ITUbee密码代数旁路攻击

李 浪[1,2] 杜国权[1]

(衡阳师范学院计算机科学与技术学院 衡阳 421002)[1] (湖南大学信息科学与工程学院 长沙 410082)[2]

摘 要 ITUbee是在2013年第二届轻量级加密安全与隐私国际研讨会上提出的轻量级密码算法,对ITUbee密码进行安全分析有着积极意义。研究了ITUbee的代数旁路攻击方法,首先构建ITUbee密码S盒的等价代数方程组;由于构造的方程组不易解,通过采集ITUbee算法的加密功耗泄露,对加密中间状态字节的汉明重进行推断,并将其转化为与密码算法联立的布尔方程组,再利用cryptominisat解析器来求解密钥。实验结果表明,按此思路构造的ITUbee攻击方法所需样本少;在已知明文和未知明密文的场景下,1次ITUbee加密、部分轮汉明重泄露的情况下可成功恢复全部初始密钥。

关键词 ITUbee,代数旁路攻击,汉明重,Cryptominisat

中图法分类号 TP309 **文献标识码** A **DOI** 10.11896/j.issn.1002-137X.2016.2.037

Algebraic Side-channel Attacks Method of ITUbee

LI Lang[1,2] DU Guo-quan[1]

(College of Computer Science and Technology, Hengyang Normal University, Hengyang 421002, China)[1]

(College of Computer Science and Electronic Engineering, Hunan University, Changsha 410082, China)[2]

Abstract ITUbee was proposed in the second lightweight cryptography for security and privacy 2013. It has great significance to do security analysis about ITUbee. The algebraic side-channel attacks methods of ITUbee were researched. First, we constructed the equivalent-algebraic equations of ITUbee S-box. But, it is difficult to work out the structured equations set. The leakage of cryptographic power consumption of ITUbee algorithm was collected. The Hamming weight of the encryption middle status byte was inferred. Then, the simultaneous Boolean equations set with the cipher algorithm was conversed. At last, we used the cryptominisat to solve the key. Experiment results show that it only needs less samples to gain the successful attack. The initial keys can be derived via analyzing the part HW (Hamming weight) leakages of the first round in the scene of the known-plaintext and the unknown ciphertext.

Keywords ITUbee, Algebraic side-channel attack, Hamming weight, Cryptominisat

1 引言

在第五届信息安全与密码学国际会议(INSCRYPT 2009)上,Renauld等人引入一类新的攻击,该类攻击被称为代数旁路攻击,它是经典代数攻击和旁路攻击的结合[1,2]。旁路攻击的重点是采集加密操作过程中设备运行的物理信息泄漏[3],这些泄露信息通过使用的物理测量识别关键字节进行采集。代数旁路攻击的出现弥补了传统旁路攻击所需样本量大、旁路信息利用率低的缺陷,提高了代数攻击求解方程组的速率,降低了求解方程组的复杂度。代数旁路攻击由密码算法和系统泄漏信息组成代数方程组进行求解,求方程组的解相当于恢复密码算法的密钥。然而,找到一个解决方案是不容易的,我们主要关注的是SAT解析器求解。在过去几十年里,SAT被用来解决该领域大量的研究问题,是目前解决代数攻击最快捷的技术,将分组密码和泄漏模型方程组转化为一个可满足性问题后利用SAT求解器求解。代数旁路攻击对密码算法具有极大威胁,PRESENT在2011年、AES算法在2009年均被成功进行过代数旁路攻击,还有轮密钥间相关性很弱的SMS4、LBlock算法以及没有密钥扩展的LED算法在2013年进行过相同攻击[4-8]。代数旁路攻击被提出后,其相关研究对于优化密码算法、提高安全性具有重要意义,因此代数旁路攻击引起了密码学者的广泛关注。

ITUbee密码算法是在第二届轻量级加密安全与隐私国际研讨会(LightSec 2013)上提出的,是一种面向软件的轻量级加密算法[9]。与传统分组密码相比,ITUbee具有执行效率更高、计算资源消耗更少、更适合普适计算等资源受限环境的优点;同其他轻型分组密码相比,ITUbee具有算法没有密钥扩展、低功耗、更少的内存要求等特点,是轻量级密码算法的佼佼者。目前对ITUbee安全分析的相关研究较少,在代数旁路攻击方面尚未有公开发表的结果。

论文基于布尔理论构造了简洁的ITUbee算法代数方程表达式;通过采集ITUbee在微控制器运行过程中泄露的信

到稿日期:2015-02-01 返修日期:2015-04-18 本文受国家自然科学基金资助项目(61572174),湖南省自然科学基金资助项目(2015JJ4011),湖南省教育厅资助科研重点项目(15A029)资助。

李 浪(1971—),男,博士,教授,硕士生导师,主要研究领域为嵌入式计算与信息安全,E-mail:lilang911@126.com;杜国权 女,主要研究领域为信息安全。

息，结合汉明重模型将其转化成密码中间状态汉明重值，并转化为相应代数方程组，同 ITUbee 密码算法代数方程组联立求解恢复密钥。实验结果表明：ITUbee 容易受到代数旁路攻击的威胁，在已知明文与未知密钥的情况下，且部分轮数的汉明重泄露的情况下可恢复完整的初始密钥。

2 ITUbee 算法

ITUbee 算法采用经典 Feistel 结构，分组长度为 80 位，密钥长度为 80 位，整个加密由 20 轮组成。为了提高加密速度并减小硬件实现面积，ITUbee 采用了无密钥生成的策略。图 1 为 ITUbee 算法运算结构图。

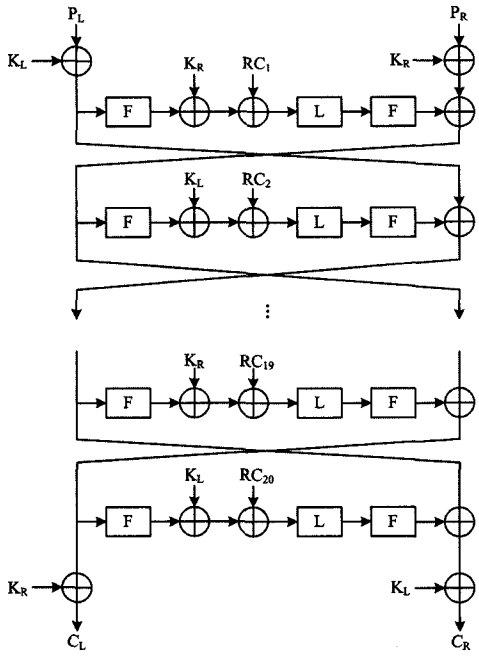

图 1 ITUbee 算法结构图

从图 1 可知，轮变换包括 F 函数、轮密钥加、常数加、L 函数。轮变换过程如图 2 所示，其中模块(1)和模块(2)都为 F 函数模块，并且模块(2)复用模块(1)。

ITUbee 轮变化描述如下：

1)轮函数 F

轮函数 F 是一个将 40bit 输入进行非线性变换的函数，由非线性替换函数 S、线性代换函数 L、非线性替换函数 S 复合而成，即

$F(X)=S(L(S(X)))$

2)S 盒

在 ITUbee 算法中唯一的非线性部分是 S 盒操作。分枝数的线性层 F 为 4，这意味着一个 F 运算时至少 4 个 S 盒是活跃的。加密一轮，如果左半部分的输入是活跃的，那么将有 2 个活性的 F 函数。在 Feistel 结构中如果产生的输出对应到右半部分的功能是一对一的，则左半部的 2 次输入会导致至少连续 3 轮有差异。在密码运算中，这个函数是一一对应的，所以 16 个 S 盒连续活跃 3 轮，至少有 4 个活性的 F 函数。

3)L 函数

L 函数主要满足如下运算规则：

$L(a\|b\|c\|d\|e)=(e\oplus a\oplus b)\|(a\oplus b\oplus c)\|(b\oplus c\oplus d)\|(c\oplus d\oplus e)\|(d\oplus e\oplus a)$

4)轮密钥加

轮密钥的选取规则描述如下：

```
for i= 1,…,20 do
  if i∈{1,3,…,19}
  RK←K_R
else
  RK←K_L
```

5)轮常数加

在 ITUbee 算法中，每轮运行一个轮常数($RC[i]$)。其中 $RC[i]$如表 1 所列。

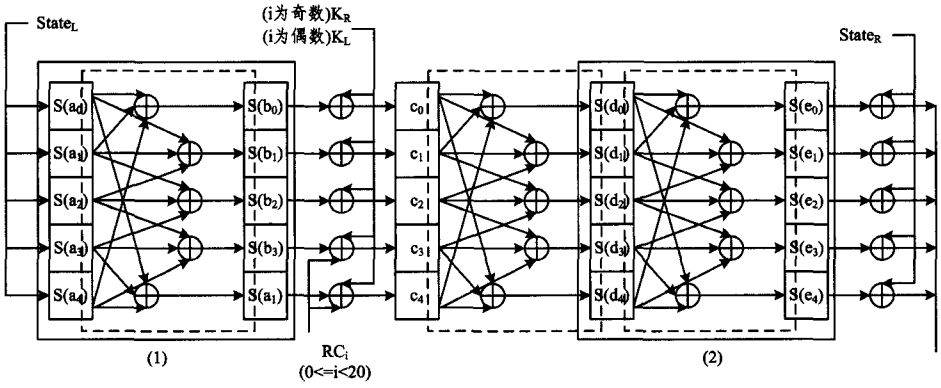

图 2 轮变化过程

表 1 轮常数 $RC[i]$

i	RC_i	i	RC_i	i	RC_i	i	RC_i
1	1428	6	0f23	11	0a1e	16	0519
2	1327	7	0e22	12	091d	17	0418
3	1226	8	0d21	13	081c	18	0317
4	1125	9	0c20	14	071b	19	0216
5	1024	10	0b1f	15	061a	20	0115

3 ITUbee 代数旁路攻击

在代数旁路攻击中，通过获得泄露信息如汉明重量的中间值建立多项式系统，这些泄露信息的存在可能会引起一些独立的线性关系，因此可以进行一个有效的代数攻击[10]。

代数旁路攻击分为代数方程组的构建、旁路泄漏信息采集及利用和求解代数方程组 3 个步骤。

1)密码代数方程组的构建

第一步的关键是用一组多项式方程来描述目标系统，得到密钥的关键是求解系统方程。为了达到这个目的，将寻找一组涉及明文 P、密文 C 和密钥 K 这些变量的方程来描述 ITUbee。最简单的解决方案是建立一个系统形式的方程。在密码算法代数方程组的构建中，关键是如何构造非线性部分 S 盒的代数方程。

2)获取和利用旁路泄漏

由于直接求解代数方程组是一个 N-P 难题，因此需要利用算法加密操作字节中间状态相关的汉明重泄露信息建立额外的方程组来加速求解方程组。

3)求解代数方程组

将汉明重表示的代数方程组同密码算法方程组联立后，密钥恢复攻击等价于方程组的求解问题。求解代数方程组通常包括基于 Gröbner 基方法和线性化方法、SAT 问题求解。本文主要基于 SAT 解析器来进行代数方程求解，其中解析器选择常用的 Cryptominisat。

3.1 ITUbee 代数攻击方程构建

S 盒是 ITUbee 算法的唯一非线性部分，它的代数方程构造是重难点。假设 S 盒输入为$(x_0,x_1,x_2,x_3,x_4,x_5,x_6,x_7)$，输出为$(y_0,y_1,y_2,y_3,y_4,y_5,y_6,y_7)$，有以下定义。

定义 1 在有限域 $GF(q^n)$上，$x_i\in GF(q)$和域元素 x 之间的关系为：$x_i=\sum_{j=0}^{n-1}a_j x^{q^j}$。

定义 2 有限域的元素 $\forall x=(x_{n-1},x_{n-2},\cdots,x_0)=\sum_{i=0}^{n-1}x_i a^i$，$x_i\in GF(q)$，$i=0,1,2,\cdots,n-1$，$x_i$ 在 X 的函数表达式 $x_i=\sum_{j=0}^{n-1}a_j x^{q^j}$，$a_j\in GF(q^n)$的系数为 0 或 1。

S 盒的变换可用下列 $GF(2^8)$上的二次方程来描述：

$$\begin{cases}x_0y_0=1\\(y_m)^2=y_{m+1},\quad 0\leqslant m\leqslant 7\end{cases}\tag{1}$$

其中，$m+1$ 为模 8 加。上述方程(1)假定 S 盒的输入加密过程中不出现零元，这个假设在加密过程中超过一半的概率是正确的，即使假设是无效的，上述方程也只有第一个错误。从方程(1)可知，S 盒有 9 个二次方程，方程量太少不够超定(XSL 技术要求描述 S 盒的二次方程组是超定的)，可以在不增加变量的情况下增加如下二次方程：

$$x_0y_1=y_0\tag{2}$$

通过假设$(x_0)^2=x_1$ 和$(x_1)^2=x_2$ 来增加两个变量 x_1，x_2，以得到代数方程组。

Mathematica 科学计算软件重要的特征之一是它不仅能做数值计算，还能进行符号运算。该软件内嵌强大的符号计算功能，但其在密码学研究领域的应用并不多见。本文利用 Mathematica 软件来验证 ITUbee 的代数攻击方程组的准确性。

在构建代数方程时，利用 Mathematica 软件中的符号计算功能。这一工具使列方程的复杂度大为降低，只要将算法描述出来，即可得到该算法的代数方程组。

根据定义，利用 Mathematica 软件可以计算获得 ITUbee 算法 S 盒的布尔方程组。ITUbee 算法 S 盒的布尔方程组构造如下[11-14]。

$x_0y_2+x_0y_5+x_0y_7+x_1y_1+x_1y_4+x_1y_6+x_2y_0+x_2y_3+x_2y_5+x_3y_2+x_3y_4+x_3y_7+x_4y_1+x_4y_3+x_4y_6+x_5y_0+x_5y_1+x_5y_2+x_5y_4+x_5y_5+x_5y_6+x_6y_0+x_6y_3+x_6y_5+x_6y_6+x_6y_7+x_7y_2+x_7y_4+x_7y_5+x_7y_6+x_7y_7+x_0+x_6=1$

$x_0y_0+x_0y_3+x_0y_6+x_1y_1+x_1y_2+x_1y_4+x_1y_5+x_1y_6+x_1y_7+x_2y_0+x_2y_1+x_2y_3+x_2y_4+x_2y_5+x_2y_6+x_3y_0+x_3y_2+x_3y_3+x_3y_4+x_3y_5+x_3y_7+x_4y_1+x_4y_2+x_4y_3+x_4y_4+x_4y_6+x_5y_0+x_5y_2+x_5y_0+x_5y_2+x_5y_3+x_5y_4+x_5y_5+x_6y_1+x_6y_2+x_6y_3+x_6y_4+x_6x_7+x_7y_0+x_7y_2+x_7y_3+x_7y_4+x_1+x_6+x_7=0$

$x_0y_1+x_0y_4+x_0y_7+x_1y_0+x_1y_3+x_1y_6+x_2y_1+x_2y_2+x_2y_4+x_2y_5+x_2y_6+x_2y_7+x_3y_0+x_3y_1+x_3y_3+x_3y_4+x_3y_5+x_3y_6+x_4y_0+x_4y_2+x_4y_3+x_4y_4+x_4y_5+x_4y_7+x_5y_1+x_5y_2+x_5y_3+x_5y_4+x_5y_6+x_5y_7+x_6y_0+x_6y_2+x_6y_3+x_6y_4+x_6y_5+x_7y_1+x_7y_2+x_7y_3+x_7y_4+x_7y_7+x_0+x_2+x_7=0$

$x_0y_0+x_0y_2+x_0y_5+x_1y_6+x_1y_7+x_2y_5+x_2y_6+x_3y_1+x_3y_5+x_3y_6+x_4y_0+x_4y_4+x_4y_5+x_5y_1+x_5y_3+x_5y_6+x_5y_7+x_6y_0+x_6y_1+x_6y_2+x_6y_4+x_6y_5+x_7y_0+x_7y_3+x_7y_6+x_7y_7+x_1+x_3+x_6=0$

$x_0y_1+x_0y_3+x_0y_6+x_1y_0+x_1y_1+x_1y_2+x_1y_4+x_1y_5+x_1y_6+x_2y_0+x_2y_3+x_2y_5+x_2y_6+x_2y_7+x_3y_2+x_3y_4+x_3y_5+x_3y_6+x_3y_7+x_4y_3+x_4y_5+x_5y_1+x_5y_2+x_5y_6+x_6y_0+x_6y_1+x_6y_5+x_7y_0+x_7y_1+x_7y_6+x_7y_7+x_2+x_4+x_6+x_7=0$

$x_0y_2+x_0y_4+x_0y_7+x_1y_1+x_1y_3+x_1y_6+x_2y_0+x_2y_1+x_2y_2+x_2y_4+x_2y_5+x_2y_6+x_3y_0+x_3y_3+x_3y_5+x_3y_6+x_3y_7+x_4y_2+x_4y_4+x_4y_5+x_4y_6+x_4y_7+x_5y_3+x_5y_5+x_6y_1+x_6y_2+x_6y_6+x_7y_0+x_7y_1+x_7y_5+x_3+x_5+x_7=0$

$x_0y_0+x_0y_3+x_0y_5+x_1y_2+x_1y_4+x_1y_7+x_2y_1+x_2y_3+x_2y_6+x_3y_0+x_3y_1+x_3y_2+x_3y_4+x_3y_5+x_3y_6+x_4y_0+x_4y_3+x_4y_5+x_4y_6+x_4y_7+x_5y_2+x_5y_4+x_5y_5+x_5y_6+x_5y_7+x_6y_3+x_6y_5+x_7y_1+x_7y_2+x_7y_6+x_4+x_6=0$

$x_0y_1+x_0y_4+x_0y_6+x_1y_0+x_1y_3+x_1y_5+x_2y_2+x_2y_4+x_2y_7+x_3y_1+x_3y_3+x_3y_6+x_4y_0+x_4y_1+x_4y_2+x_4y_4+x_4y_5+x_4y_6+x_5y_0+x_5y_3+x_5y_5+x_5y_6+x_5y_7+x_6y_2+x_6y_4+x_6y_5+x_6y_6+x_6y_7+x_7y_3+x_7y_3+x_7y_5+x_5+x_7=0$

$x_0y_2+x_0y_3+x_0y_5+x_1y_2+x_1y_4+x_1y_5+x_2y_2+x_2y_5+x_2y_7+x_3y_0+x_3y_4+x_3y_6+x_3y_7+x_4y_1+x_4y_3+x_4y_5+x_5y_0+x_5y_3+x_5y_7+x_6y_0+x_6y_3+x_6y_7+x_7y_0+x_7y_3+x_7y_6+x_3+x_6=1$

$x_0y_0+x_0y_2+x_0y_6+x_1y_0+x_1y_2+x_1y_5+x_2y_1+x_2y_3+x_2y_6+x_3y_0+x_3y_1+x_3y_2+x_3y_4+x_3y_5+x_3y_6+x_3y_7+x_4y_0+x_4y_4+x_4y_5+x_5y_0+x_5y_2+x_5y_5+x_6y_4+x_6y_6+x_7y_1+x_7y_4+x_7y_6+x_0+x_6=1$

$x_0y_1+x_0y_3+x_0y_7+x_1y_0+x_1y_2+x_1y_6+x_2y_0+x_2y_1+x_2y_3+x_2y_4+x_2y_5+x_2y_6+x_2y_7+x_3y_1+x_3y_3+x_3y_5+x_4y_1+x_4y_3+x_4y_6+x_5y_1+x_5y_3+x_5y_5+x_5y_6+x_6y_4+x_6y_5+x_7y_3+x_7y_4+x_7y_5+x_3+x_5+x_6=1$

$x_0y_1+x_0y_4+x_0y_5+x_1y_0+x_1y_3+x_1y_4+x_2y_1+x_2y_3+x_2y_5+x_3y_0+x_3y_3+x_3y_4+x_4y_0+x_4y_3+x_4y_5+x_4y_6+x_5y_2+x_5y_3+x_5y_6+x_5y_7+x_6y_0+x_6y_1+x_6y_5+x_7y_2+x_7y_5+x_7y_6+x_3+x_4=1$

半字节的 S 盒可以由 $GF(2)$上 21 个二次方程描述，任何半字节 S 盒可由至少 21 个这样的方程来描述。整个 ITUbee 密码算法可以通过 $E=n\times 21$ 个二次方程构造，二次方程由 $V=n\times 8$个变量描述，其中 n 是在 ITUbee 加密算法中使用的 S 盒数量。在 ITUbee 算法中：$n=20\times 16$，因此整个系统由 6720 个二次方程、2560 个变量组成，这是一个典型的超定多变元高次方程组，即方程数多于变量数。在密钥求解时，求解一个这样的超定多变量二次方程组问题是 NP Hard，复杂度

是关于 n 的指数。因此，为了快速有效地求解上述方程组，利用 ITUbee 在运算时泄露的信息再构造一个方程组，基于汉明重泄露的方程组构建可以有效加速上述方程组的求解，从而高效破解密钥。

3.2 汉明重的提取及利用

1)功耗泄露信息采集

实验过程为在 8 位 AVR 微控制器上实现 ITUbee 加密程序，系统晶振为 20MHz。为能够方便测量 ITUbee 加密运算时的功耗变化，在微控制器和接地端(GND)之间串联一个 18Ω 电阻进行信号放大并采集，通过 USB 数据线将采集到的功耗轨迹传到上位 PC 机处理。实际操作中，电压设置为 5V，微控制器为 8MHz，示波器采样频率为 100MS/s。

ITUbee 算法的汉明重泄露点选取原理如图 3 所示，每轮采集 28 个功耗点，整个加密过程共有 20×28+16=576 个功耗点。由于环境噪声及测试计量仪器精度的影响，部分中间状态字节的汉明重推断值可能会存在误差，因此对同一明文操作采集 3 次并求平均。同时 SAT 求解器对于输入错误非常敏感，1 bit 输入错误可能导致解析器无解，所以 ITUbee 20 轮加密过程中的汉明重信息只有部分可以利用，相当于从全部 20 轮中离散选取汉明重消息[15-19]。

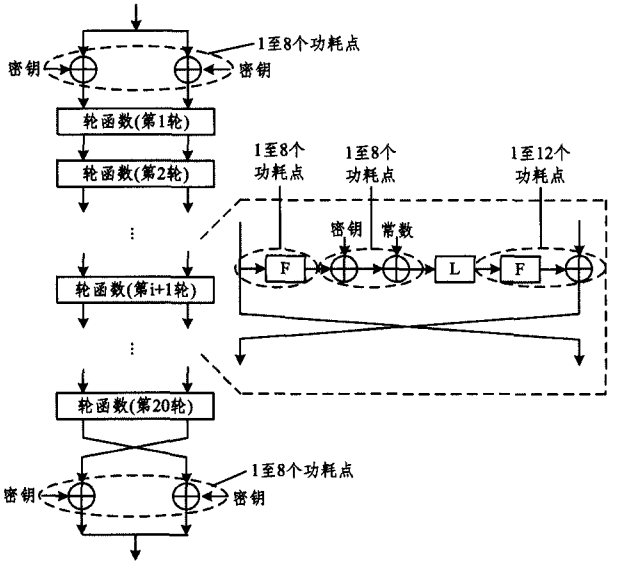

图 3 ITUbee 汉明重泄露点选取

2)汉明重推断

假设得到的汉明重是连续的，因此要找到一个最大信息集的多项式方程来描述汉明重；如果假定的汉明重是不连续的，则要选择包括所有汉明重的类。实验均在理想状况下运行，即得到的汉明重泄露模型能正确描述设备的功耗。

密码算法运行过程中泄露的功耗信息和操作数的汉明重之间具有很强的关联性。因此，通过采集到的功耗信息可推断加密过程中各操作对应中间状态字节的汉明重。为推断 ITUbee 加密各个字节的汉明重，首先需要建立各个汉明重量值对应的功耗轨迹模板，接着利用模板匹配的方式进行汉明重推断。在模板分析法中的模板搭建阶段，须根据已知中间状态字节的汉明重和对应功耗轨迹，为 9 个不同汉明重分别构建由均值向量和噪声向量方差矩阵组成的二元组$\langle \overline{t_i}, G_i \rangle$(0≤$i$≤8)，其中$\overline{t_i}$为均值向量；$G_i$ 为噪声协方差矩阵。模板搭建完成后，待匹配向量 t'和搭建模板间的极大似然度为：

$$p(t';\langle \overline{t_i}, G_i \rangle)=\frac{1}{\sqrt{(2\pi)^m |G_i|}}\exp\left(-\frac{1}{2}(t'-\overline{t_i})^{\mathrm{T}} G_i^{-1}(t'-\overline{t_i})\right)$$

根据极大似然法则，$p(t';\langle \overline{t_i}, G_i \rangle)$最大值对应的汉明重即为该中间状态字节汉明重推断值。

3)汉明重的布尔函数表示

设 n 为正整数，对任意的$(x_1, x_2, \cdots, x_n) \in GF(2)$和 0≤$k$≤$n$−1，定义 $p_n^k(k)=(p_n^k(x_1), p_n^k(x_2), \cdots, p_n^k(x_n))$，其中 $p_n^k(x_i)=\begin{cases} x_{i+k}, & i+k\leqslant n \\ x_{i+k-n}, & i+k>n \end{cases}$。如果对任意的$(x_1, x_2, \cdots, x_n) \in GF(2)$都有 $f(p_n^k(x))=f(x)$，0≤k≤n−1，则称 $f(x)$为旋转对称布尔函数。易知，$\{p_n^k \mid 0\leqslant k\leqslant n-1\}$是循环群。令 $G_n(x)=\{p_n^k \mid 0\leqslant k\leqslant n-1\}$表示在该循环群作用下由向量 x 生成的轨道，定义轨道的重量为向量 x 的汉明重量。若

$$f(x)=\begin{cases} 0, & W_H(x)<n/2 \\ b_x\in\{0,1\}, & W_H(x)=n/2 \\ 1, & W_H(x)>n/2 \end{cases}$$

则称它为择多函数。

以下为汉明重布尔方程构造方法：

(1)取定 $f(x)$为任一 n 元择多函数。

(2)任取 $GF^n(2)$中一个重量为 $n/2-1$ 的向量 $x^{(0)}$，生成 $G_n(x^{(0)})$。

(3)取 $GF^n(2)$中一个重量为 $n/2+1$ 的向量 $y^{(0)}$，生成 $G_n(y^{(0)})$，使得对每一个$\hat{x} \in G_n(x^{(0)})$，都存在唯一的$\hat{y} \in G_n(y^{(0)})$，满足 $WS(\hat{x}) \subset WS(\hat{y})$。

(4)构造布尔函数 $F(x)$：

$$F(x)=\begin{cases} f(x)+1, & \text{如果 } x\in G_n(x^{(0)}) \cup G_n(y^{(0)}), \\ f(x), & \text{其它} \end{cases}$$

当 n=6 时，不存在满足上述方法(3)。然而，对任意的 n≥8，都能找到满足上述方法(3)。事实上，取 $GF^n(2)$中一个重量为 $n/2-1$ 的向量 $x^{(0)}$，使得 $WS(x^{(0)})=\{i: 1\leqslant i\leqslant n/2-1\}$，再取 $GF^n(2)$中一个重量为 $n/2+1$ 的向量 $y^{(0)}$，使得 $WS(y^{(0)})=WS(x^{(0)}) \cup \{n/2+1, n/2+2\}$。当 n≥8 时，$|G_n(x^{(0)})|=|G_n(y^{(0)})|=n$，且对任意的$\hat{x} \in G_n(x^{(0)})$，都存在唯一的$\hat{y} \in G_n(y^{(0)})$，使得 $WS(\hat{x}) \subset WS(\hat{y})$。因此，布尔函数 $f(x)$的变元个数大于或等于 8。

对于长度为 n 的二值向量 $X=(x_1, x_2, \cdots, x_n)$，其可能的汉明重值有 $n+1$ 个；可用长度为 k 的二值向量 $h=(h_1, h_2, \cdots h_n)$，对其汉明重 $HW(X)$表示如下：

$$h_l=\sum_{t=1}^{p} a_t x_{i_1} x_{i_2} \cdots x_{i_q} \quad (1\leqslant i_1<i_2<\cdots<i_q\leqslant n)$$

其中，$p=\binom{n}{2^{k-l}}$，$q=2^{k-l}$，$k=[\log_2^{n+1}]$，1≤l≤k。对于 8bit 的中间状态字节 X，有如表 2 所列的 9 种可能的汉明重 $HW(X)$，且 0≤$HW(X)$≤8。其汉明重 $HW(X)$可以用 4bit 二值向量 $h=(h_1, h_2, h_3, h_4)$进行表示：

$$\begin{cases} h_0 = \prod_{i=1}^{8} x_i \\ h_1 = \sum_{t=1}^{70} a_t x_i x_j x_m x_n \ (1 \leqslant i \leqslant j < m < n \leqslant 8) \\ h_2 = \sum_{t=1}^{28} a_t x_i x_j \ (1 \leqslant i < j \leqslant 8) \\ h_3 = \sum_{i=1}^{8} x_i \end{cases}$$

表 2　中间状态不同汉明重推断极大似然值

中间状态	最大似然值									推断值
	0	1	2	3	4	5	6	7	8	
X_0^2	0.15	0.32	0.48	0.51	0.63	0.77	0.64	0.57	0.43	5
X_1^2	0.63	0.79	0.60	0.49	0.41	0.34	0.27	0.21	0.15	1
X_0^3	0.57	0.63	0.67	0.74	0.57	0.51	0.43	0.39	0.21	3
X_1^3	0.18	0.35	0.41	0.54	0.63	0.68	0.66	0.51	0.41	5
X_0^4	0.54	0.63	0.69	0.51	0.44	0.37	0.36	0.26	0.15	2
X_1^4	0.27	0.41	0.52	0.58	0.67	0.54	0.43	0.37	0.21	4
X_0^5	0.53	0.61	0.59	0.52	0.45	0.38	0.33	0.21	0.16	1
X_1^5	0.48	0.55	0.59	0.51	0.44	0.39	0.31	0.22	0.18	2

3.3　基于 SAT 求解的密钥恢复

由于直接进行 ITUbee 密码代数方程组求解是一个 NP 难问题，因此利用旁路泄露信息构建联立方程组来加速求解。将旁路信息表示成方程并同密码代数方程联立后，密钥恢复攻击等价于方程组求解问题，采用 Cryptominisat 解析器求解[20,21]。实际攻击中首先将布尔方程组通过变量与索引值的代换，生成符合 Cryptominisat 输入格式的文本文件，然后以命令行方式调用 crypt. exe 程序进行求解，结果如表 3 所列。表 3 中编号 2－81 一次表示为 128bits 的初始密钥比特值。编号为正表示对应密钥为 1，否则为 0。根据表 3 求解的密钥为 11000101001110001011110110010010100010011000001000101011111001000011001101100011，将其转换为 16 进制即 0xc538bd9289822be43363，这同真实密钥一致，故攻击成功。

表 3　实例中 Cryptominisat 的求解结果

编号	K	编号	K	编号	K	编号	K
2	1	22	1	42	1	−62	0
3	1	23	1	−43	0	63	1
−4	0	−24	0	−44	0	−64	0
−5	0	25	1	−45	0	−65	0
−6	0	26	1	−46	0	−66	0
7	1	−27	0	−47	0	−67	0
−8	0	−28	0	48	1	68	1
9	1	29	1	−49	0	69	1
−10	0	−30	0	−50	0	−70	0
−11	0	−31	0	−51	0	−71	0
12	1	32	1	52	1	72	1
13	1	−33	0	−53	0	73	1
14	1	34	1	54	1	−74	0
−15	0	−35	0	−55	0	75	1
−16	0	−36	0	56	1	76	1
−17	0	−37	0	57	1	−77	0
18	1	38	1	58	1	−78	0
−19	0	−39	0	59	1	−79	0
20	1	−40	0	60	1	80	1
21	1	41	1	−61	0	81	1

ITUbee 密码能被代数旁路攻击成功，在已知明文攻击场景下，1 条 ITUbee 加密过程功耗轨迹中对应的部分轮汉明重泄露即可恢复全部初始密钥，在汉明重推断存在部分错误时也能成功实施攻击。图 4 为不同错误率下所需的攻击时间，图 5 为不同错误率、不同轮数泄露时所需的攻击时间。随着错误率的增加，解析时间呈增长趋势；错误率相同时，泄露轮数越多，所需时间也越长。

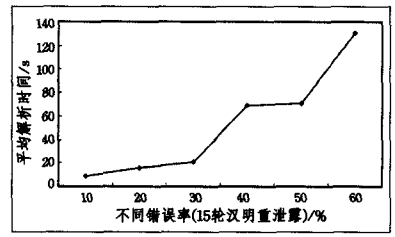

图 4　不同错误率下的攻击时间

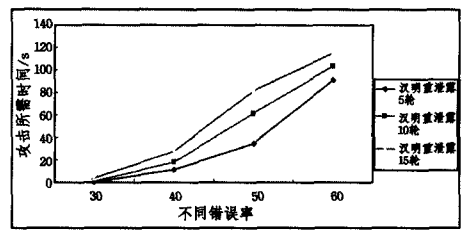

图 5　不同错误率、不同轮数泄露时所需的攻击时间

结束语　由于 ITUbee 是在 LightSec 2013 国际会议上提出的轻量级密码算法，对新的密码算法进行充分的安全分析与评估有助于算法更好地应用于实际中。本文给出了一种 ITUbee 密码算法的代数旁路攻击方法，采用了 ITUbee 运行时泄漏的旁路信息建立方程组，该方程组联合 ITUbee 代数方程组可以快速恢复密钥。实验结果证明未加防护的 ITUbee 能被代数旁路攻击成功，且攻击所需样本量小。

参 考 文 献

[1] Renauld M, Standaert F X. Algebraic side-channel attacks[C]// Proceedings of Information Security and Cryptology. Heidelberg: Springer Berlin, 2009: 393-410

[2] Renauld M, Standaert F-X. Representation, leakage and cipher-dependencies in algebraic side-channel attacks[C]// Proceedings of Industrial Track of ACNS. Heidelberg: Springer Berlin, 2010: 1-18

[3] Brier E, Clavier C, Olivier F. Correlation power analysis with a leakage model[C]// Proceedings of Cryptography Hardware and Embedded Systems. Heidelberg: Springer Berlin, 2004: 16-29

[4] Renauld M, Standaert F X. Algebraic side-channel attacks on the AES: Why time also matters in DPA[C]// Proceedings of cryptography Hardware and Embedded Systems 2009. Heidelberg: Springer Berlin, 2009: 97-111

[5] Mohamed M S E, Bulygin S, Zohner M, et al. Improved algebraic side channel attack on AES[J]. Journal of Cryptographic Engineering, 2013, 3(3): 139-156

[6] Schramm K, Leander G, Felke P, et al. A collision-attack on AES combining side channel and differential attack[C]// Proceedings of Cryptography Hardware and Embedded Systems. Heidelberg: Springer Berlin, 2004: 163-175

[7] Liu Hui-ying, Zhao Xin-jie, Wang Tao, et al. Research on Hamming Weight-based Algebraic Side-Channel attacks on SMS4 [J]. Chinese Journal of Computers, 2013, 36(6): 1183-1193(in Chinese)
刘会英，赵新杰，王韬，等. 基于汉明重 SMS4 密码代数旁路攻击研究[J]. 计算机学报，2013，36(6)：1183-1193

[8] Ji Ke-ke, Wang Tao, Guo Shi-ze, et al. Research of Hamming Weight-based algebraic side－channel attack on LED[J]. Journal on Communications, 2013, 34(7): 134-142(in Chinese)
冀可可，王韬，郭世泽，等. 基于汉明重的 LED 代数旁路攻击研

究[J]. 通信学报,2013,34(7):134-142

[9] Ferhat K. ITUbee: A Software Oriented Lightweight Block Cipher[C]// Proceedings of Lightweight Cryptography for Security and Privacy 2013. Heidelberg: Springer Berlin, 2013: 16-27

[10] Carlet C, Faugère J-C, Goyet C, et al. Analysis of the algebraic side channel attack[J]. Journal of Cryptographic Engineering, 2012, 2(1): 45-62

[11] Zhang Guo-ji, Xiao Huang-pei. Quadratic Equations on S-Boxes and a New S-Box Design Criterion[J]. Journal of South china University of Technology(Natural Science Edition), 2008, 36(8): 140-144(in Chinese)
张国基,肖黄培. S盒的二次方程及一个新的设计准则[J]. 华南理工大学学报,2008,36(8):140-144

[12] Fischer S, Meier W. Algebraic immunity of S-boxes and augmented functions[C]// Proceedings of Foundations of Software Engineering. Heidelberg: Springer Berlin, 2007: 366-381

[13] Armknecht F, Krause M. Constructing single and multi-output Boolean functions with maximal immunity[C]// Proceedings of International Colloquium on Automata, Languages and Programming. Heidelberg: Springer Berlin, 2006: 180-191

[14] Carlet C. On the algebraic immunities and higher order nonlinearities of vectorial Boolean functions[C]// Proceedings of NATO Science for Peace and Security Series, D: Information and Communication Security. Heidelberg: Springer Berlin, 2009: 104-116

[15] Oren Y, Kirschbaum M, Popp T, et al. Algebraic side-channel analysis in the presence of errors[C]// Proceedings of Cryptography Hardware and Embedded Systems. Heidelberg: Springer Berlin, 2010: 428-442

[16] Bogdanov A, Kizhvatov I, Pyshkin A. Algebraic Methods in Side-Channel Collision Attacks and Practical Collision Detection [C]// Proceedings of INDOCRYPT. Berlin: Springer, 2008: 251-265

[17] Moradi A, Mischke O, Eisenbarth T. Correlation-enhanced power analysis collision attack[C]// Proceedings of Cryptography Hardware and Embedded Systems. Heidelberg: Springer Berlin, 2010: 125-139

[18] Oren Y, Kirschbaum M, Popp T, et al. Algebraic side channel analysis in the presence of errors[C]// Proceedings of Cryptography Hardware and Embedded Systems. Heidelberg: Springer Berlin, 2010: 428-442

[19] Whitnall C, Oswald E, Mather L. An exploration of the kolmogorov-smirnov test as competitor to mutual information analysis [EB/OL]. [2011-03-08]. http://eprint.iacr.org/2011/380.pdf

[20] Knudsen L R, Miolance C V. Counting equations in algebraic attacks on block ciphers[J]. International Journal of Information Security, 2010, 9(2): 127-135

[21] Soos M, Nohl K, Castelluccia C. Extending SAT solvers to cryptographic problems[C]// Proceedings of Lecture Notes in Computer Science. Heidelberg: Springer Berlin, 2009: 244-257

(上接第 162 页)

参 考 文 献

[1] Song D X, Wagner D, Perrig A. Practical techniques for searches on encrypted data[C]// Proceedings IEEE Symposium on Security and Privacy. 2000: 44-55

[2] Boneh D, Di Crescenzo G, Ostrovsky R, et al. Public key encryption with keyword search[M]// Advances in Cryptology-Eurocrypt. Springer Berlin Heidelberg, 2004: 506-522

[3] Chen F, Liu A X. Privacy-and integrity-preserving range queries in sensor networks[J]. IEEE/ACM Transactions on Networking, 2012, 20(6): 1774-1787

[4] Shi E, Bethencourt J, Chan T H H, et al. Multi-dimensional range query over encrypted data[C]// IEEE Symposium on Security and Privacy(P'07). IEEE, 2007: 350-364

[5] Li Shuang, Xu Mao-zhi. Attribute-based public encryption with keyword search[J]. Chinese Journal of Computers, 2014, 37(5): 1017-1024(in Chinese)
李双,徐茂智. 基于属性的可搜索加密方案[J]. 计算机学报,2014,37(5):1017-1024

[6] Boneh D, Waters B. Conjunctive, subset, and range queries on encrypted data[M]// Theory of Cryptography. Springer Berlin Heidelberg, 2007: 535-554

[7] Li J, Wang Q, Wang C, et al. Fuzzy keyword search over encrypted data in cloud computing[C]// INFOCOM. IEEE, 2010: 1-5

[8] Yang Y, Lu H, Weng J. Multi-user private keyword search for cloud computing[C]// IEEE Third International Conference on Cloud Computing Technology and Science (CloudCom). IEEE, 2011: 264-271

[9] Bao F, Deng R H, Ding X, et al. Private query on encrypted data in multi-user settings[M]// Information Security Practice and Experience. Springer Berlin Heidelberg, 2008: 71-85

[10] Curtmola R, Garay J, Kamara S, et al. Searchable symmetric encryption: improved definitions and efficient constructions[C]// Proceedings of the 13th ACM Conference on Computer and Communications Security. ACM, 2006: 79-88

[11] Shen Zhi-rong, Xue Wei, Shu Ji-wu. Survey on the research and development of searchable encryption schemes[J]. Journal of Software, 2014, 25(4): 880-895(in Chinese)
沈志荣,薛巍,舒继武. 可搜索加密机制研究与进展[J]. 软件学报,2014,25(4):880-895

[12] Gu C, Zhu Y. New Efficient Searchable Encryption Schemes from Bilinear Pairings[J]. International Journal of Network Security, 2010, 10(1): 25-31

[13] Wang C, Cao N, Li J, et al. Secure ranked keyword search over encrypted cloud data[C]// IEEE 30th International Conference on Distributed Computing Systems(ICDCS). 2010: 253-262

[14] Fang Zhong-jin, Zhou Shu, Xia Zhi-hua. Research on fuzzy search over encrypted cloud data based on keywords[J]. Computer Science, 2015, 42(3): 136-139(in Chinese)
方忠进,周舒,夏志华. 基于关键字的加密云数据模糊搜索策略研究[J]. 计算机科学,2015,42(3):136-139

[15] Zheng Q, Xu S, Ateniese G. VABKS: Verifiable Attribute-based Keyword Search over Outsourced Encrypted Data[J]. IACR Cryptology ePrint Archive, 2013: 462

[16] Zheng Yu, Gao Bao. Method based on homomorphic hash for data changes capture of LDAP directory[J]. Application Research of Computers, 2013, 30(7): 2007-2009(in Chinese)
郑煜,高宝. 基于同态哈希的目录服务数据变化捕获方法[J]. 计算机应用研究,2013,30(7):2007-2009