doi:10.3969/j.issn.1000-436x.2014.05.011

对 ECC 算法实现的选择明文攻击方法

罗鹏[1,2]，李慧云[3]，王鲲鹏[4]，王亚伟[5]

(1. 中国科学院 软件研究所，北京 100190；2. 中国科学院大学，北京 100049；3.中国科学院深圳先进技术研究院，广东 深圳 518055；
4. 中国科学院 信息工程研究所，北京 100093；5. 北京华大信安科技有限公司，北京 100015)

摘 要：提出了一种新型的基于构造输入点 y 坐标的 CSPA 方法，使得 ECC 密码算法中标量乘的点加和点倍运算产生明显的能量消耗差别，从而获取密钥信息。对多种 ECC 密码算法不同实现的芯片的实际分析结果表明，该方法具有较强的实用性，能够有效地识别出标量乘运算过程中的点加运算。研究提到的方法均在素数域的 ECC 密码算法上实现。

关键词：ECC 算法；标量乘法；选择明文攻击；侧信道分析

中图分类号：TP309 **文献标识码**：A **文章编号**：1000-436X(2014)05-0079-09

Chosen message attacks method against ECC implementations

LUO Peng[1,2], LI Hui-yun[3], WANG Kun-peng[4], WANG Ya-wei[5]

(1. Institute of Software, Chinese Academy of Sciences, Beijing 100190, China;
2. University of Chinese Academy of Sciences, Beijing 100049, China;
3. Shenzhen Institutes of Advanced Technology, Shenzhen 518055, China;
4. Institute of Information Engineering, Chinese Academy of Sciences, Beijing 100093,China; 5. HUADA INFOSEC, Beijing 100015,China)

Abstract: Chosen-message side-channel analysis attacks for public-key cryptosystems based on scalar multiplications was proposed, where special input point P is chosen to have the features of special y-coordinate to generate noticeable variations on power consumption or other side-channel information, between point doubling and addition operations, rendering distinguishable key bit information. Experimental results demonstrate that the proposed chosen message with simple side channel analysis method could effectively generate the noticeable variations. The method applies to the prime filed ECC cryptographic algorithms.

Key words: Elliptic curve cryptosystem; scalar multiplication; chosen message attack; side channel analysis

1 引言

相对于对称密码系统，公钥密码系统如 RSA、椭圆曲线密码系统(ECC)的密钥管理更具灵活性。传统的公钥密码系统的安全性一般是基于因式分解问题，给定 2 个或多个大素数 p,q 的乘积 $n=pq$，求素数 p,q；而 ECC 密码系统的安全性则是基于椭圆曲线离散对数问题，已知生成元 G 和积 $y=xG$，求被乘数 x。由于椭圆曲线离散对数问题(ECDLP)是全指数问题，而传统密码系统的整数因式分解问题(IFP)或离散对数问题(DLP)是亚指数问题，因此，对具有同等安全性的 2 个系统，随圆曲线密码系统(ECC)的密钥长度要比传统公钥密码系统小得多，如 256 bit ECC 与 3 072 bit RSA 具有同等的安全性。鉴于 ECC 具备更短的密钥长度，更快的运行速度，节约了大量的内存、运算能量和通信带宽，ECC 算法在实际 IT 系统中得到了广泛应用。

目前，对于各种软件、硬件实现的公钥密码系统，侧信道分析已具备实际攻击的能力[1]。其中，常用的侧信道分析技术之一是简单能量分析(SPA)

收稿日期：2013-03-18；**修回日期**：2013-09-20

基金项目：中国科学院青年创新促进会基金资助项目(2013)；国家高技术研究发展计划(“863”计划)基金资助项目(2012AA01A403)

Foundation Items: The Fund of CAS Youth Exchange Promotion Association(2013); The National High Technology Research and Development Program of China (863 Program) (2012AA01A403)

攻击，SPA 主要通过观察设备在某一段时间内运行公钥密码算法的能量消耗来获取密钥。对于 RSA 密码系统的平方—乘法算法，攻击者使用 SPA 能够区分平方和乘法运算，从而推断出私钥的二进制比特。同样地，由于标准的 ECC 二进制算法与 RSA 的平方—乘法算法是相似的，若攻击者能区分出 ECC 算法标量乘运算中的点加和点倍运算，也能得到私钥的二进制比特。

为了抵抗上述 SPA 或其他简单能量分析方法，RSA 算法实现一般采用相同的指令序列执行乘法和平方运算，以增加区分两者的难度。相同的乘法和平方是非常有效的 SPA 攻击防护措施。但是，文献[2~5]中提出了对 RSA 输入特殊数据的选择明文攻击方法，这些方法可攻击上述防护措施。其中，Yen 提出了使用特殊输入值 x 和 $-x$ 的选择明文攻击方法[2]，并证明了其对大多数常用的 SPA 防护措施均是有效的。RSA 选择明文攻击为了增大乘法和平方运算之间能量消耗特征的差异，输入特殊数据 $-x$，因 $-x \bmod n=(n-x)$，则在求幂运算中，乘法和平方运算的输出分别为 $-x \bmod n=(n-x)$ 和 $x \bmod n$，综上所述，乘法和平方运算可分为 3 类：平方运算后进行乘法运算(M)、乘法运算后进行平方运算(S1)、平方运算后再进行平方运算(S2)，分别如下

$$z=x\times(-x) \bmod n=-x^2 \bmod n \tag{1}$$

$$z=(-x)\times(-x) \bmod n=x^2 \bmod n \tag{2}$$

$$z=x\times x \bmod n=x^2 \bmod n \tag{3}$$

上述 3 式中若有一个运算的侧信道信息泄露不同于其他 2 个，攻击者便可从中推断出指数密钥比特。而在实际的硬件实现中，M 和 S1 的运算消耗大于 S2，因此通过特殊输入的选择明文 SPA，可成功获得 RSA 密钥比特序列[3]。此外，若存在特殊的 RSA 运算顺序，Yen 方法也同样适用于相同的平方和乘积算法。若攻击者能够观察出 $\mathrm{M}\rightarrow\mathrm{S2}$ 的次序，则能识别出当前乘法运算为伪操作，从而判断当前密钥比特为 1。

同样，为了抵抗简单能量分析方法，ECC 算法实现也广泛使用相同的指令序列进行点加和点倍运算。鉴于选择明文攻击对 RSA 防御的成功攻击，攻击者也企图通过该方法对 ECC 进行攻击。但是文献[2, 3]表明，由于 ECC 模乘一致的特殊性，对 RSA 算法的选择明文攻击并不适用于 ECC 类的密码系统。因此，Fan 等人提出了针对 ECC 的错误攻击方法[6]，该方法通过在病态曲线上构造低阶点(一般为 4)，可识别出密钥的 0 bit，从而获取整个密钥比特序列。但是，该方法不适用于带参数有效校验的 ECC 实现，同时，重复的错误注入需要精确的时间限定，平均每一个特定字节需要试验 256 次错误注入，而有限的试错防御机制将极大地限制该方法的实现。

本文提出了一种基于选择 y 坐标的对 ECC 的选择明文的简单能量分析方法，可攻击当前常见的抗 SPA 的防护措施。基于选择 y 坐标的选择明文攻击的原理是：当使用大量具有不同值域的 y 坐标点进行标量乘运算时，点倍和点加运算产生的能量信息泄露有明显的差别，基于这一特性可成功进行选择明文攻击。该方法不需要构造低阶点，且构造的均为椭圆曲线上的有效点，可以通过 ECC 算法的点的有效性检查，从而可对带有错误注入响应及域参数有效校验等防护措施的算法实现进行攻击。

2 ECC 算法简介

2.1 椭圆曲线密码系统

椭圆曲线密码系统利用了椭圆曲线离散对数问题：在一个循环群 G 中，g 为生成元，且 g 的阶为 n，对于给定的元素 $y=g^x\in G$，求 x 的值。令 p 为素数，F_p 为整数模 p 的有限域，$E(F_p)$ 为域 F_p 上椭圆曲线 E 的所有点集，则 $E(F_p)$ 所有点构成了一个交换群，其中，无穷远点 O 为单位元。若点 $P\in E(F_p)$，且 P 的阶 n 为素数，则由 P 生成的循环群 $<P>=\{O,P,2P,3P,\cdots,(n-1)P\}$ 为 $E(F_p)$ 的循环子群。在 ECC 密码系统中，素数 p、域 F_p 的椭圆曲线方程、基点 P 及阶 n 均为公开的域参数，任选私钥 $d\in[1,n-1]$，则相应的公钥 $Q=dP$[7]。

有限域 F_p 的椭圆曲线为平面曲线，由满足 Weierstrass 方程：$y^2+a_1xy+a_3y=x^3+a_2x^2+a_4x+a_6$ 的点组成，其中，$a_i\in F_p$。若特征 $(\mathrm{char})F_p\neq 2,3$，则 Weierstrass 方程可转化成：$y^2=x^3+ax+b$，其中 $a,b\in F_p$，椭圆曲线由常数 a,b 确定。若大素数 n 为 $E(F_p)$ 的子群大小，且 $nP=O$，根据 Lagrange 定理可得

$$h=\frac{\#E(F_p)}{n} \tag{4}$$

其中，$\#E(F_p)$ 为椭圆曲线 $E(F_p)$ 上点的个数，h 为

余因子。$\#E(F_p)$ 可由 Schoof-Elkies-Atkin (SEA)关于素数域点计算法得到。若 h 为小因子，应满足 $hP \neq O$，以避免受到小子群攻击[7]。

除了素数有限域 F_p，ECC 有限域也可以为二进制域 F_{2^m}，F_{2^m} 包括一个次数的基域生成多项式 $f(z)$，域 F_{2^m} 中元素为二进制多项式：$f'(z)=a_{m-1}z^{m-1}+a_{m-2}z^{m-2}+\cdots+a_2z^2+a_1z^1+a_0$，$a_i \in \{0,1\}$，$f'(z)$ 的次数 $\deg(f'(z)) < m$。

2.2　素数域椭圆曲线上的运算

在 $E(F_p)$ 上定义的加法运算使用弦切线法则，则 $E(F_p)$ 为加法交换群，无穷远点 O 为单位元。对 $E(F_p)$ 上两点 P、Q 之和 $P+Q$，若 $P \neq Q$,连接 P、Q 的直线交 E 于点 R,则 R 关于 x 轴的对称点 R' 即为 $P+Q$ 之和，称为点加运算；若 $P=Q$，作 P 点的切线交 E 于点 R，则 R 关于 x 轴的对称点 R' 即为 $2P$，称为点倍运算。特征 $(\mathrm{char})F_p \neq 2,3$ 的有限域 F_p 中，点加和点倍运算具体如下。

点加：令 $P=(x_1,y_1)\in E(F_p)$，$Q=(x_1,y_1)\in E(F_p)$，且 $P \neq Q$，则 $P+Q=(x_3,y_3)$，其中

$$x_3=\left(\frac{y_2-y_1}{x_2-x_1}\right)^2-x_2-x_1 \tag{5}$$

$$y_3=\left(\frac{y_2-y_1}{x_2-x_1}\right)(x_1-x_3)-y_1 \tag{6}$$

点倍：令 $P=(x_1,y_1)\in E(F_p)$，$P \neq -P$,则 $2P=(x_3,y_3)$，其中

$$x_3=\left(\frac{3x_1^2+a}{2y_1}\right)^2-2x_1 \tag{7}$$

$$y_3=\left(\frac{3x_1^2+a}{2y_1}\right)(x_1-x_3)-y_1 \tag{8}$$

k 个相同点 P 之和用 kP 表示，称为椭圆曲线上的标量乘。标量乘是 ECC 中与密钥相关的基本运算，经常作为侧信道攻击的主要对象。标量乘运算最简单的实现算法是二进制算法[8]。图 1 为二进制算法的标量乘运算(从左到右)。为防止侧信道攻击，还可在点加点倍操作中采用原子算法[9]，使得点加点倍的侧信道信息基本相同。

3　点加运算在不同坐标系下的实现方式分析

3.1　仿射坐标

在二进制算法（如图 1 所示）点加运算 $Q=Q+P$ 中，$P(x_1,y_1)$ 为选择明文输入，$Q(x_2,y_2)$ 由 $P(x_1,y_1)$ 经过多次点加和点倍得到，y_2 在 $(0,p)$ 内可视为随机分布。

输入：$k=(k_{n-1},\cdots,k_1,k_0)_2(k_{n-1}=1),P\in E(F_p)$
输出：kP
1) $Q=P$ $\deg(f(z))=m$
2) $i=n-2$,⋯,0，重复执行
2.1) $Q=2Q$　　//点倍
2.2) 若 $k_i=1$，则 $Q=Q+P$　　//点加
3) 返回(Q)

图 1　标量乘的二进制算法(从左到右)

在仿射坐标下，由式(5)和式(6)的点加公式可得 $P+Q=(x_3,y_3)$ 的分步计算式

$$\begin{aligned}&A_1=y_2-y_1;A_2=x_2-x_1;I=A_2^{-1};\\&M_1=A_1\times I;M_2=M_1^2;A_3=M_2-x_1;\\&x_3=A_3-x_2;A_5=x_1-x_3;\\&M_3=M_1\times A_5;y_3=M_3-y_1\end{aligned} \tag{9}$$

其中，A 代表域上的加法运算，M 代表域上的乘法运算，I 代表模逆运算，下同。为节省存储空间和减少重复运算步骤，式(9)可优化为表 1 所示的迭代运算步骤。

表 1　　仿射坐标下的点加运算步骤

步骤	运算式	步骤	运算式
1	$T_1=y_2-y_1$	6	$x_3=x_3-x_1$
2	$T_2=x_2-x_1$	7	$x_3=x_3-x_2$
3	$T_2=T_2^{-1}$	8	$T_1=x_1-x_3$
4	$T_2=T_1T_2^{-1}$	9	$y_3=T_1T_2$
5	$x_3=T_2^2$	10	$y_3=y_3-y_1$

如表 1 所示，式(9)共需要进行 1 次模逆，3 次域乘法，6 次域加法。其中，y_1 直接参与运算为第 1 步的 $A_1=y_2-y_1$，因此这一步的运算可通过输入不同的 y_1 构造选择明文攻击。需注意的是，因为第 3 步的 $I=A_2^{-1}$ 运算量远大于域乘法和域加法，实际的 ECC 算法实现较少选择在仿射坐标下实现。

3.2　标准射影坐标

仿射坐标下的椭圆曲线包含无穷远点，实现较为不便，另外在点加和点倍运算需要进行求逆运算，而一般情况下求逆运算要比乘法运算耗时得

多，因此 ECC 算法实际实现时，通常将仿射坐标映射成标准射影坐标、雅克比坐标及雅克比—仿射混合坐标等方式。

在标准射影坐标下，由仿射坐标(x,y)与标准射影坐标(X,Y,Z)的映射关系$(x,y)\to(X=xZ,\ Y=yZ,Z),\ X,Y,Z\in F_p$，有$P(x_1,y_1)\to P(X_1,Y_1,Z_1)$，$Q(x_2,y_2)\to Q(X_2,Y_2,Z_2)$，对图 1 中的点加运算$Q=Q+P$，可得标准射影坐标下的点加$Q+P=R(X_3,Y_3,Z_3)$的分布运算式

$$\begin{aligned}&\lambda_1=X_2Z_1;\lambda_2=X_1Z_2;\lambda_3=\lambda_1-\lambda_2;\\&\lambda_4=Y_2Z_1;\lambda_5=Y_1Z_2;\lambda_6=\lambda_4-\lambda_5;\\&\lambda_7=\lambda_1+\lambda_2;\lambda_8=Z_1Z_2;\lambda_9=\lambda_3^2;\\&\lambda_{10}=\lambda_9\lambda_3;\lambda_{11}=\lambda_6^2\lambda_8-\lambda_7\lambda_9;X_3=\lambda_3\lambda_{11};\\&Y_3=\lambda_6(\lambda_9\lambda_2-\lambda_{11})-\lambda_{10}\lambda_5;Z_3=\lambda_{10}\lambda_8\end{aligned}\tag{10}$$

式(10)可优化为表 2 所示的迭代运算步骤。

表 2　标准射影坐标下的点加运算步骤

步骤	运算式	步骤	运算式
1	$T_1=X_2Z_1$	12	$C=CT_7$
2	$T_2=X_1Z_2$	13	$C=C-T_1$
3	$T_3=T_1-T_2$	14	$X_3=T_3C$
4	$T_4=Y_2Z_1$	15	$T_2=T_2T_6$
5	$T_5=Y_1Z_2$	16	$T_6=T_6T_3$
6	$T_4=T_4-T_5$	17	$T_5=T_6T_5$
7	$T_6=T_3^2$	18	$Y_3=T_2-C$
8	$T_1=T_1+T_2$	19	$Y_3=Y_3T_4$
9	$T_1=T_6T_1$	20	$Y_3=Y_3-T_5$
10	$T_7=Z_1Z_2$	21	$Z_3=T_6T_7$
11	$C=T_4^2$		

如表 2 所示，除去域坐标在kP运算开始和结束时的转换过程，式(10)共使用了 15 次域乘法和 6 次域加法。对于如图 1 所示的算法来说，由于每次kP运算中P为固定值，Z_1也为固定值，通常设置为 1，并且$Y_1=y_1Z_1 \bmod p$通常在kP运算前计算得到；由于Q在kP运算过程中不断变化，Z_2在$(0,p)$内可视为随机分布。因y_1通过Y_1间接参与第 5 步的$\lambda_5=Y_1Z_2$域乘法，因此这一步的运算可通过输入不同的y_1构造选择明文攻击。

3.3　雅克比坐标

在雅克比坐标下，由仿射坐标(x,y)与雅克比坐标(X,Y,Z)的映射关系$(x,y)\to(X=xZ^2,Y=yZ^3,Z),\ X,Y,Z\in F_p$，有$P(x_1,y_1)\to P(X_1,Y_1,Z_1)$、$Q(x_2,y_2)\to Q(X_2,Y_2,Z_2)$，对图 1 中的点加运算$Q=Q+P$，可得雅克比坐标下的点加$Q+P=R(X_3,Y_3,Z_3)$的分布运算式：

$$\begin{aligned}&\lambda_1=X_2Z_1^2;\lambda_2=X_1Z_2^2;\lambda_3=\lambda_1-\lambda_2;\\&\lambda_4=Y_2Z_1^3;\lambda_5=Y_1Z_2^3;\lambda_6=\lambda_4-\lambda_5;\\&\lambda_7=\lambda_1+\lambda_2;\lambda_8=\lambda_4+\lambda_5;\\&X_3=\lambda_6^2-\lambda_7\lambda_3^2;\lambda_9=\lambda_7\lambda_3^2-2X_3;\\&Y_3=\frac{\lambda_6\lambda_9-\lambda_8\lambda_3^3}{2};Z_3=\lambda_3Z_1Z_2\end{aligned}\tag{11}$$

式(11)可优化为表 3 所示的迭代运算步骤。

表 3　雅克比坐标下的点加运算步骤

步骤	运算式	步骤	运算式
1	$T_1=Z_1^2$	13	$T_2=T_2T_4$
2	$T_2=T_1X_2$	14	$X_3=T_6^2$
3	$T_3=Z_2^2$	15	$X_3=X_3-T_2$
4	$T_4=T_3X_1$	16	$T_4=T_4T_5$
5	$T_5=T_2-T_4$	17	$T_1=T_1+T_3$
6	$T_1=T_1Z_1$	18	$T_4=T_4T_1$
7	$T_1=T_1Y_2$	19	$Y_3=T_2-2X_3$
8	$T_3=T_3Z_2$	20	$Y_3=Y_3T_6$
9	$T_3=T_3Y_1$	21	$Y_3=(Y_3-T_4)/2$
10	$T_6=T_1-T_3$	22	$Z_3=Z_1Z_2$
11	$T_2=T_2+T_4$	23	$Z_3=Z_3T_5$
12	$T_4=T_5^2$		

如表 3 所示，除去域坐标在kP运算开始和结束时的转换过程，式(11)共使用了 16 次域乘法和 7 次域加法。对于图 1 所示的算法来说，由于每次kP运算中P为固定值，Z_1也为固定值，并且$Y_1=y_1Z_1^3 \bmod p$通常在kP运算前计算得到；由于Q在kP运算过程中不断变化，Z_2在$(0,p)$内可视为随机分布。因y_1通过Y_1间接参与第 9 步的$\lambda_5=Y_1Z_2^3$域乘法，因此这一步的运算可通过输入不同的y_1构造选择明文攻击。

3.4　雅克比—仿射混合坐标

雅克比—仿射混合坐标可简化雅克比坐标中运算的运算过程。在雅克比—仿射混合坐标下，点加中的$Q(x_2,y_2)\to Q(X_2,Y_2,Z_2)$为雅克比坐标，$P(x_1,y_1)$为仿射坐标，对图 1 中的点加运算$Q=$

$Q+P$，可得雅克比—仿射混合坐标下点加 $Q+P=R(X_3,Y_3,Z_3)$ 的分布运算式：

$$\begin{aligned}&\lambda_1=x_1Z_2{}^2;\lambda_2=X_2-\lambda_1;\lambda_3=y_1Z_2{}^3\\&\lambda_4=Y_2-\lambda_3;Z_3=\lambda_2Z_2;\lambda_5=\lambda_2{}^2\\&\lambda_6=\lambda_2\lambda_5;\lambda_7=X_2\lambda_5\\&X_3=\lambda_4{}^2-2\lambda_7+\lambda_6\\&\lambda_8=\lambda_7-X_3;Y_3=\lambda_8\lambda_4-\lambda_6Y_2\end{aligned}\tag{12}$$

式(12)可优化为表 4 所示的迭代运算步骤具体迭代步骤。

表 4　雅克比—仿射混合坐标下的点加运算步骤

步骤	运算式	步骤	运算步骤
1	$T_1=Z_2{}^2$	10	$T_3=T_3X_2$
2	$T_2=T_1Z_2$	11	$T_1=2T_3$
3	$T_1=x_1T_1$	12	$X_3=T_2^2$
4	$T_2=y_1T_2$	13	$X_3=X_3-T_1$
5	$T_1=X_2-T_1$	14	$X_3=X_3+T_4$
6	$T_2=Y_2-T_2$	15	$T_3=T_3-X_3$
7	$Z_3=T_1Z_2$	16	$T_3=T_3T_2$
8	$T_3=T_1{}^2$	17	$T_4=T_4Y_2$
9	$T_4=T_1T_3$	18	$Y_3=T_3-T_4$

如表 4 所示，除去域坐标在 kP 运算开始和结束时的转换过程，式(12)共使用了 11 次域乘法和 6 次域加法。对于图 1 的算法来说，由于 Q 在 kP 运算过程中不断变化，Z_2 在 $(0,p)$ 内可视为随机分布。因 y_1 直接参与第 4 步的 $\lambda_3=y_1Z_2{}^3$ 的域乘法，因此这一步的运算可通过输入不同的 y_1 构造选择明文攻击。

4　选择明文攻击方法

通过第3节对ECC算法在常用坐标系下实现方式的分析，发现可以利用 y_1 构造选择明文攻击。在仿射坐标下，由于对椭圆曲线上的任意 x_1，均对应着 2 个不同的 y_1,y_1'，且 $y_1+y_1'=p$，因此有 $y_1\in(0,\frac{1}{2}p),y_1'\in(\frac{1}{2}p,p)$。同样地，在标准射影坐标下，$Y_1=y_1Z_1,Y_1'=y_1'Z_1$，有 $Y_1\in(0,\frac{1}{2}p),Y_1'\in(\frac{1}{2}p,p)$；

在雅克比坐标下，$Y_1=y_1Z_1{}^3,Y_1'=y_1'Z_1{}^3$，有 $Y_1\in(0,\frac{1}{2}p),Y_1'\in(\frac{1}{2}p,p)$；雅克比—仿射混合坐标下的情况与仿射坐标相同。利用 y 坐标的对称分布特性，攻击者可以选取一组随机的 x 坐标，获得对应的两组 y 坐标，形成两组 P 点作为选择明文输入。在实际应用中，ECC 算法通常选取 $p>2^{160}$，根据 Hasse 定理可知，$p+1-2\sqrt{p}\leqslant\#E(F_p)\leqslant p+1+2\sqrt{p}$，由于 $2\sqrt{p}<p$，可得 $\#E(F_p)\approx p$，$\#E(F_p)>2^{160}$。曲线上存在大量的有效点，利用这些有效点和 y 坐标的特征可以构建选择明文攻击方法。

4.1　对域加法的攻击

对仿射坐标下的 ECC 算法实现可以从域加法运算入手构造选择明文攻击。

算法实现通常会对式(9)第 1 步的 $A_1=y_2-y_1$ 运算输出结果进行正负检查，如输出结果为负，会将负值通过加上 p 调整为正值。在这种情况下增加了一个算术加操作，这将造成能量消耗的差异。

1) 当 $0<y_1<\frac{1}{2}p$ 时，分以下 2 种情形。

当 $0<y_2<\frac{1}{2}p$ 时，$-\frac{1}{2}p<A_1<\frac{1}{2}p$，

$$P(-\frac{1}{2}p<A_1<0)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4},$$

$$P(0<A_1<\frac{1}{2}p)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4}$$

当 $\frac{1}{2}p<y_2<p$ 时，$0<A_1<p$

$$\begin{aligned}&P(0<A_1<\frac{1}{2}p)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4}\\&P(\frac{1}{2}p<A_1<p)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4}\end{aligned}\tag{13}$$

由式(13)，有

$$P(A_1<0)=\frac{1}{4},P(A_1>0)=\frac{3}{4}\tag{14}$$

其中，$P(X)$ 表示条件 X 存在的概率。

2) 当 $\frac{1}{2}p<y_1<p$ 时，分以下 2 种情形。

当 $0<y_2<\frac{1}{2}p$ 时，$-p<A_1<0$，

$$P(-p<A_1<-\frac{1}{2}p)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4},$$

$$P(-\frac{1}{2}p<A_1<0)=\frac{1}{2}\times\frac{1}{2}=\frac{1}{4}$$

当$\frac{1}{2}p < y_2 < p$时，$-\frac{1}{2}p < A_1 < \frac{1}{2}p$

$$P(-\frac{1}{2}p < A_1 < 0) = \frac{1}{2}\times\frac{1}{2} = \frac{1}{4}$$
$$P(0 < A_1 < \frac{1}{2}p) = \frac{1}{2}\times\frac{1}{2} = \frac{1}{4} \tag{15}$$

由式(15)有

$$P(A_1 < 0) = \frac{3}{4}, P(A_1 > 0) = \frac{1}{4} \tag{16}$$

由式(14)和式(16)可知，对 t 次 kP 运算，如果 y_1 在 $(0, p)$ 上随机选取，$\frac{1}{2}p < y_1 < p$ 时 A_1 运算步骤将比 $0 < y_1 < \frac{1}{2}p$ 时多 $\frac{1}{2}t$ 次算术加操作，对应的平均能量消耗将出现差异。由于点倍操作处理的点每次都发生变化，在能量消耗上没有与 y_1 相关的统计特征，因此通过对能量迹上能量消耗分类比较，即可以识别出 kP 运算过程中的点加操作，从而推断出平均能量消耗具有差异部分进行的为点加操作，进而推断出密钥比特。

4.2 对域乘法的攻击

对标准射影坐标、雅克比坐标及雅克比—仿射混合坐标下的 ECC 算法实现可以从域乘法运算入手构造选择明文攻击。

对标准射影坐标下的 ECC 算法实现，由于 Z_2 经过多次域上运算得到，可视为在 $(0, p)$ 范围内随机分布。对多次不同输入的 kP 运算，由于 Z_1 固定，当 $0 < y_1 Z_1 \bmod p < \frac{1}{2}p$ 时，Y_1 绝对值的数学期望为

$$\mathrm{E}(|Y_1|) = \mathrm{E}(y_1 Z_1 \bmod p) = \frac{1}{4}p \tag{17}$$

当 $\frac{1}{2}p < y_1 Z_1 \bmod p < p$ 时

$$\mathrm{E}(|Y_1|) = \mathrm{E}(y_1 Z_1 \bmod p) = \frac{3}{4}p \tag{18}$$

对常见的运算数扫描方式实现和积扫描方式实现的乘法器来说，式(10)中 $\lambda_5 = Y_1 Z_2$ 的运算复杂度为 $O(mn)$，其中，m 为 Y_1 二进制表示的比特长度，n 为 Z_2 二进制表示的比特长度。当然，对其他类型的乘法实现来说，$\lambda_5 = Y_1 Z_2$ 的运算复杂度略有不同，比如采用 Karatsuba-Ofman 乘法，但均与 Y_1, Z_2 的二进制比特长度直接相关。

当 $0 < y_1 Z_1 \bmod p < \frac{1}{2}p$ 时，由式(17)可得，多次 $\lambda_5 = Y_1 Z_2$ 的平均运算复杂度为

$$O_1(Y_1 Z_2) = O(\left\lceil \mathrm{lb}(\frac{1}{4}p) \right\rceil \times \left\lceil \mathrm{lb}(\frac{1}{2}p) \right\rceil)$$
$$= O((r-2)\times(r-1)) \tag{19}$$

当 $\frac{1}{2}p < y_1 Z_1 \bmod p < p$ 时，由式(18)可得，多次 $\lambda_5 = Y_1 Z_2$ 的平均运算复杂度为

$$O_2(Y_1 Z_2) = O(\left\lceil \mathrm{lb}(\frac{3}{4}p) \right\rceil \times \left\lceil \mathrm{lb}(\frac{1}{2}p) \right\rceil)$$
$$\approx O((r-1)^2) \tag{20}$$

式(19)和式(20)中，$\lceil t \rceil$ 为 $t + \frac{1}{2}$ 的四舍五入整数值，$\mathrm{lb}(x) = \mathrm{lb}x$，$r$ 为 p 的二进制比特表示长度。为定量比较运算复杂度的差异，在此定义复杂度差异

$$\Delta O(O_1(t_1), O_2(t_2)) = 1 - \frac{t_1}{t_2} \tag{21}$$

借助复杂度差异，在不考虑芯片运行过程中消耗能量的固定分量和噪声等情况下，可以近似地估计对不同的输入数运算的能量消耗的差异。

由式(19)~式(21)可得，复杂度比率 $\Delta O(O_1, O_2) = 1 - \frac{(r-2)\times(r-1)}{(r-1)^2} = \frac{1}{r-1}$。

例如，如果 p 为 256 bit 的素数，对分布在 $\left(0, \frac{1}{2}p\right)$ 和 $\left(\frac{1}{2}p, p\right)$ 两组不同的 $y_1 Z_1 \bmod p$，$\lambda_5 = Y_1 Z_2$ 的 $\Delta O = \frac{1}{256-1} \approx 0.003\,9$。

$\lambda_5 = Y_1 Z_2$ 的运算复杂度与消耗的能量直接相关，同时，由于点倍运算的中间点 Q 始终变化，点倍运算中涉及到 Q 点 y 坐标的操作消耗的能量不会出现统计特性，通过对不同的 y_1 坐标的 P 点的 $\lambda_5 = Y_1 Z_2$ 运算的能量消耗进行观察，即可推断出进行的点加操作。需要注意的是，这里和通常的能量分析关注的中间变量对应的能量消耗不一样，通常的能量分析关注于在不同能量模型下某一时刻中间变量与能量耗值的对应关系[10]，此处只关注对不同的输入在能量迹上某一时间段内能量消耗的绝对值。

在雅克比坐标下，可构造相同数量的两组具有不同 y 坐标的 P 点输入，其中一组的 $y_1 Z_1^3 \bmod$

$p<\frac{1}{2}p$，另外一组的 $y_1 Z_1^3 \bmod p>\frac{1}{2}p$。由于 Y_2 经过多次域上运算得到，可视为在 $(0,p)$ 范围内随机分布。后续分析可仿造标准射影坐标进行。

在雅克比—仿射混合坐标下，可构造相同数量的两组具有不同 y 坐标的 P 点输入，其中一组的 $0<y_1<\frac{1}{2}p$，另外一组的 $\frac{1}{2}p<y_1<p$。由于 Z_2^3 经过多次域上运算得到，可视为在 $(0,p)$ 范围内随机分布。后续分析可仿造标准射影坐标进行。

5　实验数据

5.1　仿射坐标下软件实现的 ECC 算法

在此对一款在仿射坐标下软件实现了 ECC 算法的芯片进行分析。为便于分析，选择了一条 p 的二进制长度为 32 bit 的曲线，曲线参数为

$p=0x$FFFFFFFFFFFFFFC5

$a=0x$FFFFFFFFFFFFFFC2

$b=0x$8353C13A26610485

$g_x=0x$A6250461F6CDAEFC

$g_y=0x$F30A2B57E068C810

实验中选取 kP 的 $k=7$，共采样 2 400 条能量迹，$P(x_1,y_1)$ 的 y_1 坐标随机选取，且前 1 200 条的 $y_1<\frac{1}{2}p$，后 1200 条 $y_1>\frac{1}{2}p$。$k=7$ 时，运算过程共进行两次 $Q=Q+P$，图 1 为第 2 次点加运算的整体能量迹，图 3 是第 2 次点加运算中进行域加法部分的能量迹。图 3 中，深色线部分是前 1 200 条能量迹的平均能量迹，浅色线部分是后 1 200 条能量迹的平均能量迹。

由图 3 可明显看出对不同的 y_1 取值范围，$\frac{1}{2}p<y_1<p$ 时平均能量迹波形在域加法部分明显大于 $0<y_1<\frac{1}{2}p$，从而识别出点加运算。对能量迹上其他非点加中域加法部分进行分析，未见明显差异。实验结果证明对域加法进行的选择明文攻击有效。

5.2　雅克比—仿射混合坐标下硬件实现的 ECC 算法

在此对一款在雅克比—仿射混合坐标下硬件实现的 ECC 算法的芯片进行分析。曲线参数为

$p=0x$FFFFFFFF00000001000000000000000000000000FFFFFFFFFFFFFFFFFFFFFFFF

$a=0x$FFFFFFFF00000001000000000000000000000000FFFFFFFFFFFFFFFFFFFFFFFC

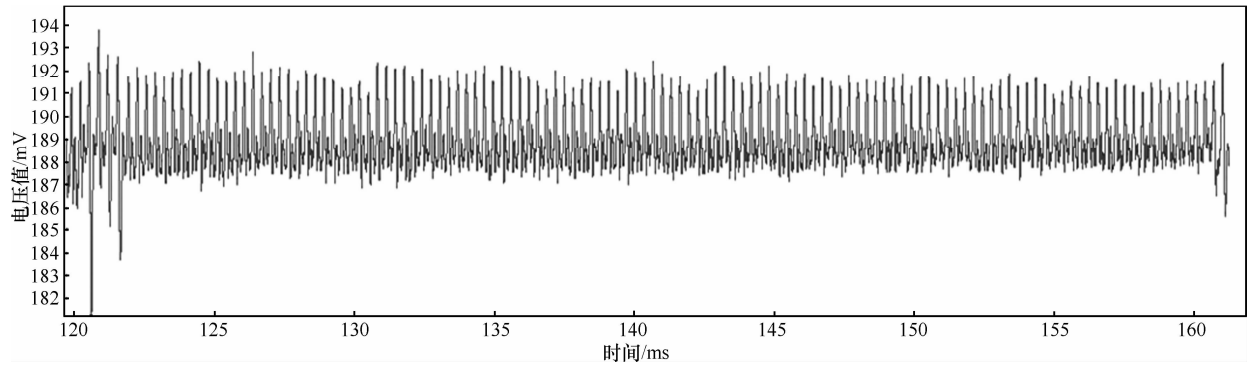

图 2　仿射坐标 k=7 时第 2 次点加的能量迹

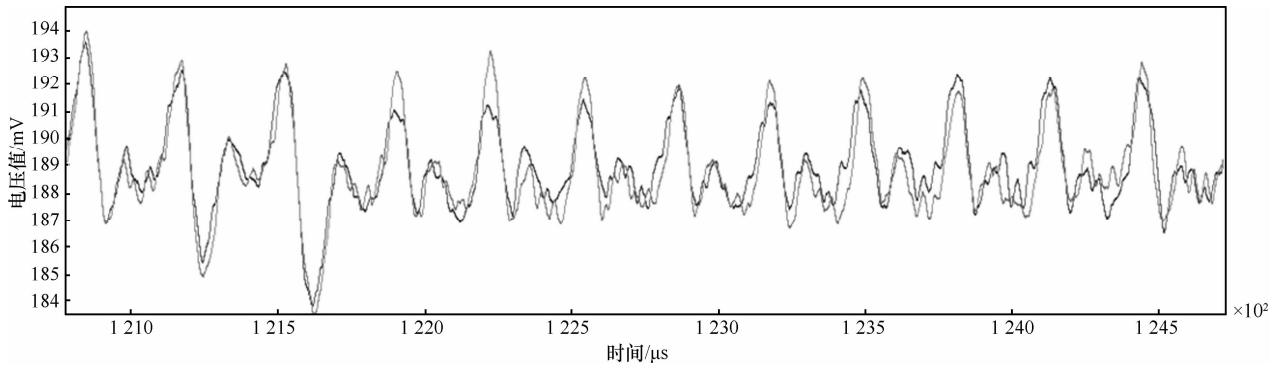

图 3　仿射坐标 k=7 时第 2 次点加中域加法部分的能量迹

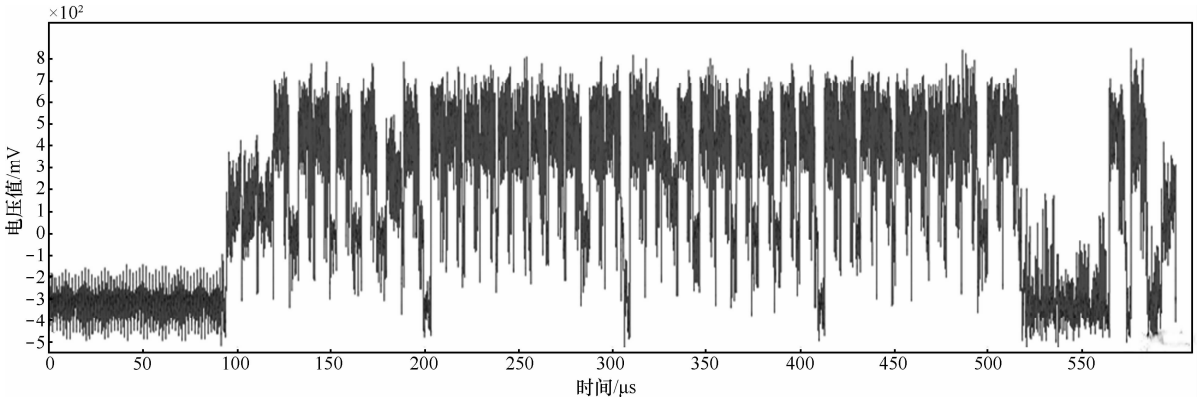

图 4　雅克比—仿射坐标混合 k=7 时 kP 运算能量迹

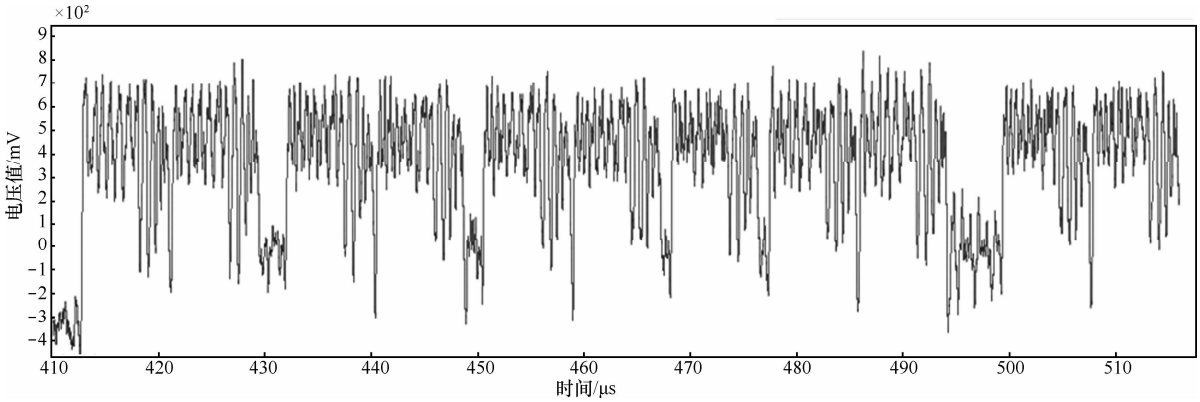

图 5　雅克比—仿射混合坐标第 2 次点加运算的能量迹

$b = 0x5AC635D8AA3A93E7B3EBBD55769886$
$BC651D06B0CC53B0F63BCE3C3E27D2604B$
$g_x = 0x6B17D1F2E12C4247F8BCE6E563A440$
$F277037D812DEB33A0F4A13945D898C296$
$g_y = 0x4FE342E2FE1A7F9B8EE7EB4A7C0F9$
$E162BCE33576B315ECECBB6406837BF51F5$

实验中选取 kP 的 $k=7$，共采样 5000 条能量迹，$P(x_1, y_1)$ 的 y_1 坐标随机选取，且前 2 500 条的 $y_1 < \frac{1}{2}p$，后 2 500 条 $y_1 > \frac{1}{2}p$，图 3 为 kP 运算的整体能量迹。运算过程共进行两次点加，图 4 是第 2 次点加的能量迹，主要由 11 次域乘法组成。

对 11 次域乘法的平均能量迹进行能量消耗统计，如表 5 所示。

由表 5 可发现，第 4 次域乘法消耗的能量差异明显大于其他域乘法。由表 4 可知，第 4 次域乘法正是进行的 $\lambda_3 = y_1 Z_2^3$。考虑到噪声和固定能量消耗等其他因素，实际的 ΔO 和 4.2 节中计算出的 0.003 9 比较接近。对能量迹上非点加部分进行分析，未见这种差异。实验结果证明对域乘法进行的选择明文攻击有效。

表 5　　11 次域乘法能耗差异

模乘编号	$y_1 < \frac{1}{2}p$ 点的能耗	$y_1 > \frac{1}{2}p$ 点的能耗	ΔO
M1	34.210 846	34.193 375	−0.000 5
M2	31.655 19	31.637 566	−0.000 6
M3	27.302 038	27.287 497	−0.000 5
M4	28.069 105	28.116 11	0.001 7
M5	27.624 77	27.602 726	−0.000 8
M6	29.530 966	29.529 814	0.000 0
M7	29.179 401	29.161 39	−0.000 6
M8	29.386 608	29.377 132	−0.000 3
M9	24.827 814	24.828 201	0.000 0
M10	28.082 44	28.057 52	−0.000 9
M11	30.132 78	30.145 34	0.000 4

6　结束语

本文提出了一种新型的对素数域上的 ECC 算法进行选择明文 SPA 分析的方法。通过操纵 kP 运算过程中输入点 $P(x_1, y_1)$ 中的 y_1，攻击者可以通过统计仿射坐标下的域加法、标准射影坐标、雅克比坐标及雅克比—仿射混合坐标下的域乘法的能量消耗判别出 kP 运算过程中的点加运算，从而推断出

密钥比特。因为攻击依赖于每次 kP 运算过程中的不变量 P，本文中提出的方法适用于基于从左至右方式实现的各种 ECC 标量乘算法。实验结果表明，本文中的方法对多种 ECC 算法软硬件实现均有效。

参考文献：

[1] KOCHER P, JAFFE J, JUN B. Timing attacks on implementations of Diffie-Hellman, RSA, DSS, and other systems[A]. Proceedings of 16th International Advances in Cryptology Conference-CRYPTO ‘96[C]. 1996.104-113.

[2] YENS M, LIEN W C, MOONS J, *et al*. Power analysis by exploiting chosen message and internal collisions-vulnerability of checking mechanism for RSA decryption[A]. MYCRYPT 2005[C]. 2005. 183-195.

[3] MIYAMOTO A, HOMMA N, AOKI T, *et al*. Enhanced power analysis attack using chosen message against RSA hardware implementations[A]. IEEE International Conference on Field Programmable Logic and Applications[C]. 2008.3282-3285.

[4] HOMMA N, MIYAMOTO A, AOKI T, *et al*. Comparative power analysis of modular exponentiation algorithms[J]. IEEE Transaction on Computers, 2010.795-807.

[5] YENSM, LIEN W C, CHEN C N. Modified doubling attack by exploiting chosen ciphertext of small order[A]. IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences[C]. 2011.1981-1990.

[6] FAN J, GIERLICHS B, VERCAUTEREN F. To infinity and beyond: combined attack on ECC using points of low order[A]. International Workshop on Cryptographic Hardware and Embedded Systems[C]. CHES, LNCS, 2011.

[7] HANKERSON D, MENEZES A, VANSTONE S. Guide to Elliptic Curve Cryptography[M]. New York, USA: Springer-Verlagn, 2003.

[8] CORON J S. Resistance against differential power analysis for elliptic curve cryptosystems[A]. International Workshop on Cryptographic Hardware and Embedded Systems[C]. CHES, LNCS, 1999.

[9] CHEN T, LI H, WU K, YU F. Countermeasure of ECC against side-channel attacks: balanced point addition and point doubling operation procedure[A]. AsiaPacific Conference on Information Processing 2009[C]. 2009. 465-469

[10] LUO P, FENG D G, ZHOU Y B. Power model in power analysis attack[J]. Journal on Communications, 2012, 33(Z1):276-282.

作者简介：

罗鹏（1978-），男，四川遂宁人，中国科学院博士生，主要研究方向为密码学、信息安全。

李慧云（1977-），女，安徽金寨人，博士，中国科学院深圳先进技术研究院副研究员，主要研究方向为集成电路设计、信息安全。

王鲲鹏（1971-），男，河北尚义人，博士，中国科学院研究员，主要研究方向为密码学、信息安全。

王亚伟（1984-），男，河南驻马店人，北京华大信安科技有限公司高级工程师，主要研究方向为安全芯片的密码系统设计、攻击及防护。

（上接第 78 页）

[16] ADILJAN Y, YOSHIHIRO H, TASUKU M, *et al*. 2-D direction histogram based entropic thresholding[J]. Neurocomputing, 2013, 120(23): 287-297.

[17] 张弘, 范九伦. 二维 Arimoto 熵直线型阈值分割法[J]. 光子学报, 2013, 42(2): 234-240.
ZHANG H, FAN J L. Two-dimensional Arimoto entropy linear-type threshold segmentation method[J]. Acta Photonica Sinica, 2013, 42(2): 234-240.

作者简介：

范朝冬[通信作者]（1984-），男，江西宜春人，湖南大学博士生，主要研究方向为图像处理、智能信息处理。E-mail: fanchd@126.com。

欧阳红林（1965-），男，湖南衡阳人，湖南大学教授、博士生导师，主要研究方向为智能信息处理、新型电力电子技术及其应用、电力传动与变频技术。

肖乐意（1986-），女，湖南常德人，硕士，长沙师范学院讲师，主要研究方向为智能课程设计、智能信息处理。