基于减轮故障的SM2解密算法选择密文组合攻击

李昊远[1,2],韩绪仓[1,2],曹伟琼[1],王 舰[1,2],陈 华[1]
(1. 中国科学院软件研究所可信计算与信息保障实验室,北京 100190;2. 中国科学院大学,北京 100049)

摘 要: SM2系列算法是由我国自主设计的商用椭圆曲线密码算法. 目前,对SM2解密算法的实现安全性分析通常遵循对椭圆曲线通用组件的研究成果,缺乏结合算法本身结构和特点而进行的实现安全性研究. 同时,SM2解密算法中的哈希和验证步骤,使大部分需要利用错误输出的故障攻击方式对于SM2解密算法并不适用. 针对该现状,本文根据SM2解密算法本身的特点,结合安全错误类故障攻击思想,提出了一种减轮故障与侧信道相结合的选择密文组合攻击. 攻击的核心是通过故障注入改变标量乘循环的轮数,然后由侧信道分析确定故障轮数的具体取值. 根据部分密钥猜测结合明文、正确密文等构建选择密文,并将其输入至具有特定故障效果的解密设备,最后通过解密设备输出验证部分密钥猜测是否正确,逐步恢复私钥. 此外,文中分析了攻击对不同标量乘法以及常见防护对策的适用性. 最后,本文在基于ARM Cortex M4核心的STM32F303微控制器芯片上,使用时钟毛刺注入和简单能量分析的方式对SM2解密算法进行了实际攻击实验并成功恢复出了私钥. 实验结果表明,该攻击方法具有可行性和实用性.

关键词: 组合攻击;减轮故障;侧信道攻击;选择密文;安全错误;SM2解密
基金项目: 国家自然科学基金(No.62172395)
中图分类号: TN918;TP309 **文献标识码:** A **文章编号:** 0372-2112(2023)11-3187-12
电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20220481

Chosen Ciphertext Combined Attack Based on Round-Reduced Fault Against SM2 Decryption Algorithm

LI Hao-yuan[1,2], HAN Xu-cang[1,2], CAO Wei-qiong[1], WANG Jian[1,2], CHEN Hua[1]
(1. *Trusted Computing and Information Assurance Laboratory*, *Institute of Software*, *Chinese Academy of Sciences*, *Beijing* 100190, *China*;
2. *University of Chinese Academy of Sciences*, *Beijing* 100049, *China*)

Abstract: SM2 algorithm is a commercial elliptic curve cryptographic algorithm designed by China. At present, the analysis of the implementation security of this algorithm usually follows the research results on the common components of elliptic curves rather than the structure and characteristics of the algorithm. At the same time, hash and verification steps in SM2 decryption algorithm make most of the fault attacks that need to exploit the error output not applicable. To solve this problem, according to characteristics of SM2 decryption algorithm, this paper proposes a chosen ciphertext combined attack that combines the round-reduced fault with side channel based on the idea of safe-error. The core of the attack is changing the number of rounds of scalar multiplication by fault injection, and determining the specific number of faulty rounds by side channel analysis. Then it constructs the chosen ciphertext based on partial key guesses combined with plaintext and correct ciphertext. And the chosen ciphertext is input to the decryption device with specific fault effect, verifying whether the partial key guess is correct by the output of the decryption device. Also, the applicability of the attack to different scalar multiplication methods and common protection countermeasures is analyzed in the paper. Lastly, we conduct practical attack experiments on the SM2 decryption algorithm with clock glitch injection and simple power analysis on an STM32F303 microcontroller chip based on the ARM Cortex M4. And we successfully recover the private key. The experimental results show that the attack method is feasible and practical.

Key words: combined attack; round-reduced fault; side channel attack; chosen ciphertext; safe-error; SM2 decryption
Foundation Item(s): National Natural Science Foundation of China (No.62172395)

收稿日期:2022-04-29;修回日期:2022-09-07;责任编辑:朱梅玉

1 引言

信息技术和芯片技术的发展使社会逐步进入数字化、智能化时代. 智慧城市、智能家居、工业互联网等一系列基于IoT(Internet of Things)的新兴事物,在人们的生产生活中扮演着越来越重要的角色. 其中,密码学作为网络安全的核心技术之一,对保障设备的安全运行起着至关重要的作用. 近年来,随着密码理论分析的日趋完善,研究者们开始大量着眼于密码设备的实现安全性研究. 其中,侧信道攻击和故障攻击是密码实现安全性研究的两大主流研究方向. 侧信道攻击[1,2]利用密码设备在执行过程中泄露的时间、功耗、电磁等物理信息对密钥信息进行恢复. 故障攻击[3,4]利用攻击者对密码运行过程的主动干扰而产生的错误输出对秘密信息进行恢复. 除了从上述两个角度分别进行的研究之外,将侧信道攻击和故障攻击相结合的攻击方式是一种更强大的分析手段,这种攻击通常被称为"组合攻击"[5].

SM2算法[6]是我国国家密码管理局于2010年发布的自主设计的商用椭圆曲线密码算法,已被国际标准化组织(International Organization for Standardization,ISO)确立为国际标准. 由于椭圆曲线密码算法具备更高的效率和更好的安全性,在我国商用密码领域,SM2算法逐渐取代RSA成为使用最广泛的公钥密码算法. SM2算法包括签名/验签算法、加密/解密算法以及密钥交换算法. 目前,国内外对椭圆曲线密码的实现安全性研究已经有了较为丰富的成果,研究者们从侧信道攻击、故障攻击,以及将二者相结合的组合攻击的角度,研究出了一套较为完整的攻击和防护框架. 作为椭圆曲线密码中核心的通用组件,在大多数情况下,对标量乘模块的攻击和防护通常可以较为容易地迁移到SM2算法中.

本文主要针对SM2解密算法实现安全性进行攻击研究. 在SM2解密算法中,私钥作为标量乘的标量参与运算. 目前,针对标量乘已经形成了较为有效的防护体系,因此直接针对SM2解密中的标量乘实施攻击难度较大. 同时,为了达到抵抗自主选择密文攻击(Adaptive Chosen Cipertext Accack,CCA2)的安全性[7],SM2解密算法中添加了哈希和验证步骤,算法的特点决定了在故障注入条件下攻击者无法获得错误的密文输出,这使大部分需要利用错误输出的故障攻击方式对于SM2解密算法并不适用. 由于上述原因,目前学术界中对SM2解密算法的实现安全性研究通常仅遵循对椭圆曲线通用组件的研究成果,然而这并不能说明SM2解密算法在实现安全性上已经有了足够的保障. 还可以从SM2解密算法本身的结构和特点出发进行研究,如史汝辉等人[8]通过对SM2解密过程中的哈希算法进行能量分析得到解密后的明文便是利用了SM2解密中存在哈希运算的特点.

为了解决SM2解密算法实现安全性研究的瓶颈,本文从SM2解密算法的整体结构出发,提出了一种基于减轮故障的组合攻击方法. 该攻击方法利用故障注入和侧信道信息的泄露,构建输入解密算法的选择密文组,结合安全错误类故障攻击思想逐步恢复解密私钥值. 本文的贡献和创新点主要包括以下几点.

(1)突破了SM2解密算法中哈希和验证步骤导致的实现安全性防护难点. 结合安全错误类故障攻击思想,针对SM2解密算法中存在的验证步骤,提出了一种恢复SM2解密私钥的选择密文组合攻击方法.

(2)将减轮故障应用于SM2解密算法,且攻击方法可适用于目前大多数采用通用标量乘算法和防护方案的SM2解密算法.

(3)该攻击方法具有较强的实用性和可行性. 在32位ARM平台上对该攻击方法进行了实验验证,结合时钟毛刺注入和简单能量分析成功恢复了SM2解密算法中的私钥.

2 相关工作

目前,对椭圆曲线算法的实现安全性研究主要从侧信道分析和故障分析两个角度着手. 在侧信道分析方面,主要研究成果包括计时攻击、能量攻击及其相应的防护对策等. 对于椭圆曲线算法,计时攻击通常是利用了标量乘过程中秘密标量与执行时间的关系,如根据不同标量比特位的对应时间关系逐比特恢复标量信息[9],以及利用标量乘执行时间与标量长度的线性关系获得标量的高位比特泄露[10]等. 对椭圆曲线算法的能量攻击主要包括简单能量分析[11]、模板攻击[12]、垂直能量分析[如DPA(Differential Power Analysis)[11]、RPA(Refined Power Analysis)[13]、ZPA(Zero-value Point Attack)[14]、Doubling Attack[15]]以及水平能量分析[16]等. 相应的侧信道防护对策通常包括采用确定时间分支平衡的标量乘运算、基点掩码、标量随机化、坐标随机化以及随机化同构曲线等[11,17].

在故障分析方面,对椭圆曲线算法的攻击主要分为安全错误类故障攻击[18]、弱曲线类故障攻击[19]、差分类故障攻击[19]等. 与侧信道分析相比,故障攻击需要更高的攻击条件,因此通常具有更大的威胁和攻击力. 其中,安全错误类故障攻击的思想是通过故障是否影响算法的输出来判断秘密信息,如计算安全错误攻击[18]和内存安全错误攻击[20]. 本文提出的攻击结合了该思想,通过分析故障注入后解密算法的输出情况对私钥进行恢复. 对于上述常见故障攻击,故障防护对策通常包括一致性校验、参数完整性校验和点校验等[19]. 此外,本文利用的减轮故障注入在对称密码中是一个常

见的故障攻击手段,因为对称密码中对相同的序列重复执行足够的轮数是确保密码理论安全的关键. 减轮攻击是Choukri等人[21]针对AES首次提出的,之后又多被用于分组密码和哈希函数分析[22,23]. 实际上,减轮攻击在公钥算法中同样可以产生强大的威胁,如本文将该故障攻击手段运用在解密算法的标量乘中. 将减轮故障应用在SM2解密算法,结合算法本身特点,可以构造出一种全新的高效攻击方法.

除了分别从侧信道和故障两个角度的分析外,组合攻击是一种更强大的安全威胁. 组合攻击的思路通常是通过故障注入影响算法的执行,同时结合被扰乱的执行所泄露的侧信道信息进行秘密恢复. 对于公钥密码算法,组合攻击通常是针对RSA中的模幂运算,或者ECC(Elliptic Curve Cryptography)中的标量乘运算而设计的. Amiel等人[5]针对RSA中使用原子防护和故障校验的模幂运算首次提出了组合攻击方法,同时指出该攻击方式可以扩展到椭圆曲线密码算法中. Fan等人[24]针对ECC算法提出了一种组合攻击方法. 他们通过故障注入使参与椭圆曲线运算中的点成为低阶点,然后根据标量乘运算过程泄露中间值成为无穷点的侧信道信息推导出秘密标量. Feix等人[25]针对一种可以抵抗组合攻击的模幂实现提出了新的组合攻击方式. 他们利用指令跳过故障和能量分析的组合,成功绕开防护限制而实施攻击. 罗鹏等人[26]提出了一种不需要错误构造低阶点的选择明文攻击方法,该方法可以有效地用于带有参数校验防护的ECC实现. 可以看出,组合攻击往往可以在单一类攻击失效时发挥其作用. 在本文所研究的组合攻击中,通过侧信道分析对故障注入产生的故障效果获得精准的确认是攻击得以成功实施的关键.

3　基础知识

3.1　椭圆曲线和标量乘算法

在椭圆曲线算法中,椭圆曲线通常基于素数域或二元扩域. 在SM2算法中,使用的是素数域上的256位椭圆曲线. 曲线E是素数域F_p上的简化Weierstrass曲线,其中,p是大于3的素数,且$\left(4a^3+27b^2\right)\bmod p\neq 0$. 该曲线可以表示为

$$E: y^2=x^3+ax+b,\ a\in F_p,\ b\in F_p \tag{1}$$

其中,a和b是曲线参数. 该曲线上的点可以构成交换群,点与点之间可以进行点加和倍点运算. 在由点构成的交换群上可以实现SM2中的核心运算标量乘,即多倍点运算. 标量乘运算是对椭圆曲线上的点进行的多次累加运算,如对于标量乘运算kG(k为标量,G为椭圆曲线上的点),等价于k个椭圆曲线上的点G依次进行点加运算. 常见的标量乘运算一般基于对标量的比特位的扫描,根据扫描的方式可以分为逐位扫描的二进制算法(binary algorithm)和逐窗口扫描的窗口算法(window algorithm). 常见的二进制标量乘算法包括Double-and-Add算法、蒙哥马利阶梯法等. 而窗口算法是在系统存在更大可用内存时的一种时间优化算法. 通过使用更多的内存存储预计算的值,算法可以在每次循环过程中计算相应窗口长度的比特位对应的运算,进而可以减少循环次数,减少计算时间. 算法1和算法2分别是从右向左和从左向右的二进制标量乘Double-and-Add always算法,算法3为蒙哥马利阶梯法,其中,∞表示椭圆曲线上的无穷远点. 算法1~3均为确定时间的可抵抗简单能量分析和计时攻击的标量乘算法.

算法1　从右向左的二进制标量乘Double-and-Add always算法[11]

输入: $k=(k_{t-1},\cdots,k_1,k_0)_2,\ G\in E$

输出: kG

1. $Q_0\leftarrow\infty,\ Q_1\leftarrow\infty,\ Q_2\leftarrow G$
2. FOR $i=0\rightarrow t-1$
3. $\quad Q_{1-k_i}\leftarrow Q_{1-k_i}+Q_2$
4. $\quad Q_2\leftarrow 2Q_2$
5. 返回 Q_0

算法2　从左向右的二进制标量乘Double-and-Add always算法[11]

输入: $k=(k_{t-1},\cdots,k_1,k_0)_2,\ G\in E$

输出: kG

1. $Q_0\leftarrow\infty,\ Q_1\leftarrow\infty$
2. FOR $i=t-1\rightarrow 0$
3. $\quad Q_0\leftarrow 2Q_0$
4. $\quad Q_{1-k_i}\leftarrow Q_{1-k_i}+G$
5. 返回 Q_0

算法3　蒙哥马利阶梯算法[27]

输入: $k=(k_{t-1},\cdots,k_1,k_0)_2,\ G\in E$

输出: kG

1. $Q_0\leftarrow\infty,\ Q_1\leftarrow G$
2. FOR $i=t-1\rightarrow 0$
3. $\quad Q_{1-k_i}\leftarrow Q_0+Q_1$
4. $\quad Q_{k_i}\leftarrow 2Q_{k_i}$
5. 返回 Q_0

3.2　SM2加解密算法

3.2.1　符号定义

G是椭圆曲线上的一个基点,阶为n. h是n的余因子. 公钥为P_B,私钥为d_B,且$P_B=d_BG$. Hash(·)为哈希函数. KDF(·)为密钥派生函数,其作用是根据相应的输入坐标值,生成一个与明文长度相同的密钥t,用于对明文或密文进行加解密.

在加密算法中,M为待加密的消息,klen为M的比特长度,C为加密得到的密文,其中C_1,C_2,C_3分别为密文的三个部分;在解密算法中,C'为待解密的密文,其

中C_1'、C_2'、C_3'分别为密文的三个部分，M'为解密得到的消息.

3.2.2 算法描述

在SM2加密算法[7]中，加密者使用公钥P_B将消息M加密成密文C；在SM2解密算法中，解密者使用私钥d_B将密文C'解密为明文M'. SM2加密算法如算法4所示，SM2解密算法如算法5所示.

算法4 SM2加密算法

输入: M,klen

输出: $C=C_1\|C_3\|C_2$

1. 用随机数发生器产生随机数$k\in[1,n-1]$
2. 计算椭圆曲线点$C_1=kG=(x_1,y_1)$,将C_1的数据类型转换为比特串
3. 计算椭圆曲线点$S=hP_B$,若S是无穷远点,则报错并退出
4. 计算椭圆曲线点$kP_B=(x_2,y_2)$,将坐标x_2和y_2的数据类型转换为比特串
5. 计算$t=\mathrm{KDF}(x_2\|y_2,\mathrm{klen})$,若$t$为全0比特串,则返回步骤1
6. 计算$C_2=M\oplus t$
7. 计算$C_3=\mathrm{Hash}(x_2\|M\|y_2)$
8. 输出密文$C=C_1\|C_3\|C_2$

算法5 SM2解密算法

输入: $C'=C_1'\|C_3'\|C_2'$,klen

输出: M'

1. 从C'中取出比特串C_1',将C_1'的数据类型转换为椭圆曲线上的点,验证C_1'是否满足椭圆曲线方程,若不满足则报错并退出
2. 计算椭圆曲线点$S=hC_1'$,若S是无穷远点,则报错并退出
3. 计算$d_BC_1'=(x_2,y_2)$,将坐标x_2和y_2的数据类型转换为比特串
4. 计算$t=\mathrm{KDF}(x_2\|y_2,\mathrm{klen})$,若$t$为全 0比特串,则报错并退出
5. 从C'中取出比特串C_2',计算$M'=C_2'\oplus t$
6. 计算$u=\mathrm{Hash}(x_2\|M'\|y_2)$,从$C'$中取出比特串$C_3'$,若$u\neq C_3'$,则报错并退出
7. 输出明文M'

3.2.3 SM2公钥加解密算法的安全性

SM2加解密算法具备抵抗自主选择密文攻击(CCA2)的安全性. CCA2是针对密文进行破解的一种攻击假设. 在CCA2模型中，攻击者任何时候都可以访问加密预言机和解密预言机，根据所掌握的信息和资源对想破解的密文给出一个答案. 为了保障SM2加解密算法具备CCA2下的理论安全性，设计者使用Hash函数的方法对与秘密信息和明文进行验证，一旦解密预言机发现错误，则拒绝输出任何信息，如算法4加密过程中的第7步和算法5解密过程中的第6步. 这样的设计，使攻击者在算法运行正常的情况下只能通过加密预言机获得有效的密文，无法伪造出任何有效的密文，保障了SM2加解密算法抵抗CCA2的安全性.

从实现安全性的角度考虑，基于Hash函数的验证方法也可以对故障攻击起到有效的防护作用. 在大部分故障攻击中，攻击者需要通过对算法的故障输出进行分析，来恢复秘密信息的值. 而在算法5的第6步中的哈希和验证的步骤使算法在解密出现错误时拒绝输出，这可以避免绝大多数潜在的故障攻击威胁.

4 选择密文组合攻击

如前文所述，对SM2解密算法的实现安全性研究较少；同时，SM2解密算法中为保证CCA2安全的验证步骤使攻击者无法获得错误输出，使需要依据故障结果的攻击无法实施. 针对以上问题，本文结合安全错误类故障攻击思想，提出了基于减轮故障的选择密文组合攻击. 本节将分别从攻击模型和原理、攻击流程、攻击分析的角度对本攻击进行说明.

4.1 攻击模型和原理

4.1.1 攻击模型

本文提出的针对SM2解密算法的选择密文组合攻击，在攻击过程中需要使用故障攻击和侧信道分析的方法. 结合实际的攻击情况，本攻击将基于以下模型与假设.

(1)攻击者了解加解密运算实现的方式，包括算法中子运算，如标量乘、哈希等运算的实施方案.

(2)攻击者可访问解密预言机，对输入的任何选择密文组都可以按照SM2解密算法执行解密.

(3)攻击者了解算法执行设备，在执行解密的过程中，攻击者可以对解密设备中的目标标量乘注入减轮故障，使标量乘循环的轮数发生整体性的减少，即故障循环过程中扫描的标量比特位为连续高位或连续低位.

(4)在已知故障效果的注入条件下，相同的故障效果可以复现. 对于减轮故障效果(故障注入后标量乘中的循环轮数)，攻击者可以从获取到的侧信道信息的分析中获得.

在SM2解密(算法5)中，在未进行标量随机化防护的情况下，私钥$d_B\in[1,n-1]$. 标量随机化防护方法是指，通过使标量与阶n的随机倍数相加，对标量进行等价变换，使随机化后的标量与原始标量在标量乘运算中可以得到相同的结果. 也就是说，当使用了该防护后，私钥的取值范围将变大. 令d_B的比特长度为dlen，则d_B的二进制形式可以表示为$d_B=(d_{\mathrm{dlen}-1},d_{\mathrm{dlen}-2},\cdots,d_1,d_0)_2$，第3步的标量乘为$d_BC_1'=(x_2,y_2)$，对应于dlen个循环. 考虑一种具体的减轮攻击：当进行故障注入后，在执行了f个循环后，标量运算提前结束，此时运算结果将变为(x_2',y_2'). 以上文中的标量乘算法1~3为例进行分析. 其中，式(2a)为算法1从右向左扫描方式对应的故障结果，仅执行低f位对应的运算；式(2b)为算法2和算法3从左向右扫描方式对应的故障结果，仅执

行高f位对应的运算.

$$\begin{cases}\left(x_2',y_2'\right)=\left(d_{f-1},d_{f-2},\cdots,d_0\right)_2C_1' & (2a)\\ \left(x_2',y_2'\right)=\left(d_{\mathrm{dlen}-1},d_{\mathrm{dlen}-2},\cdots,d_{\mathrm{dlen}-f}\right)_2C_1' & (2b)\end{cases} \tag{2}$$

上述攻击假设在实际攻击中是合理的. 首先,攻击者可以通过电压毛刺、时钟毛刺、激光或电磁等多种方式干扰设备的运行,使某些特定指令被跳过,或相关寄存器的值发生变化,从而达到轮数减少的攻击效果,如本文攻击实验中通过时钟毛刺跳过循环中的跳转指令而实现的减轮故障. 同时,在相应的攻击环境下,确定了某一故障效果对应的注入条件(注入时间、故障参数等)后,特定的故障效果可以被复现. 此外,在设备运行过程中,执行轮数与能量消耗和执行时间之间存在相关性,攻击者可以通过简单能量分析或计时的方式分析标量乘运算过程的循环执行轮数,从而确认减轮故障效果(标量乘的执行轮数).

4.1.2　攻击原理

基于上述攻击模型,结合安全错误类故障攻击的基本思想即可构造选择密文组合攻击. 安全错误类故障攻击的思想是通过算法输出正确与否对私钥信息进行判断. 本攻击通过减轮故障注入,使解密算法中的标量乘部分在故障效果下转化为较小标量的标量乘法,使攻击者可以在有限的次数内,通过迭代的方式对解密私钥进行分段逐步猜测,以恢复完整的解密私钥. 下面从安全错误的产生、选择密文组的构建以及减轮故障下的恢复等方面对攻击原理进行介绍.

(1)安全错误的产生. 从SM2加解密算法(算法4和算法5)中可以看出,在无故障的情况下,正确的解密输入(由加密算法得到)会使解密算法中的u与加密算法中C_3满足$u=C_3$;当解密算法中的第3步标量乘法被注入减轮故障时,正确的解密输入必然使$u\neq C_3$. 因此,解密算法可以以此来判断解密执行是否被减轮攻击所影响. 然而,由于u是根据解密运算中第3步的标量乘法结果以及解密明文M'计算得来的,如果攻击者已知故障注入下解密第3步的标量乘故障结果,那么攻击者便可以依据标量乘故障结果构建特定的选择密文组作为解密算法的输入,从而满足验证步骤$u=C_3'$,即解密算法可以正确输出.

(2)选择密文组的构建. 应用于该组合攻击的选择密文组是根据减轮故障下的标量乘故障结果而构建的. 在获得SM2加密得到密文C的基础上,攻击者可以使用加密算法(算法4)中的第5~7步计算故障标量乘结果所对应的密文组,以此作为特定减轮故障下的解密输入,可以使解密算法正确输出.

(3)减轮故障下的恢复. 攻击中对解密私钥的恢复是迭代进行的,恢复的关键在于私钥比特的猜测和故障效果的确认. 攻击者通过调整故障注入参数选取适用于本攻击的特定故障效果(故障执行轮数),并记录相应的故障注入参数. 其中,故障执行轮数的选择应综合考虑设备能力(详见第4.3.1节). 在已知的特定减轮故障下,标量乘的故障结果仅与标量部分比特位相关. 攻击者可以通过多次猜测相关的私钥部分比特位,结合输入的密文C_1'得到相应的故障标量乘结果,并按照上述方式构建选择密文组. 在执行相同故障下进行解密,若产生了安全错误,即解密算法正确输出,则说明部分私钥比特猜测正确. 在此基础上,攻击者可以调整减轮故障,继续恢复私钥比特,直至私钥被全部恢复. 在故障解密算法执行过程中,为了保证特定故障注入的成功,需要结合侧信道信息进行确认.

上述攻击原理基本组成了本攻击的主体部分. 根据上述原理,可以完整地构建出本组合攻击的基本框架,如图1所示,通过循环迭代可以逐步恢复私钥比特位. 基于上述攻击原理,在第4.2节将以针对算法2和算法3的组合攻击为例,对攻击流程进行详细介绍.

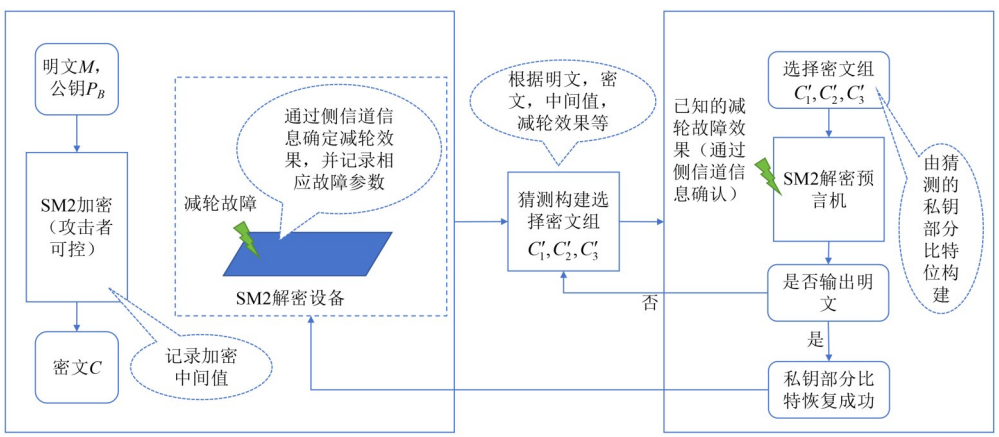

图1　选择密文组合攻击原理和流程示意图

4.2 攻击流程

以上述攻击模型和原理为基础,本节将介绍选择密文组合攻击的基本流程.针对使用不同标量乘的SM2解密算法,攻击流程中的故障标量乘结果以及选择密文的构建方法会有所不同.本文以使用算法2或算法3的标量乘法的解密算法为例,给出攻击流程以及选择密文的构建过程.其中,算法2和算法3均使用了确定时间从左向右的标量扫描方式.

4.2.1 符号定义

本章中所使用的符号在第3.2.1节的符号定义基础上进行额外定义.s是单次恢复私钥的最大长度,dlen是私钥d_B的比特长度,w是未恢复的私钥比特长度,r是已恢复的私钥比特长度,f是实际的标量乘执行轮数,Δ是每次恢复时所恢复的私钥比特长度.

4.2.2 攻击步骤

在对SM2解密算法的选择密文组合攻击中,将攻击算法分为三个部分,如图2所示.第一部分是对选择的消息进行加密,记录输入明文和输出密文.第二部分主要包括寻找合适的故障效果、构建选择密文组、在故障下对选择密文执行解密等步骤,逐步恢复私钥比特.在这一部分,攻击者需要根据标量扫描方向和故障效果构建特定的用于私钥恢复的密文.在经过了攻击的前两部分后,私钥绝大部分的比特位已经被攻击者恢复.在我们的攻击中,预先设定了单次恢复私钥的最大比特长度s,当未恢复的私钥比特长度不大于s时,通过穷举的方法猜测尚未恢复的私钥比特位,在无故障注入的前提下以原始密文作为解密算法的输入,当解密正确执行时,则说明私钥猜测正确,即恢复出完整的私钥.

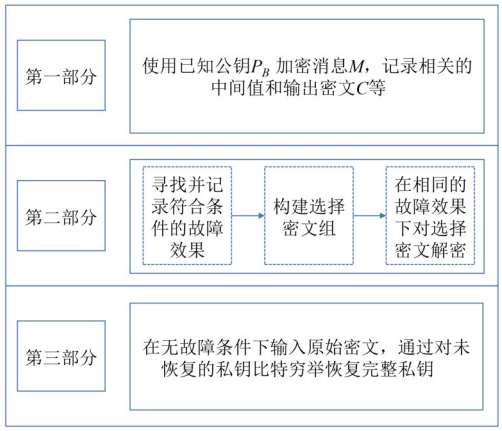

图2 选择密文组合攻击的主要阶段

在算法6中,以伪代码的形式详细介绍了针对从左到右的标量扫描方向,适用于标量乘为算法2或算法3的SM2解密算法的组合攻击方法.其中,攻击者从私钥的高位开始对私钥进行迭代恢复.在算法6,第1步对应图2中的第一部分;第2~12步对应图2中的第二部分;第13步对应图2中的第三部分.

算法6 针对SM2的组合攻击算法(从左到右的标量扫描方向)

输入: M,klen,s, dlen

输出: d_B

1. 使用算法4对明文消息M执行SM2加密,得到中间值(x_2, y_2)以及输出密文C_1,C_2,C_3
2. w = dlen, r=0. // w为未恢复的私钥比特长度,r为已恢复的私钥比特长度
3. $(x_2', y_2') = \infty$
4. WHILE $w > s$:
5. $Q = (x_2', y_2')$
6. (寻找合适的注入条件)在注入条件T,对解密设备中进行减轮故障注入,通过采集能量观察确定标量乘执行轮数f是否满足$f-r\leqslant s$.若满足,记$\Delta=f-r$;若不满足返回重新执行此步骤
7. $C_1' = C_1$
8. FOR $m = (0)_2 \rightarrow (2^{\Delta}-1)_2$ //下标2表示m对应此范围内的二进制形式
9. $(x_2', y_2') = 2^{\Delta}Q + mC_1'$
10. t= KDF(x_2'||y_2',klen)
11. $C_2' = M \oplus t$. C_3'= Hash (x_2'||M|| y_2')
12. 对$C' = C_1'||C_3'||C_2'$在执行轮数为f的故障效果下执行解密(通过能量采集或计时攻击确认故障效果),若解密运算正常输出,说明m是私钥比特的第$w-\Delta+1$位到第w位,令$w=w-\Delta$,$r=r+\Delta$,跳出循环
13. 对未恢复的长度为w的私钥部分比特进行穷举,对原始密文C_1,C_2,C_3执行解密,直至可以成功解密,并恢复出整个私钥d_B

在算法6中,攻击者通过第4~12步的循环迭代方式对私钥逐步进行恢复.其中,在每次恢复中,通过部分比特位的猜测对故障标量乘结果的计算以及选择密文组的构建为第7~11步.在攻击过程中,Q用来保存已恢复的标量位对应的标量乘结果.同时,在每次恢复中,选择密文C_1'均选择为加密对应的结果C_1.假设待恢复私钥$d_B=(d_{\text{dlen}-1},d_{\text{dlen}-2},\cdots,d_1,d_0)_2$,在进入算法6的某次循环恢复时,已恢复的私钥部分比特位$d_i=(d_{\text{dlen}-1},d_{\text{dlen}-2},\cdots,d_{\text{dlen}-r})_2$.在第5步赋给$Q$点的坐标值为$Q=d_iC_1'$(已恢复比特位与$C_1'$的标量乘结果).然后,攻击在第6步确认了本次循环要恢复的私钥比特位数为Δ以及故障执行轮数$f=\Delta+r$.接下来,攻击将对Δ位的部分私钥比特进行猜测(第8~12步),对任一猜测$m=(m_{\Delta-1},m_{\Delta-2},\cdots,m_0)_2$,以第9步中的方式构建猜测故障标量乘结果,$\left(x_2',y_2'\right)=2^{\Delta}Q+mC_1'=\left(2^{\Delta}d_i+m\right)C_1'=\left(d_{\text{dlen}-1},d_{\text{dlen}-2},\cdots,d_{\text{dlen}-r},m_{\Delta-1},m_{\Delta-2},\cdots,m_0\right)_2C_1'$.由于此时故障执行轮数$f=\Delta+r$,因此故障标量乘结果应为$(d_{\text{dlen}-1},d_{\text{dlen}-2},\cdots,d_{\text{dlen}-r},d_{\text{dlen}-r-1},\cdots,d_{\text{dlen}-r-\Delta})_2C_1'$.因此,当$m$的猜测满足$(m_{\Delta-1},m_{\Delta-2},\cdots,m_0)_2=(d_{\text{dlen}-r-1},\cdots,d_{\text{dlen}-r-\Delta})_2$时,攻击者可以得到正确的故障标量乘结果.因此,根

据第4.1.2节中的攻击原理,可以知道该猜测对应的故障标量乘结果构建的选择密文组将使具有相同故障效果的解密算法正确输出,由此便可以得到比特长度为Δ的私钥比特位. 循环执行上述恢复过程,直到未恢复的私钥比特$w\leqslant s$,便可以通过第13步中方法恢复出整个私钥.

4.3 攻击分析

4.3.1 攻击效率

从实现角度看,攻击效率主要取决于寻找合适的注入条件所需要的时间、在分段恢复密钥时进行故障注入和恢复结果确认所消耗的时间,以及最后对分段恢复后剩余的私钥比特进行恢复的时间. 对于每段比特长为Δ的密钥,攻击需要循环执行2^{Δ}次SM2解密运算,并在每次解密时成功注入故障,当解密输出正确结果时可提前跳出循环. 为了更准确地分析关键参数对攻击效率的影响,定义单次恢复私钥的最大比特长度为s,对于长度为dlen的私钥,攻击者至少需要对$\lfloor \mathrm{dlen}/s \rfloor$个分段进行注入条件的寻找和部分密钥的恢复,同时需要对剩余的长为$\mathrm{dlen}-\lfloor \mathrm{dlen}/s \rfloor \cdot s$的私钥比特进行穷举恢复. 不考虑在每段恢复中提前跳出的情况,每段恢复密钥的过程中最多需要进行2^s次成功的故障注入和解密运算. 其中,每次故障注入的成功率为p. 假设平均每次寻找注入条件的时间为t_1,每次故障注入和结果确认的执行时间为t_2,单次解密执行时间为t_3,则总消耗的时间T满足

$$T=\left\lfloor \frac{\mathrm{dlen}}{s} \right\rfloor t_1+\frac{1}{p}\left\lfloor \frac{\mathrm{dlen}}{s} \right\rfloor 2^s t_2+2^{\mathrm{dlen}-\left\lfloor \frac{\mathrm{dlen}}{s} \right\rfloor s} t_3 \qquad (3)$$

假设t_1,t_2,t_3仅受设备能力的影响,从式(3)可以看出,攻击效率与参数$s(s\geqslant 1)$和$p(0<p\leqslant 1)$密切相关. 显然,注入成功率p越大,则需要的故障注入次数越少. 由于故障注入成功率通常是由设备以及攻击者对故障参数的选择决定的,因此攻击者应在已有条件下尽可能提高注入成功率p. 此外,s越大则寻找注入条件的次数越少,但每段恢复中的故障注入和结果确认的次数也越大. 因此,对于参数s的设置,应综合考虑故障注入和运算设备的能力(即式(3)中的t_1和t_2). 通常情况下,如果使用故障设备每次寻找注入条件的时间t_1较小,那么将s设置更小将提高我们的攻击效率. 如果t_1较大,攻击者可以适当设置较大的s以减少注入条件的寻找次数,但必须充分考虑上式中第二部分的可解性.

4.3.2 攻击适用性分析

由上文中对攻击原理和步骤的描述,我们知道该攻击得以实施的前提是对SM2解密算法中的标量乘法实施可以通过能量或计时等侧信道信息识别的减轮故障注入,使标量乘法的执行结果仅与标量的部分高比特位或低比特位相关. 对于基于标量扫描的标量乘法,包括第3.1节中所介绍的逐比特扫描的标量乘算法和逐窗口扫描的标量乘算法,均是通过对标量比特位逐次判断而循环进行的. 若减轮故障使循环提前结束,会使标量乘法的执行结果仅与标量的部分高比特位或低比特位相关. 因此,对于SM2解密算法,该攻击几乎适用于所有基于标量扫描的标量乘算法. 攻击得以实施的核心就是在减轮故障下可以通过对私钥部分比特位的猜测来计算标量乘法的故障结果.

此外,从上述攻击步骤的描述可以看出,故障标量乘结果的构建需要与标量乘实现方式紧密结合,是在充分考虑标量乘的循环过程中的计算逻辑的基础上得到的. 因此,在第4.1节中提出的攻击原理的指导思想下,对于不同的标量乘实现,在应用本组合攻击时,需要结合其实现方式设计出相应的故障标量乘结果的计算方法.

对于常见的防护方案,该攻击同样具备较高的适用性. 常用的故障防护方案,如参数完整性校验、点校验、一致性校验等,均无法起到有效的防护效果. 由于减轮故障注入所产生的错误不会改变椭圆曲线的参数值,因此用来检测参数正确性的参数完整性对该攻击无效;由于减轮故障注入仅使标量乘运算的循环次数减少,而每次循环中运算仍然是正常执行的,因此整个计算过程中的中间点均是在椭圆曲线上的,点校验防护对该攻击无效;一致性校验通常用来检测多个变量之间所具备的恒定的关系是否保持,而减轮故障注入仅使轮数减少,恒定关系不会因为轮数的减少而发生变化,因此一致性校验防护对该攻击无效.

在侧信道防护方案中,大多数防护对该攻击无法起到效果. 对于私钥参与的标量乘运算,在第2节中所提到的侧信道防护方法中,包括分支平衡的标量乘运算、基点掩码、坐标随机化以及随机化同构曲线等,都不会影响在标量乘循环过程中的输出与私钥比特位间的关系,减轮故障仍然会使故障标量乘结果与私钥确定的部分位相关,因此上述侧信道防护方案无法对攻击起到有效的防护.

此外,对于另一种常见的侧信道防护方案标量随机化,攻击同样具备一定的威胁. 在未防护的情况下,通常私钥$d_B\in[1,n-1]$,而在防护下,私钥的范围会因随机数的存在而超出该范围. 但是,对于私钥参与的标量乘运算而言,随机化后的标量与原始私钥具有相同的效力. 因此,攻击恢复出任一随机化后的标量即相当于攻击成功. 对其中一次随机化标量恢复成功的难度取决于标量随机化的范围. 具体来说,攻击者可以通过随机匹配的方法,在使用同一随机数对私钥进行了随机化的多次解密运算中实施攻击. 假设攻击者通过首次恢复得到了某一随机化私钥d_r的部分比特位,攻击者在

此基础上继续猜测私钥比特并构建选择密文执行故障解密. 当某次解密中随机化私钥再次为d_r时,若此时私钥比特猜测正确即可解密成功,攻击者便可以此为基础继续对d_r进行恢复. 该方法仅适用于较小的随机化范围,当随机化范围较大时,难度将大大增加. 也就是说,标量随机化方案在随机范围较大时可以对攻击起到有一定的防护效果.

5 实验验证

根据前文所述的实验原理和流程可知,攻击过程中减轮故障注入的成功率对于攻击的可行性和实用性起到关键影响. 我们对在基于ARM核心的32位平台上实现的SM2解密算法进行了组合攻击实验. 实验使用时钟毛刺注入产生指令跳过错误,并通过示波器进行能量采集,结合简单能量分析来区分和验证减轮故障是否注入成功.

5.1 实验环境

本文实验中,SM2解密算法实现在基于ARM Cortex M4核心的STM32F303微控制器芯片上. 该芯片被安装在ChipWhisperer UFO主板上,通过主板上的接口,使用Lecroy HDO6034A示波器对目标设备实施能量采集. 此外,将ChipWhisperer-Lite组件分别与上位机和主板相连,ChipWhisperer-Lite产生频率约为7.38 MHz的时钟作为设备运行的主时钟,并通过该组件对目标设备生成时钟毛刺的故障注入,此故障通过对目标指令运行时的时钟频率进行干扰,使指令执行错误,产生指令跳过的效果. 基本的实验环境如图3所示,其中从①到④分别为含有STM32F303微控制器芯片的目标设备、ChipWhisperer UFO主板、ChipWhisperer-Lite组件、Lecroy HDO6034A示波器.

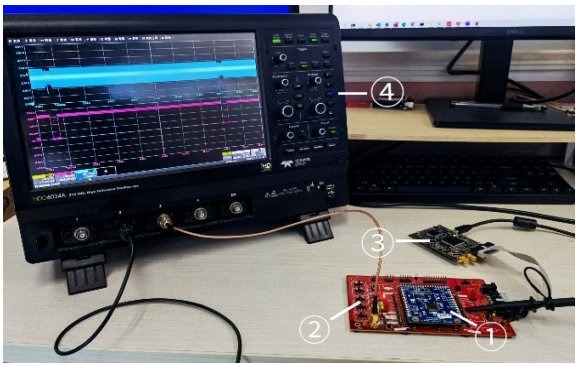

图3 选择密文组合攻击实验环境

5.2 实验原理

实验中目标设备实现的SM2解密算法是基于开源密码库micro-ecc. 其中,标量乘法使用算法3蒙哥马利阶梯法的变形,是一种恒定时间的标量乘法. 如前文所述,攻击应充分结合解密算法中标量乘实现的方式和计算逻辑,从而设计出故障标量乘结果的计算方法. 在该标量乘实现中,标量的扫描方向与算法3所示的蒙哥马利阶梯法相同,均是从标量高比特向标量低比特进行循环运算,区别是该实现的标量最高比特和最低比特的相关运算放在循环外进行,也就是说减轮故障不会影响最高比特和最低比特的运算,对该算法的减轮注入的示意图如图4所示,其中,$Q^{(i)}$表示私钥比特d_i对应的运算. 因此,对于标量乘运算dC_1,正常情况下循环共包括dlen−2轮,当注入减轮故障使循环轮数变为f时,得到的$d'C_1$如图4所示.

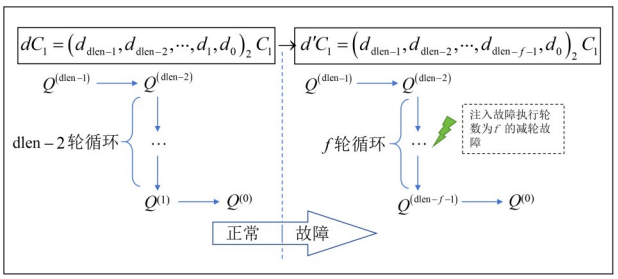

图4 减轮故障注入示意图

攻击采用算法6所示的从高位开始对私钥的恢复方式. 由于私钥最低比特位在循环外计算,减轮故障注入后的故障标量乘结果如上所示. 因此我们在通过猜测部分私钥比特位m构建故障标量乘结果时,需要考虑私钥最低比特位d_0在标量乘计算中的影响. 假设第i次恢复得到的故障标量乘结果为Q_i,应通过式(4)的方式构建第i+1次标量乘结果Q_{i+1}. 其中,Δ为第i+1次恢复的私钥比特位数,m为对Δ长度比特位对应数值的猜测.

$$Q_{i+1}=2^{\Delta}Q_i-\left(2^{\Delta}d_0-2m-d_0\right)C_1 \tag{4}$$

在本实验中,我们通过ChipWhisperer-Lite组件提供的时钟毛刺注入实现单步指令跳过的攻击效果. 攻击过程中,注入的时钟毛刺会干扰目标设备的运行主频. 在合适的故障参数设置下,时钟毛刺会使注入时刻相关的指令未执行,而利用标量乘循环过程中跳转指令的跳过错误可以有效地实施减轮故障注入. 在循环过程中,当跳转指令被跳过后,程序将跳出循环,实现减轮的效果. 图5中所标注的地址“8001d2e”对应的指令即为本实验中时钟毛刺注入影响的目标指令. 该指令为跳转指令,在循环未结束时,经过该指令后将重新进入循环. 该指令被跳过将导致循环结束,跳出循环.

在攻击过程中,我们通过设置合理的故障注入参数,在一个特定的区间内使用扫描的方式注入指令跳过故障,并通过示波器对目标设备进行能量采集来分析减轮故障注入效果. 在找到预期的减轮故障效果后,

```
8001d2a:   9b02        ldr   r3, [sp, #8]
8001d2c:   2b00        cmp   r3, #0
8001d2e:   d16e        bne.n 8001e0e <EccPoint_mult+0x126>
        XYcZ_addC(Rx[1-nb], Ry[1-nb], Rx[nb], Ry[nb]);
        XYcZ_add(Rx[nb], Ry[nb], Rx[1-nb], Ry[1-nb]);
    }
    nb = !vli_testBit(p_scalar, 0);
8001d38:   783b        ldrb  r3, [r7, #0]
8001d3a:   43db        mvns  r3, r3
8001d3c:   f003 0301   and.w r3, r3, #1
8001d40:   9303        str   r3, [sp, #12]
```

图5　指令跳过故障注入目标指令

我们通过相关私钥比特位的猜测,并使用上述故障标量乘结果的构建方式构建选择密文组. 将该密文组输入至目标设备的解密算法,并在已知的故障参数设置下尝试复现该减轮效果,通过示波器的能量采集对减轮效果进行确认.

5.3　实验关键环节验证

在攻击实验中,综合考虑计算设备的性能和减轮故障注入的精度,设置单次最大恢复长度s=8. 在执行加密算法的基础上,本节将对实验的三个关键环节结果进行展示. 第一部分是对无故障的解密进行能量采集,了解和认识曲线特征. 第二部分是循环恢复私钥比特位的过程和结果,包括对故障注入条件的寻找、故障注入成功率的分析,以及通过能量曲线对故障效果的确认等. 第三部分是对实验最终结果的恢复和总结.

5.3.1　无故障解密曲线的采集

在攻击过程中,我们首先使用50 M的采样率对正常执行的解密算法进行能量采集,通过简单能量分析的方式我们可以判断出标量乘中循环共254轮,即待恢复私钥比特长度为256(最高和最低比特在循环外处理). 图6为经过低通滤波的无故障注入的能量曲线,其中能耗较高的部分为目标标量乘区间. 可以看出,在红色虚线框内的曲线模式在整个波形中周期性出现,对应标量乘每一个循环内的运算.

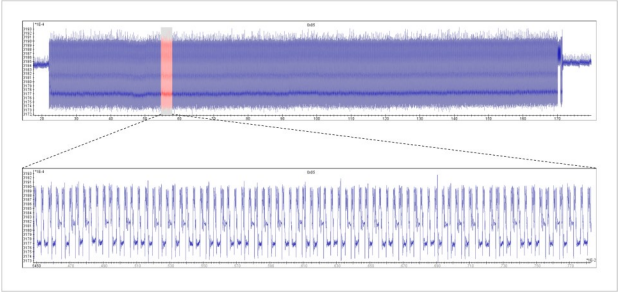

图6　无故障注入的能量曲线

5.3.2　循环恢复私钥比特位

在循环恢复私钥比特位的过程中,我们的攻击是从私钥高位开始恢复的,因此恢复的循环中故障执行轮数f应逐渐增加. 不同的故障执行轮数对应不同的注入区间. 时钟毛刺注入设置参数主要包括触发外部延时(ext-offset)、重复次数(repeat),以及时钟毛刺宽度(width)等. 表1所示为时钟毛刺注入中关键故障参数的功能和取值范围.

表1　关键故障参数

故障参数	功能	取值范围
触发外部延时(ext-offset)	表示接收到触发到注入时钟毛刺之间的时钟延时周期数	$[0, 2^{32}]$
重复次数(repeat)	表示在连续时钟周期内重复注入时钟毛刺的个数	[1, 8 192]
时钟毛刺宽度(width)	表示时钟毛刺在一个周期中所占的宽度百分比	[-49.8, 49.8]

下面以故障执行轮数f=5为例,介绍时钟毛刺注入参数的选择、参数对注入成功率的影响以及部分私钥比特的恢复. 当故障执行轮数f=5时,我们在第六轮开始执行时注入时钟毛刺,使标量乘循环提前结束. 在本实验中,为了真实模拟攻击场景,我们将毛刺注入的触发设置在解密的开始,通过触发外部延时的设置选择时钟毛刺的注入时刻,因此触发外部延时是本实验最关键的参数之一.

对于图6所示的SM2解密算法无故障执行时的能量曲线,通过周期性出现的曲线模式,我们判断出循环第六轮开始时刻位于解密开始后的363 ms附近,结合实验芯片精确的频率7.384 615 MHz,可以估算出大致的周期数为2 680 000. 因此,为了寻找准确的注入时刻,我们在触发外部延时为(2 675 000,2 685 000)的范围内进行注入扫描. 当触发外部延时为2 682 372时,故障被成功注入,获得预期的故障效果,从实时采集到的能量曲线上可以看到此时循环供执行5轮. 此外,在确定触发外部延时的过程中,由于我们没有对重复次数和毛刺宽度进行特殊的设置,而是使其在一定的范围内随机选择,所以在上述外部延时进行注入是否成功存在一定的概率. 在已知外部延时的前提下,为了确定最佳的故障注入参数,我们对不同的重复次数和毛刺宽度进行组合,得到如图7所示的时钟毛刺注入成功率结果.

由图7可知,当毛刺宽度为3,重复次数为2时,时钟毛刺注入的成功率高达79%. 由前文可知,在确定了故障执行轮数f的注入条件后,我们需要保证解密设备在选择密文输入下具备同样的故障效果,即对实验得到的故障效果进行复现. 由于算法是在恒定时间下实现的,因此不同的算法输入对算法各流程的执行时间

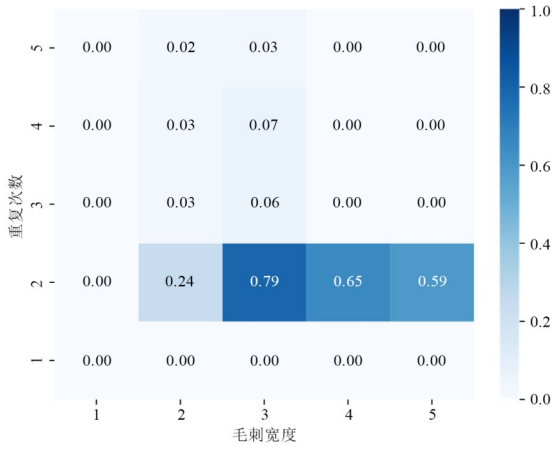

图7 时钟毛刺注入成功率

不会产生影响，也就是说上文中得到的成功率即为该故障效果的复现成功率. 此外，由于在复现的过程中我们必须通过能量曲线对注入是否成功进行确认，因此在注入失败时我们可以对同一选择密文输入的解密过程进行多次故障注入. 假设单次注入成功率为p，则注入t次时的成功率$P(t)$可由式(5)计算.

$$P(t)=1-(1-p)^{t} \tag{5}$$

因此，对于79%的单次注入成功率，在3次注入内成功的概率约为99.1%，即对于同一次部分密钥比特猜测构建的选择密文组，攻击者几乎最多需要3次注入即可成功得到预期故障效果. 图8为减轮故障注入成功后，经低通滤波处理的标量乘故障执行轮数为5的能量曲线. 可以看出，在虚线框内的曲线中存在明显周期性出现的曲线模式(图中红色标注的曲线部分，与图6中所示的曲线模式相同). 该虚线框内为标量乘中循环所对应的能量波形，虚线框前后能耗较高的部分为对高比特位和低比特位的处理阶段.

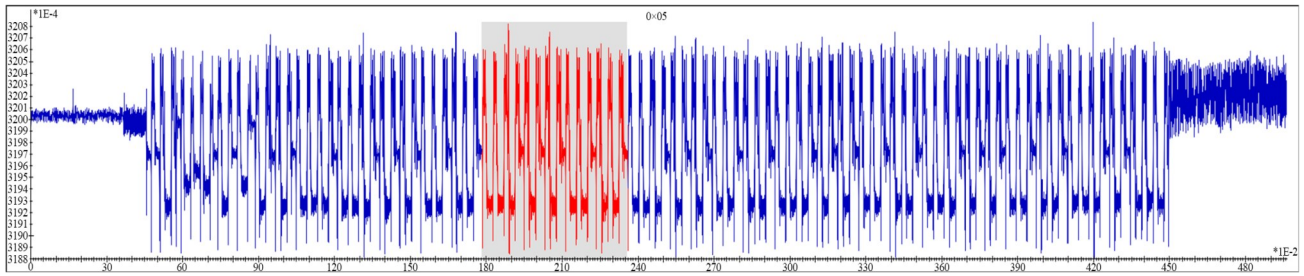

图8 故障执行轮数为5的能量曲线及单轮曲线模式

在攻击实验中，当执行轮数较多时，为了避免肉眼观测的误差，保证轮数判断的准确性，我们对每次采集到的能量曲线进行模式匹配，以判断减轮故障效果(故障执行轮数)，验证预期的减轮故障是否注入成功. 如图8所示，选中单轮能量波形(红色部分)为模板，对整个标量乘部分通过相关性计算进行模式匹配，可以得到图9所示的相关性分布图. 图中可以明显看到存在5处相关系数的尖峰，即相关性远高于其他位置的区间，这说明了该曲线对应的故障标量乘的循环执行轮数为5.

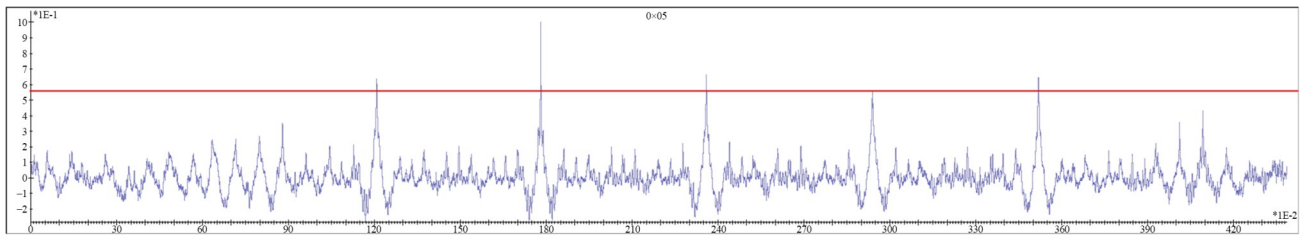

图9 单轮曲线模式匹配相关性分布图

在确认了时钟毛刺成功注入和故障效果正确的基础上，当我们对私钥部分比特m猜测正确时，可以成功观察到解密设备的正确输出，即恢复了私钥部分比特m.

5.3.3 私钥的完整恢复

以第5.3.2节中介绍的部分私钥比特的恢复为基础，在实验中逐渐增大f进行减轮故障注入，结合第4.2节所述攻击流程和第5.2节所述实验原理，即可逐步恢复出完整的SM2解密私钥. 当私钥恢复至最后，剩余待恢复私钥比特长度小于单次最大恢复长度s时，即可在无故障的条件下，输入原始加密获得的密文C，对剩余待恢复私钥比特进行穷举，直至解密算法可以正确输出，即恢复出了完整的私钥.

由于在攻击中，私钥是分段迭代进行恢复的，后续恢复的部分密钥比特位是建立在已正确恢复的比特位的基础之上，并且每次恢复是否正确是可以进行确定

性验证的,因此,本攻击对于私钥的完整恢复以及恢复结果的正确性是可以得到完美保障的. 从上述实验结果中我们可以看出,对于每次部分密钥的猜测构建的选择密文组进行解密的过程中,单次注入成功率较高,同时通过侧信道分析对减轮故障效果进行精准的确认是保证攻击效率的重要前提.

6 结论

本文从SM2解密算法的结构和特点出发对其实现安全性进行了研究,提出了基于减轮故障的选择密文组合攻击. 在该攻击中,通过猜测私钥的部分比特位,对已知的减轮故障效果注入后的标量乘结果进行预测,以此构建选择密文,并在相同的故障效果下对选择密文执行解密操作. 结合安全错误类故障攻击的思想,利用SM2解密算法中的哈希和验证步骤是否通过,判断私钥部分比特是否猜测正确,逐步恢复解密私钥. 此外,我们对攻击效率和适用性进行了分析,结果表明攻击适用于大部分标量乘算法和防护对策. 在对攻击分析的基础上,本文使用时钟毛刺注入的指令跳过故障和简单能量分析的方法进行了实际的攻击实验,实验结果表明攻击正确且有效. 通过攻击分析和实验验证,我们可以总结出本文所提攻击得以成功实施的关键点包括确认SM2解密算法中标量乘法的实现方式,以分析该攻击是否适用;根据标量乘实现方式正确构建对故障标量乘结果进行预测的计算方式;对预期故障效果具有较高的注入成功率且可以通过侧信道分析对减轮故障效果进行精准的确认.

研究表明,SM2解密算法易受选择密文组合攻击的威胁,且对于具有类似结构的CCA2安全的椭圆曲线解密算法,本文攻击思路和方法同样具有威胁. 因此,设计和实现人员应充分考虑在相应实现环境下该攻击的威胁程度,选用适当的防护对策进行防护,如随机范围较大的标量随机化等.

参考文献

[1] KOCHER P C. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[C]//Advances in Cryptology — CRYPTO'96. Berlin: Springer, 1996: 104-113.

[2] KOCHER P, JAFFE J, JUN B. Differential power analysis [C]//Advances in Cryptology — CRYPTO'99. Berlin: Springer, 1999: 388-397.

[3] BONEH D, DEMILLO R A, LIPTON R J. On the importance of checking cryptographic protocols for faults[C]// Advances in Cryptology — EUROCRYPT'97. Berlin: Springer, 1997: 37-51.

[4] 欧庆于, 罗芳, 吴晓平, 等. 基于电压毛刺故障扰动的分组密码安全性度量方法研究[J]. 电子学报, 2021, 49(3): 417-423.
OU Q Y, LUO F, WU X P, et al. Research on the metric method for the security of the block cipher based on the voltage glitch fault disturbance[J]. Acta Electronica Sinica, 2021, 49(3): 417-423. (in Chinese)

[5] AMIEL F, VILLEGAS K, FEIX B, et al. Passive and active combined attacks: Combining fault attacks and side channel analysis[C]//Workshop on Fault Diagnosis and Tolerance in Cryptography (FDTC 2007). Piscataway: IEEE, 2007: 92-102.

[6] 国家质量监督检验检疫总局, 中国国家标准化管理委员会. 信息安全技术 SM2 椭圆曲线公钥密码算法: 第 1 部分 总则: GB/T 32918.1—2016[S]. 北京: 中国标准出版社, 2017.
General Administration of Quality Supervision, Inspection and Quarantine of the People's Republic of China, Standardization Administration of the People's Republic of China. Information Security Technology—Public Key Cryptographic Algorithm SM2 Based on Elliptic Curves: Part 1 General: GB/T 32918.1—2016[S]. Beijing: Standards Press of China, 2017. (in Chinese)

[7] 汪朝晖, 张振峰. SM2椭圆曲线公钥密码算法综述[J]. 信息安全研究, 2016, 2(11): 972-982.
WANG Z H, ZHANG Z F. Overview on public key cryptographic algorithm SM2 based on elliptic curves[J]. Journal of Information Security Research, 2016, 2(11): 972-982. (in Chinese)

[8] 史汝辉, 李增局, 杜磊, 等. 一种针对SM2解密算法的侧信道攻击方法[J]. 密码学报, 2015, 2(5): 467-476.
SHI R H, LI Z J, DU L, et al. Side channel analysis on SM2 decryption algorithm[J]. Journal of Cryptologic Research, 2015, 2(5): 467-476. (in Chinese)

[9] DHEM J F, KOEUNE F, LEROUX P A, et al. A practical implementation of the timing attack[C]//Lecture Notes in Computer Science. Berlin: Springer, 2000: 167-182.

[10] BRUMLEY B B, TUVERI N. Remote timing attacks are still practical[C]//European Symposium on Research in Computer Security. Berlin: Springer, 2011: 355-371.

[11] CORON J S. Resistance against differential power analysis for elliptic curve cryptosystems[C]//Cryptographic Hardware and Embedded Systems. Berlin: Springer, 1999: 292-302.

[12] MEDWED M, OSWALD E. Template attacks on ECDSA [C]//Information Security Applications. Berlin: Springer,

2009: 14-27.
[13] GOUBIN L. A refined power-analysis attack on elliptic curve cryptosystems[C]//Public Key Cryptography — PKC 2003. Berlin: Springer, 2002: 199-211.
[14] AKISHITA T, TAKAGI T. Zero-value point attacks on elliptic curve cryptosystem[C]//Lecture Notes in Computer Science. Berlin: Springer, 2003: 218-233.
[15] FOUQUE P A, VALETTE F. The doubling attack-why upwards is better than downwards[C]//International Workshop on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2003: 269-280.
[16] BAUER A, JAULMES E, PROUFF E, et al. Horizontal collision correlation attack on elliptic curves[J]. Cryptography and Communications, 2015, 7(1): 91-119.
[17] JOYE M, TYMEN C. Protections against differential analysis for elliptic curve cryptography — an algebraic approach[C]//Cryptographic Hardware and Embedded Systems — CHES 2001. Berlin: Springer, 2001: 377-390.
[18] SUNG-MING Y, KIM S, LIM S, et al. A countermeasure against one physical cryptanalysis may benefit another attack[C]//Information Security and Cryptology — ICISC 2001. Berlin: Springer, 2002: 414-427.
[19] BIEHL I, MEYER B, MÜLLER V. Differential fault attacks on elliptic curve cryptosystems[C]//Advances in Cryptology — CRYPTO 2000. Berlin: Springer, 2000: 131-146.
[20] YEN S M, JOYE M. Checking before output may not be enough against fault-based cryptanalysis[J]. IEEE Transactions on Computers, 2000, 49(9): 967-970.
[21] CHOUKRI H, TUNSTALL M. Round reduction using faults[J]. Fault Diagnosis and Tolerance in Cryptography 2005, 2005: 13-24.
[22] PARK J. Differential fault analysis for round-reduced AES by fault injection[J]. ETRI Journal, 2011, 33(3): 434-442.
[23] JEONG K, LEE Y, SUNG J, et al. Security analysis of HMAC/NMAC by using fault injection[J]. Journal of Applied Mathematics, 2013, 2013: 1-6.
[24] FAN J F, GIERLICHS B, VERCAUTEREN F. To infinity and beyond: Combined attack on ECC using points of low order[C]//International Workshop on Cryptographic Hardware and Embedded Systems. Berlin: Springer, 2011: 143-159.
[25] FEIX B, VENELLI A. Defeating with fault injection a combined attack resistant exponentiation[C]//International Workshop on Constructive Side-Channel Analysis and Secure Design. Berlin: Springer, 2013: 32-45.
[26] 罗鹏, 李慧云, 王鲲鹏, 等. 对ECC算法实现的选择明文攻击方法[J]. 通信学报, 2014, 35(5): 79-87.
LUO P, LI H Y, WANG K P, et al. Chosen message attacks method against ECC implementations[J]. Journal on Communications, 2014, 35(5): 79-87. (in Chinese)
[27] JOYE M, YEN S M. The Montgomery powering ladder [C]//Cryptographic Hardware and Embedded Systems — CHES 2002. Berlin: Springer, 2003: 291-302.

作者简介

李昊远 男, 1995年生, 山东临沂人. 中国科学院软件研究所博士研究生. 主要研究方向为密码算法的侧信道分析与防护.
E-mail: haoyuan2019@isccas.ac.cn

曹伟琼(通讯作者) 女, 1986年生, 广西桂林人. 中国科学院软件研究所助理研究员. 研究方向为密码算法的侧信道分析与防护.
E-mail: caoweiqiong@iscas.ac.cn