48bits 密钥进行了猜测,因此剩余密钥量为 $K_R=2^{48}\times P$,也就是穷举剩余密钥的计算复杂度为 $T_R\approx 2^{32}\times K_R$.利用文献[20]的方法,为使得攻击效果更好,对攻击所需数据复杂度和计算复杂度进行折衷,选取 n=34.6,则攻击所需的计算复杂度总计为 $T=T_N+T_G+T_R=2^{n+25}+2^{n+43.09}+2^{75.63}\approx 2^{78}$ 次 15 轮 Piccolo-80 算法加密.

存储复杂度主要用于存储密文结构以及错误密钥,共需约为 $2^{60.6}$ 个字节.

综上,攻击所需数据复杂度为 $2^{58.6}$ 个选择密文,计算复杂度为 2^{78} 次 15 轮 Piccolo-80 算法加密,存储复杂度为 $2^{60.6}$ 个字节. □

2.3.2 攻击算法 2

根据 Piccolo 算法 F 函数组成部分中的 S 盒为 4bits,我们可以将算法每一轮的输入按半字节进行划分,密钥同样如此,为简化表示,每一轮的一个输入状态 $X_i^r=(X_i^{r^{L_0}}\|X_i^{r^{L_1}}\|X_i^{r^{R_1}}\|X_i^{r^{R_0}})$,将每一个主密钥定义为 $K_i=(K_i^{L_0}\|K_i^{L_1}\|K_i^{R_1}\|K_i^{R_0})$,相应的轮密钥也依此表示.通过简化密钥选取的条件,给出如下具体攻击算法.

- 攻击条件:选择第 14 轮的输入差分为$(\alpha,0,0,0,0,\beta,(\gamma\|0),0)$,密钥差分为 $\Delta K_2^L=(\gamma\|0)$.其中,α,β各表示一个差分活动字节,γ表示一个差分活动半字节.
- 攻击步骤:除数据收集阶段采用上述攻击条件,使得与算法 1 的步骤 1 略有不同外,其余步骤与攻击算法 1 基本类似.

定理 2 给出了攻击算法 2 的各项复杂度分析结果.

定理 2. 根据攻击算法 2,可恢复出 Piccolo-80 算法的全部主密钥,攻击所需数据复杂度为 $2^{62.6}$,计算复杂度为 $2^{77.93}$,存储复杂度为 $2^{64.6}$.

证明:显然对于 2^n 个结构,需要 2^{n+20} 个选择密文,相应的密钥猜测阶段所需的计算复杂度与攻击算法 1 也有所不同,具体各个环节所需的计算复杂度见表 5.

Table 5 Computation complexity analysis of attack Algorithm 2

表 5 攻击算法 2 的计算复杂度分析

具体环节	所需计算复杂度
构造明文结构	2^{n+21}
猜测密钥	$(2^4-1)\times\frac{1}{15}\times(2^{n+35}+2^{n+19}+2^{n+19}+2^{n+19})\approx 2^{n+35}$
穷举剩余密钥	$2^{48}\times(1-2^{-16})^{2^{n-25}}\times 2^{32}$

经过验证可得,选取 n=42.6,此攻击所需的数据复杂度约为 $2^{62.6}$ 个选择密文,计算复杂度为 $2^{77.93}$ 次 15 轮 Piccolo-80 算法加密,存储复杂度为 $2^{64.6}$ 个字节. □

3 21 轮 Piccolo-128 算法的相关密钥-不可能差分攻击

与第 2 节类似,本节首先分析 Piccolo-128 算法在输入状态差分和主密钥差分均仅有 1 个活动比特块的限制条件下能够具有的最长 17 轮相关密钥-不可能差分区分器,在此基础上,得到适用于攻击 21 轮含前向白化密钥的 Piccolo-28 算法的 16 轮相关密钥-不可能差分区分器,并给出相应的安全性分析结果.

© 中国科学院软件研究所 http://www.jos.org.cn

3.1 17轮相关密钥-不可能差分区分器

结合性质 1,考虑 Piccolo-28 主密钥的各个部分在各轮密钥中出现的分布情况,通过选取合适位置的密钥差分,实现与输入输出差分状态的抵消,能够构造得到的最长相关密钥-不可能差分区分器主要有 4 种情形,见表 6.

Table 6 17-round related-key impossible differential distinguishers of Piccolo-128

表 6 17 轮 Piccolo-128 算法的相关密钥-不可能差分区分器

	区分器位置	密钥差分	区分器具体表示	类似区分器总数
情形 1	第 1 轮~第 17 轮	$\Delta K_4^L=\alpha$	$(0,0,\alpha,0,0,0,0,0)\xrightarrow[\Delta K_4=(\alpha\|\|0)]{}(\alpha,0,0,0,0,0,0,0)$	2
情形 2	第 5 轮~第 21 轮	$\Delta K_0^R=\alpha$	$(0,0,0,\alpha,0,0,0,0)\xrightarrow[\Delta K_0=(0\|\|\alpha)]{}(0,0,0,0,0,\alpha,0,0)$	2
情形 3	第 9 轮~第 25 轮	$\Delta K_2^R=\alpha$	$(0,0,0,\alpha,0,0,0,0)\xrightarrow[\Delta K_2=(0\|\|\alpha)]{}(0,0,0,0,0,\alpha,0,0)$	2
情形 4	第 13 轮~第 29 轮	$\Delta K_6^L=\alpha$	$(0,0,\alpha,0,0,0,0,0)\xrightarrow[\Delta K_6=(\alpha\|\|0)]{}(\alpha,0,0,0,0,0,0,0)$	2

观察表 6 可得,区分器的密钥差分选取均位于轮密钥的左半部分.这是因为每轮轮密钥的右半部分采用的是主密钥(K_1,K_3,K_5,K_7)中的任意一个,而由性质 1,Piccolo-128 的轮密钥的右半部分是按(K_3,K_5,K_7,K_1,K_7,K_3,K_5,K_1,K_5,K_7,K_3,K_1)排列的一个 12 轮循环.其中,K_3,K_5 在轮密钥中连续出现 5 次的相距轮数为 18,K_7 在轮密钥中的相距轮数为 17,但该三者中间轮均已达到算法达到完全性的轮数要求,在截断差分条件下不能得到矛盾.K_1 在轮密钥中的相距轮数为 15<17,因此当密钥差分位于轮密钥的右半部分时,所能构造的区分器轮数定不超过 17 轮,所以未予考虑.

由此,我们可得 Piccolo-128 算法的 8 条 17 轮相关密钥-不可能差分区分器,且需要强调,情形 1 中包含了文献[19]给出的 Piccolo-128 算法的相关密钥-不可能差分区分器.

由于本文的攻击考虑前向白化密钥对算法安全性的影响,因此我们仅选取情形 1 中的区分器用于实施攻击.为了能够利用多相关密钥以及轮函数的等效结构实现攻击,我们对所得区分器进行改进,将区分器的第一轮置于密钥恢复攻击环节,也就是说改进后的区分器仅为第 2 轮~第 17 轮,表 7 给出了此 16 轮区分器的具体构造.而后,借助所得区分器向上向下分别拓展 2 轮和 3 轮,给出了 21 轮含前向白化密钥 Piccolo-128 算法的安全性分析结果.此外,情形 4 中的区分器也可用来分析 21 轮 Piccolo-128 算法仅含后向白化密钥的安全性,具体攻击过程和复杂度估计与下文给出的攻击算法 3、算法 4 基本一致.

Table 7 16-round related-key impossible differential distinguisher of Piccolo-128 with $\Delta K_4=(\alpha||0)$

表 7 Piccolo-128 在密钥差分分别选为$\Delta K_4=(\alpha||0)$时的 16 轮区分器

	加密方向			解密方向		
	第 i 轮	第 i 轮输入差分	第 i 轮密钥差分	第 i 轮	第 i 轮输入差分	第 i 轮密钥差分
密钥差分选为 $\Delta K_4=(\alpha\|\|0)$的情况	2	(0,0,0,0,0,0,0,0)	(0,0)	9	(*,0,*,*,0,*,*,*)	(0,0)
	3	(0,0,0,0,0,0,0,0)	(0,0)	10	(0,0,0,0,α,0,*,*)	(0,0)
	4	(0,0,0,0,0,0,0,0)	(0,0)	11	(0,0,α,0,0,0,0,0)	(ΔK_4,ΔK_5)=[(α||0),0]
	5	(0,0,0,0,0,0,0,0)	(0,0)	12	(0,0,0,0,0,0,0,0)	(0,0)
	6	(0,0,0,0,0,0,0,0)	(ΔK_4,ΔK_5)=[(α||0),0]	13	(0,0,0,0,0,0,0,0)	(0,0)
	7	(α,0,0,0,0,0,0,0)	(0,0)	14	(0,0,0,0,0,0,0,0)	(0,0)
	8	(*,0,0,0,0,*,α,0)	(ΔK_4,ΔK_5)=[(α||0),0]	15	(0,0,0,0,0,0,0,0)	(0,0)
	9	(*,*,0,0,*,*,*,*)	(0,0)	16	(0,0,0,0,0,0,0,0)	(0,0)
	–	–	–	17	(0,0,0,0,0,0,0,0)	(ΔK_4,ΔK_5)=[(α||0),0]
	–	–	–	18	(α,0,0,0,0,0,0,0)	–

3.2 21轮密钥恢复攻击

根据第 3.1 节所得的 16 轮区分器,向上解密 2 轮,向下加密 3 轮,得到对于 Piccolo-128 的 21 轮攻击路径,且与对 Piccolo-80 算法的攻击相同,对于 Piccolo-128 的攻击过程也由两个阶段组成:一为数据收集阶段,二为密钥猜测阶段.下文中给出两个攻击算法,其中,攻击算法 3 选取的密钥差分规模为 8bits,攻击算法 4 中密钥差分规模为 4bits.下面对算法 3 进行详细介绍,算法 4 由于步骤基本与算法 3 相似,在本节仅进行简单介绍.图 3 给出具

© 中国科学院软件研究所 http://www.jos.org.cn

体攻击路径,其中,黑色和白色标识与图 2 相同,灰色表示差分值为α的比特块.

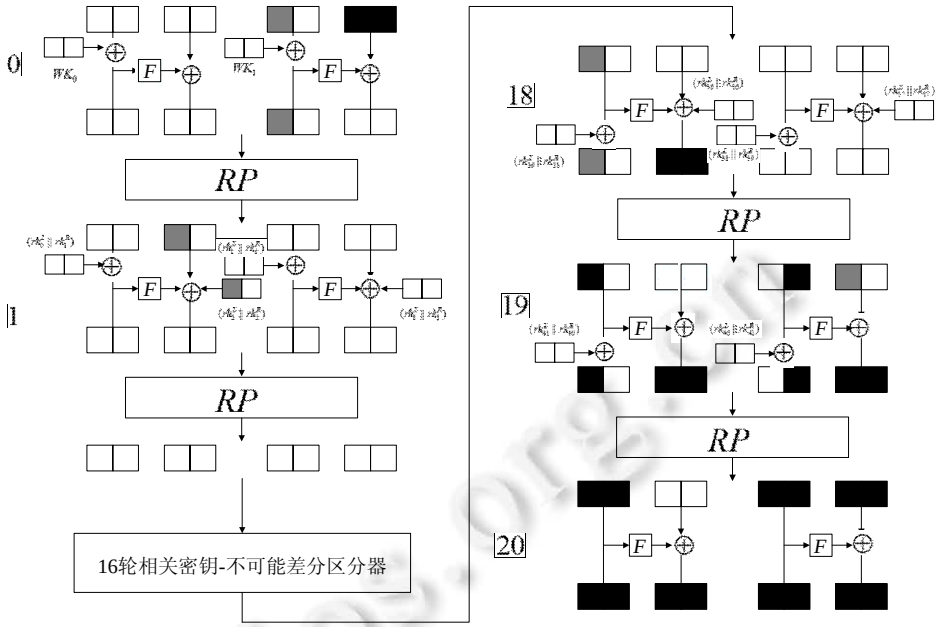

Fig.3 21-round related-key impossible differential cryptanalysis of Piccolo-128

图 3 21 轮 Piccolo-128 算法的相关密钥-不可能差分分析

3.2.1 攻击算法 3

- Step 1. 数据收集.

选择明文(P,P')的输入差分为$(0,0,0,0,\alpha,0,\beta,\gamma)$,同样,$\alpha,\beta,\gamma$表示差分活动字节.对于一个明文结构,需要选择$2^{24}$个明文,则得到$2^{47}$个明文对.当$\alpha$选定一个非零的取值时,因为有$2^8-1$种可能,所以对应的明文对数量$2^{39}$个.此时选择相应的密钥差分$\Delta K_4^L=\alpha$,基于性质 3 轮函数的等效结构,筛选满足在第 20 轮的输入差分有$\Delta X_{2,3}^{20}=0$,其余字节差分活动的明文对予以保留,构造2^n个结构,当α选定一个非零值时,可得到2^{n+23}个平均剩余明文对.

第 2 步中具体介绍密钥猜测阶段的攻击步骤,其中,步骤 2.1~步骤 2.4 是基于α的值取定的情况,$\alpha\neq 0$.

- Step 2. 密钥猜测.
  - Step 2.1. 猜测密钥 WK_1,对于选定的一个α,根据已得到的明文对,筛选使得的明文对,予以保留,则平均剩余明文对个数为$2^{n+23}\times 2^{-16}=2^{n+7}$.
  - Step 2.2. 猜测密钥(rk_{41}^L,rk_{40}^R),利用上述剩余明文对,进行第 19 轮左半部分解密,筛选$\Delta X_{2,3}^{19}=0$的明文对,予以保留,则平均剩余明文对个数为$2^{n+7}\times 2^{-16}=2^{n-9}$.
  - Step 2.3. 猜测密钥(rk_{40}^L,rk_{41}^R),利用上述剩余明文对,进行第 19 轮右半部分解密,筛选$\Delta X_6^{19}=\alpha$,$\Delta X_7^{19}=0$的明文对,予以保留,则该步骤后,平均有$2^{n-9}\times 2^{-16}=2^{n-25}$个明文对剩余.
  - Step 2.4. 猜测密钥rk_{38}^R,由步骤 2.1 已知K_1^L,根据密钥扩展算法的对应关系,也就是已知rk_{39}^L.对剩余明文对进行第 18 轮的左半部分解密,验证$\Delta X_{2,3}^{18}=0$是否成立:若此时仍有明文对通过检验,则说明此时猜测的密钥错误,将其放入错误密钥候选密钥集;否则,将其看作正确密钥的候选密钥,予以保留.
  - Step 2.5. 遍历所有可能的α值,重复进行上述 Step 2.1~Step 2.4,则对于所有可能的α值,均不通过 Step 2.4 检验的密钥可作为最终正确密钥的候选密钥.对于任一非零α值,存在满足第 2.4 步验证条件的密钥作为错误密钥筛除.上述攻击步骤已对 56bits 密钥$(rk_{41}^L,rk_{40}^L,rk_{41}^R,rk_{40}^R,rk_{38}^R,K_1^L,K_0^R)$进行了猜测,将筛选后得到的候选密钥与未猜测 72bits 密钥进行穷举,得到最终正确密钥.

© 中国科学院软件研究所 http://www.jos.org.cn

攻击所需的复杂度由定理 3 给出.

定理 3. 根据攻击算法 3,可恢复出 Piccolo-128 算法的全部主密钥,攻击所需数据复杂度为 $2^{62.3}$,计算复杂度为 $2^{82.5}$,存储复杂度为 $2^{64.3}$.

证明:攻击所需数据复杂度主要由选择明文量决定.由步骤 1 可得:对于每一个结构,选择明文数量为 2^{24},选取 2^n 个这样的结构,得到攻击所需的选择明文量为 2^{n+24},也就是数据复杂度为 2^{n+24} 个选择明文.各步骤具体的计算复杂度情况如下.

由于攻击所需选择明文为 2^{n+24},因此构造明文对所需的计算复杂度为 $T_N=2^{n+25}$.在 Step 2.1 中,攻击所需的计算复杂度为 $\frac{1}{2}\times(2^{n+23}\times 2^{16})\times 2=2^{n+39}$; Step 2.2 所需的计算复杂度为 $\frac{1}{2}\times(2^{n+7}\times 2^{32})\times 2=2^{n+39}$; Step 2.3 所需的计算复杂度为 $\frac{1}{2}\times(2^{n-9}\times 2^{48})\times 2=2^{n+39}$; Step 2.4 中,猜测密钥 rk_{38}^{R},验证 $\Delta X_{2,3}^{18}=0$ 是否成立,进行密钥筛选,则该步骤的计算复杂度为 $\frac{1}{2}\times(2^{n-25}\times 2^{56})\times 2=2^{n+31}$; Step 3 中,对 α 所有可能的 2^8-1 值进行遍历,进行密钥筛选.也就是重复上述攻击步骤 2.1~步骤 2.4,因此在密钥猜测阶段,总的计算复杂度约为

$$T_G=(2^8-1)\times\frac{1}{21}\times(2^{n+39}+2^{n+39}+2^{n+39}+2^{n+31})=2^{n+44.2}.$$

在上述攻击中,错误密钥不通过验证的概率为 $P=((1-2^{-16})^{2^{n-25}})^{2^{8-1}}$,总计对 56bits 密钥进行了猜测,因此剩余密钥量为 $K_R=2^{56}\times P$,取 n=38.3,也就是穷举剩余密钥的计算复杂度为 $T_R=K_R\times 2^{72}\approx 2^{72}$,则攻击总计所需的计算复杂度为 $T=T_N+T_G+T_R=2^{n+25}+2^{n+44.2}+2^{72}\approx 2^{82.5}$ 次 21 轮 Piccolo-128 算法加密.

存储复杂度主要用于存储明文结构以及错误密钥,共需约为 $2^{64.3}$ 个字节.

综上,攻击所需数据复杂度为 $2^{62.3}$ 个选择明文,计算复杂度为 $2^{82.5}$ 次 21 轮 Piccolo-128 算法加密,存储复杂度为 $2^{64.3}$ 个字节. □

3.2.2 攻击算法 4

- 攻击条件:与攻击算法 2 类似,对各轮输入输出状态和密钥比特块按半字节进行划分.选择明文差分为 $(0,0,0,0,(\alpha\|0),0,\beta,\gamma)$,密钥差分为 $\Delta K_4^L=(\alpha\|0)$.其中,α 表示一个差分活动半字节,β,γ 各表示一个差分活动字节.
- 攻击步骤:除数据收集阶段略异于攻击算法 3,其余步骤基本类似.

攻击算法 4 的复杂度分析结果由定理 4 给出.

定理 4. 根据攻击算法 4,恢复 Piccolo-128 算法的全部主密钥所需数据复杂度为 $2^{62.3}$,计算复杂度为 $2^{124.45}$,存储复杂度为 $2^{64.3}$.

证明:选取 n=42.3,则该攻击所需的数据复杂度约为 $2^{62.3}$ 个选择明文,计算复杂度为 $2^{124.45}$ 次 21 轮 Piccolo-128 算法加密,存储复杂度为 $2^{64.3}$ 个字节.主要证明过程与定理 2.3 类似,此处不再赘述. □

4 结束语

本文中利用相关密钥-不可能差分分析方法,结合算法密钥扩展方面的弱点,基于算法的等效结构,给出了除 Biclique 分析外,攻击轮数最长的缩减轮 Piccolo 算法的差分类分析结果.其中,对于 Piccolo-80 算法,找到所有 30 条 11 轮相关密钥-不可能差分区分器,并基于情形 1 的区分器,得到了 15 轮含前向白化密钥的攻击结果;对于 Piccolo-128 算法,找到所有 8 条 17 轮区分器,并基于改进后的 16 轮区分器,攻击了 21 轮含前向白化密钥的 Piccolo-128 算法.且根据所选取的密钥差分规模不同,对 Piccolo-80 和 Piccolo-128 分别给出了两个不同的攻击算法,攻击所需的复杂度均低于穷举分析,优于已有攻击结果.分析结果表明,15 轮 Piccolo-80 算法和 21 轮 Piccolo-128 算法在不含后向密钥的条件下均不能抵抗相关密钥-不可能差分攻击.能否在不增加算法实现代价的前提下,通过改进 Piccolo 算法的密钥扩展方式使得算法的安全性进一步提高,是下一步研究的方向.

© 中国科学院软件研究所 http://www.jos.org.cn

作者注 本文于 2018 年 5 月 22 日投稿至《软件学报》“面向自主安全可控的可信计算”专题,并于 2018 年 12 月 13 日被录用,于 2019 年正式发表.本文第一作者徐林宏的硕士学位论文于 2018 年 12 月底完成答辩,后提交至学位论文库.该硕士学位论文的部分章节内容基于本文工作成果,特此说明.

References:

[1] Bogdanov A, Knudsen LR, Leander G, *et al*. PRESENT: Anultra-lightweight block cipher. In: Paillier P, Verbauwhede I, eds. Proc. of the Int’l Workshop on Cryptographic Hardware and Embedded Systems (CHES 2007). LNCS 4727, Berlin, Heidelberg: Springer-Verlag, 2007. 450−466.

[2] Wu WL, Zhang L. LBlock: A lightweight block cipher. In: Lopez J, Tsudik G, eds. Proc. of the Int’l Conf. on Applied Cryptography and Network Security (ACNS 2011). LNCS 6715, Berlin, Heidelberg: Springer-Verlag, 2011. 327−344.

[3] Borghoff J, Canteaut A, Güneysu T, *et al*. PRINCE—A low-latency block cipher for pervasive computingapplications. In: Wang X, Sako K, eds. Proc. of the Int’l Conf. on the Theory and Application of Cryptology and Information Security—ASIACRYPT 2012. LNCS 7658, Berlin, Heidelberg: Springer-Verlag, 2012. 208−225.

[4] Beaulieu R, Treatman-Clark S, Shors D, *et al*. The SIMON and SPECK lightweight block ciphers. In: Proc. of the 2015 52nd ACM/EDAC/IEEE Design Automation Conf. (DAC). San Francisco: IEEE, 2015. 1−6. https://doi.org/10.1145/2744769.2747946

[5] Shibutani K, Isobe T, Hiwatari H, *et al*. Piccolo: An ultra-lightweight blockcipher. In: Preneel B, Takagi T, eds. Proc. of the Cryptographic Hardware and Embedded Systems (CHES 2011). LNCS 6917, Berlin, Heidelberg: Springer-Verlag, 2011. 342−357.

[6] Knudsen L. DEAL—A 128-Bit Blockcipher. AES Proposal, 1998.

[7] Biham E, Biryukov, Shanir A. Cryptanalysis of Skipjack reduced to 31 rounds using impossible differentials. In: Stern J, ed. Proc. of the Int’l Conf. on the Theory and Applications of Cryptographic Techniques—EUROCRYPTO’99. LNCS 1582, Berlin, Heidelberg: Springer-Verlag, 1999. 12−23.

[8] Mala H, Dakhilalian M, Rijmen V, *et al*. Improved impossible differential cryptanalysis of 7-round AES-128. In: Gong G, Gupta KC, eds. Proc. of the Int’l Conf. on Cryptology in India—INDOCRYPT 2010. LNCS 6498, Berlin, Heidelberg: Springer-Verlag, 2010. 282−291.

[9] Boura C, Naya-Plasencia M, Suder V. Scrutinizing and improving impossible differential attacks: Applications to CLEFIA, Camellia, LBlock and Simon. In: Sarkar P, Iwata T, eds. Proc. of the Int’l Conf. on Theory and Application of Cryptology and Information Security—ASIACRYPT 2014. LNCS 8873. Berlin Heidelberg: Springer, 2014.179−199.

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

[11] Knudsen LR. Cryptanalysis of LOKI91. In: Seberry J, Zheng Y, eds. Proc. of the Advances in Cryptology——Auscrypt’92. LNCS 718, Berlin, Heidelberg: Springer-Verlag, 1992. 196−208.

[12] Gong Z, Liu SS, Wen YM, *et al*. Biclique cryptanalysis using balanced complete bipartite subgraphs. SCIENCE CHINA Information Sciences, 2016,59(4):Article No.049101.

[13] Song J, Lee K, Lee H. Biclique cryptanalysis on lightweight block cipher: HIGHT and Piccolo. Int’l Journal of Computer Mathematics, 2013, 90(12):2564−2580.

[14] Ahmadi S, Ahmadian Z, Mohajeri J, *et al*. Low-Data complexity biclique cryptanalysis of block ciphers with application to Piccolo and HIGHT. IEEE Trans. on Information Forensics and Security, 2014,9(10):1641−1652.

[15] Azimi S, Ahmadian Z, Mohajeri J, *et al*. Impossible differential cryptanalysis of Piccolo lightweight block cipher. In: Proc. of the 2014 11th Int’l ISC Conf. on Information Security and Cryptology. Tehran: IEEE, 2014. 89−94. https://doi.org/10.1109/ISCISC.2014.6994028

[16] Tolba M, Abdelkhalek A, Youssef AM. Meet-in-the-Middle attacks on reduced round Piccolo. In: Güneysu T, Leander G, Moradi A, eds. Proc. of the Lightweight Cryptography for Security and Privacy—LightSec 2015. LNCS 9542. Cham: Springer-Verlag, 2016. 3−20.

[17] Liu Y, Cheng L, Liu ZQ, *et al*. Improved meet-in-the-middle attacks on reduced-round Piccolo. SCIENCE CHINA Information Sciences, 2018,61(3):Article No.032108.

[18] Minier M. On the security of piccolo lightweight block cipher against related-key impossible differentials. In: Paul G, Vaudenay S, eds. Proc. of the Progress in Cryptology—INDOCRYPT 2013. LNCS 8250, Cham: Springer-Verlag, 2013. 308−318.

[19] Zheng XQ, Zhang WY. Related-key impossible differential analysis of reduced round Piccolo. Application Research of Computers, 2016,33(5):1528−1532 (in Chinese with English abstract).

© 中国科学院软件研究所 http://www.jos.org.cn

[20] Boura C, Lallemand V, Naya-Plasencia M, *et al*. Making the impossible possible. Journal of Cryptology, 2018,31(1):101−133.

附中文参考文献:

[19] 郑向前,张文英.Piccolo 缩减轮数的相关密钥-不可能差分分析.计算机应用研究,2016,33(5):1528−1532.

附录 A

Table A 11-round distinguisher of Piccolo-80 with $\Delta K_2=(\gamma\|0)$ or $\Delta K_1=(0\|\gamma)$

表 A Piccolo-80 在密钥差分分别选为$\Delta K_2=(\gamma\|0)$和$\Delta K_1=(0\|\gamma)$时的 11 轮区分器

	密钥差分选为$\Delta K_2=(\gamma\|0)$的情况				密钥差分选为$\Delta K_1=(0\|\gamma)$的情况		
	第 i 轮	第 i 轮输入差分	第 i 轮密钥差分		第 i 轮	第 i 轮输入差分	第 i 轮密钥差分
加密方向	2	$(0,0,\gamma,0,0,0,0,0)$	$(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]$	加密方向	1	$(0,0,0,0,0,0,0,\gamma)$	$(\Delta K_0,\Delta K_1)=[0,(0\|\gamma)]$
	3	$(0,0,0,0,0,0,0,0)$	$(0,0)$		2	$(0,0,0,0,0,0,0,0)$	$(0,0)$
	4	$(0,0,0,0,0,0,0,0)$	$(0,0)$		3	$(0,0,0,0,0,0,0,0)$	$(0,0)$
	5	$(0,0,0,0,0,0,0,0)$	$(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]$		4	$(0,0,0,0,0,0,0,0)$	$(\Delta K_0,\Delta K_1)=[0,(0\|\gamma)]$
	6	$(\gamma,0,0,0,0,0,0,0)$	$(0,0)$		5	$(0,\gamma,0,0,0,0,0,0)$	$(0,0)$
	7	$(*,0,0,0,0,*,\gamma,0)$	$(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]$		6	$(*,0,0,\gamma,0,*,0,0)$	$(\Delta K_0,\Delta K_1)=[0,(0\|\gamma)]$
	8	$(?,*,0,0,?,*,*,*)$	–		7	$(*,?,0,0,*,?,*,*)$	–
解密方向	8	$(*,0,*,*,0,*,?,0)$	$(0,0)$	解密方向	7	$(*,0,*,?,0,*,*,*)$	$(0,0)$
	9	$(0,0,0,0,\gamma,0,*,*)$	$(0,0)$		8	$(0,0,0,0,0,\gamma,*,*)$	$(0,0)$
	10	$(0,0,\gamma,0,0,0,0,0)$	$(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]$		9	$(0,0,0,0,0,0,0,\gamma)$	$(\Delta K_0,\Delta K_1)=[0,(0\|\gamma)]$
	11	$(0,0,0,0,0,0,0,0)$	$(0,0)$		10	$(0,0,0,0,0,0,0,0)$	$(0,0)$
	12	$(0,0,0,0,0,0,0,0)$	$(\Delta K_2,\Delta K_3)=[(\gamma\|0),0]$		11	$(0,0,0,0,0,0,0,0)$	$(\Delta K_0,\Delta K_1)=[0,(0\|\gamma)]$
	13	$(\gamma,0,0,0,0,0,0,0)$	–		12	$(0,\gamma,0,0,0,0,0,0)$	–

附录 B

Table B 11-round distinguisher of Piccolo-80 with $\Delta K_4=(\gamma\|0)$

表 B Piccolo-80 在密钥差分选为$\Delta K_4=(\gamma\|0)$时的 11 轮区分器

	加密方向			解密方向		
	第 i 轮	第 i 轮输入差分	第 i 轮密钥差分	第 i 轮	第 i 轮输入差分	第 i 轮密钥差分
密钥差分选为$\Delta K_4=(\gamma\|0)$的情况	3	$(0,0,\gamma,0,0,0,\gamma,0)$	$(\Delta K_4,\Delta K_4)=[(\gamma\|0),(\gamma\|0)]$	11	$(0,0,\gamma,0,0,0,\gamma,0)$	$(\Delta K_4,\Delta K_4)=[(\gamma\|0),(\gamma\|0)]$
	4	$(0,0,0,0,0,0,0,0)$	$(0,0)$	12	$(0,0,0,0,0,0,0,0)$	$(0,0)$
	5	$(0,0,0,0,0,0,0,0)$	$(0,0)$	13	$(0,0,0,0,0,0,0,0)$	$(\Delta K_4,\Delta K_4)=[(\gamma\|0),(\gamma\|0)]$
	6	$(0,0,0,0,0,0,0,0)$	$(0,0)$	14	$(\gamma,0,0,0,\gamma,0,0,0)$	–
	7	$(0,0,0,0,0,0,0,0)$	$(0,0)$	–	–	–
	8	$(0,0,0,0,0,0,0,0)$	$(\Delta K_4,\Delta K_4)=[(\gamma\|0),(\gamma\|0)]$	–	–	–
	9	$(\gamma,0,0,0,\gamma,0,0,0)$	$(0,0)$	–	–	–
	10	$(*,*,\gamma,0,*,*,\gamma,0)$	$(0,0)$	–	–	–
	11	$(*,*,?,*,*,*,?,*)$	–	–	–	–

徐林宏(1995－),男,江苏盐城人,硕士,主要研究领域为分组密码的设计与分析.

崔竞一(1992－),男,博士生,主要研究领域为分组密码的设计与分析.

郭建胜(1972－),男,博士,教授,主要研究领域为信息安全,密码学.

李明明(1995－),男,硕士,主要研究领域为分组密码的设计与分析.

© 中国科学院软件研究所 http://www.jos.org.cn