SHACAL-2 算法中非线性函数的差分特性及其应用

沈 璇[①] 李瑞林[②] 李 超[*①] 赵光耀[③]
[①](国防科技大学理学院 长沙 410073)
[②](国防科技大学电子科学与工程学院 长沙 410073)
[③](国防科技大学计算机学院 长沙 410073)

摘 要：SHACAL-2 算法是欧洲 NESSIE 计划推荐的分组密码标准算法之一，选择函数和主函数是 SHACAL-2 算法中两类基本的非线性函数。该文分析了这两类非线性函数的差分特性，证明了当选择函数的第 1 个位置输入差分非零或者主函数的前两个位置中任意一个输入差分非零时(其它位置差分均为零)，对应差分方程解的个数仅与输入差分的重量有关。将这一特性引入到 SHACHL-2 算法的差分故障攻击中，结果表明至少需要 160 个随机故障才能使该攻击以超过 60%的成功概率恢复 512 bit 的种子密钥，至少需要 240 个随机故障才能以超过 98%的成功概率恢复 512 bit 的种子密钥。

关键词：密码学；SHACAL-2算法；选择函数；主函数；差分特性；故障分析
中图分类号：TN918 **文献标识码**：A **文章编号**：1009-5896(2014)07-1661-06
DOI: 10.3724/SP.J.1146.2013.01717

Differential Analysis of the Nonlinear Functions of SHACAL-2 Algorithm and the Application

Shen Xuan[①] Li Rui-lin[②] Li Chao[①] Zhao Guang-yao[③]
[①](*College of Science*, *National University of Defense Technology*, *Changsha* 410073, *China*)
[②](*College of Electronic Science and Engineering*, *National University of Defense Technology*, *Changsha* 410073, *China*)
[③](*College of Computer Science*, *National University of Defense Technology*, *Changsha* 410073, *China*)

Abstract: SHACAL-2 algorithm is one of the standard block ciphers recommended by European NESSIE plan. It includes two kinds of nonlinear functions, the choice function and the major function. This paper studies mainly differential properties of the two nonlinear functions, and it is shown that the number of solutions of the differential equation is only related with the weight of the input difference when the difference only appears at the first position of the choice function, or only appears at the first or the second position of the major function. This observation is applied to the differential fault analysis on SHACAL-2. The results demonstrate that at least 160 random faults are needed to obtain 512 bit key with successful probability more than 60%, while at least 240 random faults are needed to obtain 512 bit key with successful probability more than 98%.

Key words: Cryptography; SHACAL-2 algorithm; Choice function; Major function; Differential property; Fault analysis

1 引言

SHACAL-1 算法和 SHACAL-2 算法[1]是由 Handschuh 和 Naccache 提交给欧洲 NESSIE 计划的两个算法。SHACAL-1 算法由于其较为简洁的密钥扩展方案所引发的安全性问题止步于 NESSIE 计划的第 2 轮，SHACAL-2 算法则最终成为了获胜的 4 个标准算法之一。SHACAL-2 算法没有采用传统的 Feistel 结构或者 SP 结构，而是在标准函数 SHA-256 算法的基础上发展而来。

故障攻击是侧信道攻击的一种，它由Boneh等人在文献[2]中首先提出，并用于对基于CRT实现的RSA签名算法的攻击。随后，Biham 和Shamir在文献[3]中将该攻击思想进行改进，提出了分组密码的差分故障分析方法，并用于分析DES算法的安全性。近年来，国内外密码学者将这种攻击方法运用到包括分组密码和流密码在内的诸多算法，如3DES[4], SMS4[5,6], AES[7], LEX[8], Trivium[9]等算法，取得了

2013-11-04 收到，2014-03-05 改回
国家自然科学基金(61103192)和国家973计划项目(2013CB338002)资助课题
*通信作者：李超 academic_lc@163.com

很多重要的研究结果。

差分故障分析方法大都基于统计分析技术或代数分析技术。统计分析技术主要是利用密码算法的统计特性，用概率的方法进行攻击，如文献[10]利用统计方法分析了一些轻量级密码算法抵抗故障分析的能力；代数分析技术主要基于密码算法对应的代数方程组求解而进行攻击，如文献[11]利用代数方法获得了对若干轻量级密码算法较好的故障分析结果。无论是统计分析技术还是代数分析技术其核心都是对密码算法的非线性部件进行分析。对密码算法进行差分故障攻击时，一般会通过求解非线性部件来恢复内部状态或轮密钥，因此密码算法非线性部件的差分特性对其抵抗差分故障攻击的能力具有至关重要的作用。

本文主要研究了SHACAL-2算法中非线性函数的差分特性，证明了当选择函数的第1个位置输入差分非零或者主函数的前两个位置中任意一个输入差分非零时(其它位置差分均为零)，对应差分方程解的个数仅与输入差分的重量有关。该结论可用于对SHACAL-2算法的差分故障分析，攻击过程一方面利用代数分析方法求解相应的混合运算差分方程，另一方面利用概率分析的方法从理论上分析了恢复唯一种子密钥的成功概率与所需故障数目之间的关系。特别指出的是，文献[12]的差分故障分析结果是本文研究成果的一个特例，同时本文还利用选择函数的差分特性从理论上解释了SHACAL-2算法的有效差分故障位置为E。进一步，本文研究结果表明至少需要160个随机故障才能以超过60%的成功概率恢复512 bit的种子密钥，而至少需要240个随机故障才能以超过98%的成功概率恢复512 bit的种子密钥。

2 SHACAL-2算法中非线性函数的差分特性

SHACAL-2 算法包含 4 种基本函数，分别为$\Sigma_0,\Sigma_1,\mathrm{CH},\mathrm{Maj}$。本文主要对非线性函数 CH 和 Maj 的差分特性进行研究。

本文所用符号定义如下：x,y,z,δ,Δ 均为 32 bit 的字，δ 为输入差分，Δ 为输出差分，一般情形下，假设 x,y,δ,Δ 为给定的已知量，z 为未知量，X_i 表示字 X 的第 i bit，“$\overline{X}$”表示按位取反，其中 $X\in\{x,y,z,\delta,\Delta\}$。令“$\oplus$”表示按位异或，“$\boxplus$”和“$\boxminus$”表示模$2^{32}$加法和减法。“$\cdot$”表示按位取逻辑与运算(在下面证明的过程中为书写方便省略该乘号)，$W(\delta)$表示δ的重量，即δ二进制表示中所含 1 的个数，$N(z)$为满足差分方程未知量z的个数。

SHACAL-2 算法中选择函数定义为：

$$\mathrm{CH}(x,\ y,z)=x\cdot y\oplus\overline{x}\cdot z$$

主函数定义为：

$$\mathrm{Maj}(x,y,z)=x\cdot y\ \oplus x\cdot z\oplus y\cdot z$$

为分析选择函数和主函数的差分特性，首先给出如下引理：

引理[13] 给定方程$(z\oplus\delta)\boxminus z=\Delta$，其中$z,\delta,\Delta$是 32 bit 的字，假设$\delta,\Delta$已知而$z$未知，则有如下结论成立：

$$z_i=\begin{cases}0\text{或}1, & i=31\\ 0\text{或}1, & \delta_i=0,\quad 0\le i\le 30\\ \delta_{i+1}\oplus\Delta_{i+1}, & \delta_i=1,\quad 0\le i\le 30\end{cases}$$

2.1 选择函数的差分特性研究

对于选择函数$\mathrm{CH}(x,y,z)=x\cdot y\oplus\overline{x}\cdot z$，分别在$x,y,z$处诱导差分，得到如下 3 种情形的差分方程：

情形 1: $\mathrm{CH}(x\oplus\delta,y,z)\boxminus\mathrm{CH}(x,y,z)=\Delta$

情形 2: $\mathrm{CH}(x,y\oplus\delta,z)\boxminus\mathrm{CH}(x,y,z)=\Delta$

情形 3: $\mathrm{CH}(x,y,z\oplus\delta)\boxminus\mathrm{CH}(x,y,z)=\Delta$

在下面的讨论中，将根据上面 3 种情形差分方程的特点，分别推导未知量 z 的取值个数与输入差分δ之间的关系，主要结果如下：

定理 1 SHACAL-2 算法中的选择函数具有如下的差分特性：如果输入差分δ的重量为k，那么满足情形 1 方程的未知量个数为2^{32-k}，满足情形 2 方程的未知量个数为2^{32}，而满足情形 3 方程的未知量个数不能由k单独确定。

证明 首先，根据选择函数的定义，情形 1 的差分方程为

$$[(x\oplus\delta)\cdot y\oplus\overline{(x\oplus\delta)}\cdot z]\boxminus(x\cdot y\oplus\overline{x}\cdot z)=\Delta$$

注意到 $\overline{x\oplus\delta}=\overline{x}\oplus\delta$，上面的差分方程可以变为

$$\begin{aligned}\Delta&=(xy\oplus\delta y\oplus\overline{x}z\oplus\delta z)\boxminus(xy\oplus\overline{x}z)\\&=((xy\oplus\overline{x}z)\oplus\delta(y\oplus z))\boxminus(xy\oplus\overline{x}z)\end{aligned}$$

令$w=xy\oplus\overline{x}z$，则

$$\begin{aligned}&(w\oplus\delta(y\oplus z))\boxminus w=\Delta\\&(w\oplus\delta(y\oplus z))=\Delta\boxplus w\end{aligned}$$

按比特位展开

$$\left.\begin{aligned}&(w\oplus\delta(y\oplus z))_i=(\Delta\boxplus w)_i, && 0\le i\le 31\\&w_i\oplus\delta_i(y_i\oplus z_i)=\Delta_i\oplus w_i\oplus c_i, && 0\le i\le 31\end{aligned}\right\}\quad(1)$$

其中

$$\left.\begin{aligned}c_i&=w_{i-1}\Delta_{i-1}\oplus(w_{i-1}\oplus\Delta_{i-1})c_{i-1}\\&=w_{i-1}(\Delta_{i-1}\oplus c_{i-1})\oplus\Delta_{i-1}c_{i-1},\quad 1\le i\le 31\\c_0&=0\end{aligned}\right\}\quad(2)$$

将式(1)化简为

$$\delta_i(y_i\oplus z_i)=\Delta_i\oplus c_i,\quad 0\le i\le 31\qquad(3)$$

在式(3)中当$i=0$时，$\delta_0(y_0\oplus z_0)\oplus\Delta_0=c_0=0$，故

$\delta_0 z_0 = \Delta_0 \oplus \delta_0 y_0$。当 $\delta_0 = 1$ 时，z_0 可以唯一确定，当 $\delta_0 = 0$ 时，z_0 可取值 0 或 1。

当 $1 \le i \le 31$ 时

(1) $\Delta_{i-1} \oplus c_{i-1} = 1$，则 $\Delta_{i-1} c_{i-1} = 0$，由式(3)知 $1 = \Delta_{i-1} \oplus c_{i-1} = \delta_{i-1}(y_{i-1} \oplus z_{i-1})$，故

$$\begin{cases} y_{i-1} \oplus z_{i-1} = 1 \\ \delta_{i-1} = 1 \end{cases} \Rightarrow \begin{cases} z_{i-1} = \overline{y}_{i-1} \\ \delta_{i-1} = 1 \end{cases}$$

这时式(2)变为 $c_i = w_{i-1}$，代入式(3)中有

$$\left.\begin{aligned} \delta_i(y_i \oplus z_i) &= \Delta_i \oplus w_{i-1} \\ \delta_i(y_i \oplus z_i) &= \Delta_i \oplus x_{i-1}y_{i-1} \oplus \overline{x}_{i-1}z_{i-1} \\ \delta_i z_i &= \Delta_i \oplus x_{i-1}y_{i-1} \oplus \overline{x}_{i-1}z_{i-1} \oplus \delta_i y_i \\ &= \Delta_i \oplus x_{i-1}y_{i-1} \oplus \overline{x}_{i-1}\overline{y}_{i-1} \oplus \delta_i y_i, 1 \le i \le 31 \end{aligned}\right\} \quad (4)$$

式(4)中右边为已知量，当 $\delta_i = 1$ 时，z_i 可以唯一确定，当 $\delta_i = 0$ 时，z_i 可取值 0 或 1。

(2) $\Delta_{i-1} \oplus c_{i-1} = 0$，由于 Δ_{i-1} 是给定的，因此满足该条件的 c_{i-1} 是确定的，有 $c_{i-1} = \Delta_{i-1}$，$c_{i-1}\Delta_{i-1} = \Delta_{i-1}^2 = \Delta_{i-1}$。这时式(2)为 $c_i = \Delta_{i-1}$，代入式(3)有

$$\left.\begin{aligned} \delta_i(y_i \oplus z_i) &= \Delta_i \oplus c_i = \Delta_i \oplus \Delta_{i-1} \\ \delta_i z_i &= \delta_i y_i \oplus \Delta_i \oplus \Delta_{i-1} \end{aligned}\right\} \quad (5)$$

式(5)中右边为已知量，因此当 $\delta_i = 1$ 时，z_i 可以唯一确定，当 $\delta_i = 0$ 时，z_i 可取值 0 或 1。

(3)当 $\Delta_i \oplus c_i = 1$ 时，由式(3)知，$\delta_i(y_i \oplus z_i) = 1$，则此时有 $\delta_i = 1$ 且 $z_i = y_i \oplus 1 = \overline{y}_i$，故 z_i 可以唯一确定。

因此，当 $\delta_i = 1 (1 \le i \le 31)$ 时，由条件(1)和(2)知，z_i 可以唯一确定；当 $\delta_i = 0 (1 \le i \le 31)$ 时，条件(1), (2), (3)均不能将 z_i 唯一确定，z_i 可以取 0 或 1。i=0 的情况在之前已给出。

综上所述，z_i 能唯一确定的充要条件为 $\delta_i = 1$，所以当 δ 的重量为 k 时，即有 k 个分量为 1，这时 $\delta_i = 1$ 对应位置的 z 的比特分量 z_i 可以唯一确定，其余分量可取 0 或 1，因此这种情况下，满足差分方程的未知量 z 的个数为 2^{32-k}。

故对于情形 1 可以得出如下结论：满足差分方程的输入差分 δ 与未知量 z 的关系为：当 $W(\delta) = k$ 时，$N(z) = 2^{32-k}$。

其次，同样根据选择函数的定义，情形 2 的方程为

$$[x(y \oplus \delta) \oplus \overline{x}z] \boxminus (xy \oplus \overline{x}z) = \Delta$$

进一步可以转化为 $((xy \oplus \overline{x}z) \oplus x\delta) \boxminus (xy \oplus \overline{x}z) = \Delta$，令 $w = xy \oplus \overline{x}z$，则可简化为

$$(w \oplus x\delta) \boxminus w = \Delta \quad (6)$$

对于式(6)，由于 x, δ 都是给定的，w 是未知量 z 的函数，由引理可知，除去最高位外，要想确定 $z_i (0 \le i \le 30)$ 需要有 $(x\delta)_i = x_i\delta_i = 1$，即 $x_i = 1$ 且 $\delta_i = 1$，这时 $w_i = y_i$，故 z 的取值无法确定，它的每一 bit 都可以取 0 或者 1，因此满足差分方程的未知量的个数 $N(z)$ 与 δ 无关，故 $N(z) = 2^{32}$。

最后，情形 3 的方程为 $(x \cdot y \oplus \overline{x} \cdot (z \oplus \delta)) \boxminus (x \cdot y \oplus \overline{x} \cdot z) = \Delta$，进一步可转化为

$$((xy \oplus \overline{x}z) \oplus \overline{x}\delta) \boxminus (x \cdot y \oplus \overline{x} \cdot z) = \Delta$$

令 $w = xy \oplus \overline{x}z$，则方程化简为

$$(w \oplus \overline{x}\delta) \boxminus w = \Delta \quad (7)$$

对于式(7)，由于 x, δ 都是给定的，w 是未知量 z 的函数，由引理可知，除去最高位外，要想确定 $z_i (0 \le i \le 30)$ 需要有 $(\overline{x}\delta)_i = \overline{x}_i\delta_i = 1$，即 $\overline{x}_i = 1$ 且 $\delta_i = 1$，这时 $z_i = w_i = \overline{x}_{i+1}\delta_{i+1} \oplus \Delta_{i+1}$，故

$$z_i = \begin{cases} 0\text{或}1, & i = 31 \\ 0\text{或}1, & \delta_i = 0\text{或}x_i = 1, 0 \le i \le 30 \\ \overline{x}_{i+1}\delta_{i+1} \oplus \Delta_{i+1}, & \delta_i = 1\text{且}x_i = 0, 0 \le i \le 30 \end{cases}$$

这表明未知量 z 除最高位外的某一位能否被确定不仅与相应的 δ 的分量有关，而且与相应的 x 的分量有关，故满足差分方程的未知量的个数不能由 δ 单独决定。　　证毕

2.2 主函数的差分特性研究

对于主函数 $\mathrm{Maj}(x,y,z) = x \cdot y \oplus x \cdot z \oplus y \cdot z$ 的差分特性，分别在 x, y, z 处诱导差分，得到如下 3 种情形下的差分方程：

情形 4: $\mathrm{Maj}(x \oplus \delta, y, z) \boxminus \mathrm{Maj}(x,y,z) = \Delta$

情形 5: $\mathrm{Maj}(x, y \oplus \delta, z) \boxminus \mathrm{Maj}(x,y,z) = \Delta$

情形 6: $\mathrm{Maj}(x, y, z \oplus \delta) \boxminus \mathrm{Maj}(x,y,z) = \Delta$

下面根据上面 3 种情形下的差分方程的特点，分别推测未知量 z 的取值个数与输入差分 δ 之间的关系，所得结果如下：

定理 2　SHACAL-2 算法中的主函数具有如下的差分特性：如果输入差分 δ 的重量为 k，那么满足情形 4 方程的未知量个数为 2^{32-k}，满足情形 5 方程的未知量个数为 2^{32-k}，而满足情形 6 方程的未知量个数不能由 k 单独确定。

证明　根据主函数的定义，情形 4 的差分方程可以转化为

$$[(xy \oplus xz \oplus yz) \oplus \delta(y \oplus z)] \boxminus (xy \oplus xz \oplus yz) = \Delta$$

令 $w = xy \oplus xz \oplus yz$，则 $[w \oplus \delta(y \oplus z)] \boxminus w = \Delta$，这和情形 1 的结构完全相似，故用相同的方法证明可以得到同样的结果，即满足差分方程的输入差分 δ 与未知量 z 的关系为：当 $W(\delta) = k$ 时，$N(z) = 2^{32-k}$。

而情形 5 的方程可以化为 $[x(y \oplus \delta) \oplus xz \oplus (y \oplus \delta)z] \boxminus (xy \oplus xz \oplus yz) = \Delta$，　由于主函数中 x, y

是对称的，故由情形Ⅳ知，满足差分方程的输入差分 δ 与未知量 z 的关系为：当 $W(\delta)=k$ 时，$N(z)=2^{32-k}$。

最后，情形 6 的差分方程为 $[xy\oplus x(z\oplus\delta)\oplus y(z\oplus\delta)]\boxminus(xy\oplus xz\oplus yz)=\Delta$，通过简单的变量替换，上述方程转化为：$[(xy\oplus xz\oplus yz)\oplus\delta(x\oplus y)]\boxminus(xy\oplus xz\oplus yz)=\Delta$，令 $w=xy\oplus xz\oplus yz$，则 $[w\oplus\delta(x\oplus y)]\boxminus w=\Delta$。由引理知，当 $\delta_i(x_i\oplus y_i)=1$ 时，即 $\delta_i=1$且$(x_i\oplus y_i)=1$，w_i 除最高位外，能唯一确定，此时 $w_i=x_iy_i\oplus x_iz_i\oplus y_iz_i=x_iy_i\oplus(x_i\oplus y_i)z_i=z_i$，故

$$z_i=\begin{cases}0\text{或}1,\ i=31\\ 0\text{或}1,\ \delta_i=0\text{或}x_i\oplus y_i=0,\quad 0\le i\le 30,\\ \delta_{i+1}(x_{i+1}\oplus y_{i+1})\oplus\Delta_{i+1},\\ \qquad \delta_i=1\text{且}x_i\oplus y_i=1,\quad 0\le i\le 30\end{cases}$$

这表明未知量 z 除最高位外的某一位能否确定不仅与相应的 δ 的分量有关，而且与相应的 x,y 的分量有关，故满足差分方程的未知量的个数不能由 δ 单独决定。 证毕

3 选择函数的差分特性在SHACHL-2算法差分故障攻击中的应用

SHACHL-2 算法的分组长度为 256 bit，密钥长度为 512 bit，迭代次数为 64 轮。SHACHL-2 算法的一轮加密函数的更新过程如下：

$$T1_{i+1}=H_i\boxplus\Sigma_1(E_i)\boxplus\mathrm{CH}(E_i,F_i,G_i)\boxplus K_i\boxplus W_i$$
$$T2_{i+1}=\Sigma_0(A_i)\boxplus\mathrm{Maj}(A_i,B_i,C_i)$$
$$H_{i+1}=G_i;G_{i+1}=F_i;F_{i+1}=E_i;E_{i+1}=D_i\boxplus T1_{i+1}$$
$$D_{i+1}=C_i;C_{i+1}=B_i;B_{i+1}=A_i;A_{i+1}=T1_{i+1}\boxplus T2_{i+1}$$

这里 W_i 是轮常数；$\mathrm{CH},\mathrm{Maj},\Sigma_0,\Sigma_1$ 都是具有 32 bit 字输入，32 bit 字输出的函数。

在差分故障攻击中，随机选取一个明文 $P=(A_0,B_0,C_0,D_0,E_0,F_0,G_0,H_0)$，经过加密后可以获得正确密文 $Y=(A_{64},B_{64},C_{64},D_{64},E_{64},F_{64},G_{64},H_{64})$，假设中间状态某个位置上导入一个基于 32 bit 字的随机故障，并得到相应的错误密文 $Y^*=(A^*_{64},B^*_{64},C^*_{64},D^*_{64},E^*_{64},F^*_{64},G^*_{64},H^*_{64})$。

由于

$$\begin{aligned}A_{64}&=T1_{64}\boxplus T2_{64}=H_{63}\boxplus\Sigma_1(E_{63})\boxplus\mathrm{CH}(E_{63},F_{63},G_{63})\\&\quad\boxplus K_{63}\boxplus W_{63}\boxplus\Sigma_0(A_{63})\boxplus\mathrm{Maj}(A_{63},B_{63},C_{63})\\&=H_{63}\boxplus\Sigma_1(F_{64})\boxplus\mathrm{CH}(F_{64},G_{64},H_{64})\boxplus K_{63}\boxplus\\&\quad W_{63}\boxplus\Sigma_0(B_{64})\boxplus\mathrm{Maj}(B_{64},C_{64},D_{64})\end{aligned}$$

因此在上述等式中除 H_{63} 和 K_{63} 外都已知，为求解 K_{63} 只需知道 H_{63} 即可。而 $H_{63}=G_{62}$，故可以选择在倒数第 2 轮输入故障。由加密函数的更新过程有

$$\begin{aligned}A_{63}&=T1_{63}\boxplus T2_{63}=H_{62}\boxplus\Sigma_1(E_{62})\\&\quad\boxplus\mathrm{CH}(E_{62},F_{62},G_{62})\boxplus K_{62}\boxplus W_{62}\\&\quad\boxplus\Sigma_0(A_{62})\boxplus\mathrm{Maj}(A_{62},B_{62},C_{62})\end{aligned}\tag{8}$$

在式(8)中为得到 G_{62} 的值，需要在倒数第 2 轮选择一个位置诱导故障。文献[12]中提到只有 E_{62} 是有效的差分故障位置，但并没有给出理论说明。根据选择函数的差分特性，即定理 1 可知，只有在选择函数的第 1 个位置即 E_{62} 处输入故障时，才能得到 G_{62} 的有效信息，而选择 F_{62},G_{62} 时均不能得到 G_{62} 的有效信息，因此 E_{62} 是有效的差分故障位置。

如图 1 所示，在 E_{62} 处输入故障时，根据迭代函数的更新过程，倒数第 2 轮的输出值中 A_{63},E_{63},F_{63} 有差分，最后一轮的输出值中 $A_{64},B_{64},E_{64},F_{64},G_{64}$ 有差分，其余位置没有差分。当在 E_{62} 处输入故障 δ 时，得到

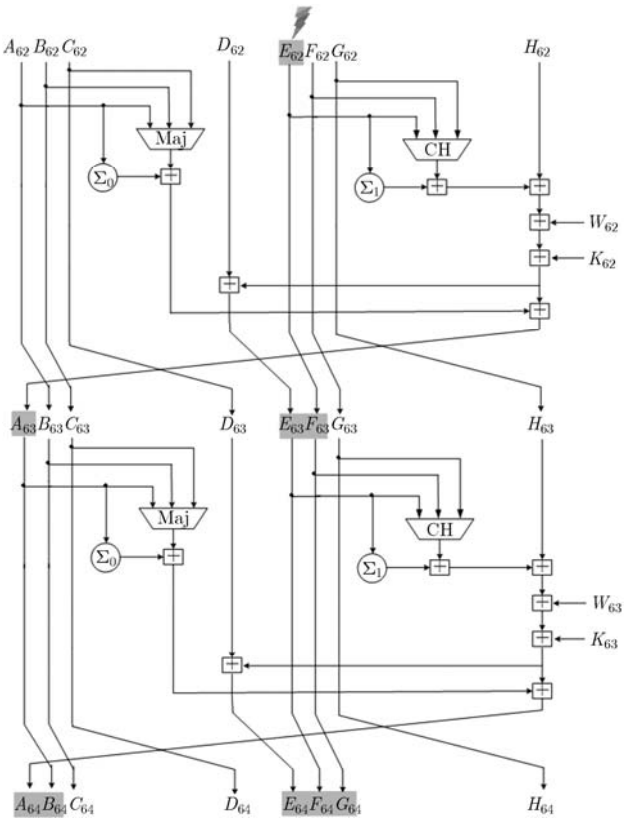

图1 对SHACAL-2算法最后两轮的差分故障攻击

$$\begin{aligned}A^*_{63}&=H_{62}\boxplus\Sigma_1(F_{63}\oplus\delta)\boxplus\mathrm{CH}(F_{63}\oplus\delta,G_{63},H_{63})\\&\quad\boxplus K_{62}\boxplus W_{62}\boxplus\Sigma_0(B_{63})\boxplus\mathrm{Maj}(B_{63},C_{63},D_{63})\end{aligned}$$

根据加密流程可知：

$$\begin{aligned}B^*_{64}&=H_{62}\boxplus\Sigma_1(G_{64}\oplus\delta)\boxplus\mathrm{CH}(G_{64}\oplus\delta,H_{64},H_{63})\\&\quad\boxplus K_{62}\boxplus W_{62}\boxplus\Sigma_0(C_{64})\boxplus\mathrm{Maj}(B_{63},C_{63},D_{63})\end{aligned}\tag{9}$$

同理，根据加密流程，式(8)可转化为

$$B_{64}=H_{62}\boxplus\Sigma_1(G_{64})\boxplus\mathrm{CH}(G_{64},H_{64},H_{63})\boxplus K_{62}\boxplus W_{62}\boxplus\Sigma_0(C_{64})\boxplus\mathrm{Maj}(B_{63},C_{63},D_{63}) \tag{10}$$

将式(9)和式(10)做模减，并整理得

$$\Delta'=\mathrm{CH}(G_{64}\oplus\delta,H_{64},H_{63})\boxminus\mathrm{CH}(G_{64},H_{64},H_{63})\boxplus(\Sigma_1(G_{64}\oplus\delta)\boxminus\Sigma_1(G_{64})) \tag{11}$$

其中 $\Delta'=B_{64}^{*}\boxminus B_{64}$，式(11)即为文献[12]中求解 H_{63} 的差分方程。

文献[12]对式(11)的求解只是通过计算机搜索给出了一个实验结果，并没有给出相应的理论分析。实际上，我们可以直接利用选择函数的差分特性给出式(11)的代数求解，下面，通过理论推导给出 SHACAL-2 算法差分故障攻击成功概率与诱导故障数目之间的关系。

令 $\Delta=\Delta'\boxminus(\Sigma_1(G_{64}\oplus\delta)\boxminus\Sigma_1(G_{64})),x=G_{64},y=H_{64},z=H_{63}$，则式(11)可变为

$$\mathrm{CH}(x\oplus\delta,y,z)\boxminus\mathrm{CH}(x,y,z)=\Delta$$

这对应于情形 1。因此，对 SHACAL-2 算法的差分故障攻击可归结于选择函数的差分特性。

如果情形 1 的差分方程记为 $(z;x,y,\delta,\Delta)$，则下面结论成立：

定理 3　对于 m 个差分方程 $(z;x,y,\delta^{(j)},\Delta^{(j)})$ $(1\le j\le m)$，其中 $\delta^{(j)}(1\le j\le m)$ 是随机选取的 32 bit 的字，则能唯一确定未知量 z 的概率为 $P_1=[1-(1/2)^m]^{32}$。

证明　由定理 1 证明过程知，未知变量 z 的每一比特取值仅仅依赖于相应 δ 的分量取值，当 $\delta_i=1(0\le i\le 31)$ 时，z_i 可唯一确定。记 A_i 表示至少存在某个整数 j，使得 $\delta_i^j=1,1\le j\le m$。注意到，每一个 δ_i^j 都是独立且随机选取的，因此，$P(A_i)=1-(1/2)^m$，由于每一个 A_i 都相互独立，所以能唯一确定未知量 z 的概率为 $P_1=\prod_{i=0}^{31}P(A_i)=[1-(1/2)^m]^{32}$

证毕

攻击者通过在适当的位置诱导 m 次故障，就可以得到m个差分方程 $(z;x,y,\delta^{(j)},\Delta^{(j)}),1\le j\le m$，此时，能通过差分方程唯一求解出未知量$z$的概率为 $[1-(1/2)^m]^{32}$，通过差分方程的解就可以确定出相应的轮密钥。根据密钥扩展方案，如果恢复出了最后16轮的轮密钥 $(K_{48},K_{49},\cdots,K_{63})$ 就可以完全恢复出种子密钥。假设连续16轮故障诱导的数目分别为 $m_1,m_2,\cdots,m_{16}$，此时需要的差分故障数目为 $n=\sum_{i=1}^{16}m_i$，而能恢复唯一种子密钥的概率为 $P_2=\prod_{i=1}^{16}[1-(1/2)^{m_i}]^{32}$，有如下结论成立。

定理 4　若以给定的成功概率 ρ 恢复出唯一的种子密钥，即有 $P_2=\prod_{i=1}^{16}[1-(1/2)^{m_i}]^{32}=\rho$，则当 $m_1=m_2\cdots=m_{16}=m$ 时，所需的总故障数 n 最小，此时，$n=\sum_{i=1}^{16}m_i=16m$。

证明　由 $\prod_{i=1}^{16}\left[1-\left(\frac{1}{2^{m_i}}\right)\right]^{32}=\rho$，则有 $\prod_{i=1}^{16}\left[1-\left(\frac{1}{2^{m_i}}\right)\right]=\rho^{1/32}$，记 $\rho_0=\rho^{1/32}$，令

$$f(m_1,m_2,\cdots,m_{16})=\prod_{i=1}^{16}\left[1-\left(\frac{1}{2^{m_i}}\right)\right]-\rho_0$$

$$g(m_1,m_2,\cdots,m_{16})=\sum_{i=1}^{16}m_i$$

则由数学分析中多元函数求条件极值的方法知，约束条件是 $f(m_1,m_2,\cdots,m_{16})=0$，目标函数是 $g(m_1,m_2,\cdots,m_{16})$。令 $h(m_1,m_2,\cdots,m_{16},\lambda)=g(m_1,m_2,\cdots,m_{16})+\lambda f(m_1,m_2,\cdots,m_{16})$，则达到极值条件时满足的方程组为

$$\left.\begin{aligned}&\frac{\partial h}{\partial m_j}=1+\lambda\prod_{i=1,i\ne j}^{16}\left(1-\frac{1}{2^{m_i}}\right)2^{-m_i}\ln 2=0,\\&\qquad j=1,2,\cdots,16\\&\prod_{i=1}^{16}\left(1-\left(\frac{1}{2^{m_i}}\right)\right)-\rho_0=0\end{aligned}\right\} \tag{12}$$

令 $1\le j_k,j_l\le 16,j_k\ne j_l$，由式(12)有

$$\left.\begin{aligned}&\frac{\partial h}{\partial m_{j_k}}=1+\lambda\prod_{i=1,i\ne j_k}^{16}\left(1-\frac{1}{2^{m_i}}\right)2^{-m_{j_k}}\ln 2=0\\&\frac{\partial h}{\partial m_{j_l}}=1+\lambda\prod_{i=1,i\ne j_l}^{16}\left(1-\frac{1}{2^{m_i}}\right)2^{-m_{j_l}}\ln 2=0\end{aligned}\right\}$$

将两式相减得

$$\lambda(2^{-m_{j_k}}-2^{-m_{j_l}})\prod_{i=1,i\ne j_l,i\ne j_k}^{16}\left(1-\frac{1}{2^{m_i}}\right)\ln 2=0$$

又 $1-1/2^{m_i}\ne 0,i=1,2,\cdots,16$，且 $\lambda\ne 0$，故有 $2^{-m_{j_k}}-2^{-m_{j_l}}=0$，因此可得 $m_{j_k}=m_{j_l}$。由 j_k,j_l 取值的任意性可知 $m_1=m_2=\cdots=m_{16}=m$，代入式(12)得 $m=-\log_2(1-\rho^{1/(32\times16)})$，此时目标函数取得最小值，即有

$$n=\sum_{i=1}^{16}m_i=16m=-16\log_2(1-\rho^{1/(32\times16)})$$

证毕

由定理 4 知，当给定成功概率 ρ 时，为使总故障数目最小，需要每轮注入的故障数 $m_i(1\le i\le 16)$ 均相等，即 $m_1=m_2=\cdots=m_{16}=m$，此时 $P_2=(1-(1/2)^m)^{32\times16}$。表 1 列出了 ρ 取不同的值时对应 m 的取值情况。

4　实验结果

为了验证上述理论的正确性，本文在 PC 机上(CPU: Pentium Dual-Core E6700 3.20 GHz, RAM: 2 GB)使用 C++语言编程(Visual C++6.0)进行故障攻击的模拟实验。当 $m=8,9,\cdots,15$，随机选取一个明文和 512 bit 的密钥，然后每轮注入相同的故障数 m，进行 1000 次实验。表 2 给出每轮的注入故障

表 1 成功概率 ρ 与每轮注入故障数 m 的取值关系

ρ (%)	m	ρ (%)	m
50	9.53	80	11.16
60	9.98	90	12.25
70	10.49	95	13.29

表 2 不同 m 取值下成功概率的理论与实验结果比较

m	ρ_m(%)		m	ρ_m(%)	
	理论	实验		理论	实验
8	13.84	13.78	12	88.25	88.65
9	36.75	37.12	13	93.94	93.73
10	60.64	60.15	14	96.92	96.19
11	77.88	77.42	15	98.45	98.72

数 m 与成功恢复出唯一 512 bit 种子密钥的概率 ρ_m 的理论结果与实验结果。文献[12]只给出 m=8 时的一个实验结果，实际上，该情形成功的概率只有 13.84%。

5 结束语

本文通过研究 SHACAL-2 算法中非线性函数的差分特性，证明了当选择函数的第 1 个位置输入差分非零或者主函数的前两个位置中任意一个输入差分非零时(其它位置差分均为零)，差分方程解的个数只与输入差分的重量有关。将这一结果运用到 SHACHL-2 算法的差分故障攻击中，从理论上解释了有效的差分故障位置为 E，并证明了至少需要 160 个随机故障才能以超过 60%的成功概率恢复 512 bit 的种子密钥，而至少需要 240 个随机故障才能以超过 98%的成功概率恢复 512 bit 的种子密钥。上述结果均在个人电脑上进行了实验验证。下一步将探讨如何利用本文的研究结果对 SHACAL-2 算法的压缩函数进行分析。

参 考 文 献

[1] Handschuh H and Naccache D. SHACAL: a family of block ciphers[OL]. https://www.cosic.esat.kuleuven.be/nessie/, 2002.

[2] Boneh D, DeMillo R A, and Lipton R J. On the importance of eliminating errors in cryptographic computations[J]. *Journal of Cryptology*, 2001, 14(2): 101–119.

[3] Biham E and Shamir A. Differential fault analysis of secret key cryptosystems[J]. *LNCS*, 1997, 1294: 513–525.

[4] Hemme L. A differential fault attack against early rounds of (Triple-) DES[J]. *LNCS*, 2004, 3156: 254–267.

[5] 张蕾, 吴文玲. SMS4密码算法的差分故障攻击[J]. 计算机学报, 2006, 29(9): 1596–1602.
Zhang Lei and Wu Wen-ling. Differential fault analysis on SMS4[J]. *Chinese Journal of Computers*, 2006, 29(9): 1596–1602.

[6] 李玮, 谷大武. 基于密钥编排故障的SMS4算法的差分故障分析[J]. 通信学报, 2008, 29(10): 135–142.
Li wei and Gu Da-wu. Differential fault analysis on SMS4 based on the key schedule[J]. *Journal on Communications*, 2008, 29(10): 135–142.

[7] Kim C H. Differential fault analysis of AES: toward reducing number of faults[J]. *Information Sciences*, 2012, 199: 43–57.

[8] 张中亚, 关杰. 对流密码算法LEX的差分故障攻击[J]. 上海交通大学学报, 2012, 46(6): 865–869.
Zhang Zhong-ya and Guan Jie. Differential fault analysis on the stream cipher LEX[J]. *Journal of Shanghai Jiaotong University*, 2012, 46(6): 865–869.

[9] Hu Y, Gao J, Liu Q, *et al*.. Fault analysis of Trivium[J]. *Designs, Codes and Cryptography*, 2012, 62(3): 289–311.

[10] Gu D, Guo Z, and Liu J. Differential fault analysis on lightweight blockciphers with statiastical cryptanalysis techniques[C]. 2012 Workshop on Fault Diagnosis and Tolerance in Cryptography, Leuven, 2012: 27–33.

[11] Zhang F, Zhao X, Guo S *et al*.. Improved algebraic fault analysis:a case study on piccolo and applications to other lightweight block ciphers[J]. *LNCS*, 2013, 7864: 62–79.

[12] 魏悦川, 李琳, 李瑞林, 等. SHACAL-2算法的差分故障攻击[J]. 电子与信息学报, 2010, 32(2): 318–322.
Wei Yue-chuan, Li Lin, Li Rui-lin, *et al*.. Differential fault analysis on SHACAL-2[J]. *Journal of Electronic & Information Technology*, 2010, 32(2): 318–322.

[13] Li R, Li C, and Gong C. Differential fault analysis on SHACAL-1[C]. 2009 Workshop on Fault Diagnosis and Tolerance in Cryptography, Lausanne, 2009: 120–126.

沈 璇： 男，1990年生，硕士生，研究方向为编码密码理论及其应用.

李 超： 男，1966年生，博士生导师，教授，研究方向为编码密码理论及其应用.