文章编号:1671-4229(2022)04-0012-09

AES不可能差分攻击的进一步改进

闫雪萍,戚文峰*,谭 林
(战略支援部队信息工程大学 网络空间安全学院,河南 郑州 450001)

摘 要:AES是目前使用最广泛的分组密码算法。不可能差分密码分析是评估分组密码算法安全性的重要方法之一,目前AES-128的7轮不可能差分密钥恢复攻击是单密钥模式下轮数最长的攻击之一。在不可能差分攻击中,为了获得满足区分器差分的数据,需要进行数据对和猜测密钥的筛选,它们之间有很强的关联性,对攻击复杂度有很大影响。通过对筛选数据对和猜测密钥进行折中可以使不可能差分攻击的时间复杂度较低。目前时间复杂度最低的7轮AES-128不可能差分攻击是2010年Mala等利用筛选数据对和猜测密钥的一个折中提出的,攻击的时间、数据和存储复杂度分别为$2^{110.1}$、$2^{106.2}$和$2^{94.2}$。如果采用只筛选数据对的方法,攻击的数据复杂度和存储复杂度相对较低。2018年,Boura等利用只筛选数据对得到时间、数据和存储复杂度分别为2^{113}、$2^{105.1}$和$2^{74.1}$(原文中的时间复杂度$2^{106.88}$被更正为2^{113})的7轮AES-128不可能差分攻击。在EUROCRYPT 2021上,Leurent等发现了AES密钥方案的新表示技术,将Boura等攻击的时间复杂度改进到$2^{110.9}$。文章将Leurent等提出的AES密钥方案的新表示技术应用于Mala等的7轮AES-128不可能差分攻击,利用改进的密钥方案筛选过程将攻击的时间复杂度从$2^{110.1}$改进到$2^{108.96}$,数据复杂度从$2^{106.2}$改进到2^{105}。文章给出的7轮AES-128不可能差分攻击的时间复杂度是上述3个攻击算法中最低的。

关键词:不可能差分分析;AES;密钥方案
中图分类号:TN 918.1 **文献标志码**:A

Further improvement of impossible differential attack on AES

YAN Xue-ping, QI Wen-feng, TAN Lin*
(School of Cyber Security, Strategic Support Force Information Engineering University, Zhengzhou 450001, China)

Abstract: The Advanced Encryption Standard (AES) is currently the most widely used block cipher. Impossible differential cryptanalysis is an important approach to evaluate the security of block ciphers. Currently, impossible differential attacks on 7 rounds of AES-128 are among the attacks with the longest rounds in the single-key setting. In an impossible differential attack, to obtain data that satisfies the difference of the distinguisher, it is necessary to filter data pairs and key guesses, and there is a close correlation between them, which has a great impact on the complexity. By making compromises between filtering data pairs and key guesses, we can reduce time complexity of an impossible differential attack. The impossible differential attack on 7-round AES-128 with the lowest time complexity was proposed by Mala, et al. in 2010 with time, data, and memory complexities of $2^{110.1}$, $2^{106.2}$, and $2^{94.2}$, respectively, using a compromise between filtering data pairs and keys guesses. If we use the method that only filters data pairs, data and memory complexities of the impossible differential attack are low. In 2018, Boura, et al. used filters data pairs to obtain an impossible differential attack on 7-round AES-128 with time, data and memory complexities of 2^{113}, $2^{105.1}$, and $2^{74.1}$ (the time complexity

收稿日期:2022-09-10;**修回日期**:2022-10-13
基金项目:国家密码发展基金资助项目(MMJJ20180204, MMJJ20170103)
作者简介:闫雪萍(1997—),女,博士研究生. E-mail:yanxueping163@163.com
* 通信作者. E-mail:wenfeng.qi@263.net
引文格式:闫雪萍,戚文峰,谭林. AES不可能差分攻击的进一步改进[J].广州大学学报(自然科学版),2022,21(4):12-20.

in the original paper $2^{106.88}$ was corrected to 2^{113}), respectively. In EUROCRYPT 2021, Leurent, et al. discovered new representations of the AES key schedule and improved the time complexity of the attack by Boura, et al. to $2^{110.9}$. This paper applies new representations of the AES key schedule proposed by Leurent, et al. to the impossible differential attack on 7 rounds of AES-128 proposed by Mala, et al., and improves the time complexity of the attack from $2^{110.1}$ to $2^{108.96}$ and the data complexity from $2^{106.2}$ to 2^{105} by the improved filtering process of key schedule. The time complexity of our attack is the lowest of the three attacks mentioned above.

Key words: impossible differential cryptanalysis; AES; key schedule

0　引　言

高级加密标准 AES[1]是目前使用最广泛、研究最多的分组密码算法。许多密码算法、Hash 函数和伪随机数发生器采用类似 AES 的结构来设计,许多密码算法甚至直接使用减轮 AES 作为算法的关键部件,如 Saturnin[2]、PRINCE[3]、LED[4]和 AEZ[5]等。从 AES 提出至今二十多年来,学者们进行了大量密码分析研究。虽然未对 AES 产生实际的威胁,但学术界持续的研究促进了人们对 AES 密码结构性质的认识和 AES 密码分析技术的发展。对 AES 主要的密码分析技术有:积分[1,6]、不可能差分[7-14]、零相关线性[15]、子空间路径[16]、混合差分[17-18]、yoyo[19]、交换攻击[20]、折回镖[21]、Boomeyong[22]和中间相遇攻击[23]等。

不可能差分分析技术[24]利用概率为 0 的差分特征对密码算法进行区分或者密钥恢复攻击,是评估分组密码算法安全性的重要方法之一。2000 年,Biham 等[7]提出 AES 的 4 轮不可能差分区分器和 5 轮密钥恢复攻击。2002 年,Cheon 等[8]将其扩展到 6 轮。Zhang 等[9]和 Bahrak 等[10]分别利用不同的 4 轮 AES 不可能差分区分器给出了 7 轮 AES-128 的密钥恢复攻击,时间复杂度均为 2^{119},数据复杂度均为 $2^{115.5}$。2008 年,Lyu 等[11]利用多个差分和密钥扩展算法将文献[10]中攻击的时间复杂度改进到 $2^{117.2}$、数据复杂度改进到 $2^{112.2}$。2010 年,Mala 等[12]提出新的 4 轮 AES 不可能差分区分器,利用数据筛选和密钥猜测的折中将 7 轮 AES-128 攻击的时间、数据和存储复杂度分别改进到 $2^{110.1}$,$2^{106.2}$和 $2^{94.2}$。2018 年,Boura 等[13]改变筛选数据对和猜测密钥的顺序,利用多个不可能差分区分器对攻击进行了改进,得到时间复杂度为 2^{113},数据复杂度为 $2^{105.1}$的 7 轮 AES-128 密钥恢复攻击(时间复杂度被文献[14]由 $2^{106.88}$更正为 2^{113})。在 EUROCRYPT 2021 上,Leurent 等[14]发现了 AES 密钥方案的新特征,通过选择不同的基表示,AES-128 的轮密钥可以表示成 4 个 32 比特的块独立地进行变换。利用密钥方案的新表示技术,文献[14]改进了 Boura 等的 7 轮不可能差分攻击,时间、数据和存储复杂度分别为 $2^{110.9}$、$2^{104.9}$和 $2^{74.9}$。与 Mala 等的攻击相比,文献[14]的攻击数据复杂度有一定减少,时间复杂度略有增加。

本文将 Leurent 等提出的密钥方案的新表示技术应用于 Mala 等的 7 轮 AES-128 不可能差分攻击,将攻击的时间复杂度由 $2^{110.1}$改进到 $2^{108.96}$,数据复杂度从 $2^{106.2}$改进到 2^{105}。表 1 给出了目前 5 轮以上 AES-128 密钥恢复攻击的主要结果,其中,时间复杂度中 XOR 表示异或,其余为相应轮数的 AES 加密,数据复杂度中 ACC 指适应性选择明密文,其余为选择明文。

表 1　5 轮以上 AES-128 密钥恢复攻击主要结果

Table 1　Key recovery attacks on 5 and more rounds of AES-128

轮数	攻击方式	时间复杂度	数据复杂度	存储复杂度	参考文献
5	混合差分	2^{24}	2^{24}	$2^{21.5}$	[17]
	折回镖	$2^{16.5}$	2^{15} ACC	2^{9}	[21]
6	积分	2^{44}	$2^{34.6}$	2^{32}	[6]
	Boomeyong	2^{78}	$2^{79.72}$ ACC	2^{28}	[22]
	不可能差分	2^{46}	$2^{114.5}$	2^{45}	[9]
	混合差分	2^{81}	$2^{27.5}$	$2^{27.5}$	[17]
	混合差分	$2^{62.62}$	$2^{44.42}$	$2^{44.42}$	[18]

(续表1)

轮数	攻击方式	时间复杂度	数据复杂度	存储复杂度	参考文献
7	中间相遇	2^{97}	2^{99}	2^{98}	[23]
	不可能差分	2^{119}	$2^{115.5}$	2^{45}	[9]
	不可能差分	2^{119}	$2^{115.5}$	2^{109}	[10]
	不可能差分	$2^{117.2}$	$2^{112.2}$	$2^{112.2}$	[11]
	不可能差分	$2^{110.1}$	$2^{106.2}$	$2^{94.2}$	[12]
	不可能差分	2^{113*}	$2^{105.1}$	$2^{74.1}$	[13]
	不可能差分	$2^{110.9}$	$2^{104.9}$	$2^{71.9}$	[14]
	不可能差分	$2^{108.96}$	2^{105}	2^{96}	第3.2节

注:文献[13]给出的时间复杂度为$2^{106.88}$,后被文献[14]更正为2^{113}

1 准备工作

1.1 AES算法简介

AES算法的分组长度为128比特,密钥长度支持128比特、192比特和256比特,相应的迭代轮数分别为10、12和14,分别用AES-128、AES-192和AES-256来表示。128比特状态通常用有限域$\mathbb{F}_{2^8}$上4×4的矩阵来描述,矩阵中字节顺序如图1所示。AES的轮变换包括以下4个操作:

(1)字节替换(SB):状态矩阵的每个字节查询同一个8比特S盒。

(2)行移位(SR):将状态矩阵的第i行循环左移i个字节,其中$i=0,1,2,3$。

(3)列混合(MC):用$\mathbb{F}_{2^8}$上一个MDS矩阵乘以状态矩阵的每一列。

(4)轮密钥加(AK):将状态与轮密钥逐比特异或。

0	4	8	12
1	5	9	13
2	6	10	14
3	7	11	15

图1 状态矩阵字节顺序

Fig. 1 Order of bytes in the state matrix

在加密时,明文先与主密钥异或,再进行相应数量的轮变换,最后一轮没有MC操作。AES一轮轮变换为$AK\circ MC\circ SR\circ SB$,当交换线性运算$AK$和$MC$的顺序时,轮变换可以表示为$MC\circ AK'\circ SR\circ SB$。AES的轮密钥由主密钥$K_0$通过密钥扩展方案生成,AES-128的密钥扩展方案如下:

$$K_{r,0}=K_{r-1,0}+S(K_{r-1,13})+C_r,$$
$$K_{r,1}=K_{r-1,1}+S(K_{r-1,14}),$$
$$K_{r,2}=K_{r-1,2}+S(K_{r-1,15}),$$
$$K_{r,3}=K_{r-1,3}+S(K_{r-1,12}),$$
$$K_{r,j}=K_{r-1,j}+K_{r,j-4},$$

其中,C_r是轮常数,$K_{r,j}$表示第r轮轮密钥K_r的第j个字节,$1\leqslant r\leqslant 10$,$0\leqslant j\leqslant 15$。关于AES算法更详细的介绍参见文献[1]。本文用$col(i)$表示矩阵的第i列,$SR(col(i))$表示矩阵的第i条反对角线,$SR^{-1}(col(i))$表示矩阵的第i条对角线,$i=0,1,2,3$。用$\boldsymbol{X}_i$表示矩阵$\boldsymbol{X}$的第i个元素,$\boldsymbol{X}_{\{i_1,i_2,\cdots,i_n\}}$表示矩阵$\boldsymbol{X}$的第$i_1,i_2,\cdots,i_n$个元素。用$\boldsymbol{X}_{col(i_1,i_2,\cdots,i_n)}$表示矩阵$\boldsymbol{X}$的第$i_1,i_2,\cdots,i_n$列上的元素,$\boldsymbol{X}_{SR(col(i_1,i_2,\cdots,i_n))}$表示矩阵$\boldsymbol{X}$的第$i_1,i_2,\cdots,i_n$条反对角线上的元素,$\boldsymbol{X}_{SR^{-1}(col(i_1,i_2,\cdots,i_n))}$表示矩阵$\boldsymbol{X}$的第$i_1,i_2,\cdots,i_n$条对角线上的元素。

1.2 AES密钥方案的新表示技术

在EUROCRYPT 2021上,通过分析AES-128密钥方案的不变子空间,Leurent等[14]发现选择$\mathbb{F}_2^8$上16维向量空间的另一组基$e_0,e_1,\cdots,e_{15}$时,密钥方案可以表示成4个块独立地进行变换。这里

$$e_0=(0,0,0,1,0,0,0,1,0,0,0,1,0,0,0,1),$$
$$e_1=(0,0,1,0,0,0,0,0,0,0,0,0,0,0,0,0),$$
$$e_2=(0,1,0,0,0,1,0,0,0,0,0,0,0,0,0,0),$$
$$e_3=(1,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0),$$
$$e_4=(0,0,1,0,0,0,1,0,0,0,1,0,0,0,1,0),$$
$$e_5=(0,1,0,0,0,0,0,0,0,0,0,0,0,0,0,0),$$
$$e_6=(1,0,0,0,1,0,0,0,0,0,0,0,0,0,0,0),$$
$$e_7=(0,0,0,1,0,0,0,0,0,0,0,1,0,0,0,0),$$
$$e_8=(0,1,0,0,0,1,0,0,0,1,0,0,0,1,0,0),$$
$$e_9=(1,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0),$$
$$e_{10}=(0,0,0,1,0,0,0,1,0,0,0,0,0,0,0,0),$$
$$e_{11}=(0,0,1,0,0,0,0,0,0,0,1,0,0,0,0,0),$$
$$e_{12}=(1,0,0,0,1,0,0,0,1,0,0,0,1,0,0,0),$$
$$e_{13}=(0,0,0,1,0,0,0,0,0,0,0,0,0,0,0,0),$$

$e_{14}=(0,0,1,0,0,0,1,0,0,0,0,0,0,0,0,0)$,

$e_{15}=(0,1,0,0,0,0,0,0,0,1,0,0,0,0,0,0)$。

对于 AES-128 第 r 轮的轮密钥 $K_r=(K_{r,0},K_{r,1},\cdots,K_{r,15})$,设 $s_r=(s_{r,0},s_{r,1},\cdots,s_{r,15})$是 K_r 在基 $e_0,e_1,\cdots,e_{15}$ 下的坐标,$0\leqslant r\leqslant 10$。对行向量和列向量不加区分,则 $K_r=\boldsymbol{A}s_r$,其中矩阵 $\boldsymbol{A}$ 是标准基到基 $e_0,e_1,\cdots,e_{15}$的过渡矩阵,

$$\boldsymbol{A}=\begin{bmatrix}
0&0&0&1&0&0&1&0&0&1&0&0&1&0&0&0\\
0&0&1&0&0&1&0&0&1&0&0&0&0&0&0&1\\
0&1&0&0&1&0&0&0&0&0&0&1&0&0&1&0\\
1&0&0&0&0&0&0&1&0&0&1&0&0&1&0&0\\
0&0&0&0&0&0&1&0&0&0&0&0&1&0&0&0\\
0&0&1&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&1&0\\
1&0&0&0&0&0&0&0&0&0&1&0&0&0&0&0\\
0&0&0&1&0&0&0&0&0&0&0&1&0&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&1\\
0&0&0&0&1&0&0&0&0&0&0&1&0&0&0&0\\
1&0&0&0&0&0&0&1&0&0&0&0&0&0&0&0\\
0&0&0&0&0&0&0&0&0&0&0&0&1&0&0&0\\
0&0&0&0&0&0&0&0&1&0&0&0&0&0&0&0\\
0&0&0&0&1&0&0&0&0&0&0&0&0&0&0&0\\
1&0&0&0&0&0&0&0&0&0&0&0&0&0&0&0
\end{bmatrix}。$$

于是有 $s_r=\boldsymbol{A}^{-1}K_r$,具体表示为

$$\begin{cases}
s_{r,0}=K_{r,15},\\
s_{r,1}=K_{r,14}+K_{r,10}+K_{r,6}+K_{r,2},\\
s_{r,2}=K_{r,13}+K_{r,5},\\
s_{r,3}=K_{r,12}+K_{r,8},\\
s_{r,4}=K_{r,14},\\
s_{r,5}=K_{r,13}+K_{r,9}+K_{r,5}+K_{r,1},\\
s_{r,6}=K_{r,12}+K_{r,4},\\
s_{r,7}=K_{r,15}+K_{r,11},\\
s_{r,8}=K_{r,13},\\
s_{r,9}=K_{r,12}+K_{r,8}+K_{r,4}+K_{r,0},\\
s_{r,10}=K_{r,15}+K_{r,7},\\
s_{r,11}=K_{r,14}+K_{r,10},\\
s_{r,12}=K_{r,12},\\
s_{r,13}=K_{r,15}+K_{r,11}+K_{r,7}+K_{r,3},\\
s_{r,14}=K_{r,14}+K_{r,6},\\
s_{r,15}=K_{r,13}+K_{r,9}。
\end{cases}\tag{1}$$

对于 AES-128 第 $r+1$ 轮的轮密钥 K_{r+1},设 s_{r+1}是 K_{r+1}在基 $e_0,e_1,\cdots,e_{15}$下的坐标,$0\leqslant r\leqslant 9$。文献[14]发现了如下特征:

$$\begin{cases}
s_{r+1,0}=s_{r,13}+S(s_{r,12}),\\
s_{r+1,1}=s_{r,14},\\
s_{r+1,2}=s_{r,15},\\
s_{r+1,3}=s_{r,12},\\
s_{r+1,4}=s_{r,1}+S(s_{r,0}),\\
s_{r+1,5}=s_{r,2},\\
s_{r+1,6}=s_{r,3},\\
s_{r+1,7}=s_{r,0},\\
s_{r+1,8}=s_{r,5}+S(s_{r,4}),\\
s_{r+1,9}=s_{r,6},\\
s_{r+1,10}=s_{r,7},\\
s_{r+1,11}=s_{r,4},\\
s_{r+1,12}=s_{r,9}+S(s_{r,8})+C_r,\\
s_{r+1,13}=s_{r,10},\\
s_{r+1,14}=s_{r,11},\\
s_{r+1,15}=s_{r,8}。
\end{cases}\tag{2}$$

根据公式(2),如果已知 s_r 的 4 个字节 $s_{r,\{4i,4i+1,4i+2,4i+3\}}$ 就能够计算 s_{r+1} 的 4 个字节 $s_{r+1,\{4(i+1),4(i+1)+1,4(i+1)+2,4(i+1)+3\}}$,这里脚标模 16,$0\leqslant i\leqslant 3$. 利用轮密钥在基 $e_0,e_1,\cdots,e_{15}$下的新表示,将 AES-128 的密钥扩展方案等价转换成 4 个块的独立运算,这一特征可以用于不可能差分攻击时验证不同轮密钥部分字节的关系。

2　7 轮不可能差分攻击的改进

本节利用文献[14]的 AES 密钥方案的新表示技术给出改进的密钥方案筛选过程,将文献[12]中攻击的时间复杂度由 $2^{110.1}$ 改进到 $2^{108.96}$,数据复杂度由 $2^{106.2}$ 改进到 2^{105}。文献[12]使用的 4 轮 AES 不可能差分区分器如图 2 所示,输入差分在某条对角线上为 0,输出差分不可能仅有 1 个字节非 0。

下面介绍文献[12]的 7 轮 AES-128 不可能差分攻击流程。如图 3 所示,记第一轮 *SB* 前的状态为 X,第一轮 *MC* 后的状态为 X^*,第二轮 *SB* 前的状态为 Y,第二轮 *MC* 后的状态为 Y^*。交换第六轮 *MC* 和 *AK*,使用等效的轮密钥加。记第六轮 *MC* 前的状态为 Z,第七轮 *SR* 后的状态为 Z^*。选取 2^n 个不同的明文结构,每个结构在第 0 条和第二条对角线遍历、其余 8 个字节取任意固定值,筛选密文差分满足 $(C_1+C_2)_{SR(col(0,1,2))}=0$ 的 $2^{n+127}\times 2^{-96}=2^{n+31}$ 个明文对 (P_1,P_2)。对这 2^{n+31} 个明文对

中的每个对(P_1,P_2)执行以下操作:首先猜测密钥$K_{0,SR^{-1}(col(0,2))}$,部分加密到状态X^*,筛选出满足状态差分$(X_1^*+X_2^*)_{\{1,3,9,11\}}=0$的$K_{0,SR^{-1}(col(0,2))}$的猜测值。再猜测密钥$K_{1,\{0,2,8,10\}}$,进一步部分加密到状态$Y^*$,筛选出满足状态差分$Y_1^*+Y_2^*$在某条对角线上为0的$K_{1,\{0,2,8,10\}}$的猜测值。猜测密钥$K_{7,SR(col(3))}$,将$(C_1,C_2)$部分解密到状态$Z$,筛选出满足状态差分$Z_1+Z_2$的第三列仅有1个字节非零的$K_{7,SR(col(3))}$的猜测值。筛选出的16字节密钥$K_{0,SR^{-1}(col(0,2))}\|K_{1,\{0,2,8,10\}}\|K_{7,SR(col(3))}$是不可能的,将被删除。在16字节密钥的一个错误猜测值下,一个明文对(P_1,P_2)的状态差分满足$(X_1^*+X_2^*)_{\{1,3,9,11\}}=0$、$Y_1^*+Y_2^*$在某条对角线为0且$Z_1+Z_2$的第三列仅有1个字节非零的概率为$2^{-32}\times4\times2^{-16}\times4\times2^{-24}=2^{-68}$。为了使得对于密钥的每个错误猜测值都平均存在一个对,可以将其进行筛选,令$2^{n+31}\times2^{-68}\geqslant1$,得$n\geqslant37$。

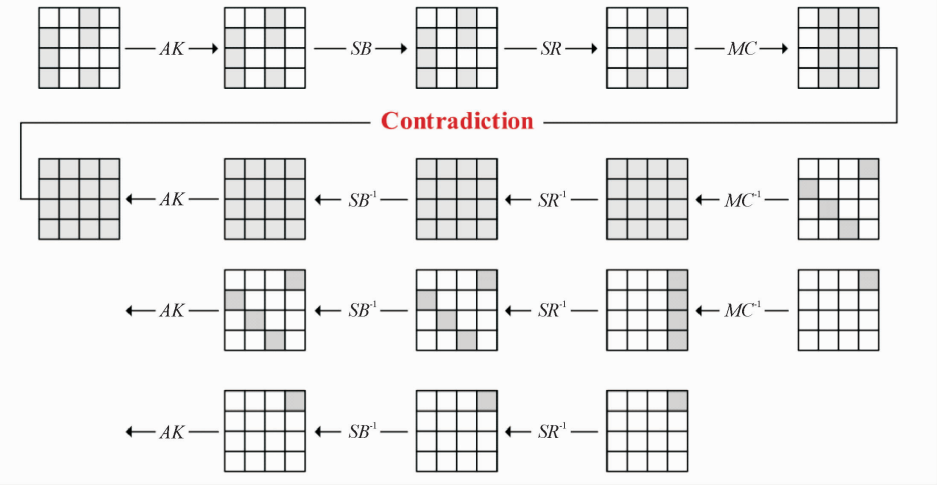

图2 AES的4轮不可能差分区分器

Fig. 2 Impossible differential distinguisher for 4-round AES

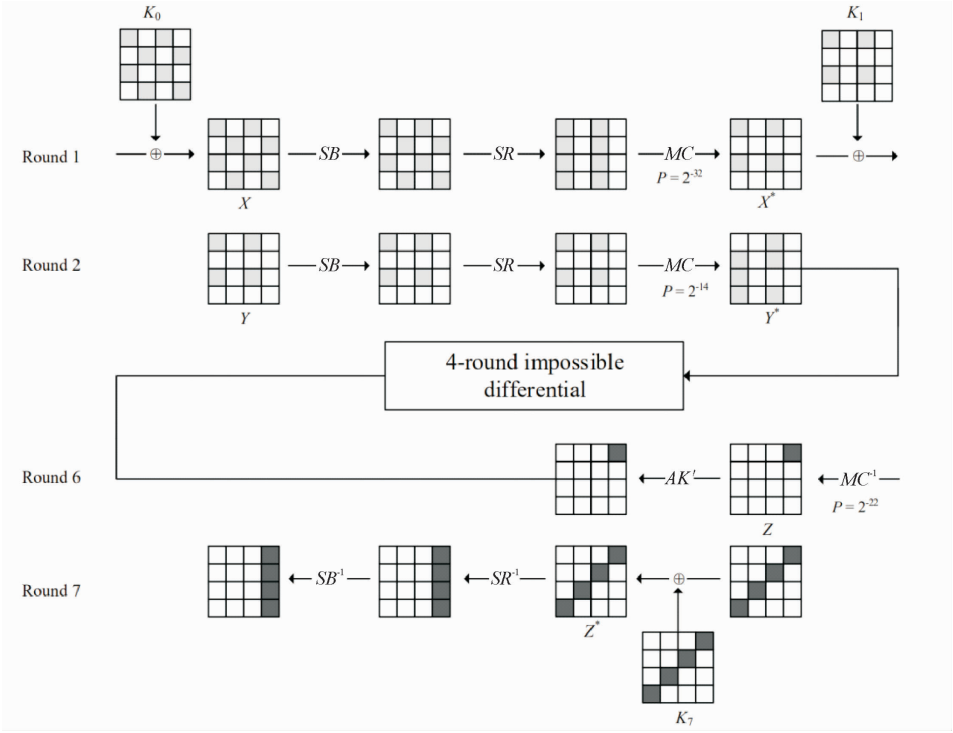

图3 7轮AES不可能差分攻击

Fig. 3 Impossible differential attack on 7-round AES

根据 AES-128 密钥方案中轮密钥之间的关系,

$$K_1,0 = K_0,0 + S(K_{0,13}) + C_1,$$

$$K_{1,2} = K_{0,2} + S(K_{0,15}),$$

猜测 $K_{0,SR^{-1}(col(0,2))}$ 之后,可以直接计算出 $K_{1,\{0,2\}}$。因此,在算法中,只需要猜测 14 字节的轮密钥 $K_{0,SR^{-1}(col(0,2))} \| K_{1,\{8,10\}} \| K_{7,SR(col(3))}$ 即可进行攻击。猜测 $K_{1,\{8,10\}}$ 之后,根据

$$K_0,4 = K_{1,8} + K_{1,0} + K_{0,8},$$

$$K_{0,6} = K_{1,2} + K_{1,10} + K_{0,10},$$

可以计算出 $K_{0,\{4,6\}}$,所以算法最终可以筛选出 $K_{0,\{0,5,10,15,2,7,8,13,4,6\}} \| K_{7,SR(col(3))}$ 的猜测值。

假设 $K_{0,\{0,5,10,15,2,7,8,13,4,6\}} \| K_{7,SR(col(3))}$ 的每个猜测值经过不可能差分区分器的筛选后被留下的概率为 P,则平均有 $2^{112}P$ 个值被留下。对于每个留下的值,首先遍历主密钥 K_0 剩余的 6 个字节 $K_{0,\{1,3,9,11,12,14\}}$,由密钥扩展方案计算出 K_7,并筛选出满足计算的 $K_{7,SR(col(3))}$ 与被留下的值相同的 $K_{0,\{1,3,9,11,12,14\}}$,这一过程被称为密钥方案筛选过程。被筛选出的 $K_{0,\{1,3,9,11,12,14\}}$ 的个数期望为 $2^{48} \times 2^{-32} = 2^{16}$,将它们与 $K_{0,\{0,5,10,15,2,7,8,13,4,6\}}$ 一起作为候选密钥进行加密验证。密钥方案筛选过程需要 $2^{112}P \times 2^{48} = 2^{160}P$ 次密钥方案的计算,最后对 $2^{112}P \times 2^{16} = 2^{128}P$ 个候选密钥进行加密验证。

2.1　改进的密钥方案筛选过程

在密钥方案筛选过程中,对于每一个可能的密钥值 $K_{0,\{0,5,10,15,2,7,8,13,4,6\}} \| K_{7,SR(col(3))}$,

(1)遍历 $K_{0,\{12,14\}}$ 的 2^{16} 个可能值,根据公式(1)计算 $s_{0,\{0,1,2,3\}}$,再根据公式(2)计算 $s_{7,\{12,13,14,15\}}$,验证 $K_{7,12} = s_{7,12}$ 是否成立。如果成立再计算 $s_{0,\{8,9,10,11\}}$ 和 $s_{7,\{4,5,6,7\}}$,验证 $K_{7,6} = s_{7,4} + s_{7,14}$ 是否成立,如果成立将 $K_{0,\{12,14\}}$ 的值保留。这步筛选后平均留下 $2^{16} \times 2^{-16} = 1$ 个 $K_{0,\{12,14\}}$ 的值。

(2)遍历 $K_{0,11}$ 和 $K_{0,1} + K_{0,9}$ 的 2^{16} 个可能值,由公式(1)计算 $s_{0,\{4,5,6,7\}}$,再由公式(2)计算 $s_{7,\{0,1,2,3\}}$,以 $s_{7,0}$ 的值为索引将 $K_{0,11}$ 和 $K_{0,1} + K_{0,9}$ 的值存储到表格 L 中。平均一个 $s_{7,0}$ 的值对应 2^8 个 $K_{0,11}$ 和 $K_{0,1} + K_{0,9}$ 的值。

(3)遍历 $K_{0,9}$ 和 $K_{0,3} + K_{0,11}$ 的 2^{16} 个可能值,由公式(1)计算 $s_{0,\{12,13,14,15\}}$,再由公式(2)计算 $s_{7,\{8,9,10,11\}}$,验证 $K_{7,9} = s_{7,8} + s_{7,15}$ 是否成立。如果成立,根据 $s_{7,0} = s_{7,13} + s_{7,10} + s_{7,7} + K_{7,3}$ 计算对应的 $s_{7,0}$。查询表格 L 得到对应的 $K_{0,11}$ 和 $K_{0,1} + K_{0,9}$,求解出 $K_{0,\{1,3,9,11\}}$。最终平均得到 $K_{0,\{1,3,9,11,12,14\}}$ 的 $2^{16} \times 2^{-8} \times 2^8 = 2^{16}$ 个可能值。

经过改进之后,密钥方案筛选过程的时间复杂度由 $2^{160}P$ 次密钥方案的计算减少到 $2^{112}P \times 2^{16} = 2^{128}P$ 次密钥方案的计算,留下的候选密钥个数不变,即 $2^{128}P$ 个。

2.2　改进的 7 轮 AES-128 不可能差分攻击方案

2.2.1　预处理阶段

在预处理阶段,得到预计算表格 L_1,$L_{2,i}$,$L_{2,j}$ 和 L_3,$i = 0,1,2,3$,$j = 8,9,10,11$。

(1) L_1:遍历 2^{64} 个所有可能的 $(X_{1,SR^{-1}(col(2))}, X_{2,SR^{-1}(col(2))})$,计算对应的差分 $\Delta X^*_{col(2)}$。如果 ΔX^* 在第 9 个字节和第 11 个字节为 0,则将 $(X_{1,SR^{-1}(col(2))}, X_{2,SR^{-1}(col(2))})$ 以 $\Delta X_{SR^{-1}(col(2))}$ 为索引存入表 L_1 中。表 L_1 共有 $2^{64} \times 2^{-16} = 2^{48}$ 个对,平均每个索引下有 $2^{48} \times 2^{-32} = 2^{16}$ 个对。

(2) $L_{2,i}$,$i = 0,1,2,3$:遍历 2^{32} 个所有可能的 $(Y_{1,\{0,10\}}, Y_{2,\{0,10\}})$,计算对应的差分 $\Delta Y^*_{col(0)}$。如果 ΔY^* 在第 i 个字节为 0,则将 $(Y_{1,10}, Y_{2,10})$ 以 24 比特级联值 $Y_{1,0} \| Y_{2,0} \| \Delta Y_{10}$ 为索引存入表 $L_{2,i}$ 中,$i = 0,1,2,3$。对每个 i,表格 $L_{2,i}$ 共有 $2^{32} \times 2^{-8} = 2^{24}$ 对 $(Y_{1,10}, Y_{2,10})$,平均每个索引下有 1 对。

(3) $L_{2,j}$,$j = 8,9,10,11$:遍历 2^{32} 个所有可能的 $(Y_{1,\{2,8\}}, Y_{2,\{2,8\}})$,计算对应的差分 $\Delta Y^*_{col(2)}$。如果 ΔY^* 在第 j 个字节为 0,则将 $(Y_{1,8}, Y_{2,8})$ 以 24 比特级联值 $Y_{1,2} \| Y_{2,2} \| \Delta Y_8$ 为索引存入表 $L_{2,j}$ 中,$j = 8,9,10,11$。对每个 j,表格 $L_{2,j}$ 共有 $2^{32} \times 2^{-8} = 2^{24}$ 对 $(Y_{1,8}, Y_{2,8})$,平均每个索引下有 1 对。

(4) L_3:将只在 1 个字节差分非零的所有 $2^{32} \times 4 \times (2^8 - 1) \approx 2^{42}$ 个 $(Z_{1,col(3)}, Z_{2,col(3)})$ 计算得到 $(Z^*_{1,SR(col(3))}, Z^*_{2,SR(col(3))})$,将 $(Z^*_{1,SR(col(3))}, Z^*_{2,SR(col(3))})$ 按照差分值 $\Delta Z^*_{SR(col(3))}$ 为索引存入表格 L_3 中,平均每个索引下有 $2^{42} \times 2^{-32} = 2^{10}$ 个对。

2.2.2　在线阶段

在线阶段攻击的流程如下:

(1)选取 2^n 个不同的明文结构,每个结构在第 0 条和第 2 条对角线遍历,其余 8 个字节取任意固定值,得到 2^{n+127} 个只在第 0 条和第 2 条对角线有差分的明文对。筛选出密文差分满足 $(C_1 + C_2)_{SR(col(0,1,2))} = 0$ 的 2^{n+31} 个明文对 (P_1, P_2) 存入表 T_1 中。

(2)猜测 $K_{0,SR^{-1}(col(0))}$,对表 T_1 中的每个明文对 (P_1, P_2),部分加密得到 $(X^*_{1,col(0)}, X^*_{2,col(0)})$。如果差分 $\Delta X^*_{\{1,3\}} = 0$,则继续执行步骤 3,否则对 T_1 中下一对明文操作。对于 $K_{0,SR^{-1}(col(0))}$ 的每个猜测值,平均有 $2^{n+31} \times 2^{-16} = 2^{n+15}$ 个明文对 (P_1, P_2) 继续执行步骤 3。

(3)对明文对 (P_1, P_2),根据 $\Delta P_{SR^{-1}(col(2))}$ 的值在表

格 L_1 中搜索得到对应的 2^{16} 个状态对($X_{1,SR^{-1}(col(2))}$, $X_{2,SR^{-1}(col(2))}$)。分别将这些状态对与($P_{1,SR^{-1}(col(2))}$, $P_{2,SR^{-1}(col(2))}$)异或得到 2^{16} 个 $K_{0,SR^{-1}(col(2))}$ 的值,并将其以明文对 (P_1,P_2) 为索引存入表 T_2 中。

(4)对表 T_2 中的每对(P_1,P_2)和每个 $K_{0,SR^{-1}(col(2))}$ 的值,根据轮密钥之间的关系由 $K_{0,SR^{-1}(col(0,2))}$ 计算 $K_{1,\{0,2\}}$。将(P_1,P_2)部分加密得到$(X^*_{1,\{8,10\}},X^*_{2,\{8,10\}})$和$(Y_{1,\{0,2\}},Y_{2,\{0,2\}})$。因为 $\Delta Y_{\{8,10\}}=\Delta X^*_{\{8,10\}}$,对于每个 $i=0,1,2,3$,根据 $Y_{1,0}\parallel Y_{2,0}\parallel \Delta Y_{10}$搜索表格 $L_{2,i}$得到1对$(Y_{1,10},Y_{2,10})$,将其与$(X^*_{1,10},X^*_{2,10})$异或得到1个 $K_{1,10}$的值。对于每个 $j=8,9,10,11$,根据 $Y_{1,2}\parallel Y_{2,2}\parallel \Delta Y_8$ 搜索 $L_{2,j}$得到1对$(Y_{1,8},Y_{2,8})$,将其与$(X^*_{1,8},X^*_{2,8})$异或得到1个 $K_{1,8}$的值。当 i, j 在一条对角线上时,将对应的 $K_{1,8}$与 $K_{1,10}$进行组合得到4个 $K_{1,\{8,10\}}$。由轮密钥之间的关系可计算对应的 $K_{0,\{4,6\}}$的值,并将其添加到表 T_2 里(P_1,P_2)和 $K_{0,SR^{-1}(col(2))}$ 对应的条目之中,记更新后的表为 T_3。

(5)对于 T_3 中的每个明文对(P_1,P_2),根据其密文差分 $(C_1+C_2)_{SR(col(3))}$ 搜索表格 L_3 得到 2^{10} 对$(Z^*_{1,SR(col(3))},Z^*_{2,SR(col(3))})$。分别将它们与$(C_{1,SR(col(3))},C_{2,SR(col(3))})$异或得到 2^{10} 个 $K_{7,SR(col(3))}$ 的值,将其添加入 T_3 中明文对(P_1,P_2)对应的条目之中,记更新后的表为 T_4。表 T_4 中 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的值与当前猜测的 $K_{0,SR^{-1}(col(0))}$一起为排除的错误密钥。

(6)将 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的所有可能值存入表格 T 中。将 T_4 中的密钥值从 T 中删去,再对 T 中留下的 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的值,与当前猜测的 $K_{0,SR^{-1}(col(0))}$一起,调用改进的密钥方案筛选过程算法得到候选密钥。

(7)对候选密钥进行加密验证。

附录中的算法1给出了改进的7轮 AES-128 不可能差分攻击。对于 $K_{0,SR^{-1}(col(0))}$ 的每个猜测值和表 T_2 中的每个明文对(P_1,P_2),平均在表 T_4 中对应 $2^{16}\times 4\times 2^{10}=2^{28}$ 个 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的不可能值。筛选后每个16字节密钥 $K_{0,\{0,5,10,15,2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的猜测值被留下的概率为 $P=(1-2^{28}\times 2^{-80})^{2^{n+15}}\approx e^{-2^{n-37}}$。在预处理阶段,构造表格 L_1 需要 $2^{64}\times 2\times 4=2^{67}$ 次S盒计算,构造 $L_{2,i}$和 $L_{2,j}$共需要 $2^{32}\times 4\times 2=2^{35}$ 次S盒计算,构造 L_3 需要 $2^{42}\times 2\times 4=2^{45}$ 次S盒计算。在线阶段,对于 $K_{0,SR^{-1}(col(0))}$ 的每个猜测值,步骤2中筛选明文对(P_1,P_2)需要 $2^{n+31}\times 2\times 4=2^{n+34}$ 次S盒计算,步骤3中构造表格 T_2 需要 $2^{n+15}\times 2^{16}=2^{n+31}$ 次内存访问(MA),相当于 2^{n+31} 次S盒计算,步骤4中构造表格 T_3 需要 $2^{n+15}\times 2^{16}\times(2+8+8)\approx 2^{n+35.17}$ 次S盒计算,步骤5中构造表格 T_4 需要 $2^{n+15}\times 2^{10}=2^{n+25}$ 次内存访问。对于每个 $K_{0,SR^{-1}(col(0))}$ 的猜测值,表格 T_4 中一共有 $2^{n+15}\times 2^{28}=2^{n+43}$ 个 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的不可能值,所以步骤6的删除操作共需要 $2^{32}\times 2^{n+43}=2^{n+75}$ 次内存访问。改进的密钥方案筛选过程的复杂度为 $2^{128}P$ 次密钥方案计算,步骤7一共需要 $2^{128}P$ 次加密。算法最终的时间复杂度为 $2^{n+75}/(20\times 7)+2^{128}P\approx 2^{n+67.87}+2^{128-1.44\times 2^{n-37}}$,当 $n=41$ 时算法时间复杂度最低,为 $2^{108.96}$。算法的数据复杂度为 $2^{n+64}=2^{105}$。筛选满足密文差分的明文对所用的 Hash 表的规模为 2^{96},对于 $K_{0,SR^{-1}(col(0))}$ 的每个猜测值,表格 T_4 的大小为 2^{n+43},表格 T 为 2^{80},所以算法的存储复杂度主要取决于 Hash 表的大小 2^{96}。

3 结束语

本文将 Leurent 等提出的 AES 密钥方案的新表示技术应用于 Mala 等的7轮 AES-128 不可能差分攻击,将攻击的时间复杂度从 $2^{110.1}$ 改进至 $2^{108.96}$,数据复杂度从 $2^{106.2}$ 改进至 2^{105}。有一些问题需要进一步研究,例如:①Leurent 等的密钥方案新表示技术将 AES-128 的密钥扩展方案等价转换成4个块的独立运算,是否存在其他新表示可以将密钥扩展方案进一步转化为更多块的独立运算;②本文对于攻击的改进幅度不大,如何更好地处理数据筛选与密钥猜测的折中,以及结合其他技术进一步改进攻击的复杂度。

参考文献:

[1] Daemen J, Rijmen V. The design of rijndael: AES—The advanced encryption standard[M]. Berlin: Springer, 2002.

[2] Canteaut A, Duval S, Leurent G, et al. Saturnin: A suite of lightweight symmetric algorithms for post-quantum security [J]. IACR Transactions on Symmetric Cryptology, 2020, S1: 160-207.

[3] Borghoff J, Canteaut A, Güneysu T, et al. PRINCE-A low-latency block cipher for pervasive computing applications[C]// ASIACRYPT 2012. Berlin: Springer, 2012: 208-225.

[4] Guo J, Peyrin T, Poschmann A, et al. The LED block cipher[C]//CHES 2011. Berlin: Springer, 2011: 26-341.
[5] Hoang V T, Krovetz T, Rogaway P. Robust authenticated-encryption AEZ and the problem that it solves[C]//EUROCRYPT 2015. Berlin: Springer, 2015: 15-44.
[6] Ferguson N, Kelsey J, Lucks S, et al. Improved cryptanalysis of Rijndael[C]//Fast Software Encryption 2015. Berlin: Springer, 2001: 213-230.
[7] Biham E, Keller N. Cryptanalysis of reduced variants of Rijndael[C]//The third AES Conference. 2000:8-9.
[8] Cheon J H, Kim M, Kim K, et al. Improved impossible differential cryptanalysis of Rijndael and Crypton[C]//ICISC 2001. Berlin: Springer, 2002: 39-49.
[9] Zhang W T, Wu W L, Feng D G. New results on impossible differential cryptanalysis of reduced AES[C]//ICISC 2007. Berlin: Springer, 2007: 239-250.
[10] Bahrak B, Aref M R. Impossible differential attack on seven-round AES-128[J]. IET Information Security, 2008, 2(2): 28-32.
[11] Lyu J Q, Dunkelman O, Keller N, et al. New impossible differential attacks on AES[C]//INDOCRYPT 2008. Berlin: Springer, 2008: 279-293.
[12] Mala H, Dakhilalian M, Rijmen V, et al. Improved impossible differential cryptanalysis of 7-round AES-128[C]//INDOCRYPT 2010. Berlin: Springer, 2010: 282-291.
[13] Boura C, Lallemand V, Naya-Plasencia M, et al. Making the impossible possible[J]. Journal of Cryptology, 2018, 31: 101-133.
[14] Leurent G, Pernot C. New representations of the AES key schedule[C]//EUROCRYPT 2021. Berlin: Springer, 2021: 54-84.
[15] Hu K, Cui T T, Gao C, et al. Towards key-dependent integral and impossible differential distinguishers on 5-round AES [C]//SAC 2018. Cham: Springer, 2018: 139-162.
[16] Grassi L, Rechberger C, Rønjom S. Subspace trail cryptanalysis and its applications to AES[J]. IACR Transactions on Symmetric Cryptology, 2016, 2: 192-225.
[17] Bar-On A, Dunkelman O, Keller N, et al. Improved key recovery attacks on reduced-round AES with practical data and memory complexities[J]. Journal of Cryptology, 2020, 33: 1003-1043.
[18] 谭林, 闫雪萍, 戚文峰. AES 和 PRINCE 的 6 轮混合差分攻击[J]. 密码学报, 2022, 9(4): 755-766.
[19] Rønjom S, Bardeh N G, Helleseth T. Yoyo tricks with AES[C]//ASIACRYPT 2017. Cham: Springer 2017: 217-243.
[20] Bardeh N G, Rønjom S. The exchange attack: How to distinguish 6 rounds of AES with $2^{88.2}$ chosen plaintexts[C]//ASIACRYPT 2019. Cham: Springer 2019: 347-370.
[21] Dunkelman O, Keller N, Ronen E, et al. The retracing boomerang attack[C]//EUROCRYPT 2020. Cham: Springer 2020: 280-309.
[22] Rahman M, Saha D, Paul G. Boomeyong: Embedding yoyo within boomerang and its applications to key recovery attacks on AES and Pholkos[J]. IACR Transactions on Symmetric Cryptology, 2021, 3: 137-169.
[23] Derbez P, Fouque P, Jérémy J. Improved key recovery attacks on reduced-round AES in the single-key setting[C]//EUROCRYPT 2013. Berlin: Springer, 2013: 371-387.
[24] Biham E, Biryukov A, Shamir A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials[C]//EUROCRYPT 1999. Berlin: Springer, 1999: 12-23.

附录:

算法 1　改进的 7 轮 AES-128 不可能差分攻击

预处理:构造表格 L_1, $L_{2,i}$, $L_{2,j}$和 L_3, $i=0,1,2,3$, $j=8,9,10,11$。

输入:2^{41}个明文结构,每个明文结构在第 0 条和第 2 条对角线遍历,其余 8 个字节取任意固定值。

输出:主密钥 K_0。

对每个明文结构,以密文的第 0、1、2 条反对角线的值为索引构建 Hash 表,将满足密文差分$(C_1+C_2)_{SR(col(0,1,2))}=0$ 的明文对(P_1,P_2)存入表格 T_1 中。

for 每个 $K_{0,SR^{-1}(col(0))}$ do
  for 每个 $(P_1,P_2)\in T_1$ do
    计算 $(X^*_{1,col(0)},X^*_{2,col(0)})$。
    if $\Delta X^*_{\{1,3\}}=0$ then
      根据 $\Delta P_{SR^{-1}(col(2))}$ 在表 L_1 中寻找对应的 2^{16} 个 $(X_{1,SR^{-1}(col(2))},X_{2,SR^{-1}(col(2))})$。分别将它们与 $(P_{1,SR^{-1}(col(2))},P_{2,SR^{-1}(col(2))})$ 异或得到 2^{16} 个 $K_{0,SR^{-1}(col(2))}$ 的值,以明文对 (P_1,P_2) 为索引存入表 T_2 中。
    end if
  end for
  for 每个 $(P_1,P_2)\in T_2$ 和 $K_{0,SR^{-1}(col(2))}$ do
    计算 $(X^*_{1,\{0,2,8,10\}},X^*_{2,\{0,2,8,10\}})$。
    计算 $K_{1,\{0,2\}}$,得到 $(Y_{1,\{0,2\}},Y_{2,\{0,2\}})$。计算 $\Delta Y_{\{8,10\}}$。
    根据 $Y_{1,0}\parallel Y_{2,0}\parallel \Delta Y_{10}$ 搜索 $L_{2,i}$ 得到 $(Y_{1,10},Y_{2,10})$,与 $(X^*_{1,10},X^*_{2,10})$ 异或得到 $K_{1,10}$ 的值,$i=0,1,2,3$。
    根据 $Y_{1,2}\parallel Y_{2,2}\parallel \Delta Y_8$ 搜索 $L_{2,j}$ 得到 $(Y_{1,8},Y_{2,8})$,与 $(X^*_{1,8},X^*_{2,8})$ 异或得到 $K_{1,8}$ 的值,$j=8,9,10,11$。
    当 i, j 属于一条对角线时,将对应的 $K_{1,8}$ 与 $K_{1,10}$ 组合得到 4 个 $K_{1,\{8,10\}}$,计算对应的 $K_{0,\{4,6\}}$,将其添加到表 T_2 中 (P_1,P_2) 和 $K_{0,SR^{-1}(col(2))}$ 对应的条目之中得到表 T_3。
  end for
  for 每个 $(P_1,P_2)\in T_3$ do
    根据 $(C_1+C_2)_{SR(col(3))}$ 搜索 L_3 得到 2^{10} 对 $(Z^*_{1,SR(col(3))},Z^*_{2,SR(col(3))})$,分别将它们与 $(C_{1,SR(col(3))},C_{2,SR(col(3))})$ 异或得到 2^{10} 个 $K_{7,SR(col(3))}$ 的值。将 2^{10} 个值添加到 T_3 中明文对 (P_1,P_2) 对应的条目中得到表 T_4。
  end for
  将 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的所有可能值存入表格 T 中。
  将 T_4 中每个 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ 的值从表格 T 中删去。
  for T 剩余的 $K_{0,\{2,7,8,13,4,6\}}\parallel K_{7,SR(col(3))}$ do
    for 每个 $K_{0,\{12,14\}}$ do
      计算 $s_{0,\{0,1,2,3\}}$ 和 $s_{7,\{12,13,14,15\}}$。如果 $K_{7,12}=s_{7,12}$,计算 $s_{0,\{8,9,10,11\}}$ 和 $s_{7,\{4,5,6,7\}}$。如果 $K_{7,6}=s_{7,4}+s_{7,14}$,将 $K_{0,\{12,14\}}$ 保留。
    end for
    for 每个 $K_{0,11}$ 和 $K_{0,1}+K_{0,9}$ do
      计算 $s_{0,\{4,5,6,7\}}$ 和 $s_{7,\{0,1,2,3\}}$,以 $s_{7,0}$ 的值为索引将 $K_{0,11}\parallel K_{0,1}+K_{0,9}$ 的值存入表 L。
    end for
    for 每个 $K_{0,9}$ 和 $K_{0,3}+K_{0,11}$ do
      计算 $s_{0,\{12,13,14,15\}}$ 和 $s_{7,\{8,9,10,11\}}$。如果 $K_{7,9}=s_{7,8}+s_{7,15}$,计算 $s_{7,0}=s_{7,13}+s_{7,10}+s_{7,7}+K_{7,3}$,对表格 L 索引得到其对应的 $K_{0,11}\parallel K_{0,1}+K_{0,9}$,计算得到 $K_{0,\{1,3,9,11\}}$。将对应的 K_0 进行加密验证。
      if K_0 通过验证 then
        return K_0。
      end if
    end for
  end for
end for

【责任编辑:孙向荣】