基于旁路攻击的 AES 算法中间变量脆弱点

方 明, 徐开勇, 杨天池, 孟繁蔚, 禹 聪
(信息工程大学 电子技术学院, 郑州 450000)

摘 要: 介绍了针对 AES 算法的旁路攻击过程,指出了可行的 SPA、DPA 以及 HDPA 攻击方法,在此基础上,指出了 AES 算法面对旁路功耗攻击的脆弱点。其过程是:根据 AES 算法的实现流程,查找出可能产生的中间变量,通过对中间变量的分析,将潜在的可以被旁路功耗攻击利用的中间变量定义为脆弱点。进一步指出了旁路功耗攻击是利用了 AES 算法实现过程中数据与密钥的相关性、时钟频率的可测性以及不同指令执行时功耗的差异性来攻破 AES 算法的。

关键词: 旁路攻击; 脆弱点识别; AES 算法
中图分类号: TP309 **文献标志码**: A **文章编号**: 1001-3695(2013)05-1536-04
doi:10.3969/j.issn.1001-3695.2013.05.066

AES intermediate variables vulnerability recognition based on side channel attacks

FANG Ming, XU Kai-yong, YANG Tian-chi ,MENG Fan-wei, YU Cong
(*Institue of Electronic Technology*, *Information Engineering University*, *Zhengzhou* 450000, *China*)

Abstract: This paper introduced the side channel attacks of the AES algorithm and pointed out the viable methods of SPA, DPA and HDPA attack. On this basis, it expatiated a vulnerability recognition process about the AES algorithm face to the side channel attacks. According to the realization of the AES algorithm, it found out the possible intermediate variables, by analysising of the intermediate variables. Further it pointed out that the side channel attacks were taking advantage of the vulnerability's correlation with the key, the clock frequency's testability and the differences of different instructions' execution to break the AES algorithm.

Key words: side channel attacks; vulnerability recognition; AES algorithm

信息安全的核心是密码学,而密码学的核心又是密码算法。近年来,人们对各种密码算法的实现进行了大量的研究,其结果表明,任何密码算法都可以用软件或硬件来实现。其中,软件加密具有灵活、轻便的优点,但是加密速度较低,并且由于软件本身运行在一个开放的环境中,被攻击和窜改的可能性较大;相对于软件加密,硬件加密的特点是加密速度快、安全性高、安装容易、总体成本低。这些优点使得密码算法的硬件产品——密码芯片具有软件加密产品不可比拟的优势,因此密码芯片已经成为了密码算法实现的重要载体。以智能卡、付费电视卡、生物特征认证卡为代表的数据存储密码芯片,正被广泛地应用于电子商务、税收、通信等领域。AES 属于对称分组密码算法。AES 将分组长度固定为 128 bit,支持 128、192、256 三种密钥长度。AES 加密算法具有运算速度快、资源占用率小、性能可靠等特点,因此 AES 算法被广泛应用在密码芯片中。

传统的数学分析方法并不能解释物理设备的泄露效应,这是由于一个实际应用的密码系统的安全性不仅仅与其密码体系的设计有关,还涉及到程序和硬件的设计等多个方面,如果密码设计者、程序员和硬件设计工程师之间互相不了解不配合,最终系统的安全性往往也是不完善、不可靠的。

旁路攻击是一种破解安全芯片内私有密钥的有效手段,包括时间分析攻击、功耗攻击、电磁辐射分析等。在各种旁路攻击技术手段中,相对而言功耗攻击的威胁最高[1,2]。功耗攻击按其强度可分为简单功耗攻击(simple power analysis,SPA)、差分功耗攻击(differential power analysis,DPA)以及高阶 DPA 攻击(high order DPA,HDPA)。在目前利用旁路功耗攻击分析 AES 密码算法的实现方法中,大多只针对 AES 算法执行过程中 S 盒的中间值进行攻击,如文献[3~7]介绍了针对 AES 算法 S 盒的第一轮输出的旁路功耗攻击,通过对 AES 的 S 盒变换中第一轮的输出进行差分功耗分析(DPA),攻击者可以获得 AES 算法的种子密钥。文献[8~11]介绍了针对 AES 算法 S 盒的高阶 DPA 攻击,通过观测多位的输出可以使得攻击结果更为精确。文献[6]指出:旁路功耗攻击是利用了 AES 算法实现过程中数据与密钥的相关性、时钟频率的可测性以及不同指令执行时功耗的差异性来攻破 AES 算法的。因此,一个运行着 AES 算法的密码系统,其中除了 S 盒之外还有其他可以被旁路攻击所利用的脆弱点,本文将说明利用这些脆弱点攻击者可以获得很多敏感信息。

1 旁路功耗攻击

1.1 简单功耗攻击(SPA)

SPA 是利用功耗曲线的特征,依据攻击者的经验直观地分

收稿日期: 2012-06-24; **修回日期**: 2012-08-19

作者简介:方明(1987-),男,硕士研究生,主要研究方向为密码系统工程(fangming_334817943@qq.com);徐开勇(1963-),男,研究员,主要研究方向为密码系统工程;杨天池(1981-),男,博士,主要研究方向为密码系统工程;孟繁蔚(1987-),男,硕士研究生,主要研究方向为密码系统工程;禹聪(1986-),女,硕士研究生,主要研究方向为信息安全.

析出加密硬件中执行的指令或操作。

为了测量密码芯片的功耗曲线,常用的方法是将一个小电阻(如 50 Ω)接至芯片电源或接地端,此时电阻两端的电压将正比于流经密码芯片的电流。SPA 通过直接分析密码芯片执行加密操作时释放出的功耗特征来获取与密钥有关的信息,对于加密中的不同操作,在功耗曲线中显示出来的特征会产生不同的变化。例如采用指令流方式运行的密码电路中,置换、移位、查表和异或等基本操作通常具有不同的功耗特征,SPA 分析者可以将这些功耗特征翻译成加密时的相关操作。

1.2　差分功耗攻击(DPA)

DPA 要比 SPA 强得多,而且更加难以预防,DPA 攻击使用统计分析和误差修正技术,提取与密钥有关的信息,与 SPA 分析相结合,DPA 具有更强的攻击能力。对于一个分组加密算法,如果攻击者猜对某一轮进入 S 盒的子密钥时,那么按一定规则差分处理后的相关曲线将显示出一些很突出的尖峰。DPA 分析需要采集多组功耗曲线,将其作为先验函数,而将猜测密钥组根据一定规则所产生的规则作为后验函数,计算两组函数所具有信息的相关性,如果相关性强,则说明猜测准确。对于每次猜测,DPA 攻击可分为三个步骤:a)功耗样本采集;b)猜测密钥;c)功耗样本分集差分。

1.3　高阶差分功耗攻击(HDPA)

DPA 也称为一阶差分功耗攻击。高阶差分功耗攻击与一阶差分功耗攻击的不同是:对于同一个功耗曲线,一阶差分功耗攻击只对其中一个单独时间点进行采样分析,而高阶则是对其两个或两个以上的时间点进行联合采样统计分析,与之相应的区分函数和统计分析过程较一阶差分功耗攻击更加复杂,分析精度比一阶差分功耗分析高,但是数据采集量和计算复杂度远远大于一阶差分功耗攻击,能有效防御一阶差分功耗攻击的方法未必对高阶差分攻击有效。差分功耗攻击的阶数越高,功耗偏差越明显,则估计值正确的概率也越大,所以选择一阶还是高阶差分功耗攻击技术,与所遇到的具体情况有关。如果信噪比较小时,可用高阶差分功耗攻击;如果可以用简单的滤波技术增加信噪比,则使用一阶差分功耗分析。功耗攻击所需设备并不复杂。

2　识别 AES 算法中间变量脆弱点

AES 算法是当前应用非常广泛的分组密码算法,因此安全芯片中通常包括 AES 算法部件。为有效抵抗功耗攻击,AES 算法部件的设计与实现必须采取有效的防护技术。

2.1　AES 算法概述

以 128 bit 分组长度的 AES 算法为例,该算法共需 10 轮迭代过程,前 9 轮迭代包含的操作为轮密钥加、字节代换、行移位、列混淆。

a)总体描述为:给定一个明文 x,先进行轮密钥加操作,即将轮密钥与 state 异或。

b)对前 9 轮中的每一轮,用 S 盒对 state 进行一次字节代换操作,并对 state 进行一次行移位操作;再对 state 进行一次列混淆操作;然后进行一次轮密钥加操作。

c)将前 9 轮迭代的结果进行最后一次迭代,在最后一次迭代中只进行字节代换、行移位和轮密钥加操作。

d)将最后一轮迭代后的结果定义为密文 y。

2.2　AES 的密钥扩展算法

对 10 轮版本的 AES 需要 11 个轮密钥,每个轮密钥由 16 Byte 组成,密钥编排算法是面向字的,每个轮密钥由 4 Byte 组成。轮密钥的并联叫做扩展密钥,共包含 44 Byte,表示为 $W[0],\cdots,W[43]$,其中每个 $W[i]$ 都是 1 Byte。

定义函数:

a)RW 操作。对 4 Byte 进行移位操作,即 $RW(B_0,B_1,B_2,B_3)=(B_1,B_2,B_3,B_0)$。

b)SW 操作。对 4 Byte 使用 AES 的 S 盒,即 $SW(B_0,B_1,B_2,B_3)=(B_0',B_1',B_2',B_3')$,其中 $B_i'=\text{subByte}(B_i)$。

c)Rcon 是一个 10 Byte 的数组 $Rcon[1],\cdots,Rcon[10]$。

算法 1　密钥扩展

输入:128 bit 种子密钥(key[0],…,key[15])。

输出:44 轮 128 bit 轮密钥。

```
Rcon[1] = 01000000      Rcon[6] = 20000000
Rcon[2] = 02000000      Rcon[7] = 40000000
Rcon[3] = 04000000      Rcon[8] = 80000000
Rcon[4] = 08000000      Rcon[9] = 1B000000
Rcon[5] = 10000000      Rcon[10] = 36000000
for i = 0 to 3
do W[i] = (key[4i],key[4i + 1],key[4i + 2],key[4i + 3])
for i = 4 to 43
   do temp = W[i - 1]
if i = 0(mod4)
then temp = SW(RW(temp))⊕Rcon[i/4]
W[i] = W[i - 4]⊕temp
return (W[0],…,W[43])
```

2.3　迭代操作过程

由于行移位和列混淆两个操作只是对 state 矩阵进行行列的重新排序,并未涉及到密钥的计算过程,并且由于 AES 算法的公开性,只要知道行移位和列混淆的输入,根据算法规则很容易推导出行移位和列混淆的输出。所以,行移位和列混淆的操作结果对抵抗旁路攻击而言价值有限,本文不予考虑。下面重点介绍字节代换和轮密钥加的操作过程。

1)字节代换　定义函数 F 为求一个域元素的乘法逆元,BTF 为把一个字节变换为一个 F28 上的域元素,FTB 为将一个 F28 上的域元素变换成 1 Byte。

算法 2　字节代换

输入:一个 8 bit 字节:$(a_7\ a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0)$。

输出:变换后的 8 bit 字节 $(b_7\ b_6\ b_5\ b_4\ b_3\ b_2\ b_1\ b_0)$。

$Z=\text{BTF}(a_7\ a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0)$

if $(Z\neq 0)$ then $Z=F(Z)$

$(a_7\ a_6\ a_5\ a_4\ a_3\ a_2\ a_1\ a_0)=\text{FTB}(Z)$

$C=(c_7\ c_6\ c_5\ c_4\ c_3\ c_2\ c_1\ c_0)=01100011$

for $i=0$ to 7

do $b_i=(a_i+a_{i+4}+a_{i+5}+a_{i+6}+a_{i+7}+c_i)\bmod 2$

return $(b_7\ b_6\ b_5\ b_4\ b_3\ b_2\ b_1\ b_0)$

2)轮密钥加　即将 128 bit 密钥与 state 按位异或。

2.4　密钥扩展算法的中间变量脆弱点识别

根据以上对 AES 算法的描述可以看出,AES 加密算法主要由密钥扩展算法和迭代算法两大部分组成。

根据对密钥扩展算法的描述,由于功耗量的存在,所暴露出的中间变量脆弱点共有四组,分别是:

a) i 在 0 ~ 3 循环中的密钥加载功耗 di;

b) i 在 4 ~ 43 循环中,当 i 为 4 的倍数时的计算功耗 ei;

c) $W[i]$ 与 temp 进行异或操作时的功耗 fi;

d) 返回 $W[0] \sim W[43]$ 时的加载功耗 gi。

对于上述四组功耗泄露点而言,每一组都有可能被攻击者所利用,由于 i 在 0 ~ 3 循环中加载的是种子密钥,因此如果攻击者可以成功利用第一组加载功耗量 di 对 AES 算法进行功耗分析,那么攻击者可以直接获得 AES 的 128 bit 种子密钥,直接攻破 AES 算法。所以,该脆弱点是最严重的。

i 在 4 ~ 43 循环中是依据种子密钥生成扩展密钥,如果攻击者能够正确地猜测出每一轮的轮密钥,由于 AES 算法的公开性,依据每一轮的密钥,攻击者可以很容易地推导出 AES 算法的种子密钥。功耗泄露量 ei、fi 和 gi 是针对轮密钥进行功耗分析的切入点。

对于第二组功耗量 ei,虽然与之相关的过程量 temp 的值具有一定的随机性,但是其前面有一个判断语句 if $i \equiv 0 \pmod 4$,这给攻击者利用 SPA 攻击识别每一轮密钥的起始点,进而为确定攻击时刻带来了极大的方便。

对于功耗泄露量 fi 而言,由于 SW 和 RW 的操作使得 temp 具有一定的随机性,然而这种随机性可以被高阶 DPA 攻击所抵消。如果攻击者可以找到一个合适的攻击模型,那么功耗量 fi 也会成为功耗攻击潜在的脆弱点。

与功耗量 di 相同,功耗量 gi 也是攻击者可以直接利用的获得轮密钥的有效脆弱点。当轮密钥计算完毕后通过加载时的能量泄露 gi,攻击者很容易获得每轮的密钥。依据以上分析,将四组能量泄露的危害程度进行排序,如图 1 所示:

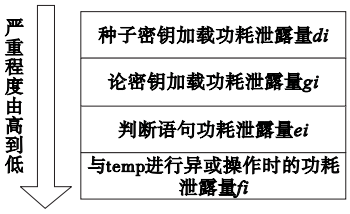

图1 脆弱点危害程度排序

根据 AES 扩展密钥生成算法流程,可以得出如下的功耗攻击模型,如图 2 所示。在实施攻击的过程中,攻击者要分析的是一串连续的功耗波形,因此首先要清楚波形中哪一部分是与其攻击目标相关的区间。为了确定这一段相关的区间,攻击者会使用 SPA 攻击方法来识别。所以,在中间值识别过程中也加入了 SPA 相关的信息泄露点。

2.5 轮密钥加的中间值识别

轮密钥加就是将每轮的 128 bit 密钥与 128 bit 的输入逐位异或的过程,在攻击者可以进行已知明文分析的前提下,第一轮的轮密钥加操作的分析过程就退化为与前文对加载功耗泄露量 di 和 gi 的分析过程一样,不同的是攻击者根据观测结果与明文输入的比较确定密钥每一位的值。因此,该中间值所泄露的敏感信息的危害也是极其严重的。

2.6 字节代换的中间值识别

在 AES 算法中,唯一的非线性计算部分就是字节代换。由于字节代换是非线性操作,将现行的 mask 技术运用在 S 盒变换时会消耗大量的系统资源。因此,在大多数的密码芯片中,不对 S 盒操作进行掩码处理,故实际的旁路攻击大多数都是利用了 S 盒操作的中间变量。另外,由于 S 盒操作是基于字节的,也就是 8 bit 的数据,根据 DPA 攻击原理,攻击者很容易穷举出 8 bit 的数据所有 256 种的可能性。文献[8]中提出:只要猜测的数据小于 32 bit,攻击者就可以有效地执行 DPA 攻击。由于难以用开支较小的保护方式和较短的操作数,以及在计算执行时与密钥的相关性,字节变换的中间值是 AES 算法脆弱点中最为核心的脆弱点。根据前面对字节代换算法的描述,对字节代换算法的功耗攻击主要过程如图 3 所示。

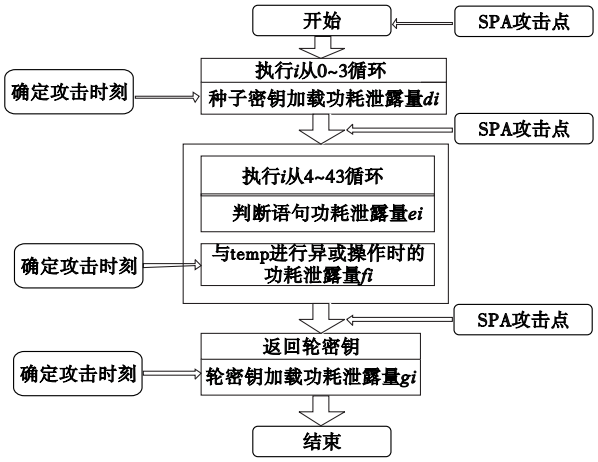

图2 密钥扩展算法脆弱点识别模型

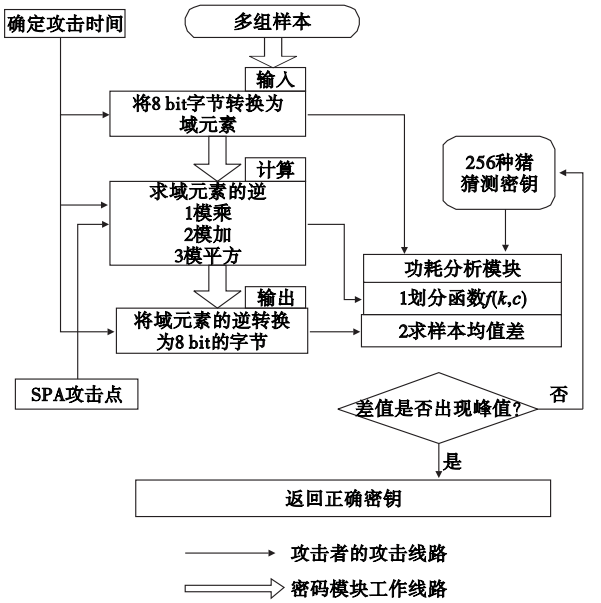

图3 脆弱点代换识别模型

在 S 盒操作过程中,最为复杂的操作就是对 GF(28)上的域元素求逆。然而直接在 GF(28)上进行求逆比较复杂,因此一般算法将 GF(28)上的求逆转换成 GF(24)和 GF(22)上的运算。GF(28)中的每一个元素都可以映射成 GF((24)2)中的一个一次二项式 $ah\ x + al, ah, al \in$ GF(24)。这样,一个在 GF(28)域上求逆的运算就分解为 GF(24)上的两个加法运算、一个平方运算、一个常量乘法运算、三个普通乘法运算和一个求逆运算。再将计算出的结果映射回 GF(28)中即可得到 GF(28)上的求逆结果。

由于加法运算、平方运算、常量乘法运算、普通乘法运算以及求逆运算在计算时所产生的功耗曲线有着可以识别的波形特征,攻击者可以根据该特征识别出密码模块正在进行的数据操作,并且根据更细粒度的模加、模乘、模平法的分析,更准确地细化划分函数 $f(C, K)$。根据上文的攻击过程可以看出,

SPA 攻击点就是在识别不同的计算方式所产生的功耗特征,并根据与特定数据的相关性建立划分函数,因此在密码模块中,每一次细粒度的模加、模乘、模平方等操作的功耗都会给攻击者泄露出重要的信息。因此安全的密码模块一定也会对细粒度的模加、模乘、模平方等操作采取一定的保护措施。

3　脆弱点验证

功耗分析攻击实验平台由 FPGA 密码芯片、示波器、稳压电源及计算机组成。为采集密码芯片在工作时的功耗变化,在密码芯片接地端与电路板接地端接一个 20 Ω 的电阻,通过探头接电阻的两端,将密码芯片的功耗信号传送给示波器进行采集,并通过 USB 传输到 PC 机存储,示波器的采集过程由 PC 机上用 LabView 编写的虚拟仪器控制平台实现自动控制。整个控制流程如下:

a)控制平台首先为 FPGA 密码芯片注入密钥。

b)通过 RS-232 接口为密码芯片提供随机明文输入。

c)当密码芯片运行 AES 时,触发示波器记录电阻两端的功耗输出(采样率 250 MHz,每条轨迹共采样 10 000 个点),并控制示波器实时向 PC 机传输功耗数据。

d)重复上述过程,从而实现数据采集的完全自动化。

利用该实验平台对 AES 算法脆弱点的验证如下:

实验 1　对轮密钥加过程中的脆弱点进行攻击

AddRoundKey 变换是明文和轮密钥的一个字节逐位异或操作,如果攻击者对轮密钥加进行已知明文攻击,对密码芯片输入全 1 的 128 bit 明文,则得到的密文就是轮密钥的取反。假设密码芯片在初始状态下各个门级电路为低电平,初始密钥为 11100011000111000110011…,则对轮密钥加的功耗攻击波形如图 4 所示。当逻辑门的初始状态为不可知时,由于逻辑门的 0 到 1 和 1 到 0 的翻转所消耗的能量不同,也会给攻击者泄露敏感数据的汉明距离信息。

图4　轮密钥加的攻击结果

通过实验可知,利用轮密钥加过程中的功耗泄露量可以有效地对密码芯片实施旁路功耗攻击。

实验 2　对 S 盒运算中的脆弱点进行攻击

首先根据 S 盒执行过程中细粒度计算执行所泄露的功耗差异构造划分函数。通过细粒度计算执行的功耗差异,可以确定整体 S 盒的执行过程由两个加法运算、一个平方运算、一个常量乘法运算、三个普通乘法运算和一个求逆运算组成。根据计算执行顺序,确定划分函数为 $f(C,K)$ 将样本测试结果分成两组,其中 C 是 S 盒的第一轮输出,K 是猜测密钥,则依据 DPA 攻击的过程,得到密钥猜测正确时的差分波形,如图 5 所示。

由于细粒度计算执行功耗量的差异,攻击者可以确定某个 S 盒具体的计算执行过程,依据该过程攻击者可以确定划分函数。

对加载功耗泄露量 di 与 gi 的攻击过程与轮密钥加的攻击过程相似,通过观察寄存器中门级电路的翻转功耗曲线,可以确定敏感数据或敏感数据的汉明距离。

实验 3　判断语句功耗泄露量的识别

密钥扩展算法中的判断语句所产生的功耗波形如图 6 所示,根据该波形可以确定出每一轮密钥的起始位置。

从图中可以看出,由于判断条件成立时的操作与条件不成立时的操作不同,每隔 4 个时钟周期,功耗曲线都会出现一个高点尖峰。

图5　密钥猜测正确时的差分波形

图6　判断语句功耗波形

由于判断语句的存在,通过直观地观察加密过程中的功耗波形,攻击者可以确定出每一轮密钥的起始时刻。

4　结束语

本文在研究 AES 算法产生的所有中间值的基础上,结合 SPA、DPA 以及 HDPA 的攻击过程,筛选出与敏感数据相关、有利于确定攻击时刻以及能够识别出密码芯片中执行不同操作的脆弱点,并验证了攻击这些脆弱点的有效性,得出如下结论:旁路功耗攻击不仅可以对 AES 算法的 S 盒产生威胁,而且对密钥扩展算法以及密钥加载过程也会产生极大的威胁。

参考文献:

[1] MANGARD S. Securing implementations of block ciphers against side-channel attacks[D]. Austria:Graz University of Technology, 2004.

[2] KOCHER P C,JAFFE J, JUN B. Differential power analysis[C]// Proc of the 19th Annual International Cryptology Conference on Advances in Cryptology. London:Springer,1999: 388-397.

[3] SCHEIDER H. Analysis of the resistance of different logic styles against SPA & DPA attacks[D]. [S.l.]:Institute for Applied Information Processing and Communications, Graz University of Technology,2003.

[4] KOCHER P, JAFFE J, JUN B. Introduction to differential power analysis and related attacks[R]. San Francisco: Cryptography Reseach,1998.

[5] CLAVIER C, CORON J, DABBOUS N. Differential power analysis in the presence of hardware counter measures[C]//Proc of the 2nd International Workshop Worcester on Cryptographic Hardware and Embedded Systems. Berlin:Springer-Verlag,2000:252-263.

[6] 童元满,戴葵,陆洪毅,等. 基于细粒度任务调度的防功耗分析模幂方法[J]. 计算机工程,2006,32(24):15-16,33.

[7] 尹文龙,丁国良,刘昌杰,等. 一种对嵌入式加密芯片的增强 DPA 攻击方法[J]. 计算机应用研究,2010,27(2):712-713,717.

[8] JOYE M, PAILLIERi P, SCHOENMAKERS B. On second-order differential power analysis[C]//Proc of the 7th International Workshop on Cryptographic Hardware and Embedded Systems. Berlin:Springer-Verlag,2005:293-308.

[9] MESSERGES T S . Using second-order power analysis to attack DPA resistant software [C]//Proc of the 2nd International Workshop Worcester on Cryptographic Hardware and Embedded Systems. Berlin: Springer-Verlag,2000:27-28.

[10] MANGARD S, OSWALD E, POPP T. 能量分析攻击[M]. 冯登国,周永彬,刘继业,等译. 北京:科学出版社,2009.

[11] BOSCHER A, HANDSCHUH H. Masking does not protect against differential fault attacks[C]//Proc of the 5th Workshop on Fawlt Diagnosis and Tolerance in Cryptography. Piscataway: IEEE Press, 2008:35-40

[12] 刘政林,韩煜,邹雪城,等. AES 能量攻击的建模与分析[J]. 计算机工程与科学,2008,30(3):17-20.