模乘碰撞攻击的分析方法改进

陈艾东[1,2]，陈 运[2]，曹娜娜[2]

(1. 电子科技大学计算机科学与工程学院 成都 611731; 2. 成都信息工程学院信息安全研究所 成都 610225)

【摘要】针对真实环境中用直接差分的方法寻找模乘碰撞较困难的问题，提出了一种*K*均值聚类算法。该算法可以自适应找出模乘碰撞。在搭建的真实攻击环境下，验证了文献[8]中的碰撞攻击在ASIC真实环境中攻击效果并不明显的结论。应用改进的方法后，一对功耗曲线样本便可恢复出88%以上的密钥，实现了小样本量曲线的RSA指数的提取。讨论了对这种攻击方法的防御方案。

关 键 词 选择明文攻击; 碰撞攻击; 密码分析; *K*均值聚类; 边信道攻击

中图分类号 TN918.91 **文献标识码** A doi:10.3969/j.issn.1001-0548.2012.05.008

Improved Analysis Method of the Modular Multiplication Collision Attack

CHEN Ai-dong[1,2], CHEN Yun[2], and CAO Na-na[2]

(1. School of Computer Science & Engineering, University of Electronic Science and Technology of China Chengdu 611731;

2. Information Security Institute, Chengdu University of Information Technology Chengdu 610225)

Abstract The simple power analysis attacks (SPA) of chosen-message contrary pairs is proposed by Miyamoto，which is an attack method based on searching the collision of modular multiplication. But in the real environment searching the collision is difficult. For this problem, the *K*-means clustering algorithm is proposed which can identify the modular multiplication collision automatically. The insignificant effects of collision attack are validated in the ASIC environment which suggested by Miyamoto. After the improvement method, by using a couple of power consumption curves it can recover more than 88% of the secret keys. Finally, the countermeasure of this attack method is discussed.

Key words chosen-message attack; collision attack; cryptanalysis; *K*-means clustering; side channel attack

边信道攻击是一种从密码设备中获取秘密信息的密码攻击方法。常见的边信道攻击有时间攻击[1](timing attack)、能量分析攻击[2](power analysis attack)、电磁攻击[3](electric magnetic attack)和故障攻击[4](fault attack)等。能量攻击是边信道攻击的一种，它是根据密码系统的能耗与被处理数据之间的依赖性进行攻击的[5]。选择明文攻击是一种传统的密码攻击方法，将其与能量攻击相结合，在被控设备上输入特定的明文或消息，在设备加解密过程中捕获功耗曲线，寻找其功耗特征与秘密信息的相关性，进而恢复密钥。

文献[6]提出了一种选择*N*–1(*N*为模数)作为*M*的方法，针对RSA从左向右的二元算法实现，使其每步的中间结果都可能有两个值，攻击者可以利用该原理从后向前推出密钥。但并未给出实验及结果。文献[7]在Xilinx的平台FPGA上实现了文献[6]中的选择明文攻击，推理出3种密钥组合与3种波形的对应关系，并给出攻击过程和结果。但对于此种特殊明文，只需对*N*–1进行屏蔽便可组织此类攻击。文献[8]提出了前一种的改进方法，选取*X*和–*X*作为输入，分别采集两条功耗曲线，对两条曲线做差找到碰撞区域，既而提取指数，该实验环境由FPGA实现并搭建。本文实现了RSA算法的8051芯片搭建实验平台，进行真实环境的边信道攻击。

1 模幂运算与功耗分析模型

1.1 RSA算法

RSA密码系统中模幂运算的形式如下：

加密： $c = m^e \bmod n$

解密： $m = c^d \bmod n$

签名： $s = m^d \bmod n$

验证： $m = s^e \bmod n$

收稿日期：2010 – 11 – 29; 修回日期：2011 – 09 – 21

基金项目：国家自然科学基金(60873216)；四川省应用基础研究基金(2008JY0078)；四川省应用基础研究基金(2011JY0100)

作者简介：陈艾东(1978 –)，女，博士生，主要从事边信道攻防方面的研究.

其中，c是密文，m是明文，e和n是公钥，d是密钥，s为签名。

根据指数的扫描方向，RSA算法的实现可分为从左至右(L-R)、从右到左(R-L)、左右混合(RAD)等3种形式，以下以L-R的二元表示法为例[8]。

算法　MSB FIRST模幂算法
输入：　$X, N, E=(e_t e_{t-1}\cdots e_1 e_0)_2$
输出：　$Z=X^E \bmod N$
Z: =1;
for　$i=k-1$　down to　0
　　Z: =$Z*Z \bmod N$;　/squaring
　　if $(e_i=1)$　then
　　Z: =$Z*X \bmod N$;　/multiplication
　　end if
end for

1.2　功耗模型分析

在真实环境下，进行功耗曲线的采集要受到设备、环境等多方面的影响，具体的功耗的组成如下：

$$P_{\text{total}}=P_{\text{op}}+P_{\text{data}}+P_{\text{el.noise}}+P_{\text{const}} \quad (1)$$

式中，P_{total}为总功耗；P_{op}为操作依赖分量；P_{data}为数据依赖分量；$P_{\text{el.noise}}$为电子噪声；P_{const}为恒定分量。

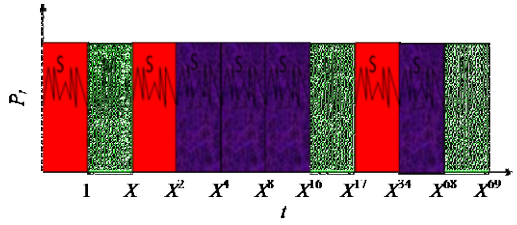
$$P_{\text{op}}+P_{\text{data}}=P_{\text{exp}}+P_{\text{sw.noise}} \quad (2)$$

式中，P_{exp}为能量消耗分量；$P_{\text{sw.noise}}$为转换噪声。如果保证P_{op}不变，P_{data}应该成为影响功耗的主要因素。对于乘法$a\times b$和乘法$c\times d$，如果采用相同的操作数(如$a=b$且$c=d$)，将产生相似的功耗，即 $P_{a\times b}=P_{c\times d}$，进而使总体的功耗$P_{\text{total}}$也相近。

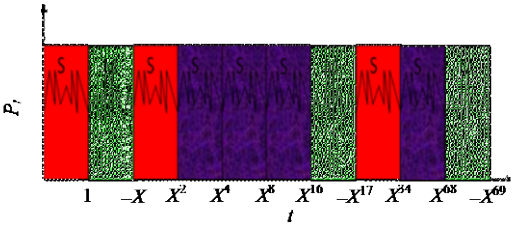
2　选择相反数对的功耗分析攻击

2.1　Miyamoto的选择相反数对攻击

在算法中，如果参与Z: = $Z*Z$ mod N和Z: = $Z*X*$ mod N的值一样，将产生相似的功耗。



a. x为明文的功耗



b. $-x$为明文的功耗

图1　选择相反数对攻击

选择相反数对攻击情况如图1所示，取任意x及其模N相反数$N-x$作为明文，进行算法的模乘运算。图例中将在第4、5、6和9个模乘出现碰撞(即出现相同的参与模乘运算的操作数)。这样，在对应的几个模乘处将呈现相似的功耗。

将图1中两条曲线相减，得到图2的功耗曲线。因为在第4、5、6和9个模乘处出现了碰撞，有相似的功耗，所以相减后在这几个模乘处的功耗明显低于其他模乘。如在第4个模乘出现了碰撞，根据算法就可以判定在第3个模乘和第4个模乘位置连续执行平方运算，那么e_3=0。依此类推，在第4、5和第8个模乘对应的密钥比特也都是0，其余的是1。

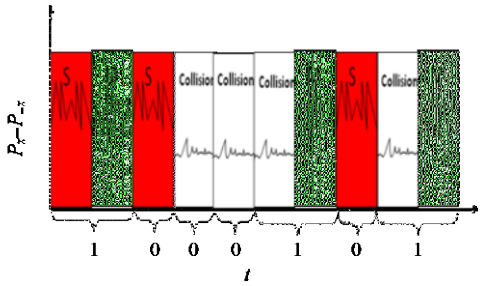


图2　P_x-P_{-x}后的功耗

2.2　真实环境中攻击的问题

真实环境下，进行选择相反数对的攻击发现，由于噪声和模乘对齐问题的干扰，在寻找碰撞的过程中，并不像文献[8]所述的很容易找到碰撞的模乘。将两个功耗曲线相减之后，不能区分出发生碰撞的模乘的位置，无法实现攻击。

2.3　寻找碰撞K均值聚类算法

对输入明文x，$N-x$采集的功耗曲线进行处理，构造如下矩阵[9-12]：

$$\boldsymbol{X}=\begin{pmatrix} x_{11} & x_{12} & \cdots & x_{1p} \\ x_{21} & x_{22} & \cdots & x_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ x_{n1} & x_{n2} & \cdots & x_{np} \end{pmatrix}$$

$$\boldsymbol{Y}=\begin{pmatrix} y_{11} & y_{12} & \cdots & y_{1p} \\ y_{21} & y_{22} & \cdots & y_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ y_{n1} & y_{n2} & \cdots & y_{np} \end{pmatrix}$$

式中，行向量代表一个模乘；n代表曲线中模乘的数量；p代表每个模乘的点数。矩阵为：

$$\boldsymbol{Z}=\boldsymbol{X}-\boldsymbol{Y}=\begin{pmatrix} z_{11} & z_{12} & \cdots & z_{1p} \\ z_{21} & z_{22} & \cdots & z_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ z_{n1} & z_{n2} & \cdots & z_{np} \end{pmatrix}$$

式中，行向量代表两曲线对应模乘的差。

寻找碰撞K均值聚类算法如下：

输入：数据集Z，聚类数目为2。

输出：两个类簇C_1与C_2。

1) 初始化，随机指定两个聚类中心m_1和m_2。

2) 分配z_i，对每个样本z_i计算$d(z_i,m_j)$，i=1,2,⋯,n，j=1,2为判断z_i与m_j之间距离的函数。如果$d(z_i,m_j)=\min\{d(z_i,m_j),j=1,2\}$，则$z_i\in C_j$。

3) 修正簇中心，重新计算各簇中心，对每个簇C_j计算$m_j=\dfrac{1}{N_j}\sum_{t=1}^{N_j}z_t$。$t$为模乘在新簇中的编号，$N_j$为第$j$个簇中模乘的个数。

4) 计算偏差$J=\sum_{j=1}^{2}\sum_{t=1}^{N_j}\left\|z_{jt}-m_j\right\|^2$。

5) 收敛判断，如果J值收敛，则输出C_1与C_2，算法终止；否则转步骤2)。

3 实 验

3.1 功耗测试环境

实验在自主开发的功耗分析平台上对8051芯片进行测试。功耗分析平台中，工作站与示波器USB相连，对示波器与接口板进行设置。发送指令或者数据到接口板，接收返回数据。示波器采用Tektronix PPO4032，接收指令采集功耗曲线和触发信号。

3.2 实验数据预处理

在测试平台上，分别采集x和N–x作为明文参与模乘运算的功耗曲线。应用3.1节中Miyamoto的攻击方法，将两条曲线相减，如图3所示。

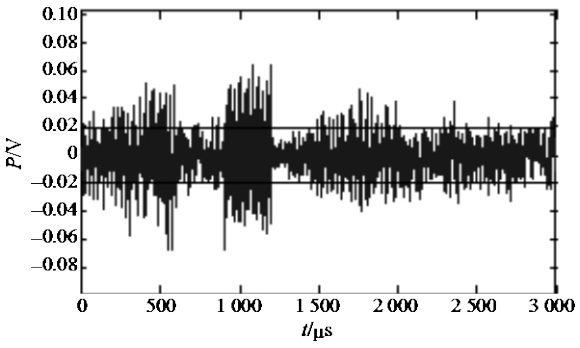


图3 P_x–P_{-x}的实测功耗

由图3可知无法得到文献[8]中的实验效果，应出现碰撞处的功耗并不是都很低。图4为应出现碰撞的第4个模乘，尖峰出现的主要原因是因为每个蒙哥马利运算的子轮没有对齐。这样发生小小的错位后，高功耗减低功耗处便出现了上尖峰，低功耗减高功耗，产生了下尖峰。

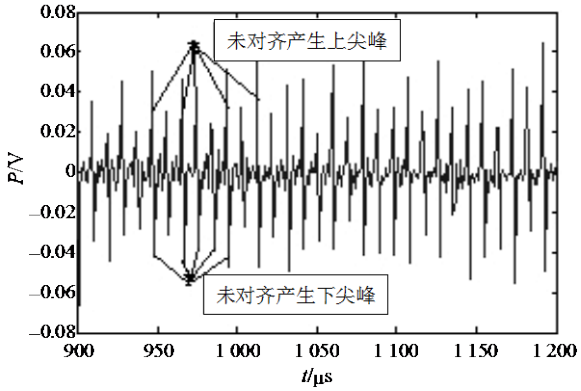


图4 第4个模乘的功耗

3.3 K均值聚类寻找碰撞

以阈值0.02滤波，统计功耗在[–0.02, 0.02]内的每个模乘点的个数，如图5所示。

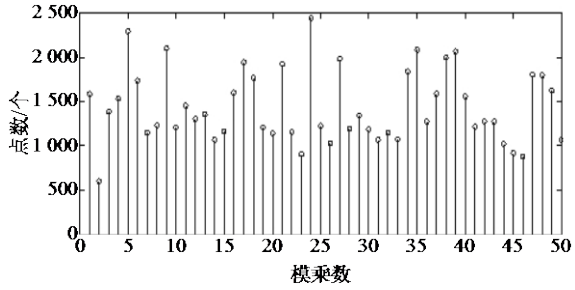


图5 各个模乘在阈值[–0.02,0.02]之间点的个数

图中，横坐标为模乘的序号，纵坐标为滤波后模乘中在阈值方位内的点的数量(即较靠近横轴的点的数量)。纵坐标的值越大，说明在对应模乘中靠近横轴的点的数量越多，即是说明在做差之前两个模乘的功耗越接近。虽然可以区分部分模乘的差别，但是在不具有先见知识的前提下，无法确定碰撞模乘的位置。将曲线进行滑动平均滤波(阈值为5个点)，如图6所示。

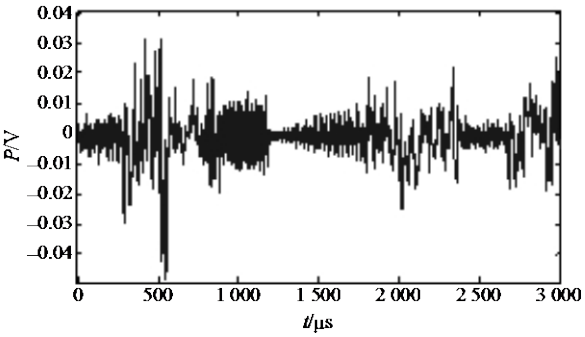


图6 滑动平均滤波后的功耗波形

选择第2个模乘和第5个模乘作为聚类的起始点，应用本文提出的*K*均值聚类算法寻找碰撞。因为密钥首位一定为‘1’，那么如果与第2个模乘分在一类的就不是碰撞，与第2个模乘不属于同一类就是碰撞。那么碰撞模乘的前一个模乘一定是指数‘0’出现的模乘。由此推出1 024位密钥。

在聚类的过程中采用不同的距离函数$d(z_i,m_j)$，比较结果如表1所示。

表1　应用不同距离函数的*K*均值聚类的效果对比

$d(z_i,m_j)$	攻击成功百分比/(%)
平方欧氏距离	83.9
绝对值距离	85.6
夹角余弦	86.2
相关系数	88.3

由表1可知，对于找相似功耗模乘，相关系数距离函数效果最好。

3.4　防御对策

在底数掩码的算法中，对底数做了随机，选择的明文在随机数的作用下与实际参与运算的值没有直接的关系，所以攻击无效。防范本文提出的攻击也可以采用底数掩码的方法进行防御。

4　结　论

本文针对真实环境中用直接差分的方法寻找模乘碰撞较困难的问题，提出一种*K*均值聚类算法寻找模乘碰撞的方法。在搭建的真实攻击环境下，验证了文献[8]提出的碰撞攻击在ASIC真实环境中攻击效果并不明显。应用本文的改进方法后，一对功耗曲线样本便可恢复出88%的密钥，实现了小样本量曲线的RSA指数的提取。

参　考　文　献

[1] KOCHER P, JAFFEN J, JUN B. Differential power analysis[C]//Advances in Cryptology — CRYPTO’ 99 19th Annual International Crytology Conference. Santa Barbara, California, USA: LNCS, 1999.

[2] MESSERGES T S, DABBISH E A, SLOAN R H. Investigations of power analysis attacks on smartcards[C]// Proceedings of the USENIX Workshop on Smartcard Technology. Chicago, Illinois, USA: USENIX Association Berkeley, 1999.

[3] RAO J R, ROHATGI P. EM powering side-channel attacks[R]. Preliminary Technical Report. IBM T J Watson Research Center, 2001.

[4] BIHAM E, SHAMIR A. Differential fault analysis of secret key cryptosystems[C]//Proc of Crypto’97. Santa Barbara, California, USA: [s. n.], 1997.

[5] MANGARD S, OSWALD E, POPP T. Power analysis attacks: Revealing the secrets of smart cards[M]. Berlin: Springer, 2007.

[6] YEN S M, LIEN W C, MOON S J, et al. Power analysis by exploiting chosen message and internal collisions vulnerability of checking mechanism for RSA-decrytion[C] //Progress in Cryptology-Mycrypt 2005 First International Conference on Cryptology in Malaysia. Kuala Lumpur, Malaysia: LNCS, 2005: 183-195.

[7] MIYAMOTO A, HOMMA N, AOKI T. Choosen-message SPA attacks against FPGA-based RSA hardware implementations[C]//Field Programmable Logic and Applications. Heidelberg, Germany: FPL, 2008.

[8] MIYAMOTO A, HOMMA N, AOKI T, et al. Enhanced power analysis attack using chosen message against RSA hardware implementations[C]//ISCAS 2008. Seattle, WA, USA: Circuits and Systems, 2008: 3282-3285.

[9] MENEAES A J. 应用密码学手册[M]. 胡磊，译. 北京: 电子工业出版社, 2005.
MENEAES A J. Handbook of applied cryptography[M]. Translated by HU Lei. Beijing: Publishing House of Electronic Industry, 2005.

[10] JAIN A K, MURTY M N, FLYNN P J. Data clustering: a review[J]. ACM Computing Surveys, 1999, 31(3): 264-323.

[11] LIKAS A, VLASS M, VERBEEK J. The global *K*-means clustering alogorithm[J]. Pattern Recognition, 2003, 36(2): 451-461.

[12] HAE-SANG P, JUN C H. A simple and fast algorithm for *K*-medoids clustering[J]. Expert Systems with Application, 2009, 36(2): 3336-3341.

编　辑　黄　莘