【信息科学与控制工程】　　doi：10.11809/scbgxb2016.08.033

针对 LBlock 算法踪迹驱动 Cache 攻击 *S* 盒特性分析

于　茜[a]，蔡红柳[a]，陈财森[b]

（装甲兵工程学院　a. 信息工程系；b. 科研部，北京　100072）

摘要：针对轻量级密码 LBlock 算法的 Cache 计时研究，着重分析密码算法中 *S* 盒的非线性结构特征。基于其结构特征推导出 *S* 盒的真值表，求解得出 *S* 盒输入输出关系的代数表达式；再结合 LBlock 算法的加密过程和轮函数 *F* 的结构，推导出每个轮运算的表达式以及 *S* 盒查找索引的代数表达式；结合踪迹驱动 Cache 计时攻击的攻击原理与模型，总结得出针对 LBlock 算法 Cache 攻击中密钥分析的核心表达式，结果表明 LBlock 算法存在遭受 Cache 计时攻击的可能性。

关键词：LBlock 算法；Cache 计时攻击；代数表达式；*S* 盒；特性分析

本文引用格式：于茜，蔡红柳，陈财森. 针对 LBlock 算法踪迹驱动 Cache 攻击 *S* 盒特性分析[J]. 兵器装备工程学报，2016(8)：146－150.

Citation format：YU Xi，CAI Hong-liu，CHEN Cai-sen. Completeness Analysis on S-Box of Trace Driven Cache Timing Attack against LBlock Algorithm[J]. Journal of Ordnance Equipment Engineering，2016(8)：146－150.

中图分类号：TP391　　**文献标识码**：A　　**文章编号**：2096－2304(2016)08－0146－06

Completeness Analysis on *S*-Box of Trace Driven Cache Timing Attack against LBlock Algorithm

YU Xi[a]，CAI Hong-liu[a]，CHEN Cai-sen[b]

（a. Department of Information and Communication Engineering；
b. Department of Science Research，Academy of Armored Forces Engineering，Beijing 100072，China）

Abstract：Aiming at the study of the cache timing attack for lightweight block cipher called LBlock，we focused on the analysis of the nonlinear structure characteristics of *S* box in cryptographic algorithms. Firstly，we derived the truth-table of *S* box based on its structure feature to obtain the relation algebra expression between inputs and outputs of *S* box. Secondly，with reference of encryption process of the LBlock algorithm and the structure of round function *F*，the operation expression of each round and the algebra expressions of look-up index for *S* box were deduced. Finally，we summarized the core expression of the analysis of the key in the cache attack for LBlock algorithm on the basis of the principle and model of the trace-driven cache timing attack. The final conclusion shows that the LBlock algorithm has the possibility of the cache timing attack.

Key words：LBlock algorithm；Cache timing attack；algebra expression；S box；characteristic analysis

随着信息安全的地位日益重要，轻量级密码算法在 RFID 电子标签、无线传感器网络、移动智能终端等资源受限设备的应用越来越广，是目前密码安全研究的一个热点领域。LBlock 算法[1]是吴文玲和张蕾 2011 年提出的一种基于 32 轮类 Feistel 结构的轻量级分组密码，采用与传统分组密码类似的迭代结构，即将明文用简单的轮函数在密钥的作用

收稿日期：2016－02－17；**修回日期**：2016－04－10

作者简介：于茜(1992—)，女，硕士研究生，主要从事网络安全与对抗研究。

下进行多次迭代最终得到密文,轮密钥则通过密钥调度算法由主密钥生成,其密钥和分组长度分别为 80 和 64 比特。

轻量级密码算法[2]具有处理数据规模小、数据吞吐量低、实现占用内存空间小等特点,在保证安全性的前提下,为提高实现效率,目前主要采用利用现有算法结构的健壮性和安全性改进现有的密码算法,如改善算法的布尔函数,使算法的 S 盒相同,以降低算法实现的资源需求等。基于分组密码的迭代结构特性,传统的安全性分析方法主要有基于统计方法的线性分析[3]、差分分析[4]、立方攻击[5]和基于解方程组的代数攻击[6]等。近年来,随着旁路攻击在密码分析领域的研究进展,攻击者可通过获取密码算法在执行过程中泄露的功耗、电磁、时间等旁路信息,利用这些旁路信息与密钥的相关性分析获取密钥,是密码分析研究领域的热点[7]。将旁路攻击与传统密码分析方法相结合,提出了代数旁路攻击、立方攻击与侧信道相结合的攻击方法。Shamir 等[8]提出的侧信道立方攻击,其思想是假设攻击者通过侧信道获取密码算法中间状态的某个比特,将该比特信息与立方攻击相结合从而获取密钥信息,2009 年 Yang 等[9]对 PRESENT 算法进行了侧信道立方攻击的验证。自 Kocher[10]和 Kelsey[11]等人提出将高速存储器 Cache 的行为信息作为旁路泄露信息的思想以来,Cache 攻击成为旁路攻击的新热点,其主要思想是通过获取密码算法在执行过程中对 Cache 的访问行为,以及 Cache 计时信息与密钥信息的相关性分析获取密钥信息,对基于 S 盒查找的分组密码算法构成了安全性威胁,如 DES、AES、ARIA 算法,同时也可基于滑动窗口算法实现的 RSA 公钥密码算法进行攻击。

在 Cache 计时攻击过程中,如何准确获取 Cache 访问行为信息和 S 盒查找索引与子密钥的相关性分析是密钥分析的关键因素。本文在踪迹驱动 Cache 计时攻击原理的基础上,结合 LBlock 算法实现过程的代数性质分析,给出 LBlock 算法的非线性组件 S 盒的代数表达式,结合算法的加密过程和轮函数 F 的结果,推导出每个轮运算的表达式以及 S 盒查找索引的代数表达式,由轮子密钥的代数表达式,给出通过 Cache 访问信息与 S 盒查找索引推导出子密钥信息的核心表达式,从理论上论证了 LBlock 存在遭受 Cache 计时攻击的可能性。

1　LBlock 密码算法实现分析

1.1　算法概述

LBlock 算法的加密过程主要是一个 32 位的类 Feistel 结构迭代。设 $M = X_1 \| X_0$ 表示 64 比特明文,其加密过程如下:

1) 对于 $i = 2,3,\cdots,33$,执行:

$$X_i = F(X_{i-1},K_{i-1}) \oplus (X_{i-2} <<< 8)$$

2) 输出 64 比特密文 $C = X_{32} \| X_{33}$。LBlock 加密结构如图 1 所示。

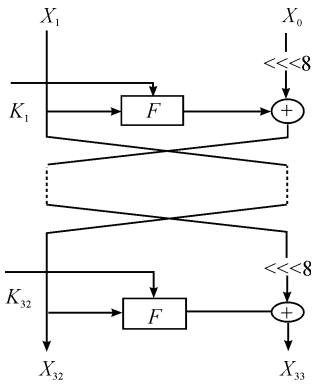

图 1　LBlock 加密结构

其中,每轮加密运算的轮函数 F 及其涉及的混淆函数 S 和扩散函数 P 定义如下。

1) 轮函数 F:

$$F(X,K_i) = P(S(X \oplus K_i)) \in \{0,1\}^{32}$$

2) 混淆函数 S:

$S:\{0,1\}^{32} \rightarrow \{0,1\}^{32}$

$Y = Y_7 \| Y_6 \| Y_5 \| Y_4 \| Y_3 \| Y_2 \| Y_1 \| Y_0 \rightarrow$

$Z = Z_7 \| Z_6 \| Z_5 \| Z_4 \| Z_3 \| Z_2 \| Z_1 \| Z_0$

$Z_7 = s_7(Y_7), Z_6 = s_6(Y_6)$

$Z_5 = s_5(Y_5), Z_4 = s_4(Y_4)$

$Z_3 = s_3(Y_3), Z_2 = s_2(Y_2)$

$Z_1 = s_1(Y_1), Z_0 = s_0(Y_0)$

混淆函数中 S 盒的具体变换规律如表 1 所示。

表 1　LBlock 的 S 盒

S 盒	表元素的值
S_0	14,9,15,0,13,4,10,11,1,2,8,3,7,6,12,5
S_1	4,11,14,9,15,13,0,10,7,12,5,7,2,8,1,3
S_2	1,14,7,12,15,13,0,6,11,5,9,3,2,4,8,10
S_3	7,6,8,11,0,15,3,14,9,10,12,13,5,2,4,1
S_4	14,5,15,0,7,2,12,13,1,8,4,9,11,10,6,3
S_5	2,13,11,12,15,14,0,9,7,10,6,3,1,8,4,5
S_6	11,9,4,14,0,15,10,13,6,12,5,7,3,8,1,2
S_7	13,10,15,0,14,4,9,11,2,1,8,3,7,5,12,6
S_8	8,7,14,5,15,13,0,6,11,12,9,10,2,4,1,3
S_9	11,5,15,0,7,2,9,13,4,8,1,12,14,10,3,6

3) 扩散函数 P:

$P:\{0,1\}^{32} \rightarrow \{0,1\}^{32}$

$Z = Z_7 \| Z_6 \| Z_5 \| Z_4 \| Z_3 \| Z_2 \| Z_1 \| Z_0 \rightarrow$

$U = U_7 \| U_6 \| U_5 \| U_4 \| U_3 \| U_2 \| U_1 \| U_0$

$Z_7 = U_6, Z_6 = U_4, Z_5 = U_7, Z_4 = U_5$

$Z_3 = U_2, Z_2 = U_0, Z_1 = U_3, Z_0 = U_1$

轮函数 F 的具体结构如图 2 所示。

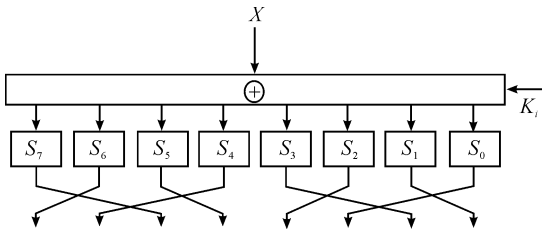

图 2 LBlock 轮函数 F 的结构

根据 LBlock 轮函数 F 的结构图,对应 8 个 S 盒,将每轮参与轮函数 F 的左 32 比特 X_i 按 4 比特一组,从左到右依次记为 $a_{i,7},a_{i,6},a_{i,5},a_{i,4},a_{i,3},a_{i,2},a_{i,1},a_{i,0}$;相应地,轮密钥 K_i 也按 4 比特一组,从左到右依次记为 $K_{i,7},K_{i,6},K_{i,5},K_{i,4},K_{i,3},K_{i,2},K_{i,1},K_{i,0}$。

假设第一轮右 32 比特 X_0 为全 0,那么以 S_7 为例,则可知第一轮的查找索引为 $a_{1,7}\oplus K_{1,7}$;第二轮的查找索引为 $a_{2,7}\oplus K_{2,7}$。

由 LBlock 算法加密过程,每轮左 32 位的数学表达为

$$X_i = P(S(X_{i-1},K_{i-1}))\oplus(X_{i-2}<<<8)$$

则有:

$$X_2 = P(S(X_1,K_1))\oplus(X_0<<<8) = P(S(X_1,K_1))$$

由轮函数 F 的结构图可知,第一轮 S_6 盒的输出对应于第二轮 S_7 盒的索引,即 $a_{2,7}=S_6(a_{1,6}\oplus K_{1,6})$。

那么可得出,第二轮 S_7 盒的查找索引为 $S_6(a_{1,6}\oplus K_{1,6})\oplus K_{2,7}$。

因此通过以上分析得出如下结论:

假设 LBlock 算法第一轮右 32 比特 X_0 为全 0,由算法加密结构可得第一轮查找 S_7 盒的索引为 $a_{1,7}\oplus K_{1,7}$;第二轮查找 S_7 盒的索引为 $S_6(a_{1,6}\oplus K_{1,6})\oplus K_{2,7}$。

该结论也是通过获取 S 盒索引值分析推导相关密钥位的关键因素。

1.2 子密钥生成算法

LBlock 算法的子密钥生成,主要用于如何从 80 比特的主密钥中依次生成 32 个 32 位的轮子密钥。假设初始时 80 比特的主密钥存储于密钥寄存器 K 中,记为

$$K = k_{79}k_{78}\cdots k_0$$

其中,k_0 表示最低位。

算法步骤具体如下:

1) 输出 K 的最左边 32 比特作为轮子密钥 K_1;

2) 对于 $i=1,2,\cdots,31$,执行:

a) $K<<<29$;

b) $[k_{79}k_{78}k_{77}k_{76}]=s_9[k_{79}k_{78}k_{77}k_{76}]$,

$[k_{75}k_{74}k_{73}k_{72}]=s_8[k_{75}k_{74}k_{73}k_{72}]$;

c) $[k_{50}k_{49}k_{48}k_{47}]=[k_{50}k_{49}k_{48}k_{47}]\oplus[i]_2$;

d) 输出当前最左边 32 位最为轮子密钥 K_{i+1}。

利用同样的方法依次生成各个轮子密钥,用于轮计算。

2 LBlock 算法中非线性 S 盒的特性分析

2.1 S 盒的真值表推导

LBlock 算法的 S 盒输入、输出均为 4 位二进制数。假设输入 $X=x_3x_2x_1x_0$,其中 x_3 为最高位,x_0 为最低位,输出 $Y=y_3y_2y_1y_0$,其中 y_3 为最高位,y_0 为最低位。根据表 1 中 LBlock S 盒的变换规律,以 S_0 为例:当输入 $X(x_3x_2x_1x_0)=0000$ 时,输出 $Y(y_3y_2y_1y_0)=1110$,依次可得 S_0 的真值表,如表 2 所示。

表 2 S_0 的真值表

项目	X 与 Y 的数值对应关系			
$X(x_3x_2x_1x_0)$	0000	0001	0010	0011
$Y(y_3y_2y_1y_0)$	1110	1001	1111	0000
项目	X 与 Y 的数值对应关系			
$X(x_3x_2x_1x_0)$	0100	0101	0110	0111
$Y(y_3y_2y_1y_0)$	1101	0100	1010	1011
项目	X 与 Y 的数值对应关系			
$X(x_3x_2x_1x_0)$	1000	1001	1010	1011
$Y(y_3y_2y_1y_0)$	0001	0010	1000	0011
项目	X 与 Y 的数值对应关系			
$X(x_3x_2x_1x_0)$	1100	1101	1110	1111
$Y(y_3y_2y_1y_0)$	0111	0110	1100	0101

由表 2 S_0 的真值表可分别得到 LBlock 算法中 S_0 盒 4 个对应于 $x_3x_2x_1x_0$(从 0000 到 1111)的输出 y_0,y_1,y_2,y_3 的真值表,如表 3 所示。

表 3 LBlock S 盒(S_0)中输出 y_0,y_1,y_2,y_3 的真值表

	y_0	y_1	y_2	y_3
输出 y 的真值	011010011	101000110	101011000	111010110
	0011001	1011100	0001111	0100010

假设 LBlock 算法中 S 盒的输出 $Y(y_3y_2y_1y_0)$ 与输入 $X(x_3x_2x_1x_0)$ 的布尔函数关系为

$$y_i = f_i(x_3,x_2,x_1,x_0)$$

其中,$i=0,1,2,3$,以 S_0 盒为例,根据 4 个输出 y_0,y_1,y_2,y_3 的真值表求解其布尔函数表达式,以 y_0 为例,真值表中共有 8 项为 1,可表示成:

$$y_0 = \sum_j m_j = m_{0001}\oplus m_{0010}\oplus m_{0100}\oplus m_{0111}\oplus m_{1000}\oplus m_{1011}\oplus m_{1100}\oplus m_{1111}$$

其中,$j=x_3x_2x_1x_0$,即 $m_{0001},m_{0010},m_{0100},m_{0111},m_{1000},m_{1011},$

m_{1100},m_{1111}分别是 y_0 的真值表中为 1 的项。将这 8 项逐一展开,以 m_{0001}为例,此时 $j = x_3x_2x_1x_0 = 0001$,其中,将为 1 的项用 x_i 表示,为 0 的项用 $\bar{x}_i$ 表示[12],则 $m_{0001} = \bar{x}_3\bar{x}_2\bar{x}_1x_0$,再将 $\bar{x}_i = x_i \oplus 1$ 带入其中,可得:$m_{0001} = (x_3 \oplus 1)(x_2 \oplus 1)(x_1 \oplus 1)x_0$

展开后得:

$$m_{0001} = x_3x_2x_1x_0 \oplus x_3x_2x_0 \oplus x_3x_1x_0 \oplus x_2x_1x_0 \oplus x_3x_0 \oplus x_2x_0 \oplus x_1x_0 \oplus x_0$$

同理可得:

$$m_{0010} = x_3x_2x_1x_0 \oplus x_3x_2x_1 \oplus x_3x_1x_0 \oplus x_2x_1x_0 \oplus x_3x_1 \oplus x_2x_1 \oplus x_1x_0 \oplus x_1$$

$$m_{0100} = x_3x_2x_1x_0 \oplus x_3x_2x_1 \oplus x_3x_2x_0 \oplus x_2x_1x_0 \oplus x_3x_2 \oplus x_2x_1 \oplus x_2x_0 \oplus x_2$$

$$m_{0111} = x_3x_2x_1x_0 \oplus x_2x_1x_0m_{1000} = x_3x_2x_1x_0 \oplus x_3x_2x_1 \oplus x_3x_2x_0 \oplus x_3x_1x_0 \oplus x_3x_2 \oplus x_3x_1 \oplus x_3x_0 \oplus x_3$$

$$m_{1011} = x_3x_2x_1x_0 \oplus x_3x_1x_0$$

$$m_{1100} = x_3x_2x_1x_0 \oplus x_3x_2x_1 \oplus x_3x_2x_0 \oplus x_3x_2$$

$$m_{1111} = x_3x_2x_1x_0$$

全部相加并约去 0 项即可得到 y_0,同理可得其他 3 个输出 y_1,y_2,y_3 关于输入的表达式。

即可得出如下结论:

设 $X = x_3x_2x_1x_0$ 为 S 盒的输入,$Y = y_3y_2y_1y_0$ 为 S 盒的输出,利用 LBlock 算法中 S_0 的真值表,可计算出:

$$y_0 = x_0 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_2x_3$$

$$y_1 = 1 \oplus x_0 \oplus x_2 \oplus x_0x_2 \oplus x_1x_2 \oplus x_3$$

$$y_2 = 1 \oplus x_0x_2 \oplus x_1x_2 \oplus x_3 \oplus x_0x_3 \oplus x_2x_3 \oplus x_0x_2x_3 \oplus x_1x_2x_3$$

$$y_3 = 1 \oplus x_0x_1 \oplus x_0x_2 \oplus x_3 \oplus x_1x_3 \oplus x_0x_2x_3$$

其他 S 盒(S_1 到 S_9)的 4 个输出的代数表达式都可用类似方法求得,这里不再赘述。

2.2　轮子密钥的代数表达式

根据上述章 1.2 中描述的子密钥生成算法和章 2.1 中推导的 S 盒输入输出代数表达式可以得到 LBlock 算法的每一轮子密钥的代数表达式。

设主密钥

$$K = k_{79}k_{78}\cdots k_0$$

则第二轮子密钥为

$$K_2 = \{1 \oplus k_{47} \oplus k_{49} \oplus k_{47}k_{49} \oplus k_{48}k_{49} \oplus k_{50},\ k_{47} \oplus k_{48} \oplus k_{49} \oplus k_{50} \oplus k_{49}k_{50},\ 1 \oplus k_{47} \oplus k_{47}k_{49} \oplus k_{48}k_{49} \oplus k_{50} \oplus k_{47}k_{50} \oplus k_{49}k_{50} \oplus k_{47}k_{49}k_{50} \oplus k_{48}k_{49}k_{50},\ 1 \oplus k_{47}k_{48} \oplus k_{47}k_{49} \oplus k_{50} \oplus k_{48}k_{50} \oplus k_{47}k_{49}k_{50},\ 1 \oplus k_{43} \oplus k_{43}k_{45} \oplus k_{44}k_{45} \oplus k_{43}k_{46} \oplus k_{45}k_{46}k_{43}k_{45}k_{46} \oplus k_{44}k_{45}k_{46} \oplus k_{43} \oplus k_{44} \oplus k_{43}k_{44}k_{45} \oplus k_{43}k_{45} \oplus k_{44}k_{46} \oplus k_{45}k_{46} \oplus k_{43}k_{45}k_{46},\ k_{43} \oplus k_{44} \oplus k_{45} \oplus k_{46} \oplus k_{45}k_{46},\ k_{43} \oplus k_{45} \oplus k_{43}k_{45} \oplus k_{44}k_{45} \oplus k_{46},\ k_{42},\ k_{41},\ k_{40},\ k_{39},\ k_{38},\ k_{37},\ k_{36},\ k_{35},\ k_{34},\ k_{33},\ k_{32},\ k_{31},\ k_{30},\ k_{29},\ k_{28},\ k_{27},\ k_{26},\ k_{25},\ k_{24},\ k_{23},\ k_{22},\ k_{21},\ k_{29},\ k_{19}\}$$

同理可得第三轮的子密钥 K_3,以此类推。

3　基于 S 盒索引特性的踪迹驱动 Cache 攻击密钥分析

3.1　踪迹驱动 Cache 攻击原理与 S 盒索引特性

踪迹驱动 Cache 攻击是一种非常有效的旁路分析技术[13],其原理是:在执行密码算法的过程中,常常多次查表访问同一个 S 盒,利用多次查表产生的 Cache 命中和失效情况推测密钥。其中,查找所需元素在 Cache 中的情况称为 Cache 命中,查找所需元素不在 Cache 中的情况称之为 Cache 失效,此时处理器会把所需元素所在内存块的所有对应元素加载到某一 Cache 行。

实验中,先对 Cache 进行清空,则第一次查表通常会发生 Cache 失效;第二次查表时,Cache 可能产生 Cache 命中也可能产生 Cache 失效,若命中,说明所需元素一定在第一次查表后加载元素的 Cache 行中。而每次查表需要一个查表索引值,该索引值的高两位对应 Cache 行,低两位对应行内地址,即两次查表索引值的高两位相等。

由此得出结论:

设 Y 为查找 S 盒的索引值,其高两位表示为$\langle Y\rangle$,若两次查找同一 S 盒的索引值分别为 Y_1,Y_2,且二次查找发生 Cache 命中,则有

$$\langle Y_1\rangle = \langle Y_2\rangle$$

3.2　LBlock 密钥与 S 盒索引值相关性分析

通过构造 LBlock 算法加密过程中的 S 盒查找命中,利用上述得出的 3 个结论理论上对 LBlock 密钥与 S 盒索引值相关性进行分析[14]。

假设第一轮右 32 比特 X_0 为全 0,随机选择第一轮加密的高四位明文 $a_{i,7}$和次高四位明文 $a_{i,6}$,使得第二轮查找 S_7 发生 Cache 命中。则由节 1.1 与 3.1 得出的结论可推导出:

$$S_6(a_{1,6} \oplus K_{1,6}) \oplus K_{2,7} = a_{1,7} \oplus K_{1,7} \quad (1)$$

将明文比特和密钥比特分别代入式(1),可得:

$$S_6(a_{1,6} \oplus k_{75}k_{74}k_{73}k_{72}) \oplus k_{50}k_{49}k_{48}k_{47} = a_{1,7} \oplus k_{79}k_{78}k_{77}k_{76}$$

此时 S_7 索引的高两位比特只与 10 位密钥 k_{79},k_{78}, k_{75},k_{74},k_{73},k_{72},k_{50},k_{49},k_{48},k_{47}有关,其中最高位只与 9 位密钥 k_{79},k_{75},k_{74},k_{73},k_{72},k_{50},k_{49},k_{48},k_{47}有关,次高位只与 9 位密钥 k_{78},k_{75},k_{74},k_{73},k_{72},k_{50},k_{49},k_{48},k_{47}有关。对此,选出发生命中的 12 组明文,由这 12 组等式可以确定唯一解,可以恢复与高两位比特相关的 10 位密钥。综上可知,至多进行 $12 \times 2^9 + 12 \times 2^9 = 12 \times 2^{10}$ 次判别运算可恢复上述 10 位密钥 k_{79},

k_{78}, k_{75}, k_{74}, k_{73}, k_{72}, k_{50}, k_{49}, k_{48}, k_{47}。

同理,分析第一轮第二轮 S_6 盒查表索引的高两位比特,可唯一确定 8 位密钥,即 k_{67}, k_{66}, k_{65}, k_{64}, k_{46}, k_{45}, k_{44}, k_{43}。

进一步分析其他 S 盒查表索引的高两位,最终可恢复 49 位密钥,具体每次 S 盒查表索引的高两位对应密钥位和密钥个数如表 4 所示。

表 4 第二轮命中时 S 盒查表索引的高两位对应的密钥个数和密钥位

S 盒	每次查找表索引的高两位所对应的密钥位	密钥个数
S_7	k_{79}, k_{78}, k_{75}, k_{74}, k_{73}, k_{72}, k_{50}, k_{49}, k_{48}, k_{47}	10
S_6	k_{67}, k_{66}, k_{65}, k_{64}, k_{46}, k_{45}, k_{44}, k_{43}	8
S_5	k_{77}, k_{76}, k_{71}, k_{70}, k_{42}, k_{41}	6
S_4	k_{69}, k_{68}, k_{38}, k_{37}	4
S_3	k_{63}, k_{62}, k_{59}, k_{58}, k_{57}, k_{56}, k_{34}, k_{44}	8
S_2	k_{51}, k_{30}, k_{29}	3
S_1	k_{61}, k_{60}, k_{55}, k_{54}, k_{26}, k_{25}	6
S_0	k_{53}, k_{52}, k_{22}, k_{21}	4

如果每次选择的明文的组数都比每次 S 盒索引所涉及的密钥数多 2,则从表 4 中易知此时共需 $12+10+8+6+10+5+8+6=65$ 个选择明文,大约 $12\times 2^{10}+10\times 2^{8}+8\times 2^{6}+6\times 2^{4}+10\times 2^{8}+5\times 2^{3}+8\times 2^{6}+6\times 2^{4}\approx 2^{14.18}$ 次判别操作。

因为 LBlock 算法有 32 轮,每一轮都有 8 个 S 盒参与运算,同时每一次判别操作至多有 2 个 S 盒参与运算,所以 1 次判别运算可视为 $2/(32\times 8)=1/128$ 次 LBlock 加密运算。综上所述,共需约 $2^{7.18}$ 次 LBlock 加密运算,可恢复表 1 中的 49 bit 密钥。

同理,对 S 盒进行第三轮第四轮分析,可获得更多的相关密钥位,如表 5 所示。

表 5 第四轮命中时 S 盒查表索引的高两位对应的密钥个数和密钥位

S 盒	每次查找表索引的高两位所对应的密钥位	密钥个数
S_7	k_{20}, k_{19}, k_{18}	3
S_6	k_{36}, k_{35}, k_{17}, k_{16}, k_{15}, k_{14}	6
S_5	k_{13}, k_{12}	2
S_4	k_{40}, k_{39}, k_{9}, k_{8}	4
S_3	k_{28}, k_{27}, k_{5}, k_{4}	4
S_2	k_{1}, k_{0}	2
S_1	k_{32}, k_{31}	2
S_0	k_{24}, k_{23}	2

同样假设每次选择的明文的组数都比每一次索引所涉及的密钥数多 2,则分析第 3 轮只需 $5+8+4+6+6+4+4+4=41$ 个选择明文,判别时间可忽略,最后剩余的 6 bit 密钥可以用穷举的方法来得到,至多需要 2^6 次 LBlock 运算。

综上所述,整个攻击一共需要 106 个选择明文,约 $2^{7.18}+2^6\approx 2^{7.71}$ 次加密运算可恢复 LBlock 的全部密钥。

因此,LBlock 算法存在 Cache 计时攻击的可行性。

4 结论

本文通过深入分析 LBlock 算法的实现过程及算法函数结构,依据踪迹驱动 Cache 计时攻击原理,针对 LBlock 算法的 S 盒特性展开研究,首先分析得出其非线性 S 盒的代数表达式,依据加密过程和轮函数 F 的结构,推导出每个轮运算的表达式以及 S 盒查找索引的代数表达式;然后基于 S 盒索引特性分析,提出了针对 LBlock 算法 Cache 攻击中密钥分析的核心表达式,从而论证了 LBlock 算法 Cache 计时攻击的可行性,为其他轻量级密码算法的实现安全性分析提供有价值的参考借鉴。

参考文献:

[1] WU W L, ZHANG L. LBlock: A Lightweight Block Cipher [C]//Proceedings of the 9th International Conference on Applied Cryptography and Network Security. Berlin, Germany: Springer-Verlag, 2011: 327-344.

[2] 吴文玲,范伟杰,张蕾. 轻量级分组密码研究进展[J]. 中国密码学发展报告,2010,140:159.

[3] MATSUI M. Linear cryptanalysis method for DES cipher [C]//Advances in Cryptology-EUROCRYPT' 93. Springer Berlin Heidelberg, 1994: 386-397.

[4] BIHAM E, SHAMIR A. Differential cryptanalysis of the data encryption standard [C]//Advances in Cryptology-CRYPTO 1990, LNCS 537, 1991: 2-21.

[5] DINUR I, SHAMIR A. Cube attacks on tweakable black box polynomials [C]//Advances. in Cryptology-EUROCRYPT 2009. Springer Berlin Heidelberg, 2009: 278-299.

[6] ALBRECHT M. Alorithmic algebraic techniques and their application to block cipher cryptanalysis [D]. Royal Holloway: University of London, 2010.

[7] 郭世泽,王韬,赵新杰. 密码旁路分析原理与方法[M]. 北京:科学出版社,2014.

[8] LI Z Q, ZHANG B, YAO Y, et al. Cube Cryptanalysis of LBlock with Noisy Leakage [C]//Proceedings of the 15th International Conference on Information Security and Cryptology. Berlin, Germany: Springer-Verlag, 2012: 141-155.

(下转第 180 页)

软件显示的结果中可以查看工期频率分布直方图、累计频率分布图、统计表等。在累计频率图下端的文本框中输入期望的工期,也可以移动标尺坐标选择,即可得到按此工期完工的概率。1#因素影响下关键路经工期处于期望值附近的完成概率为39.52%,2#因素影响下关键路经工期处于期望值附近的完成概率为49.47%,显然,因为1#因素和2#因素的影响,极大降低了生产按期完成的概率,而且1#因素比2#因素对关键路径工期的影响更大。

4 结论

本文提出基于计划评审技术(PERT)的不确定因素影响下的生产进度评估方法。该方法将工作持续时间分解为理想工作计划时间与不确定因素导致的延误时间,对影响生产进度的不确定因素进行识别,将不确定因素对工作造成的延误时间看作符合β分布的随机变量,对其进行三时估计,二者相加后继续经典计划评审技术生产进度评估流程后续步骤。实例表明,运用此方法并借助软件仿真获得不确定因素影响下的各路径工期期望、方差及按时完工概率等参数,可以量化不确定因素对生产进度的影响大小,为生产进度控制提供了依据。

参考文献:

[1] 李阳,于海山,沈琴,等.改进的PERT项目工期估算方法[J].工业工程与管理,2007(4):38-42.

[2] 刘武,杜志达,刘祥瞻.PERT网络活动时间参数估计的改进[J].统计与决策,2008(4):150-153.

[3] 王涛,蔡建锋.改进的PERT项目工期方差的估算方法[J].工业工程与管理,2012,17(1):36-39.

[4] 周华任,赵颖,周生.运筹与优化[M].北京:清华大学出版社,2012.

[5] 张怀强,李积源,等.武器系统研制进度风险分析方法研究[J].海军工程大学学报,2000,19(2):93-96.

[6] 严武,程振源,李海东.风险统计与决策分析[M].北京:经济管理出版社,1999.

[7] 陈志诚,齐欢,狄鹏.基于蒙特卡罗仿真的风险评估系统设计与实现[J].项目管理技术,2010,8(7):64-67.

(责任编辑 杨继森)

(上接第150页)

[9] BOGDAOV A,KUNDSEN L R,LEANDER G,et al. PRESENT:An Ultra-lightweight Block Cipher[C]//Proceedings of the 9th International Workshop on Cryptographic Hardware and Embedded Systems. Berlin, Germany: Springer-Verlag,2007:450-466.

[10] KOCHER P. Timing attacks on implementations of Diffie-Hellman,RSA,DSS,and other systems[C]//Advances in Cryptology-CRYPTO 1996,LNCS 1109,1996:104-113.

[11] KELSEY J,SCHNEIER B,WAGNER D,et al. Side channel cryptanalysis of product ciphers[C]//Peoceeding of the 5th European Symposium on Research in Computer Security-ESORICS 1998,LNCS 1485,1998:97-110.

[12] 温巧燕,钮心忻,杨义先.现代密码学中的布尔函数[M].北京:科学出版社,2000.

[13] 谷晓辰,丁文霞. 基于混沌Lorenz系统的S盒设计[J].重庆理工大学学报(自然科学),2013(3):97-103.

[14] 朱嘉良,韦永壮.针对LBlock算法的踪迹驱动Cache攻击[J].计算机工程,2015,47(5):153-158.

[15] 彭昌勇,祝跃飞,顾纯祥,等.1~5轮LBlock的多项式表示及完全性分析[J].计算机工程,2012,38(9):155-157.

(责任编辑 杨继森)