云南大学数学与统计学院
上机实践报告

课程名称：近代密码学实验	**年级**：2015 级	**上机实践成绩**：
指导教师：陆正福	**姓名**：刘鹏	
上机实践名称：椭圆曲线离散对数问题实验	**学号**：20151910042	**上机实践日期**：2018-06-03
上机实践编号：No.06	**组号**：	**上机实践时间**：07:31

一、实验目的

1. 熟悉椭圆曲线离散对数问题（ECDLP）及其有关的密码体制；
2. 实现与 ECDLP 有关的基本算法；
3. 了解参数与参数规模

二、实验内容

1. 了解椭圆曲线离散对数问题（ECDLP）有关的算法
2. 编程实现 Diffie-Hellman 密钥交换协议的椭圆曲线版本。
3. 编程实现 ElGamal 加密体制的椭圆曲线版本。

说明：基础有限域为素域$\mathrm{GF}(p)$（p为大素数）的情形为必做实验；基础有限域为$\mathrm{GF}(2^m)$的情形为选做实验

三、实验平台

Windows 10 Pro Workstation 1803；

SageMath version 8.2, Release Date: 2018-05-05；

四、实验记录与实验结果分析

1 题

了解与椭圆曲线离散对数（ECDLP）问题相关的算法。

Solution:

大多数使用公钥密码学进行加密和数字签名的产品和标准都是用 RSA 算法。近年来，为了保证 RSA 使用安全性,密钥的位数一直在增加，这对于使用 RSA 体制的应用而言是一项巨大的负担，对进行大量安全交易的电子商务与银行系统而言更是如此。近你来出现的椭圆曲线密码学（ECC）对 RSA 提出了挑战。ECC 的主要优势在于，它可以使用比 RSA 短得多的密钥得到相同安全性，减少处理荷载。

椭圆曲线并不是椭圆，之所以称之椭圆曲线为这一类方程的样式，与计算椭圆周长的方程类似，也使用三次方程来表示的。一般，椭圆曲线的三次方程形式为

$$\mathrm{y^2+axy+by=x^3+cx^2+dx+e}$$

其中，a，b，c，d和e是实数，x和y是取值在实数集上的变量。在椭圆曲线加密中，并不需要这种普通形式，下述形式已经足够：

$$y^2=x^3+ax+b$$

这是一个三次方程。椭圆曲线的定义中，还需要一个称作无穷远点或者零点的元素，记作O。

当方程满足$\mathrm{4a^3+27b^2\neq 0}$时，以椭圆曲线上的所有点作为集合，可以定义一种加法，进而作出一个阿贝尔群，即一个符合封闭性、加法结合律、加法单位元、逆元存在、加法交换律这 5 条性质的代数群。该加法是这样描述的：

（1） 无穷远点O被称为该加法的单位元，在椭圆曲线上面任取一点P，都有$P+O=O+P=P$；
（2） $\forall P=(x,\ y)$，其逆元为$-P=(x,\ -y)$；
（3） 若椭圆曲线上的三个点在共线，则认为这三个点的和为1，即若P，Q，R三点共线，则$P+Q+R=O$，进一步$P+Q=-R$。也就是说，两个不在同一条竖直线上的点，和为与之共线且在椭圆曲线上的第三点相对于横轴的镜像对称点。
（4） 对于同一个点，计算其 2 倍，只需做出该点的切线，并由此寻找该切线另外的与椭圆曲线相交的点的横轴景象。

很显然，这是一个阿贝尔群（在一些其他条件下），即交换群。因为任取两点，相加的顺序与第三点存在的位置无关。

以上是实数域上的椭圆曲线描述，在椭圆曲线密码体制中，使用的变元和系数均为有限域中元素的椭圆曲线。椭圆曲线密码体制使用两种椭圆曲线，分别是适合软件实现的定义在Z_p上的素曲线（prime curve）和适合硬件实现的定义在$\mathrm{GF}(2^m)$上的二元曲线。

先讨论素曲线的情形。此时变量和系数均在Z_p里面取值

$$y^2 \bmod p=(x^3+ax+b) \bmod p$$

这样的代数系统可以记为$\mathrm{E_p(a,\ b)}$。作为有限域，必然包含加法与乘法这两种运算。把 ECC 中的加法运算与 RSA 中的模乘运算相对应，将 ECC 中的乘法运算与 RSA 中的模幂运算对应。如果建立基于椭圆曲线的密码体制，需要类似因子分解两个素数之积或求离散对数这样的难题。考虑方程$Q=k\cdot P$，其中$Q,\ P\in E_p(a,\ b)$且$k<p$，对于给定的k和P计算Q比较容易，而对给定的Q和P计算k比较困难。这就是椭圆曲线的离散对数问题。

可以考虑，若给定了k和P，则可以通过类似快速幂取模算法的步骤，迅速得出Q，但是通过碰撞的方式去解k，就不得不一次一次累加在Z_p上计算一次椭圆曲线加法是比较消耗时间的。若$P=(x_P,\ y_P)$，$Q=(x_Q,\ y_Q)$，且$P\neq Q$，则$R=P+Q=(x_R,\ y_R)$由下列规则确定：

$$\begin{aligned} x_R &= (\lambda^2-x_P-x_Q) \bmod p \\ y_R &= (\lambda(x_P-x_R)-y_P) \bmod p \end{aligned}$$

其中

$$\lambda=$$

bash 命令：

程序代码 1

ssh 界面

安装过程分析：

六、实验体会

七、参考文献