关于LBLOCK-S的相关密钥不可能回旋密码分析

介绍：随着电子信息的迅速发展和射频识别技术（RFID）等的普及，传统的分组密码已经不适合资源受限的环境。因此，在资源受限的环境下，兼顾安全性和硬件性能的轻量级分组密码成为了研究的热点。

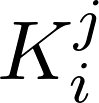
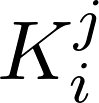
Lblocks作为和凯撒竞争的LAC的核心分组密码，是吴文玲等人在2011年的ANCS会议上提出的LBlock的改进版本。与LBlock算法类似，它也使用了feistel结构以及32轮加密，并且分组长度和密钥长度分别为64比特和80比特。LBlock-s算法和LBlock算法不同的地方在于LBlock-s采用了一种改进的密钥调度算法，具有更快的扩散速度，并将S盒中的10个不同的S盒改为了相同的盒，以降低硬件和软件的成本。在安全性方面，Shan[3]等人表明32轮相关差分验证中至少有32个活跃的盒子中的概率不会高于2-64，并给出轮了10轮和11轮的相关密钥差分特征。肖[4]利用差分密码分析给出了一个16轮的差分路径，李[5]等人用改进的多维零相关线性密码分析对LBlock-s进行了23轮的攻击。利用LBlock人员设计给出的LBlock的14轮微分可能特性，贾[6]对LBlock-s进行了21轮的攻击，他们给出了22轮和23轮的攻击结果，但没有给出具体的分析。并且结果表明，22轮的攻击时间复杂度是278.86次22轮加密的时间，接近穷举搜索的时间复杂度，远高于本文的相应攻击结果。

相关的密钥分析分别由Knudsen[7]和Biham[8]独立引进的。该技术的基本思想就是攻击者发现密钥调度算法的弱点，选择合适的密钥关系，然后预测这些密钥加密后的密文。Knudsen和Biham先后针对Skipjack提出了不可能差分密码分析，攻击者试图找到概率为0的差分特征，以消除错误的密钥，然后恢复正确的密钥。相关密钥密码分析和不可能差分密码分析都是分析各种分组密码安全性的有力技术，对蜂鸟-2，TEA，LBlock，MIBS等许多分组密码都有很好等攻击效果。回旋密码分析是Wagner在1999年提出的，它是差分密码分析的一个变种。回旋密码分析的基本思想就是利用概率较大的短差分特征形成概率较大的长差分特征。相关的不可能回旋密码分析是通过结合使用这三种攻击而获得的。迄今为止，利用这种密码分析技术在AES和LBlock上取得过许多令人满意的分析结果。

本文首次从相关密钥不可能回旋密码分析的角度研究了LBlock-s的安全性。通过分析LBlock-s的轮函数以及密钥调度函数的性质，实现了一个15轮相关的密钥不可能回旋识别器，得到轮一个22轮相关的密钥不可能回旋特征来回复68比特的密钥，时间复杂度为268.76，选择了258个明文。到目前为止，这是对22轮lBlock-s算法最佳对攻击效果。

概论：第二节对LBlock-s进行了描述，并给出了一些本文使用的一些符号。在第三节中，我们研究了相关的密钥不可能回旋密码分析。在第四节中，我们介绍了一个与15轮相关的密钥不可能回旋识别器和22轮的攻击，并在第五节给出结论。

1. ：LBlock-s的描述

* 符号：
* P，C：P为64比特明文和C为64比特密文
* *K*，*K*：80比特的主密钥以及和密钥的差异和密钥的变化率
* *Ki*，*Ki*：第i轮的次密钥以及密钥的变化率
* ，：*Ki*的第*j*次nibble攻击以及*Ki*的第*j*次nibble攻击
* *Xi*，*Xi* ：第*i*轮输入的左半边字符以及它的变化
* *X*0：第一轮输入的明文的右半边
* *Xij*，*Xij*：*Xi*的第j次nibble攻击和*Xi*的第j次nibble攻击
* *Xi* **||** *Xj*：*Xi*和*Xj*的拼接的组合
* *X*<<< *i：X*字符串循环向左移位i个比特
* X>>> i：*X*字符串循环向右移位i个比特
* [*i*]2：整数*i*的二进制形式
* LBlock-s算法概述：

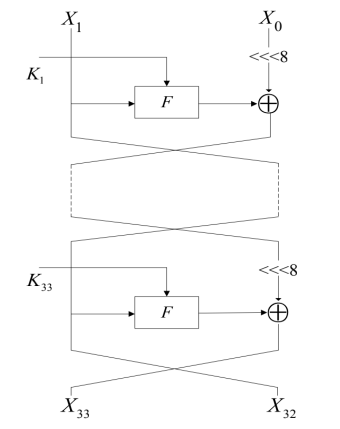
LBlock-s是LBlock的改进版本，具有Feistel结构的变体。它由32轮迭代组成，块长64位，主密钥长80位。加密过程如下，图1示出了一般结构。

输入：X=*X*1 **||** *X*0

for *i* in [2,33]:

*Xi*=*F*(*Xi*-1,*Ki*-1) **⊕** *Xi-2*<<<8

输出：C=*X*33 **||** *X*32

图1

F轮函数定义为： (见图2)。它包括三个基本函数：键加法层、非线性变换和线性扩散函数。混淆函数S包括8个相同的并行4位S盒，这是LBlock中使用的第一个S盒(见表一，16进制表示法)，并且函数P是图2中详细说明的4位逐字排列。

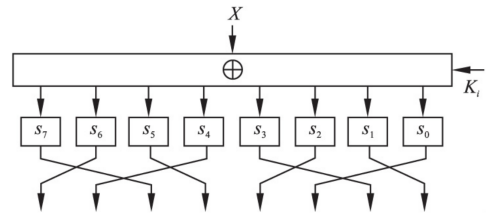
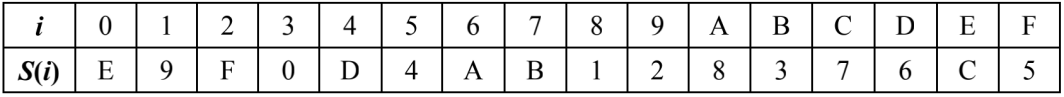
图2

表1 S盒的成分

LBlock-s的解密算法与加密算法相反。因此，这里仅作简要说明：我们让 C = *X*32 **||** *X*33表示密文，随后进行循环运算：

for *i* in [31,0]:

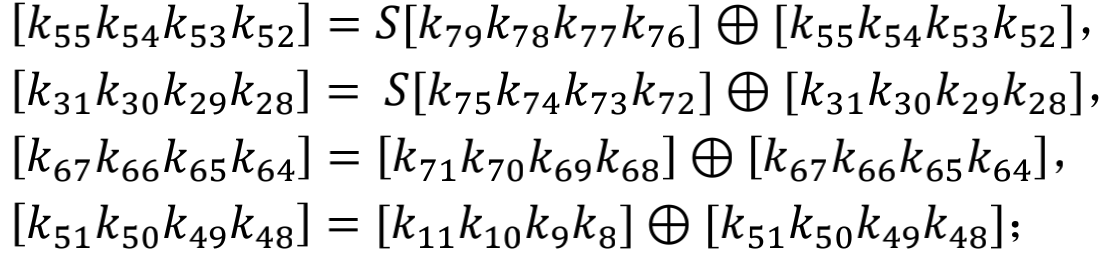
*Xi*=*F*(*Xi+*1,*Ki+*1) **⊕** *Xi+2>>>*8

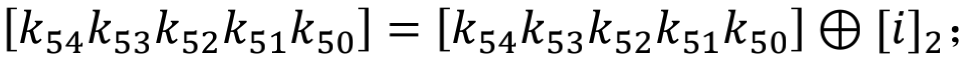
输出：P=*X*1 **||** *X0*

* 密钥调度算法

LBlock-s采用了一种改进的密钥调度算法，该算法具有更好的抗双液密码分析的扩散性能。LBlock-s的密钥调度每次更新16位，这些更新的位受密钥寄存器的32位的影响。相反，原始密钥调度仅基于密钥寄存器的13位更新13位。主密钥 存储在密钥寄存器中。输出最左边的32位作为子密钥*K1*。对于*for i* =1,2，⋯，31，按如下方式更新密钥寄存器：

1.

2.

3.

1. 输出寄存器K最左边的32位作为轮次密钥.

其中S盒与加密算法中使用的S盒相同。

1. :相关密钥不可能回旋密码分析

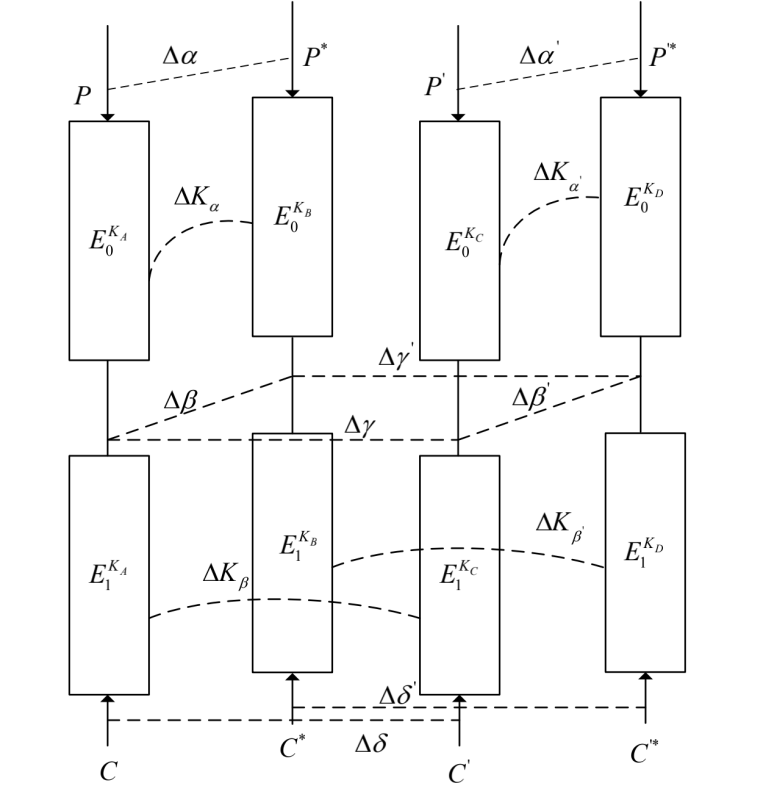
相关密钥不可能回旋密码分析包括相关密钥密码分析、不可能差分密码分析和回旋密码分析。如图3所示，该技术将密码*E*视为两个子密码*E*0和*E*1，即。通常*E=E*0**||** *E*1，每个子密码包含两个相关的密钥微分特性，概率为1。

它们如下所示：

* α→β：是*E0*的第一个相关键差分特性
* α′→β′：是*E0*的第二个相关键差分特性
* δ→γ：是*E1*-1的第一个相关键差分特性
* δ′→γ′：是*E1*-1的第二个相关键差分特性

这些α、β、α′、β′、δ、γ、δ′、γ′都是*n*比特的块。四个特征中使用的对应密钥分别是：*K*A,*K*B,*K*C,*K*D，当β、β′、γ、γ′满足条件*β***⊕***β*′**⊕***γ***⊕***γ*′≠0时，构造了相应的键不可能回旋识别器。然后可以通过在顶部和底部添加轮数来发动攻击。利用扩展部分来猜测密钥，可以剔除满足整体特征的密钥，直到只剩下正确的密钥为止。

相关的密钥不可能回飞棒密码分析充分利用了这三种密码分析方法。在这种攻击中，攻击者需要找到四个密钥差分特征，而不是使用*E*概率较大的单个密钥差分特征，这些特征是*K*A⊕*K*B=*K*α，*K*C⊕*K*D=*K*α’，*K*A⊕*K*C=*K*β，*K*B⊕*K*D=*K*β’，分别的对于*E*0和*E*1，它们不一定相关。换句话说，对于为*E*0选择的任何可能的关键差异，可以选择为*E*1选择的各种关键差异。因此，与相关密钥、不可能差分密码分析相比，相关密钥不可能回旋密码分析更有利于发现具有更好扩散特性的密钥调度算法的密码的相关密钥差分特征。

图3

1. ：LBlock-s的相关密钥不可能回旋密码分析

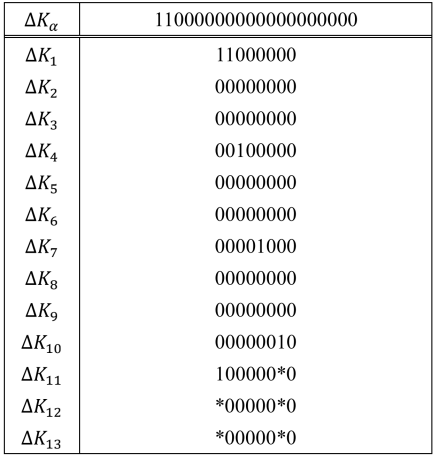
本节详细介绍了LBlock-s上的相关键不可能回旋攻击。在本文的其余部分中，“\*”用于表示非零的4位半字节，而“？”用于表示可以采用任何值的4位半字节。通过对LBlock-s的结构和圆函数的分析，我们得到了一个15轮相关的键不可能回旋识别器：

wpsoffice((00000000,00000000),(00000000,00000000)) ((00000\*00,00000000),(\*0000000,00000000))，并有密钥的差异：*K*α=*K*α′= (11000000000000000000)和*K*β=*K*β′=(00000000000000000000)，通过在该识别器的顶部延伸4轮，在底部延伸3轮，可以攻击22轮LBlock-s算法。

4.1 15轮相关键不可能回旋识别器

通过对LBlock-s密钥调度算法的深入研究，我们发现，如果没有非零差通过S盒，对于某些主密钥差会出现两个零密钥差，一个非零密钥差。要减少活动的S盒的数量，我们挑选密钥的差异*K*α=*K*α’=(110000000000 00000000)来构造低权重密钥差分特征。通过Kα产生的第1轮到第13轮的子密钥差异性由表2显示。由于密钥调度算法具有较快的扩散速度，我们选择其他密钥差异*K*β=*K*β’=(000000000000 00000000)来增加攻击的轮数。也就是说，我们构造了一个相关的键不可能回旋识别器，使得*K*A=*K*C和*K*B=*K*D，这意味着识别器包含两个密钥。

表2 1到13轮到子密钥

结合选取的关键差异，我们仔细选择了输入差异和输出差异。设*α→β*和*α*′*→β*′的输入差都是(00000000,00000000), 那么第五轮和第六轮就不会出现非零差扩散。同时让*E*1的第一个输入差为(00000\*00,00000000)，然后第二个输入差为(\*0000000,00000000)，然后，在同一轮中，相关密钥差分特征*δ→*γ的一些非零半字节的位置将与其他相关密钥差分特征*δ*′*→γ*′的位置相同，这使得在每一步中过滤更多的四分之一用于子密钥恢复。

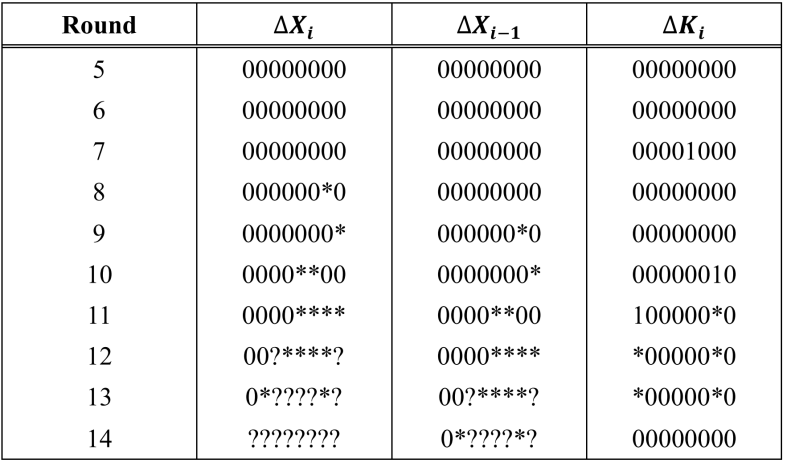
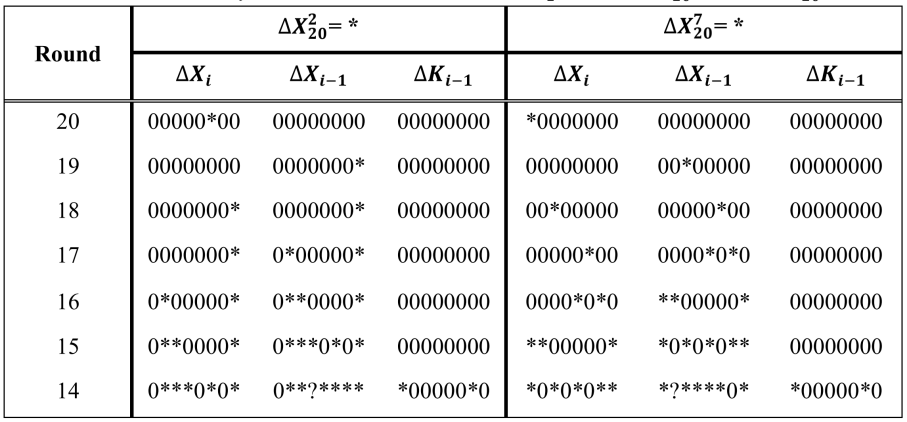
在这次攻击中，*E*表示LBlock-s的15轮相关键不可能回旋识别器，*E*0表示第5轮到第13轮，*E*1表示第14轮到第19轮。

定理1，假设*E*0两个相关的密钥差分特征*α→β*和*α*′*→β*′都是(00000000,00000000) → (????????,0\*????\*?)，并且密钥的差异*K*α=*K*α’= (11000000000000000000)，E1-1第一相关差分特性*δ→*γ是(00000\*00,00000000)→(0\*\*\*0\*0\*,0\*\*?\*\*\*\*)，密钥的差异*K*β= (00000000000000000000)，E1-1第二相关差分特性*δ*′*→*γ′是(\*0000000,00000000) → (\*0\*0\*0\*\*,\*?\*\*\*\*0\*0)，密钥差异为*Kβ*’=(00000000000000000000)，然后，这四个微分特征组成了一个15轮相关的密钥不可能回旋区分LBlock-s。

表3和表4分别显示了𝐸𝐸*E*0和𝐸𝐸E1−1的详细相关关键差异特性。

证明：从加密的方向看，在*α→β*和*α*′*→β*′中*X*713都为0，至于解密的方向，*X*713在第一相关差异*δ→*γ中为0，但是在第二相关差异*δ*′*→*γ′中是\*。很明显，0⊕0⊕0⊕\*=\*，不可能等于0，因此，当这四个相关的密钥差分特征在中间相遇时就产生了冲突，这满足了相关密钥不可能回旋密码分析的原则。

表3 相关密钥差特性

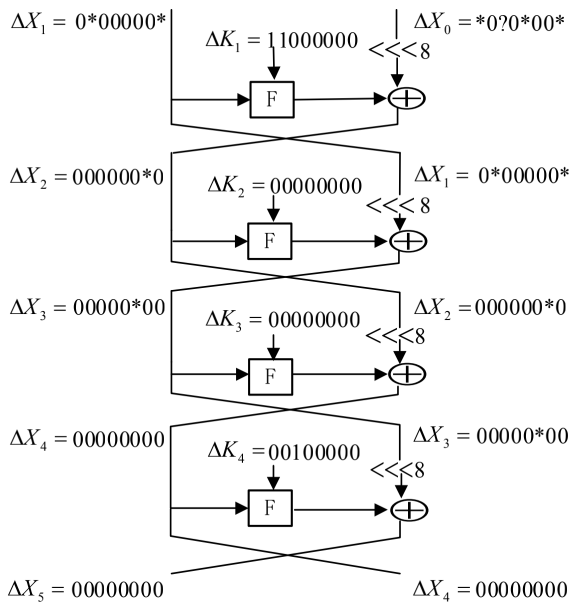
表4 当*X*220=\*并且*X*720=\*时E1-1的相关密钥差特性

wpsoffice因此，上述四个相关的密钥差特性构成了LBlock-s的15轮相关键不可能回旋识别器：((00000000,0000 0000),(00000000,00000000)) ((00000\*00,00000000),(\*0000000,00000000))

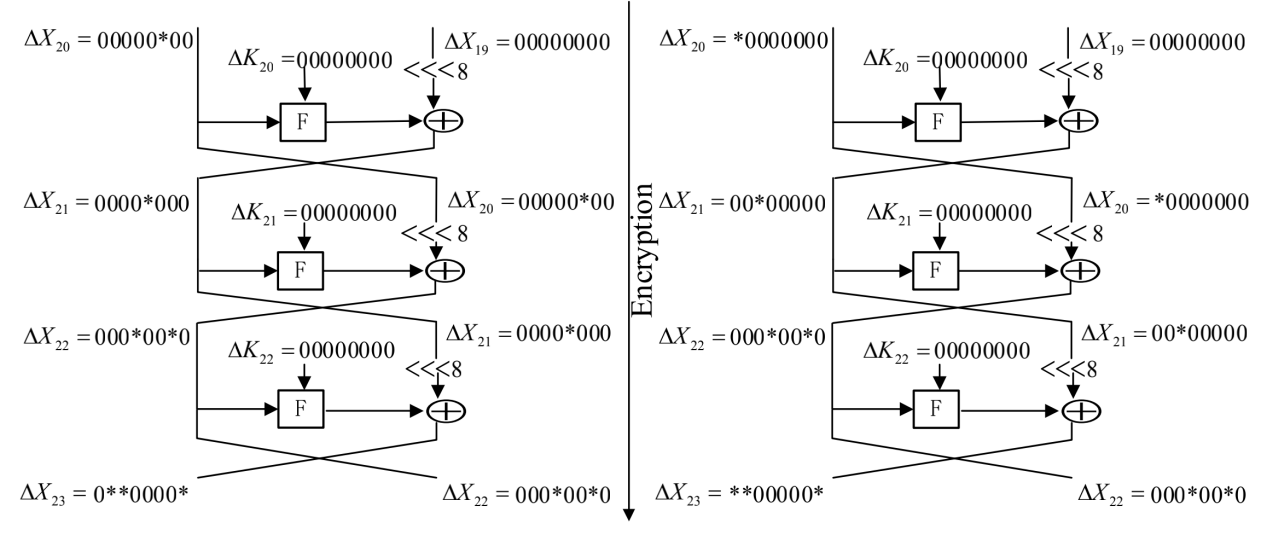
密钥差*K*α=*K*α’= (11000000000000000000)，并且*K*β=*K*β’=(00000000000000000000)。

4.2 对22轮LBlock-s的相关密钥的不可能回旋攻击

对22轮LBlock-s的攻击是通过在上述识别器的顶部加4轮和底部加3轮来实现的。扩展差分特性如图4和图5所示。由于*E*0的两个相关键差分特征被选择为相同的，因此两个特征的输入差异都被给出为（*X*0，*X*1）=(0\*00000\*,\*0?0\*00\*)。攻击程序可详述如下：

图4 添加在区分器顶部的4轮

解密

图5 在识别器底部添加3轮

步骤1:选择一组224个明文来生成一个结构，其中半字节 和f 取 的所有可能值，其他半字节取常量。因此，每个结构包含224×224×＝247个明文对。选择上面描述的2*n*个结构，那么将会有2*n*×247=2*n*+46.1个明文对，表示为(P,P\*)。同样的，选择2*n*个以上的结构，其中就包含个2*n*+47个明文对，称为(P′,P′\*)。然后构造一个四重文本((P,P\*)(P′,P′\*))。我们采用2*n*+1个结构，于是可以得到22*n*+94个四重文本。

步骤2:用密钥*K*A,*K*B,*K*C,*K*D加密这些四重文本22轮，以获得相应的四重密文((C,C\*)(C′,C′\*))。然后通过检查四重密文的差值((0\*\*0000\*,000\*00\*0),(\*\*00000\*,000\*00\*0))，来过滤四重密文。然后剩下的四重文本数为22*n*+94×2-4×22=22*n*+6。

步骤3:猜测子密钥 和

1. 对剩余四分之一的C和C′部分解密一轮，并用方程过滤四分之一：



用子密钥 对C\*和C′\*做同样的操作。

1. 对于剩余的四重密文，对一轮的C和C′进行部分解密，并用方程对四分位数进行过滤：



用子密钥 对C\*和C′\*做同样的操作。

因此大约有22*n*+6×2-4×4=22*n*-10个四重文本，时间复杂度为(22*n*+6×24+22*n*-2×28)×× ≈ 22*n*+2.63

步骤4:猜测子密钥 和

a. 部分加密剩余四分之一组的P和P\*一轮，并用公式过滤四分之一组:

用子密钥对P\*和P′\*做同样的操作。

b. 对于其余的四分之一组，对P和P\*进行一轮部分加密，并通过公式对四分之一组进行过滤:

用子密钥对P\*和P′\*做同样的操作。

c. 然后对剩余的纯文本四分之一部分加密P和P\*一轮，并用公式过滤四分之一:



用子密钥对P\*和P′\*做同样的操作。

因此剩余四重文本数为22*n*-10×2-4×6=22*n*-34，时间复杂度为：

( 22*n*-10×212+22*n*-18×216+22*n*-26×220 ) × × ≈ 22*n*-5.37

步骤5:猜测子密钥 和

对剩余的纯文本四分之一部分加密P和P\*两轮，并用公式过滤四分之一：

并且

用子密钥对P\*和P′\*做同样的操作。

因此剩余四重文本数为22*n*-34×2-4×2=22*n*-42，时间复杂度为：22*n*-34×228 × × ≈ 22*n*-5.37

步骤6:猜测子密钥 和

对剩余的明文四分之一部分加密P和P\*三轮，并用公式过滤四分之一。

 并且

用子密钥对P\*和P′\*做同样的操作。

因此剩余四重文本数为22*n*-42×2-4×2=22*n*-50，时间复杂度为：22*n*-34×228 × × ≈ 22*n*-7.88

步骤7:猜测子密钥 和

1. 对剩余四重密文的部分解密C和C′，并用方程对四分位数进行过滤：



并且

(b) 然后对剩余四分之一的C\*和C′\*部分解密两轮，并用方程过滤四分之一：

并且：

因此剩余四重文本数为22*n*-50×2-4×2=22*n*-58，时间复杂度为：

(22*n*-50×258+22*n*-54×256) × × ≈ 22*n*-4.37

步骤8:猜测子密钥 和 ，因为子密钥 和 已经被猜测了，所以只需要猜测剩余的8位密钥。

1. 对剩余四分之一的C和C′部分解密三轮，并用方程过滤四分之一

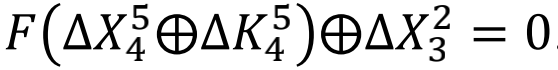
 并且

(b) 然后部分解密剩余四重文本的C\*和C′\*三个回合，并用等式过滤四重文本

 并且

因此剩余四重文本数为22*n*-58×2-4×2=22*n*-66，时间复杂度为：

(22*n*-58×260+22*n*-62×264) × × ≈ 22***n***- 2.88

步骤9:猜测半字节 以确定子密钥 。由于 中的 有24个可能值，因此本次攻击中的 有24×24=28个可能值。对剩下的四个四分之一部分加密四轮，只保留满足条件

的四重文本。根据S盒的差分分布表，上述方程以1/6的概率成立，因此剩余四重文本的数量为22*n*-66×28×1/6≈22*n*-60.58，时间复杂度为：

22*n*-64×28×264×1/8×1/22≈22*n*-1.46。

如果过滤后仍有四重文本，则表明所猜测的子密钥满足相关的密钥不可能回旋的区分，因此需要删除这些子密钥，并尝试其他候选子密钥。

如果选择234个结构，即n=33，则该攻击的时间复杂度约为：

22n+2.63+22*n*-5.37+22*n*-12.46+22*n*-7.88+22*n*-4.37+22*n*-2.88+22*n*-1.46 =22*n*+2.76=268.76

22轮加密，数据复杂度为2*n*+1+24=258选明文。

表5显示了步骤3-9中猜测的位数的摘要。

表5 步骤3-9中猜测的位数

