软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(8):2337−2348 [doi: 10.13328/j.cnki.jos.005759]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

Midori-64 算法的截断不可能差分分析*

李明明, 郭建胜, 崔竞一, 徐林宏

(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: 分析了 Midori-64 算法在截断不可能差分攻击下的安全性.首先,通过分析 Midori 算法加、解密过程差分路径规律,证明了 Midori 算法在单密钥条件下的截断不可能差分区分器至多 6 轮,并对 6 轮截断不可能差分区分器进行了分类;其次,根据分类结果,构造了一个 6 轮区分器,并给出 11 轮 Midori-64 算法的不可能差分分析,恢复了 128 比特主密钥,其时间复杂度为 $2^{121.4}$,数据复杂度为 $2^{60.8}$,存储复杂度为 $2^{96.5}$.

关键词: 密码分析;轻量级分组密码;Midori;不可能差分分析;截断不可能差分区分器

中图法分类号: TP309

中文引用格式: 李明明,郭建胜,崔竞一,徐林宏.Midori-64 算法的截断不可能差分分析.软件学报,2019,30(8):2337−2348. http://www.jos.org.cn/1000-9825/5759.htm

英文引用格式: Li MM, Guo JS, Cui JY, Xu LH. Truncated impossible differential cryptanalysis of Midori-64. Ruan Jian Xue Bao/Journal of Software, 2019,30(8):2337−2348 (in Chinese). http://www.jos.org.cn/1000-9825/5759.htm

Truncated Impossible Differential Cryptanalysis of Midori-64

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

(Information Engineering University, Zhengzhou 450001, China)

Abstract: The security of Midori-64 under truncated impossible differential cryptanalysis was studied. Firstly, by analyzing the differential paths of Midori in encryption and decryption direction, it was proved that the upper bound for the rounds of impossible differential distinguishers in single-key setting for Midori is 6. And the 6-round truncated impossible differential distinguisher was classified. Secondly, based on the classification, a 6-round distinguisher was constructed. At the same time the impossible differential attack on 11-round Midori-64 was given to recover the 128-bit master key with a time complexity of $2^{121.4}$ 11-round encryptions, a data complexity of $2^{60.8}$ chosen-plaintexts, and a memory complexity of $2^{96.5}$ 64-bit blocks.

Key words: cryptanalysis; lightweight block cipher; Midori; impossible differential cryptanalysis; truncated impossible differential distinguisher

不可能差分分析,由 Knudsen[1]和 Biham[2]两人分别独立提出,是目前最常用的密码分析方法之一.Kim 等人[3]提出了μ方法.该方法通过计算机搜索找出各种算法结构中所固有的不可能差分形式,这些固有的不可能差分只与算法的结构有关,而与算法所采用的 S 盒无关,这也就是所谓的截断不可能差分.后来,Luo 等人[4]在μ方法的基础上改进,提出了 UID 方法.不同于μ方法,UID 方法利用中间相错的思想将密码算法结构分解成两部分进行处理.此外,孙兵等人[5]基于整环上的矩阵论证明了 SPN 结构及嵌套 SPN 的 Feistel 结构的截断不可能差分区分器的上界取决于线性层的本原指数,并以 AES 算法、ARIA 算法和 Camellia 算法为实例,分别证明了 AES 算

* 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: 2018-04-24; 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 09:16:21, http://kns.cnki.net/kcms/detail/11.2560.TP.20190329.0915.010.html

法和 ARIA 算法的截断不可能差分区分器都至多 4 轮、Camellia 算法(不考虑 FL/FL^{-1} 层)不可能差分区分器至多 8 轮.

随着物联网的不断发展,物联网的安全问题日益严峻.由于物联网上的加密器件大多存储空间小、计算能力弱,因此,传统的密码算法不适合用来保护物联网上的信息安全.在这种情况下,大量轻量级密码算法应运而生,典型的算法有 PRESENT[6],MIBS[7],GOST[8],KLEIN[9],LED[10],Piccolo[11],LBlock[12],PRINCE[13]等.

这些轻量级密码算法大部分关注的是硬件实现电路的面积,但 Banik 等人[14]认为,制约轻量级算法应用的因素是算法的功耗,低功耗的密码算法能够更为广泛地应用于多种设备中,并在 Asiacrypt 2015 会议上提出专注于低功耗的 Midori 算法.由于 Midori 算法受限于轻量级使用环境,故通常不对其在已知密钥、相关密钥、选择密钥条件下的安全性进行分析,更多考察算法在单密钥条件下的安全性.

目前,对于 Midori 算法在单密钥条件下有多个安全性分析结果.2017 年,Lin 等人[15]利用中间相遇攻击方法,通过构造 5 轮与 6 轮区分器,分别给出了 10 轮~12 轮 Midori-64 算法的安全性分析结果.2015 年,Guo 等人[16]对 Midori-64 算法的弱密钥集进行了分析,利用 Subspace 攻击给出了全轮 Midori-64 算法的分析结果;2017 年,Chen 等人[17]构造了第一个 6 轮截断不可能差分区分器,并给出首个不含入口白化密钥的 10 轮 Midori-64 算法不可能差分分析.

Midori 算法的设计者曾估计:在单密钥条件下,该算法不可能差分区分器的最大轮数是 7 轮,其 7 轮或 7 轮以上截断不可能差分区分器是否存在,是尚未解决的问题.若存在,可以利用该不可能差分区分器分析更高轮数 Midori 算法;若不存在,如何将 Midori 算法 6 轮截断不可能差分区分器都找出来并进行分类,及是否存在更优的区分器使得对 Midori-64 算法分析结果更好.这些问题就是本文研究的重点.

本文首先证明了在单密钥条件下,Midori 算法截断不可能差分区分器的轮数至多 6 轮,并根据该证明过程,对 6 轮截断不可能差分区分器进行了分类;其次,根据 Midori 算法 6 轮截断不可能差分区分器的分类,构造了一个 6 轮区分器,并给出了 11 轮 Midori-64 算法的不可能差分分析,恢复了 128 比特主密钥.本文得到的结果与在单密钥条件下现有分析结果对比见表 1.

Table 1 Comparison of attacks on Midori-64

表 1 Midori-64 算法分析结果对比

攻击方法	轮数	恢复密钥规模	时间复杂度	数据复杂度	存储复杂度	出处
MITM	10	128	$2^{99.5}$	$2^{61.5}$	$2^{92.7}$	文献[15]
ID	10	72	$2^{80.81}$	$2^{62.4}$	$2^{65.13}$	文献[17]
MITM	11	128	2^{122}	2^{53}	$2^{89.2}$	文献[15]
ID	11	128	$2^{121.4}$	$2^{60.8}$	$2^{96.5}$	本文
MITM	12	128	$2^{125.5}$	$2^{55.5}$	2^{106}	文献[15]
Subspace(2^{32} 个弱密钥)	16	128	2^{16}	2^{1}	–	文献[16]

本文第 1 节主要介绍 Midori 算法及其相关性质.第 2 节证明 Midori 算法截断不可能差分区分器至多 6 轮.第 3 节对 Midori 算法的 6 轮截断不可能差分区分器进行分类.第 4 节给出 11 轮 Midori-64 算法的不可能差分分析.第 5 节总结全文.

1 Midori 算法介绍

1.1 符号说明

- P:明文.
- ΔC:输出密文差分.
- x_i:第 i 轮经过密钥异或后的状态.
- y_i:第 i 轮经过非线性变换后的状态.
- z_i:第 i 轮经过移位变换后的状态.
- w_i:第 i 轮经过列混合变换后的状态.

- (z_i', z_i''):表示第 i 轮经过移位变换后输出的数据对.
- k_{ei}:第 i 轮加密密钥.
- $k_{ei}[j]$:第 i 轮加密密钥第 j 个字节.
- k_{ei}^*:第 i 轮加密密钥经过 $SC^{-1}\cdot MC^{-1}$ 变换后状态,即 $k_{ei}=MC\cdot SC(k_{ei}^*)$.

1.2 Midori算法

Midori 算法分为 Midori-64 与 Midori-128 两种,密钥规模都是 128 比特,分组规模分别为 64 比特与 128 比特,对应的迭代轮数为 16 轮与 20 轮.分组状态表示为 16 个比特块的形式,分组规模为 64 比特时,每个比特块为半字节;分组规模为 128 比特时,每个比特块为字节,如图 1 所示.

$$\begin{pmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{pmatrix}$$

Fig.1 State of Midori

图 1 Midori 分组状态

Midori 算法采用 SPN 结构,其轮函数依次由非线性变换 SubCell、移位变换 ShuffleCell、列混合变换 MixColumn、密钥异或 KeyAdd 这 4 个变换组成.这 4 个变换及密钥扩展算法的具体介绍如下.

- 非线性变换 SubCell:在 Midori 算法中共有 2 个不同的对合 4 比特 S 盒 Sb_0,Sb_1,即 $Sb_0,Sb_1:\{0,1\}^4\to\{0,1\}^4$.在 Midori-64 中使用一个 S 盒 Sb_0.在 Midori-128 中使用由两个 Sb_1 与 4 个不同的置换生成 4 个 8 比特的 S 盒 Sb_0,Sb_1,Sb_2,Sb_3.Sb_0,Sb_1 以 16 进制表示形式给出,见表 2.

Table 2 S-boxes Sb_0, Sb_1 of Midori

表 2 Midori 中使用的 S 盒 Sb_0,Sb_1

x	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
$Sb_0[x]$	c	a	d	3	e	b	f	7	8	9	1	5	0	2	4	6
$Sb_1[x]$	1	0	5	3	e	2	f	7	d	a	9	b	c	8	4	6

- 移位变换:将 16 个比特块的顺序进行重新排列,其变换及逆变换具体如下:

$$\begin{pmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{pmatrix} \overset{SC}{\Longrightarrow} \begin{pmatrix} s_0 & s_{14} & s_9 & s_7 \\ s_{10} & s_4 & s_3 & s_{13} \\ s_5 & s_{11} & s_{12} & s_2 \\ s_{15} & s_1 & s_6 & s_8 \end{pmatrix}, \begin{pmatrix} s_0 & s_5 & s_{15} & s_{10} \\ s_7 & s_2 & s_8 & s_{13} \\ s_{14} & s_{11} & s_1 & s_4 \\ s_9 & s_{12} & s_6 & s_3 \end{pmatrix} \overset{SC^{-1}}{\Longleftarrow} \begin{pmatrix} s_0 & s_4 & s_8 & s_{12} \\ s_1 & s_5 & s_9 & s_{13} \\ s_2 & s_6 & s_{10} & s_{14} \\ s_3 & s_7 & s_{11} & s_{15} \end{pmatrix}.$$

- 列混合变换:使用对合二元阵 M 对状态中每一列进行如下操作,即 $M\cdot(s_i,s_{i+1},s_{i+2},s_{i+3})'\to(s_i,s_{i+1},s_{i+2},s_{i+3})'$,其中,$i$=0,4,8,12,$M$ 及其逆矩阵 M^{-1} 如下:

$$M=\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}, M^{-1}=\begin{pmatrix} 0 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 1 \\ 1 & 1 & 1 & 0 \end{pmatrix}.$$

- 密钥异或:将状态字节与密钥字节对应位置进行模 2 加运算.

密钥扩展算法:Midori 算法使用 128 比特主密钥,其密钥扩展算法较为简单.其中,Midori-64 算法将 128 比特主密钥分为两部分 $K=k_0\|k_1$,算法的入口与出口白化密钥均为 $WK=k_0\oplus k_1$,圈子密钥 $k_{ei}=k_{(i+1)\bmod 2}\oplus\alpha_i$,$1\leqslant i\leqslant 15$,其中,$\alpha_i$ 是轮常数.而 Midori-128 算法的入口与出口白化密钥均为 $WK=K$,圈子密钥 $k_{ei}=K\oplus\beta_i$,$1\leqslant i\leqslant 19$,其中,β_i 为轮常数.且当 $1\leqslant i\leqslant 15$ 时,满足 $\alpha_i=\beta_i$.

Midori 算法的加密需要在算法的初始位置异或一个白化密钥 WK,为确保 Midori 算法加解密算法结构相似性,算法在最后一轮省略移位变换与列混合变换.16 轮 Midori-64 算法的加密流程如图 2 所示.

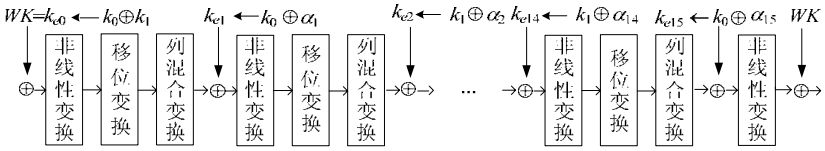

Fig.2 Encryption process of Midori-64

图 2 Midori-64 算法加密流程

1.3 Midori算法相关性质

性质 1.1. 在 Midori-64 算法中,若已知白化密钥,当再得到任意一个奇数轮圈子密钥或者任意一个偶数轮圈子密钥时,即可恢复主密钥;在 Midori-128 算法中,若已知任意一个圈子密钥,则可以恢复主密钥.

性质 1.2[15]. 给定 Midori 算法 S 盒的一对对应的输入、输出差分,平均可以确定一对 S 盒的输入与输出.

性质 1.3[18]. 对于 Midori 算法列混合变换,当输入状态的某列仅有 1 个比特块差分活动时,输出状态必在相同列的其余 3 个比特块有差分活动,且差分值相等;当输入状态的某列有两个比特块差分活动且两个差分值相同,则输出状态必定与输入状态相同;当输入状态的某列有 3 个比特块差分活动且差分值相同,则输出状态仅在相同列的另一个比特块上差分活动.

性质 1.4. 原在同一列的半字节,经两次 Midori 算法移位变换后,必定在同一行:

$$\begin{pmatrix} 0 & 4 & 8 & 12 \\ 1 & 5 & 9 & 13 \\ 2 & 6 & 10 & 14 \\ 3 & 7 & 11 & 15 \end{pmatrix} \xrightarrow{SC} \begin{pmatrix} 0 & 14 & 9 & 7 \\ 10 & 4 & 3 & 13 \\ 5 & 11 & 12 & 2 \\ 15 & 1 & 6 & 8 \end{pmatrix} \xrightarrow{SC} \begin{pmatrix} 0 & 2 & 3 & 1 \\ 12 & 14 & 15 & 13 \\ 4 & 6 & 7 & 5 \\ 8 & 10 & 11 & 9 \end{pmatrix}.$$

性质 1.5. 设 y_i 仅在某一列有 3 个比特块差分活动(差分值可能不同),则 y_i 经一轮 Midori 算法加密,再经一次移位变换后输出状态 z_{i+1} 必定其中 3 列有两个比特块差分活动,一列有 3 个比特块差分活动.

证明:y_i 经移位变换输出状态 z_i 在不同的 3 列各有一个比特块差分活动.再经列混合变化输出状态 w_i.其中,3 列有 3 个比特块差分活动,其余一列没有差分活动比特块.由于经密钥异或变换及非线性变换不改变原状态截断差分活动位置,故 y_{i+1} 的差分状态与 w_i 一致.对于 y_{i+1} 经移位变换的输出状态 z_{i+1},不妨先考虑 $y_{i+1}+z_i$ 经移位变换后的输出状态.因为 $y_{i+1}+z_i$ 满足其中 3 列每比特块都有差分活动,其余一列没有差分活动比特块,故经 SC 变化输出状态必定每一列都有 3 个比特块差分活动.又因为由性质 1.4 可知,z_i 再经一次移位变换后输出状态有差分活动的比特块必定在同一行,所以 z_{i+1} 必定其中 3 列有两个比特块差分活动,一列有 3 个比特块差分活动. □

性质 1.6. 设 y_i 仅在某一列有 2 个比特块差分活动(差分值可能不同),其余列没有差分活动比特块,则 y_i 经一轮 Midori 算法加密,再经一次移位变换后输出状态 z_{i+1} 必定其中两列各有两个比特块差分活动,其余两列各有一个比特块差分活动.

证明过程与性质 1.5 的证明类似,故略.

2 Midori 算法截断不可能差分区分器最大轮数的分析

2016 年,Chen 等人在单密钥条件下构造了第一个 6 轮截断不可能差分区分器.对于 7 轮及 7 轮以上截断不可能差分区分器是否存在,是本文研究的重点内容之一.

对于 Midori 算法截断不可能差分区分器至多 6 轮的证明的基本思想:证明任意(非零)初始差分状态经过 3.5 轮 Midori 算法加密或者 3.5 轮 Midori 解密算法后的输出状态的每个比特块都可能差分活动,这样就不可能构造出 7 轮或者 7 轮以上的截断不可能差分区分器.当然,7 轮截断不可能差分区分器不仅仅只能由向下加密 3.5 轮、向上解密 3.5 轮的差分路径组成,也可以由加密(解密)4.5 轮和解密(加密)2.5 轮的差分路径构成,但由于若某个状态其每个比特块都可能存在差分活动,该状态再经 1 轮加密或解密 Midori-64 算法,输出状态必定不存在确定差分活动或差分不活动的比特块,这样就不能构成一条截断不可能差分区分器.所以,只需证明任意初始差分状态经过 3.5 轮 Midori 算法加密或者 3.5 轮 Midori 解密算法后的输出状态的每个比特块都可能差分活动

即可.

为便于区分器的分类,不妨以状态 z_1 为初始输入状态,按加密过程和解密过程,证明分为第 2.1 节与第 2.2 节两部分.

2.1 Midori算法加密过程差分路径

引理 2.1. 初始输入状态 z_1 只在某一列存在差分活动的比特块时,经过 3.5 轮 Midori 算法加密后,输出状态 w_4 的每个比特块都可能差分活动.

为证明该引理,分别讨论初始输入状态 z_1 只在某一列存在 1 个差分活动、2 个差分活动、3 个差分活动、4 个差分活动比特块这 4 种情况.本文中如不加特殊说明,我们考虑的初始状态 z_1 其差分活动比特块的差分值都是相同的,这是因为差分值不同先于差分值相同的情况输出每个比特块都可能差分活动的状态.由于 *KeyAdd* 不改变原状态截断差分活动位置,为表述简洁,性质 2.1~性质 2.4 中都省去了最后 1/4 轮.

性质 2.1. 初始输入状态 z_1 只存在 1 个差分活动比特块时,经过 2.5 轮 Midori 算法加密后,输出状态 w_3 的每个比特块都可能差分活动.

证明:设 z_1 唯一的差分活动比特块在第 i 列,其中,$0\leqslant i\leqslant 3$.z_1 经 3/4 轮 Midori 算法加密后,非线性变换 *SB* 输出状态 y_2 必定仅在第 i 列有 3 个比特块差分活动.以 z_1 仅在第 6 个比特块有差分活动为例,给出其 2.5 轮 Midori 算法加密过程,如图 3 所示,其中,白色表示差分为 0 的半字节,黑色表示差分活动的半字节,斜线表示可能存在差分的半字节.

由性质 1.5 可知,y_2 经一轮 Midori 算法加密后,再经 *SC*,输出状态 z_3 必定其中 3 列有两个比特块差分活动,一列有 3 个比特块差分活动.z_3 再经 *MC*,输出状态 w_3 的每个比特块都可能差分活动.证毕. □

性质 2.2. 初始输入状态 z_1 只在某一列存在 2 个比特块差分活动时,经过 3.5 轮 Midori 算法加密后,输出状态 w_4 的每个比特块都可能差分活动.

证明:z_1 经 3/4 轮 Midori 算法加密后,输出状态 y_2 必定仅在某一列有 2 个比特块差分活动,其余列没有比特块差分活动.以 z_1 仅在第 5、第 6 个比特块有差分活动为例,给出其 3.5 轮 Midori 算法加密过程,如图 4 所示.

由性质 1.6 可知,y_2 经一轮 Midori 算法加密后,再经一次 *SC*,输出状态 z_3 必定其中两列各有两个比特块差分活动,其余两列各有一个比特块差分活动.故易知:再经 1.5 轮,输出状态 w_4 不存在确定有无差分活动的比特块.证毕. □

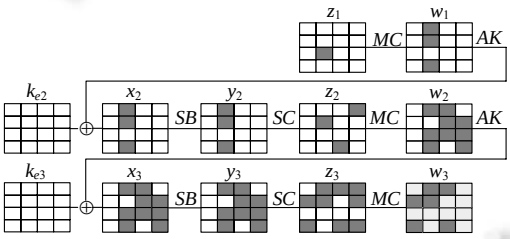

Fig.3 2.5-round differential path of Midori in encryption direction I

图 3 2.5 轮 Midori 算法加密方向差分路径 I

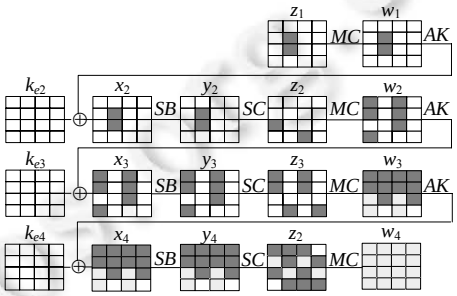

Fig.4 3.5-round differential path of Midori in encryption direction I

图 4 3.5 轮 Midori 算法加密方向差分路径 I

性质 2.3. 初始输入状态 z_1 只在某一列存在 3 个比特块差分活动时,经过 3.5 轮 Midori 算法加密后,输出状态 w_4 的每个比特块都可能差分活动.

证明:z_1 经一轮 Midori 算法加密后,输出状态 z_2 必定只有 1 个比特块差分活动,由性质 2.2 可知,z_2 再经 2.5 轮 Midori 算法加密,输出状态 w_4 的每个比特块都可能存在差分活动.以 z_1 仅在第 5~第 7 个比特块差分活动为例,给出其 3.5 轮 Midori 算法加密过程,如图 5 所示. □

性质 2.4. 初始输入状态 z_1 只在某一列存在 4 个比特块差分活动,经过 3.5 轮 Midori 算法加密后,输出状态 w_3 的每个比特块都可能差分活动.

证明:z_1 经一轮 Midori 算法加密后,输出状态 z_2 必定每一列各有一个比特块差分活动.易知:再经 1.5 轮算法加密,输出状态 w_3 不存在确定有无差分活动的比特块.以 z_1 仅在第 4~第 7 个比特块有差分活动为例,给出其 3.5 轮 Midori 算法加密过程,如图 6 所示. □

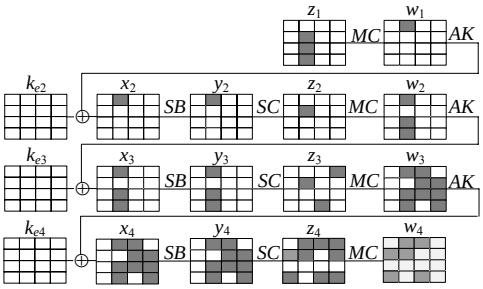

Fig.5 3.5-round differential path of Midori in encryption direction II

图 5 3.5 轮 Midori 算法加密方向差分路径 II

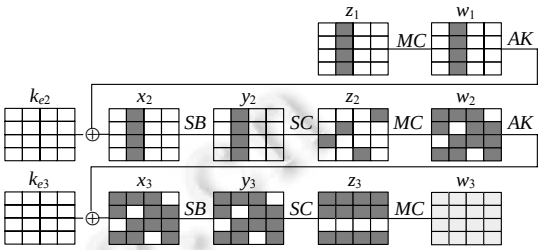

Fig.6 2.5-round differential path of Midori in encryption direction II

图 6 2.5 轮 Midori 算法加密方向差分路径 II

引理 2.2. 假设初始输入状态 z_1 只在某一列存在比特块差分活动时,若经过 n+1/2 轮 Midori 算法后,输出状态 w_{n+1} 的每个比特块都可能差分活动,则任意增加 z_1 中其他差分为零列的差分活动得到状态 z_1',z_1' 经过 n+1/2 轮 Midori 算法后的输出状态 w_{n+1}' 也必定每个比特块都可能差分活动.

证明:由 *SC*,*SB*,*KeyAdd*,*MC* 的性质可知,任意状态,其列与列之间的这 4 种变换是相互独立的.故 z_1 和 z_1' 经相同的变换, z_1' 经变换后,输出状态有差分活动的比特块集合必定包含 z_1 的输出状态有差分活动的比特块集合,所以该结论是显然的. □

由引理 2.1 和引理 2.2 可得出如下定理.

定理 2.1. 任一初始输入状态 z_1 经过 3.5 轮 Midori 算法加密后,输出状态 w_4 的每个比特块都可能差分活动.

2.2 Midori算法解密过程差分路径

引理 2.3. 状态 z_m 只存在 1 个比特块差分活动时,经过 3.5 轮 Midori 解密算法后,输出状态 x_{m-3} 的每个比特块都可能差分活动.

证明:z_m 经一轮 Midori 算法解密后,输出状态 z_{m-1} 必定其中一列有 3 个比特块差分活动,其余列无差分活动比特块.依据性质 1.5,同理可证:再经过 1.5 轮,输出状态 x_{m-2} 必定其中 3 列有两个比特块差分活动,一列有 3 个比特块差分活动.故 x_{m-2} 再经一轮解密算法,输出状态 x_{m-3} 每个比特块都可能差分活动.以 z_m 仅在第 5 个比特块有差分活动为例,给出其 3.5 轮 Midori 算法解密方向差分路径,如图 7 所示. □

引理 2.4. 假设状态 z_m 只存在 1 个比特块差分活动时,若经过 n+1/2 轮 Midori 解密算法后,输出状态 x_{m-n} 的每个比特块都可能差分活动,则任意增加 z_m 上其他比特块的差分得到的状态 z_m',经过 n+1/2 轮 Midori 解密算法后,输出状态 x_{m-n}' 每个比特块也必定都可能差分活动.

证明:不妨将状态上可能有差分活动的所有比特块用集合 H 表示,例如 z_m 上可能有差分活动的所有个比特块集合用 H_{z_m} 表示,状态 z_m' 上可能有差分活动的所有个比特块用集合 $H_{z_m'}$ 表示,显然有 $H_{z_m} \subset H_{z_m'}$. 由 *SC*,*SB*,*KeyAdd*,*MC* 的性质可知, $H_{y_m} \subset H_{y_m'}$, $H_{x_m} \subset H_{x_m'}$, $H_{w_{m-1}} \subset H_{w_{m-1}'}$, $H_{z_{m-1}} \subset H_{z_{m-1}'}$.故 z_m 和 z_m' 经相同的变换, z_m' 经变换后输出状态有差分活动的比特块集合,必定包含 z_m 经变换后的输出状态有差分活动的比特块集合,所以该结论是成立的. □

由引理 2.3 和引理 2.4 可得出如下定理.

定理 2.2. 任一状态 z_n 经过 3.5 轮 Midori 解密算法后,输出状态 x_{n-3} 的每个比特块都可能差分活动.

再由定理 2.1 和定理 2.2 可得出本文一个重要的结论,如定理 2.3 所示.

定理 2.3. Midori 算法在单密钥条件下的截断不可能差分区分器至多 6 轮.

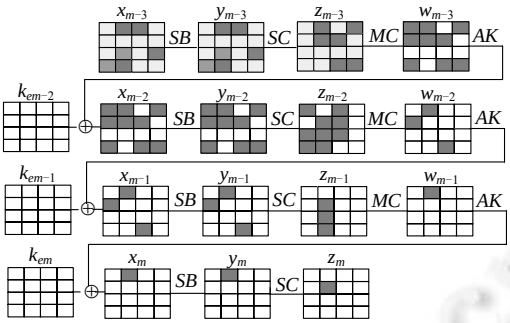

Fig.7 3.5-round differential path of Midori in decryption direction

图 7 3.5 轮 Midori 算法解密方向差分路径

3 Midori 算法 6 轮截断不可能差分区分器的分类

在接下来分类过程中将用到一个概念——列最小差分活动比特块数,其定义如下.

定义 3.1. 设状态 z 各列的差分活动比特块数分别为 $t_{col(0)},t_{col(1)},t_{col(2)},t_{col(3)}$,该状态的列最小差分活动比特块数,是指非零差分列中最小差分活动的比特块数,即 $\min\{t_{col(i)}|t_{col(i)}>0,i=0,1,2,3\}$.

接下来,我们以输入状态 z_1 的列最小差分活动比特块数进行分类讨论如下.

性质 3.1. 输入状态 z_1 的列最小差分活动比特块数为 1 时,不可能构造出 6 轮截断不可能差分区分器.

证明:由性质 2.1 和引理 2.2 可知,此时,输入状态 z_1 经过 2.5 轮 Midori 算法加密后,输出状态 w_3 的每个比特块都可能差分活动;再由定理 2.2 可知,任意状态经 3.5 轮解密算法后,输出状态的每个比特块都可能差分活动,故不可能构造出一条 6 轮截断不可能差分路径. □

性质 3.2. 输入状态 z_1 的列最小差分活动比特块数为 2 时,若要构造一条 6 轮不可能差分路径,输出状态 z_7 必定仅有 1 个比特块差分活动.

证明:要想构造出一条不可能差分路径,由性质 2.2 可知,输入状态 z_1 只能向下加密 2.5 轮,所以输出状态 z_7 要向上解密 3.5 轮,且满足存在确定有无差分活动的比特块,故 z_7 必定仅有 1 个比特块差分活动. □

性质 3.3. 输入状态 z_1 的列最小差分活动比特块数为 3 时,要构造一条 6 轮不可能差分路径,若 z_1 经过 2.5 轮 Midori 算法加密,则输出状态 z_7 必定仅有 1 个比特块差分活动.若输入状态 z_1 经过 3.5 轮 Midori 算法加密,则 z_7 经一次 SC^{-1} 输出状态 y_7 有如下 4 种情形:(1) y_7 只有 1 列有比特块差分活动;(2) y_7 有两列有比特块差分活动,且其中一列差分活动比特块数必定为 1;(3) y_7 有 3 列有比特块差分活动,且其中两列差分活动比特块数必定为 1;(4) y_7 每一列都有比特块差分活动时,且必定有 3 列其差分活动比特块数为 1.

证明:若输入状态 z_1 经过 2.5 轮 Midori 算法加密,输出状态 z_7 必定仅有 1 个比特块差分活动.若输入状态 z_1 经过 3.5 轮 Midori 算法加密,则加密后输出状态 w_4 的每个比特块都可能存在差分活动,则 z_7 必定满足解密 2.5 轮后存在一定没有差分活动的比特块.对于输出状态 z_7 经一次 SC^{-1} 的输出状态 y_7 分如下 4 种情形讨论.

- 情形 1:当 y_7 只有 1 列有比特块差分活动时,显然是满足要求的.
- 情形 2:当 y_7 仅有 2 列有比特块差分活动时,若这两列都有两个以上比特块差分活动,容易推导出 x_5 每个比特块都可能存在差分活动,故不符合要求,所以此时 y_7 必须有一列差分活动比特块数为 1.
- 情形 3:当 y_7 仅在 3 列有比特块差分活动时,若有两列差分活动比特块数超过一个,由情形 2 可知不符合要求,所以 y_7 其中两列差分活动比特块数必定为 1.
- 情形 4:当 y_7 在每一列都有比特块差分活动时,同样由情形 2 可知,有 3 列差分活动比特块数必定为 1.

综上所述,性质 3.3 得证. □

性质 3.4. 输入状态 z_1 列最小差分活动半字节数为 4 时,不可能构造出 6 轮截断不可能差分区分器.

该证明过程同性质 3.1.

综上所述,Midori 算法 6 轮截断不可能差分区分器可分为性质 3.2 与性质 3.3 两大类.由引理 2.2 和引理 2.4 可知,z_1,z_7 仅有 1 列有比特块差分活动的情况下,区分器能向下加密及向上解密更多轮数.故简单考虑 z_1,z_7 仅有 1 列有比特块差分活动的情况,此时按输入状态 z_1 存在的差分活动比特块数将 Midori 算法 6 轮截断不可能差分区分器分为如下两大类.

1) 输入状态 z_1 仅在某一列存在 2 个比特块差分活动时,输出状态 z_7 必定仅有 1 个比特块差分活动.具体不可能差分差分路径为输入状态 z_1 向下加密 2.5 轮,输出状态 z_7 向上解密 3.5 轮.
2) 输入状态 z_1 仅在某一列存在 3 个比特块差分活动时:(1) 若输入状态 z_1 经过 2.5 轮 Midori 算法加密,则输出状态 z_7 必定仅有 1 个比特块差分活动;(2) 若输入状态 z_1 经过 3.5 轮 Midori 算法加密,则输出状态 z_7 必定仅有 1 个或者 2 个比特块差分活动(证明略).

4 Midori-64 算法的不可能差分分析

根据 Midori 算法 6 轮截断不可能差分区分器的分类结果,可构造相应的 6 轮不可能差分区分器.第 4.1 节将给出一个 6 轮不可能差分区分器;第 4.2 节进一步给出 Midori-64 算法的 11 轮不可能差分分析.

4.1 Midori算法的6轮不可能差分区分器

定理 4.1. 当初始输入状态 z_1 仅在第 0~第 2 这 3 个比特块差分活动,且满足$\Delta z_1[0]=\Delta z_1[1]=\Delta z_1[2]$时,经过 6 轮 Midori 算法加密后,输出状态 z_7 仅在第 6、第 7 比特块处差分活动,且满足$\Delta z_1[6]=\Delta z_1[7]$是不可能的.具体形式如图 8 所示.

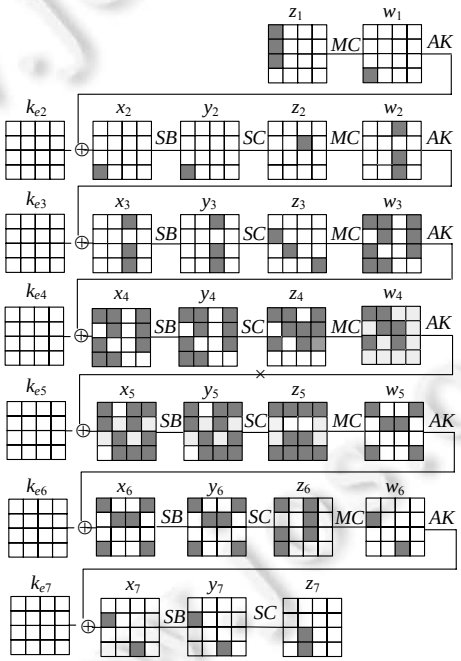

Fig.8 6-round impossible differential distinguisher of Midori

图 8 Midori 算法的 6 轮不可能差分区分器

证明:当输入状态 z_1 仅在第 0~第 2 这 3 个比特块差分活动,且满足$\Delta z_1[0]=\Delta z_1[1]=\Delta z_1[2]$时,经 3.5 轮 Midori 算法加密后,输出状态 w_4 的第 4 比特块必定差分活动;当输出状态 z_7 仅在第 6、第 7 比特块处差分活动,且满足$\Delta z_1[6]=\Delta z_1[7]$时,经 2.5 轮 Midori 算法解密后,输出状态 x_5 的第 4 比特块差分为 0,故产生矛盾,所以结论成立.

证毕. □

4.2 11轮Midori-64算法的不可能差分分析

利用定理 4.1 中给出的 Midori 算法 6 轮不可能差分区分器,向上解密 1.5 轮,向下加密 3.5 轮,给出 11 轮 Midori-64 算法的不可能差分分析结果.攻击过程分为预计算与在线攻击两个阶段.

(一) 在预计算阶段,共需要预计算并存储 4 个表.

- 表 T_1:仅在 $w'_9[1,3]$ 处差分活动的 $\Delta w'_{9,col(0)}$ 共有 2^8 种取值,故$\Delta x_{10,col(0)}$有 2^8 种取值.以$\Delta y_{10,col(0)}$的 2^{16} 种可能取值为索引,根据性质 2.2,平均每个索引值对应 2^8 个 $y_{10,col(0)}$及对应的 $w'_9[1,3]$.因$\Delta y_{10,col(0)}=\Delta C_{col(0)}$和 $w'_9[1,3]=z'_9[10,15]$,以 2^{16} 个密文差分$\Delta C_{col(0)}$为索引,平均可以得到 2^8 个 $y_{10,col(0)}$和 $(z'_9[10,15],z''_9[10,15])$,将其存储在表 T_1 中.
- 表 T_2:仅在 $w'_9[4]$ 处差分活动的 $\Delta w'_{9,col(1)}$ 共有 2^4 种取值.Δy_{10} 第 1 列中仅在第 5~第 7 半字节差分活动,以 2^{12} 个$\Delta C_{col(1)}$为索引,存储 2^4 个 $y_{10}[5,6,7]$和 $(z'_9[14],z''_9[14])$ 在表 T_2 中.
- 表 T_3:在 $w'_9[8]$ 处差分活动的 $\Delta w'_{9,col(2)}$ 共有 2^4 种取值.Δy_{10} 第 2 列中仅在第 9~第 11 半字节差分活动,以 2^{12} 个$\Delta C_{col(2)}$为索引,存储 2^8 个 $y_{10}[9,10,11]$和 $(z'_9[9],z''_9[9])$ 在表 T_3 中.
- 表 T_4:在 $w'_9[13,15]$ 处差分活动的 $\Delta w'_{9,col(3)}$ 共有 2^8 种取值.以 2^{16} 个$\Delta C_{col(3)}$为索引,存储 2^8 个 $y_{10,col(3)}$和 $(z'_9[8,13],z''_9[8,13])$ 在表 T_4 中.

(二) 在线攻击阶段.

1. 选择 2^n 个明文结构,其中的明文满足在第 1、第 3、第 5、第 6、第 9~第 11、第 14、第 15 半字节上取所有的值,其余半字节取固定值.故一个明文结构包含 2^{36} 个明文,可以构造 2^{71} 个明文对.因此,攻击的选择明文量为 2^{n+36},其中包含 2^{n+71} 个明文对.
2. 运用快速排序算法筛选明文对.筛选出密文在第 4、第 8 半字节差分不活动,在第 1、第 3、第 5~第 7、第 9~第 11、第 13、第 15 半字节差分活动的明文对,剩余 2^{n+63} 个明文对.以明文对序号作索引,将筛选得到的明文对存储在表Ω中,只需要存储明文第 1、第 3、第 5、第 6、第 9~第 11、第 14、第 15 半字节与密文第 1~第 3、第 5~第 7、第 9~第 15 半字节.
3. 猜测出口白化密钥 $k_{e11}[0,1,2,3]$.对于表Ω中的 2^{n+63} 个明文对,利用明文对序号可以得到其对应的密文差分$\Delta C_{col(0)}$.利用$\Delta C_{col(0)}$查表 T_1,得到 2^8 个 $y_{10,col(0)}$和 $(z'_9[10,15],z''_9[10,15])$,可以确定密钥 $k_{e11}[0,1,2,3]=C_{col(0)}\oplus y_{10,col(0)}$.以 2^{16} 个 $k_{e11}[0,1,2,3]$的可能值为索引,平均存储 $2^{n+63}\times 2^8/2^{16}=2^{n+55}$ 个明文对序号和 $(z'_9[10,15],z''_9[10,15])$ 在表Ω_1 中.
4. 猜测出口白化密钥 $k_{e11}[5,6,7]$.已知 $k_{e11}[0,1,2,3]$,对表Ω_1 中的 2^{n+55} 个明文对,利用明文对序号可得其对应的密文差分$\Delta C_{col(1)}$.利用$\Delta C_{col(1)}$查表 T_2,得到 2^4 个 $y_{10}[5,6,7]$和 $(z'_9[14],z''_9[14])$,可确定密钥 $k_{e11}[5,6,7]$.以 $k_{e11}[5,6,7]$为索引,平均存储 $2^{n+55}\times 2^4/2^{12}=2^{n+47}$ 个明文对序号和 $(z'_9[10,14,15],z''_9[10,14,15])$ 在表Ω_2 中.
5. 猜测出口白化密钥 $k_{e11}[9,10,11]$.已知 $k_{e11}[0,1,2,3,5,6,7]$,对于表Ω_2 中的 2^{n+47} 个明文对,利用明文对序号可以得到其对应的密文差分$\Delta C_{col(2)}$.利用$\Delta C_{col(2)}$查表 T_3,得到 2^8 个 $y_{10}[9,10,11]$和 $(z'_9[9],z''_9[9])$,可以确定密钥 $k_{e11}[9,10,11]$;以 $k_{e11}[9,10,11]$为索引,平均存储 $2^{n+47}\times 2^4/2^{12}=2^{n+39}$ 个明文对和 $(z'_9[9,10,14,15],z''_9[9,10,14,15])$存储在表$\Omega_3$ 中.
6. 猜测出口白化密钥 $k_{e11}[12,13,14,15]$.已知 $k_{e11}[0,1,2,3,5,6,7,9,10,11]$,对于表$\Omega_3$ 中的 2^{n+39} 个明文对,可以得到其对应的密文差分$\Delta C_{col(3)}$.利用$\Delta C_{col(3)}$查表 T_4,得到 2^8 个 $y_{10,col(3)}$和 $(z'_9[8,13],z''_9[8,13])$,可以确定密钥 $k_{e11}[12,13,14,15]$;以 $k_{e11}[12,13,14,15]$为索引,平均存储 $2^{n+39}\times 2^8/2^{16}=2^{n+31}$ 个明文对序号和 $(z'_9[8,9,10,13,14,15],z''_9[8,9,10,13,14,15])$ 存储在表Ω_4 中.
7. 猜测 $k^*_{e10}[8,9,10,13,14,15]$.已知 $k_{e11}[0,1,2,3,5,6,7,9,10,11,12,13,14,15]$,利用表$\Omega_4$ 中 2^{n+31} 个明文对,穷举 $\Delta z'_8[6,7]$,根据性质 1.2 可以得到 2^8 个 $y_9[8,9,10,13,14,15]$,故可以确定 $k^*_{e10}[8,9,10,13,14,15]=y_9[8,9,10,13,14,15]\oplus z'_9[8,9,10,13,14,15]$.以 $k^*_{e10}[8,9,10,13,14,15]$ 为索引,表Ω_5 平均存储 $2^{n+31}\times 2^8/2^{24}=2^{n+15}$ 个明文

序号和 $(z_8'[6,7], z_8''[6,7])$.

8. 猜测 $k_{e9}^*[2,3]$.已知 $k_{e11}[0,1,2,3,5,6,7,9,10,11,12,13,14,15]$与 $k_{e10}^*[8,9,10,13,14,15]$,利用表$\Omega_5$ 中 2^{n+15} 个明文对穷举 $\Delta z_7'[1,11]$,可得 2^4 个 $k_{e9}^*[2,3]$.以 $k_{e9}^*[2,3]$ 为索引,表Ω_6 中存储 $2^{n+15}\times 2^4/2^8=2^{n+11}$ 个明文对序号.
9. 已知 $k_{e11}[0,1,2,3,5,6,7,9,10,11,12,13,14,15]$与 $k_{e10}^*[8,9,10,13,14,15]$ 及 $k_{e9}^*[2,3]$ 取值,根据密钥扩展算法,由出口白化密钥 $k_{e11}[0,1,2,3,5,6,7,9,10,11,12,13,14,15]$可以直接得到入口白化密钥 $k_{e0}[1,3,5,6,9,10,11,14,15]$,利用上述密钥对$\Omega_6$ 中 2^{n+11} 个明文对加密 1 轮,筛选出满足使得仅在$\Delta w_0[0,5,10]$处活动的明文对.以 $k_{e9}^*[2,3]$ 为索引,表Ω_7 中存储 $2^{n+11}\times 2^{-24}=2^{n-13}$ 个明文对序号.
10. 已知密钥 $k_{e11}[0,1,2,3,5,6,7,9,10,11,12,13,14,15]$与 $k_{e10}^*[8,9,10,13,14,15]$ 及 $k_{e9}^*[2,3]$ 取值,穷举 2^{12} 个密钥 $k_{e1}[0,5,15]$.利用上述密钥对Ω_7 中 2^{n-13} 个明文对加密 1/2 轮,以 2^{-8} 的概率得到不可能差分区分器的输入.将通过检测的密钥列为候选密钥,最后对候选密钥穷举 10 个半字节密钥 $k_{e10}^*[0,1,4,5,6,7,11,12]$ 及 $k_{e11}[4,8]$,进行 11 轮 Midori 算法加密检测得到的 128 比特密钥是否正确.

具体的攻击路径如图 9 所示,其中,网状表示猜解的密钥半字节.

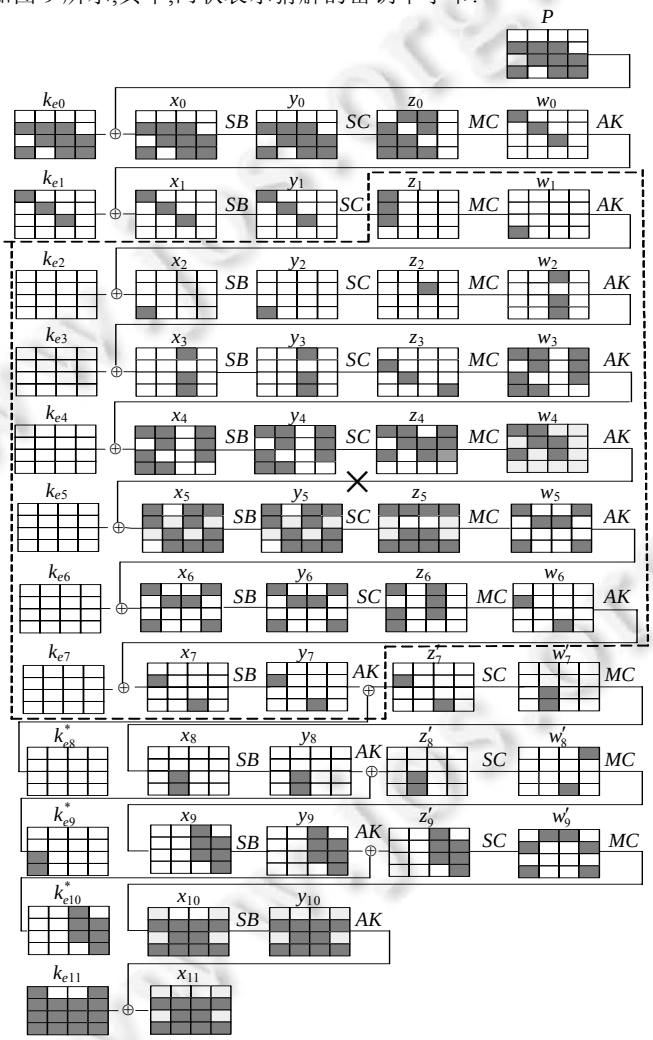

Fig.9 11-round impossible differential cryptanalysis of Midori-64

图 9 11 轮 Midori-64 算法不可能差分分析

由定理 4.2 给出上述 11 轮 Midori-64 算法不可能差分分析的复杂度分析结果.

定理 4.2. 利用 6 轮不可能差分区分器对 11 轮 Midori-64 算法进行不可能差分分析,恢复 128 比特主密钥,其时间复杂度为 $2^{121.4}$ 次 11 轮 Midori-64 算法加密,数据复杂度为 $2^{60.8}$ 个选择明文,存储复杂度为 $2^{96.5}$ 个 Midori-64 状态.

证明:下面给出 11 轮 Midori-64 算法不可能差分分析中步骤 1~步骤 9 各步骤的复杂度,见表 3.证毕. □

Table 3 Complexity of 11-round impossible differential cryptanalysis on Midori-64

表 3 11 轮 Midori-64 算法不可能差分分析复杂度

步骤	时间复杂度	存储复杂度
1	–	$2^{n+71}\times 9$
2	$2^{n+71}\times 2^{6.2}$	$2^{n+63}\times(9+13)$
3	$2^{n+63}\times 2^{8}$	$2^{16}\times 2^{n+55}\times(4+(n+63)/4)$
4	$2^{16}\times 2^{n+55}\times 2^{4}$	$2^{12}\times 2^{n+47}\times(6+(n+63)/4)$
5	$2^{28}\times 2^{n+47}\times 2^{4}$	$2^{12}\times 2^{n+39}\times(8+(n+63)/4)$
6	$2^{40}\times 2^{n+39}\times 2^{8}$	$2^{16}\times 2^{n+31}\times(12+(n+63)/4)$
7	$2^{56}\times 2^{n+31}\times 2^{8}$	$2^{24}\times 2^{n+15}\times(4+(n+63)/4)$
8	$2^{80}\times 2^{n+15}\times 2^{4}$	$2^{8}\times 2^{n+11}\times(n+63)/4$
9	$2^{88}\times 2^{n+11}$	$2^{8}\times 2^{n-13}\times(n+63)/4$

由表 3 可知,前 9 个分析步骤,其时间复杂度大约为 $2^{n+96.6}$ 次 11 轮 Midori-64 算法加密.对于第 10 步,在 88 比特已知密钥下,对 2^{n-13} 个明文对加密 0.5 轮的时间复杂度为 $2^{88}\times 2^{12}\times 2^{n-13}\times 0.5\times 2/11\approx 2^{n+83.5}$ 次 11 轮 Midori-64 算法加密;经过检测后候选密钥集规模 $\varepsilon=2^{100}\times(1-2^{-8})^{2^{n-13}}$,对剩余 40 比特密钥进行穷举恢复主密钥的时间复杂度为 $\varepsilon\times 2^{40}$.当 n=24.8 时,11 轮 Midori-64 算法不可能差分分析总时间复杂度为 $2^{96.6+24.8}+2^{83.5+24.8}+2^{100}\times(1-2^{-8})^{2^{24.8-13}}\times 2^{40}\approx 2^{121.4}$ 次 11 轮 Midori-64 算法加密,数据复杂度为 $2^{60.8}$ 个选择明文,存储复杂度约为 $2^{95.8}\times(4+87.8/4)/16\approx 2^{95.5}$ 个 Midori-64 状态.

5 结 论

本文证明了在单密钥条件下 Midori 算法的截断不可能差分区分器至多 6 轮,并对 6 轮截断不可能差分区分器进行分类;其次,根据 Midori 算法 6 轮截断不可能差分区分器的分类构造了一个较优的 6 轮区分器,并给出了 11 轮 Midori-64 算法的不可能差分分析,其时间复杂度为 $2^{121.4}$,数据复杂度为 $2^{60.8}$,存储复杂度为 $2^{96.5}$.该结果表明,本文给出了一个较好的 11 轮 Midori-64 算法的分析.对于 Midori-128 算法是否也可以根据 Midori 算法 6 轮截断不可能差分区分器的分类构造出较优的区分器,以得到较好的不可能差分分析结果,这是我们下一步要考虑的问题.

作者注 本文是我们于 2018 年 4 月 24 日投到《软件学报》的论文.该文是战略支援部队信息工程大学郭建胜老师(本文第二作者)指导的 2018 届(2018 年 12 月)毕业的研究生李明明(本文第一作者)的硕士学位论文《典型轻量级分组密码算法不可能差分分析研究》工作成果的一部分,特此说明.

References:

[1] Knudsen L. DEAL—A 128-bit block cipher. Technical Report, No.151, Department of Informatics, University of Bergen, 1998.

[2] Biham E, Biryukov A, Shamir A. Cryptanalysis of skipjack reduced to 31 rounds using impossible differentials. In: Proc. of the EUROCRYPT'99. Berlin, Heidelberg: Springer-Verlag, 1999. 12–23.

[3] Kim J, Hong S, Lim J. Impossible differential cryptanalysis using matrix method. Discrete Mathematics, 2010,310(5):988–1002.

[4] Luo YY, Lai XJ, Wu ZM, Gong G. A unified method for finding impossible differentials of block cipher structures. Information Sciences, 2014,263(1):211–220.

[5] Sun B, Liu M, Guo J, Rijmen V, Li RL. Provable security evaluation of structures against impossible differential and zero correlation linear cryptanalysis. In: Proc. of the Advances in Cryptology (EUROCRYPT 2016). Berlin, Heidelberg: Springer-Verlag, 2016. 196–213.

[6] Bogdanov A, Knudsen LR, Leander G, Paar C, Poschmann A, Robshaw MJB, Seurin Y, Vikkelsoe C. PRESENT: An ultra-lightweight block cipher. In: Proc. of the CHES 2007. Berlin, Heidelberg: Springer-Verlag, 2007. 450–466.

[7] Izadi M, Sadeghiyan B, Sadeghian SS, Khanook HA. MIBS: A new lightweight block cipher. In: Proc. of the CANS 2009. Berlin, Heidelberg: Springer-Verlag, 2009. 334–348.

[8] Dolmatov V. GOST 28147-89 encryption, decryption and MAC algorithms. RFC 5830, IETF, 2010. http://tools.ietf.org/html/rfc5830

[9] Gong Z, Nikova S, Law YW. KLEIN: A new family of lightweight block ciphers. In: Proc. of the Int'l Workshop on Radio Frequency Identification: Security and Privacy Issues. Berlin, Heidelberg: Springer-Verlag, 2011. 1–18.

[10] Guo J, Peyrin T, Poschmann A, Robshaw M. The LED block cipher. In: Proc. of the CHES 2011. Berlin, Heidelberg: Springer-Verlag, 2011. 326–341.

[11] Shibutani K, Isobe T, Hiwatari H, Mitsuda A, Akishita T, Shirai T. Piccolo: An ultra-lightweight block cipher. In: Proc. of the CHES 2011. Berlin, Heidelberg: Springer-Verlag, 2011. 342–357.

[12] Wu WL, Zhang L. LBlock: A lightweight block cipher. In: Proc. of the Int'l Conf. on Applied Cryptography and Network Security. Berlin, Heidelberg: Springer-Verlag, 2011. 327–344.

[13] Borghoff J, Canteaut A, Güneysu T, Kavun EB, Knezevic M, Knudsen LR, Leander G, Nikov V, Paar C, Rechberger C, Rombouts P, Thomsen SS, Yalçin T. PRINCE—A low-latency block cipher for pervasive computing applications. In: Proc. of the Int'l Conf. on the Theory and Application of Cryptology and Information Security. Berlin, Heidelberg: Springer-Verlag, 2012. 208–225.

[14] Banik S, Bogdanov A, Isobe T, Shibutani K, Hiwatari H, Akishita T, Regazzoni F. Midori: A block cipher for low energy. In: Proc. of the ASIACRYPT 2015. Berlin, Heidelberg: Springer-Verlag, 2015. 411–436.

[15] Lin L, Wu WL. Meet-in-the-Middle attacks on reduced-round Midori64. IACR Trans. on Symmetric Cryptology, 2017,2017(1): 215–239.

[16] Guo J, Jean J, Nikolić I, Qiao K, Sasaki Y, Sim SM. Invariant subspace attack against full midori64. In: Proc. of the IACR Cryptology ePrint Archive. 2015. https://eprint.iacr.org/2015/1189.pdf

[17] Chen Z, Wang XY. Impossible differential cryptanalysis of Midori. In: Proc. of the Int'l Conf. on Mechatronics and Automation, World Scientific. 2017. 221–229. [doi: 10.1142/9789813208537_0028]

[18] Cui JY. Research on cryptanalysis based on meet-in-the-middle [MS. Thesis]. Zhengzhou: Information Engineering University, 2017 (in Chinese with English abstract).

附中文参考文献:

[18] 崔竞一.基于中间相遇思想的攻击方法研究[硕士学位论文].郑州:信息工程大学,2017.

李明明(1995－),男,湖南衡阳人,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.

徐林宏(1995－),男,硕士,主要研究领域为分组密码的设计与分析.