设计任务之四：

椭圆曲线加密算法(Elliptic Curve Cryptosystem， ECC)的设计与实现

email： xjfang@aliyun.com
personal website：http://star.aust.edu.cn/~xjfang

date: 2016-12-25

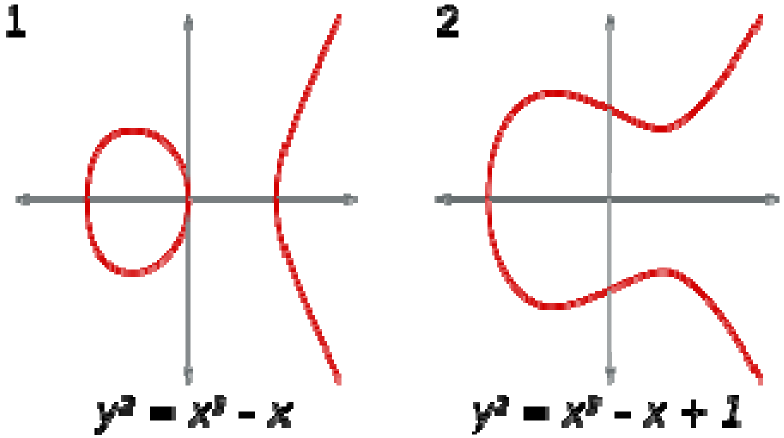
一、什么是椭圆曲线

- 椭圆曲线是指威尔斯特拉(Weierstrass)方程所确定的平面曲线

$$E: y^2 + axy + by = x^3 + cx^2 + dx + e$$

其中a,b,c,d,e属于域F, F可以是有理数域、复数域或有限域GF(p)。

椭圆曲线有一个特殊的点，记为O，它并不在椭圆曲线E上，此点称为无限远的点(the point at infinity) 。在xOy平面上，可以看作是平行于y轴的所有直线的集合的一种抽象。



二、有限域GF(p)上的椭圆曲线

- 密码学中普遍采用有限域上的椭圆曲线，它是指椭圆曲线方程的定义中，所有系数、方程的根都是某一有限域GF(p)中的元素。其最简单的表示为：

$$E: y^2 = x^3 + ax + b \pmod{p}$$

也记为 $E_p(a,b)$

其中p是一个大素数，a, b, x, y均在GF(p), 且满足$4a^3+72b^2 \pmod{p} \neq 0$，以保证在GF($p$)有限域中，E上的所有点构成一个Abel群。

二、有限域GF(p)上的椭圆曲线

定理：$E_p(a, b)$上的点，对于如下定义的加法规则构成一个Abel群。

（一）加法规则：

① O+O=O；

② 对任意P=(x,y)∈ $E_p(a, b)$, 有P+O=O+P=P;

③ 对任意P=(x,y)∈ $E_p(a, b)$, 有P+(-P)=O， 即P的逆元为-P =(x,-y)

④ 令P=(x1,y1) ∈ $E_p(a, b)$， Q=(x2, y2) ∈ $E_p(a, b)$, 则P+Q=R=(x3,y3)
其中：

$$x_3 = \lambda^2 - x_1 - x_2$$

$$y_3 = \lambda(x_1 - x_3) - y_1$$

$$\lambda = \begin{cases} \dfrac{y_2 - y_1}{x_2 - x_1}, \text{若} P \neq Q \\ \dfrac{3x_1^2 + a}{2y_1}, \text{若} P = Q(\text{倍点规则}) \end{cases}$$

二、有限域GF(p)上的椭圆曲线

定理：$E_p(a, b)$上的点，对于如下定义的加法规则构成一个**Abel群（交换群）**。

（一）加法规则：

⑤ 对所有的点P, Q, 满足加法交换律， 即P+Q=Q+P;

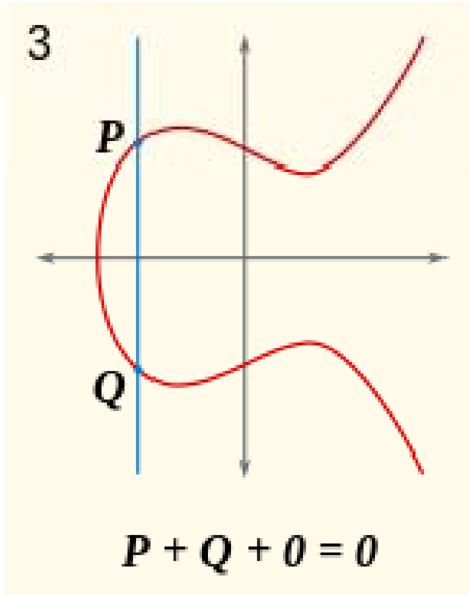
⑥ 对所有的点P, Q, R, 满足加法结合律，即P+(Q+R)=(P+Q)+R

二、有限域GF(p)上的椭圆曲线

（二）$E_p(a, b)$上的点在Abel群上加法规则的几何意义

① O是单位元；

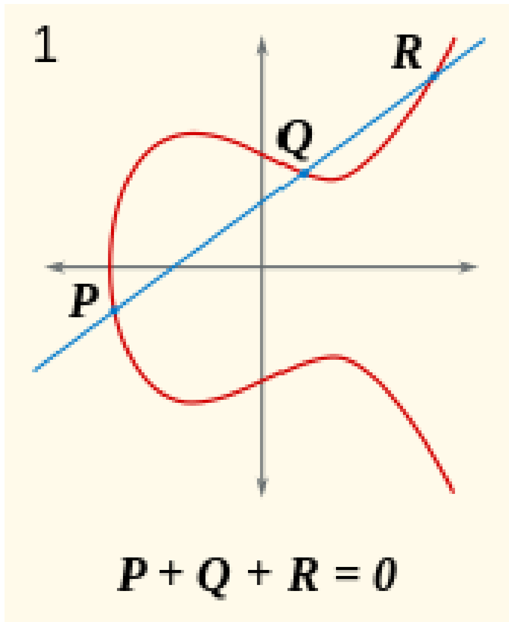
② (互为逆元点相加)一条与X轴垂直的线与曲线相交于两个点，这两个点的横坐标相同，即P=(x, y), Q=(x, -y), 同时它也与曲线相交于无穷远点O， 因此Q=-P。故椭圆曲线的性质决定P与其逆元成对地出现在椭圆曲线上。



二、有限域GF(p)上的椭圆曲线

（二）$E_p(a, b)$上的点在Abel群上加法规则的几何意义

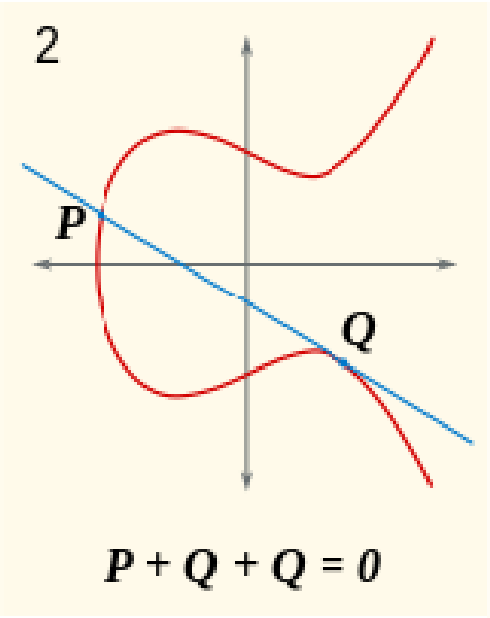
③ (不同点相加)横坐标不同的两个点P，Q相加时，先在它们之间画一条直线并求直线与曲线的第三个交点R, 则P+Q+R=O, 即P+Q=-R.



二、有限域GF(p)上的椭圆曲线

（二）$E_p(a, b)$上的点在Abel群上加法规则的几何意义

④ (相同点相加)两个相同的点Q相加时，通过该点画一条切线，切线与曲线交于另一个点P, 则Q+Q=2Q=-P.



二、有限域GF(p)上的椭圆曲线

（三）椭圆曲线点乘规则

① kP=P+P+…+P (k个P相加）

② s, t为整数，(s+t)P=sP+tP,
s(tP)=(st)P

定义1 <u>**椭圆曲线的阶**</u>：**椭圆曲线**E_p(a, b)在有限域GF(p)所有离散点的个数，记为N，称为椭圆曲线的阶。

定义2 <u>**点的阶**</u>：**P=(x,y)∈ E_p(a, b), 若存在最小的整数n，使得*n*P=O, 则称*n*为椭圆曲线上点P的阶。**

定义3 <u>**生成元**</u>：**除了无穷远点O之外，椭圆曲线上任何可以生成所有点的点都可称为椭圆曲线E的生成元，但并不是所有点都是生成元。**

三、椭圆曲线上点的计算

- **1、Hasse's theorem on elliptic curves**
- Hasse's theorem on elliptic curves, also referred to as the Hasse bound, provides an estimate of the number of points on an elliptic curve over a finite field, bounding the value below.
- If N is the number of points on the elliptic curve E over a finite field with *p* elements, then Helmut Hasse's result states that

$$|N-(p+1)| \le 2\sqrt{p}$$

三、椭圆曲线上点的计算

- **2、the generation algorithm for pointers on $E_p(a,b)$**

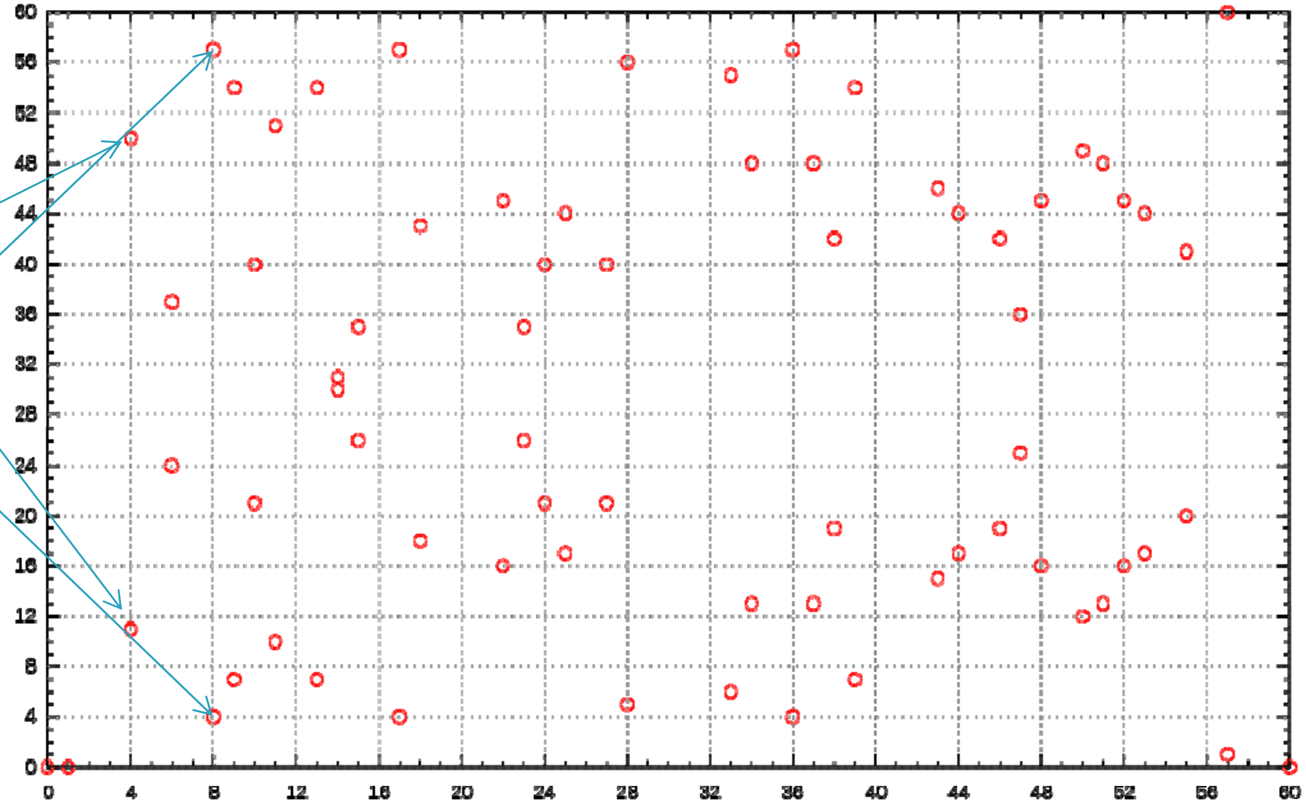
Step1: 对x=0,1,…, p-1计算$x^3+ax+b \pmod p$

Step2: 对step1得到的每一结果确定它是否有一个模p的平方根，如果没有，则$E_p(a,b)$中没有以该结果相应的x为横坐标的点；如果有，就有两个平方根y和p-y，从而点(x, y)和(x, p-y)都是$E_p(a,b)$上的点。

三、椭圆曲线上点的计算

- e.g.椭圆曲线$E_{61}(-1,0)$
- E: $y^2=x^3-x \bmod 61$

点集合
(0,0)
(1,0)
(4,11)
(4,50)
(8,4)
(8,57)
…….



三、椭圆曲线上点的计算

- e.g.椭圆曲线$E_{23}(1,1)$
- E: $y^2=x^3+x+1 \bmod 23$

表4-8　椭圆曲线上的点集 $E_{23}(1,1)$

(0, 1)	(0, 22)	(1, 7)	(1, 16)	(3, 10)	(3, 13)	(4, 0)	(5, 4)	(5, 19)
(6, 4)	(6, 19)	(7, 11)	(7, 12)	(9, 7)	(9, 16)	(11, 3)	(11, 20)	(12, 4)
(12, 19)	(13, 7)	(13, 16)	(17, 3)	(17, 20)	(18, 3)	(18, 20)	(19, 5)	(19, 18)

三、椭圆曲线上点的计算

- **e.g.椭圆曲线$E_{23}(1,1)$**
- **E: $y^2=x^3+x+1 \bmod 23$**

例：仍以$E_{23}(1,1)$为例，设P=(3,10)，Q=(9,7)，则

在mod 23互为逆元

$$\lambda=\frac{7-10}{9-3}=\frac{-3}{6}=\frac{-1}{2}\equiv(-1)\cdot 2^{-1}\equiv 22\cdot 12\equiv 11 \bmod 23$$

$$x_3=11^2-3-9=109\equiv 17 \bmod 23$$

$$y_3=11(3-17)-10=-164\equiv 20 \bmod 23$$

-164=-164+8*23 mod 23=20 mod 23

所以P+Q=（17,20），仍为$E_{23}(1,1)$中的点。

三、椭圆曲线上点的计算

- **e.g.椭圆曲线$E_{23}(1,1)$**
- **E: $y^2=x^3+x+1 \bmod 23$**

若求2P则

$$\lambda=\frac{3\cdot 3^2+1}{2\times 10}=\frac{5}{20}=\frac{1}{4}\equiv 6 \bmod 23$$

$$x_3=6^2-3-3=30\equiv 7 \bmod 23$$

$$y_3=6(3-7)-10=-34\equiv 12 \bmod 23$$

所以2P=(7,12)。

三、椭圆曲线上点的计算

【例 4-30】已知 $y^2 \equiv x^3 - 2x - 3$ 是系数在 $GF(7)$上的椭圆曲线，P=(3,2)是其上一点，求 10P。

$E_7(-2,-3)$

解：$2P = P+P=(3,2)+(3,2)=(2,6)$,

$3P = P+2P=(3,2)+(2,6)=(4,2)$,

$4P = P+3P=(3,2)+(4,2)=(0,5)$,

$5P = P+4P=(3,2)+(0,5)=(5,0)$,

$6P = P+5P=(3,2)+(5,0)=(0,2)$,

$7P = P+6P=(3,2)+(0,2)=(4,5)$,

$8P = P+7P=(3,2)+(4,5)=(2,1)$,

$9P = P+8P=(3,2)+(2,1)=(3,5)$,

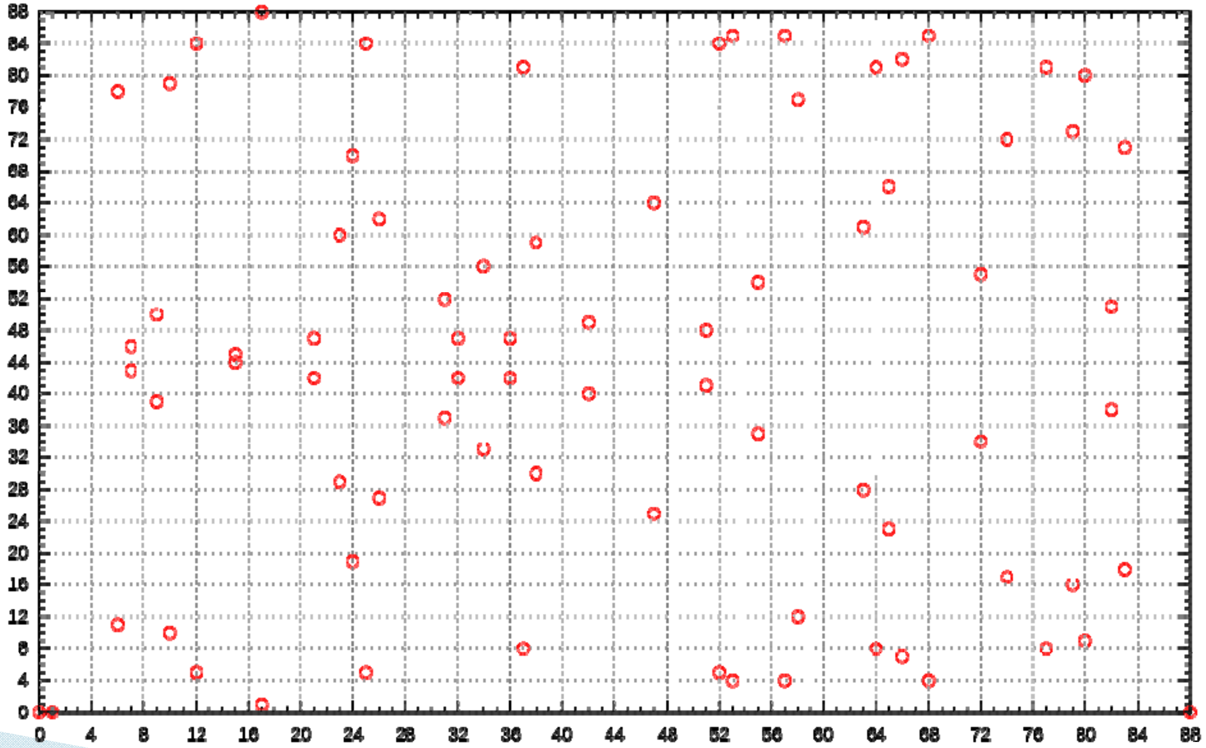
$10P = P+9P=(3,2)+(3,5)= O$.

可见P(3,2)是生成元，P的阶为10

四、ECC的密钥生成算法

椭圆曲线上所有点都落在某一个区域内，组成一个Abel群，与秘钥长度对应，秘钥长度越长，这个长度越大，这个区域越大，安全层次越高，但计算速度慢；反之亦然。

Set of affine points of elliptic curve $y^2 = x^3 - x$ over finite field F89(p=89).



Source from: https://en.wikipedia.org/wiki/Elliptic_curve

四、ECC的密钥生成算法

在$E_p(a,b)$构成的**Abel群**中，考虑方程$Q=kP$，其中$P\in E_p(a,b)$且为生成元，Q为P的倍点，即存在正整数$k(k<p)$，则由k和P易求Q。**由P、Q求k称为椭圆曲线上的离散对数问题**。事实上，对大素数构成的群E，目前还不存在多项式时间算法求解椭圆曲线上的离散对数问题，所以是一个数学难题。

四、ECC的密钥生成算法

生成一个用户B的公钥、私钥对的算法如下：

（1）选择一个椭圆曲线E：$y^2=x^3+ax+b \pmod p$，构造一个椭圆Abel群$E_p(a, b)$;

（2） 在$E_p(a, b)$中挑选生成元$G=(x_0,y_0)$, G应使得满足$nG=O$的最小n (n是G的阶)是一个非常大的素数；

（3）选择一个小于n的整数n_B作为私钥，公钥为$P_B=n_BG$, 即：

用户B的public key=$(n, G, P_B, E_p(a, b))$

用户B的secure key=n_B(小于n)

五、椭圆曲线密码体制的加解密算法

假设(发送端)A→B(接收端)实现保密通信。

（1）发送端A的加密算法

step1：将明文消息编码成一个数$m<p$，将m镶嵌到曲线上得点$P_m=(x_m,y_m)$，再对点P_m做加密变换。

step2: 在[1,n-1]内选取一个随机整数k, 计算点$P_1=kG$。k是保密的，但接收端无需知道。

step3: 根据B的公钥P_B, 计算点$P_2=kP_B$。

step4: A端传送加密数据$C_m=\{P_1, P_m+P_2\}$, 其为2个点。

五、椭圆曲线密码体制的加解密算法

假设(发送端)A→B(接收端)实现保密通信。

（2）用户B端的解密算法

step1：接收方B接收到的是2个点kG, P_m+kP_B，其用自己的私钥n_B做如下计算：

$P_m+P_2-n_BP1$即可解密，因为

$P_m+k\mathbf{P_B}-n_BkG=P_m+k\mathbf{n_BG}-n_BkG=P_m$

step2: 根据P_t，接收端再根据发送方的明文编码的镶嵌方法即可得到明文编码m，进一步得到明文。

五、椭圆曲线密码体制的加解密算法

（3）消息如何镶嵌到椭圆曲线上

step1：将明文消息编码为一个整数m，要求m<p;

step2: 对椭圆曲线E: $y^2=x^3+ax+b \pmod p$, 设置一个足够大的整数r, 将m镶嵌到椭圆曲线上，r可以在30~50之间，计算一系列x

$x=\{m*r+j,\ j=0,1,\dots,r\}$, 直到$x^3+ax+b \pmod p$是一个数的平方，即得到椭圆曲线上的点：

$$P_m=(x,\sqrt{x^3+ax+b})$$

五、椭圆曲线密码体制的加解密算法

（3）消息如何镶嵌到椭圆曲线上

反过来，通过椭圆曲线上的点(x,y)，也可计算明文消息编码

$$m = \left\lfloor \frac{x}{r} \right\rfloor$$

那么是不是一定能将m镶嵌到椭圆曲线上呢？因为在0~p的整数中，有一半是mod p的平方剩余，有一半是mod p的非平方剩余。所以在r次找到x，使得$x^3+ax+b \pmod{p}$是一个数的平方的概率不小于$1-2^{-k}$

五、椭圆曲线密码体制的加解密算法

（3）消息如何镶嵌到椭圆曲线上

例：E: $y^2=x^3+3x \pmod{4177}$，m=2174

$x=\{r*m+j=30*2174+j, j=0,1,\ldots\}$，当j=15时，

$x=30*2174+15=65235$, $x^3+3x=1444 \bmod 4177=38^2$

因此m=2174就镶嵌到椭圆曲线上的点

$$P_m=(65235, 38)$$

反之，已知椭圆曲线上的点(65235, 38)，可计算明文消息编码

$$m=\left\lfloor \frac{65235}{30} \right\rfloor = \lfloor 2174.5 \rfloor = 2174$$

六、椭圆曲线密码体制设计任务

（1）给定椭圆曲线

椭圆曲线为$E_{89}(-1,0)$： $y^2=x^3-x \pmod{89}$

六、椭圆曲线密码体制设计任务

（2）设计任务

a) 编程计算该椭圆曲线上所有在有限域GF(89)上的点；

b) 编程实现椭圆曲线上任意一个点P(例如P=(12,5))的倍点运算的递归算法，即计算k*P(k=2,3,…)；（重点！）

c) 利用此递归算法找出椭圆曲线上的所有生成元G以及它们的阶n，即满足n*G=O；

六、椭圆曲线密码体制设计任务

（2）设计任务

d) 设计实现某一用户B的公钥、私钥算法，即得到public key=(n, G, P_B, E_p(a, b))

secure key=n_B(小于n)

d) 假如用户A发送明文消息“yes”并加密传输给用户B，用户B接收消息后要能解密为明文。试用ECC密码体制实现此功能。

Thanks !