GF(p) 上安全椭圆曲线及其基点的选取¹

张方国 王常杰 王育民

(西安电子科技大学 ISN 国家重点实验室 西安 710071)

摘要 椭圆曲线密码体制的研究与实现已逐渐成为公钥密码体制研究的主流, 适用于密码的安全椭圆曲线及其基点的选取, 是椭圆曲线密码实现的基础。在该文中, 作者讨论了大素数域上的安全椭圆曲线的选取算法和基点的选取, 并借助于 MIRACL 系统利用标准 C 语言对它们成功实现。

关键词 椭圆曲线密码体制, 安全曲线, 基点, 实现

中图分类号 TN918.1

1 引言

由 Koblitz 和 Miller 开创性的工作^[1,2], 使得被数学家研究了一百多年的椭圆曲线在密码领域中得以发挥重要作用。椭圆曲线密码体制的数学基础是利用椭圆曲线上的有理点构成的 Abelian 加法群构造的离散对数的计算困难性。由于椭圆曲线密码体制比原有的密码体制(如 Rivest Shamir Adleman, RSA 等)更具优越性, 所以对这一密码体制的研究和实现迅速普及。

1999 年 8 月 22 日, 来自 6 个不同国家的科学家们在 CWI(CWI 是在荷兰的一个数学和计算机科学的国家研究学会)的 Herman te Riele 的带领下, 在对模长是 155 位的十进制 RSA-155 的攻击中, 利用数域筛法(NFS)成功地分解出了 512-bit RSA 模的素因子。密码学家们建议使用 1024bit 模长, 预计要保证 20 年的安全性, 就要选择 1280bit 的模长, 增大 RSA 的模长带来了实现上的难度。椭圆曲线上的离散对数的计算要比有限域上的离散对数的计算更困难, 160bit 的椭圆曲线密码体制可以提供与 1024bit 的 RSA 或 DSA(Digital Signature Algorithm)同样的安全度, 从而能设计出密钥更短的公钥密码体制, 因而受到了国际上的广泛关注。

从目前 ECC 的实现结果来看, 基于大素数域的椭圆曲线密码用软件实现要比基于特征 2 的 ECC 实现要快的多。在本文中, 我们讨论了大素数域的椭圆曲线密码实现的基础性工作: 安全椭圆曲线的选取算法和基点的选取, 并借助于 MIRACL 系统^[3]利用标准 C 语言对它们成功实现。

2 ECDLP 攻击

在椭圆曲线密码体制实现中, 首先要选取安全的椭圆曲线, 所谓安全, 就是指建立在所选取的椭圆曲线上的密码体制能够抗击目前存在的所有攻击。目前对椭圆曲线离散对数的主要攻击可见表 1。

表 1 目前对椭圆曲线离散对数的主要攻击

攻击法	发表时间及作者	适用的曲线及效率
小步大步法 (BSGS)	1969 年 Shanks	所有椭圆曲线 $O(\sqrt{n})$
Pollard 方法	1978 年 Pollard	所有椭圆曲线
SPH 法	1978 年 Silver, Pohlig, Hellman	对 $\#E(F_q)$ 无大素因子非常有效
MOV 攻击 ^[4]	1991 年 Menezes, Okamoto, Vanstone	对超奇异曲线非常有效
SSSA 攻击	1997 年 Semaev, Smart, <i>hbox</i> Satoh, Araki	对非正规曲线是多项式时间的
Xedni 计算法	1998 年 Silverman	所有曲线, 但效率不高

¹ 2000-09-01 收到, 2001-03-06 定稿

国家自然科学基金重大项目资助, 批准号是: (19931010)

2000年4月4日,由 Robert Harley, Damien Doligez, Daniel de Rauglaudre 和 INRIA(法国)的 Xavier Leroy 领队的国际研究小组宣布解决了 Certicom 的 ECC2K-108 挑战。解决该挑战历时4个月,动用了9,500台机器。解决 ECC2K-108 花费了的工作量大约为分解 RSA-155(512bit RSA 挑战数)的50倍。运用这些方法解决更大参数的问题所花费的努力提供了进一步的证据说明了 Certicom 和其它公司所推荐的 163bit ECC 至少和 1024-bit RSA 的安全性相当。关于 ECC2K-108 的解的详细信息参见: <http://cristal.inria.fr/~harley/ecd17/>。

3 GF(p) 上安全椭圆曲线的选取

为了抗击上面提到的各种攻击,我们在有限域 GF(p) 上找的椭圆曲线 $E_{(a,b)}: y^2 = x^3 + ax + b$, 应该满足下列要求: $\#E_{(a,b)}$ 具有大的素因子 n , $n > 2^{160}$ 且 $n > 4\sqrt{p}$; $E_{(a,b)}$ 是非超奇异的; $E_{(a,b)}$ 不是非正规的。(有关 ECC 的基本理论可以参看文献 [5-7])

构造有限域 GF(p) 上的安全椭圆曲线的方法一般有两种: 随机曲线法 (或 Schoof-Atkin-Elkies, SEA 法) 和 CM 法。随机曲线法是通用的方法,它可以构造出许多安全的椭圆曲线,只是实现起来较复杂; CM 法相对比较简单,实现速度也比较快,但是由这种方法找出的椭圆曲线具有复乘 (Complex Multiplication), 并且和虚二次域的某个阶 (Order) 有内在的联系,所以建立在它上面的密码体制可能会存在潜在的不安全性。因此从长远的角度或从不断出现的对 ECC 的攻击来看,利用随机曲线法选择安全椭圆曲线是非常必要的。

下面给出在 GF(p) 上构造安全椭圆曲线的算法:

输入 有限域的大小 p (在我们的实现中取了 $p = 2^{176} - 233$, $p = 2^{224} - 2^{10} - 1$ 和 $p = 2^{192} - 2^{16} - 1$, 这些素数都形如 $p = 2^n - c$ (c 是小整数)。GF(p) 上 ECC 实现的最基本的运算是 GF(p) 中的域元素运算,而 GF(p) 中域元素运算用的最多的是两个数的模乘运算。当 p 是形如 $p = 2^n - c$ 的素数时, A 、 B 是 GF(p) 中的两个元素, $C = AB$ 是一个 $2n$ 比特的数, C 可写成 $C = 2^n H + L$ ($H, L < 2^n$), 则 $C \equiv cH + L \pmod{p}$, 这是我们为什么选取这类素数的原因)。

输出 椭圆曲线 $E_{(a,b)}$, $\#E_{(a,b)} = nh$, n 是大的素因子, $n > 2^{160}$ 且 $n > 4\sqrt{p}$ 。

- (1) 随机产生 GF(p) 上的一条椭圆曲线 $E_{(a,b)}: y^2 = x^3 + ax + b$;
- (2) 利用 SEA 算法计算出 $\#E_{(a,b)}$;
- (3) 利用大整数分解算法分解 $\#E_{(a,b)}$, 并检测 $\#E_{(a,b)}$ 的分解中是否有 $> 2^{160}$ 的素因子 n 。如果没有, 返回 (1);
- (4) 检测 $n \neq p$, 否则返回 (1);
- (5) 检测 $p^k \neq 1 \pmod{n}$, $k = 1, 2, \dots, 20$ 。否则返回 (1);
- (6) 输出 $a, b, \#E_{(a,b)}, n, h = \#E_{(a,b)}/n$ 。

实现上面算法的最关键的一步就是计算椭圆曲线点的个数 $\#E_{(a,b)}$, 即 SEA 算法的实现。在有限域 GF(p) 上, 这里 $p > 3$, 给定一条椭圆曲线 $y^2 = x^3 + ax + b$, $a, b \in GF(p)$, 且 $4a^3 + 27b^2 \neq 0 \pmod{p}$, 计算它在这个域上的点的个数是一个非常重要和非常有意义的工作。这不仅是一个数论中的重要问题, 而且也是密码学, 特别是现代公钥密码学中一个急需解决的问题。当有限域比较小时, 我们可以直接遍历这个有限域, 找到适合椭圆曲线的点, 从而确定椭圆曲线的点的个数。这种方法需要 p^2 次算术运算, 当 p 非常大时, 这种方法显然是不可能的。1985年, Schoof^[8] 提出了一个计算复杂度是多项式时间的算法, 但由于整个算法需要的

存储量相当大,所以在具体实现算法时仍然不太可行。之后,Atkin 和 Elkies 对 Schoof 算法进行了一些改进,即 SEA 算法,使得这一算法在实际中实现成为可能^[9,10]。

由 Hasse 定理知: $\#E(\text{GF}(p)) = p+1-t$, 其中 $|t| \leq 2\sqrt{p}$, 并且满足 $\phi^2 - t\phi + p = 0$ 。这里 ϕ 是 Frobenius 变换, 即 $\phi(x, y) = (x^p, y^p)$ 。所以如果能确定出 t , 也就确定出了 $\#E_{(a,b)}$ 。SEA 算法的基本思想是: 先对小素数 l_i 计算出 $t_i \equiv t \pmod{l_i}$, $i = 1, 2, \dots, s$, 且 $\prod_{i=1}^s l_i > 4\sqrt{p}$; 然后利用中国剩余定理计算出 t 。其中 l_i 可以从 $2, 3, 5, 7, \dots$ 取起。

输入 有限域的大小 p , 椭圆曲线 $E_{(a,b)}$ 的参数 $a, b \in \text{GF}(p)$, $j(E) = 12^3 \times \frac{4a^3}{4a^3 + 27b^2}$, $F(x, y) = y^2 - x^3 - ax - b$ 。

输出 $\#E_{(a,b)}$ 。

(1) 令 $l = 1; M = 1; i = 1$;

(2) while($M < 4\sqrt{p}$) do

(a) $l = \text{nextprime}(l)$;

(b) 计算第 l 个模多项式 $\Phi_l(x, y) \in \text{GF}(p)[x, y]$;

(c) 计算 $\Phi_l(x, j(E)) \in \text{GF}(p)[x]$;

(d) 检验 $\text{GCD}(x^p - x, \Phi_l(x, j(E))) = G_l(x) \neq 1$, 否则 go to (a);

(e) 计算除多项式 $d_l(x)$ 的一个 $(l-1)/2$ 次因式 $g_l(x)$;

(f) 计算 λ_l , 使得 $(x^p, y^p) = \lambda_l(x, y) \pmod{(F(x, y), g_l(x))}$;

** 这里 $\text{mod}(F(x, y), g_l(x))$ 表示计算 (x^p, y^p) , $\lambda_l(x, y)$ 的过程中, x, y 的坐标按模 $F(x, y)$ 和模 $g_l(x)$ 运算 **

(g) 计算 $t_l = (\lambda_l^2 + p)/(\lambda_l \pmod{l})$, 即得 $t_l \equiv t \pmod{l}$;

(h) $M = M \times l$, $i++$ 。

(3) 利用中国剩余定理计算 t

(a) for($j = 1, j \leq i, j++$)

$L_j = M/l_j$; $y_j = L_j^{-1} \pmod{l_j}$ 。

(b) 计算 $t = t_1 L_1 y_1 + t_2 L_2 y_2 + \dots + t_i L_i y_i$

(4) 输出 $\#E_{(a,b)} = p + 1 - t$ 。

在上面的过程中, 比较困难的是计算第 l 个模多项式 $\Phi_l(x, y) \in \text{GF}(p)[x, y]$ 和计算除多项式 $d_l(x)$ 的一个 $(l-1)/2$ 次因式 $g_l(x)$ 。对于模多项式这一步, 我们可以预计算。在我们的实现中, 我们预计算了小于 200, 180 和 160 的所有素数的模多项式, 并把它们分别存入文件 GFp224.pol, GFp192.pol 和 GFp176.pol 中, 在 SEA 算法中直接调用即可。我们借助于 MIRACL 系统^[3], 用标准 C 成功实现了上面寻找安全曲线的算法。我们的程序取素域 $p = 2^{176} - 233$, $p = 2^{192} - 2^{16} - 1$ 和 $p = 2^{224} - 2^{10} - 1$, 用 Pentium-II350MHz 计算机运行计算, 并在随机选取曲线时我们固定 $a = -3$ (因为 $a = -3$ 时, 在射影坐标下, 椭圆曲线的倍点运算比一般选取的 a 少了 2 次 $\text{GF}(p)$ 中元素乘法运算)。当 $p = 2^{224} - 2^{10} - 1$ 时, 程序运行了 40 多个小时, 共测试了 $\text{GF}(p)$ 上 $(p = 2^{224} - 2^{10} - 1)237$ 条椭圆曲线, 输出结果有 16 条满足要求的安全曲线。 $p = 2^{224} - 2^{10} - 1$ 时是最费时的。在 $p = 2^{176} - 233$ 时, 大约平均 4-5min 测试一条曲线; $p = 2^{192} - 2^{16} - 1$ 时约为 9min。

4 基点的选取

要构造基于离散对数的密码体制,就必须找出椭圆曲线加法群的大素因子子群的一个生成元,即基点。为找基点,我们先找椭圆曲线上任意一个随机点。

下面算法是寻找有限域 $\text{GF}(p)$ 上的给定椭圆曲线上的一个随机点(无穷远点除外)的方法[7]:

输入 一个素数 $p > 3$, $\text{GF}(p)$ 上椭圆曲线 E 的参数 a, b ;

输出 E 上一个随机点(无穷远点除外)。

(1) 随机选取 x , $0 \leq x < p$;

(2) 令 $\alpha \leftarrow x^3 + ax + b \pmod{p}$;

(3) 如果 $\alpha = 0$, 输出 $(x, 0)$ 并停止;

(4) 利用模 p 的平方根方法求 α 的一个平方根或判断它不存在;

(5) 如果(4)中的结果没有平方根存在,则返回(1)。否则(4)输出一个整数 β , $0 < \beta < p$ 使得 $\beta^2 \equiv \alpha \pmod{p}$;

(6) 生成一个随机比特 μ , 令 $y \leftarrow (-1)^\mu \beta$;

(7) 输出 (x, y) 。

设 $\#E(\text{GF}(q)) = n = hk$, k 是椭圆曲线 E 的阶的大素因子(至少 160bit), 下面算法给出找 E 上一个具有 k 阶的点, 即所用椭圆曲线群的基点。

输入 h, k , $\text{GF}(q)$ 上椭圆曲线 E , $\#E(\text{GF}(q)) = hk$;

输出 E 上阶为 k 的点 G 。

(1) 用上面算法生成 E 上一个随机点 P (不是 O);

(2) 令 $G \leftarrow hP$;

(3) 如果 $G = O$, 则返回(1);

(4) 输出 G 。

注: 判断(3)中的 G 是否为 O , 只需验证 $G = (x, y)$ 是否满足椭圆曲线方程 $y^2 = x^3 + ax + b$, 如果不满足, 则认为 $G = O$ 。

5 结 束 语

在本文中,我们讨论了大素数域上的安全椭圆曲线的选取算法和基点的选取,并借助于 MIRACL 系统利用标准 C 语言对它们成功实现。有了安全椭圆曲线和基点,就有了椭圆曲线密码的基础,从而可以很容易地实现 ECC, 例如 Diffie-Hellman 密钥分配协议、ElGamal 加解密、ECDSA(椭圆曲线数字签名算法)等等。任何一个公钥密码在实际中实现所面对的一个实际问题就是实现速度, ECC 也不例外。标量乘运算是影响 ECC 实现的一个关键步骤。加速标量乘, 从而提高 ECC 的实现速度是目前 ECC 实现中的一个重要研究课题。

参 考 文 献

- [1] N. Koblitz, Elliptic curve cryptosystems, Mathematics of Computation, 1987, 45(177), 203-209.
- [2] V. Miller, Uses of elliptic curves in cryptography, Advances in Cryptology-CRYPTO' 85, LNCS218, Santa Barbara, Calif., Springer-Verlag, 1986, 417-426.
- [3] Multiprecision Integer and Rational Arithmetic C/C++ Library(MIRCAL), available at <http://indigo.ie/~mscott/>.

- [4] A. Menezes, T. Okamoto, S. Vanstone, Reducing elliptic curve logarithms to logarithms in a finite field, *IEEE Trans. on Information Theory*, 1993, 39(5), 1639–1646.
- [5] N. Koblitz, *A Course in Number Theory and Cryptography*, 2nd edition, Springer-Verlag, 1994, Ch.6.
- [6] N. Koblitz, *Algebraic Aspects of Cryptography, Algorithms and Computation in Math.* Editors: E. Becker, M. Bronstein, H. Cohen, 3(1998), Berlin Heidelberg, Springer-Verlag, 1998, Ch.6.
- [7] IEEE P1363, *Standard Specifications for Public-Key Cryptography*, Ballot Draft, 1999, Drafts available at <http://indigo.ie/~mscott/>.
- [8] R. Schoof, Elliptic curves over finite fields and the computation of square roots mod p , *Math. Comp.*, 1985, 44(170), 483–494.
- [9] T. Izu, J. Kogure, M. Noro, K. Yokoyama, *Efficient Implementation of Schoof's Algorithm*, *Asiacrypt'98*, Berlin Heidelberg, Springer-Verlag, 1998, 66–79.
- [10] Lehmann, Maurer, Mueller, Shoup, Counting the number of points on elliptic curves over finite fields of characteristic greater than three, *Proc. 1st Algorithmic Number Theory Symposium (ANTS)*, Berlin Heidelberg, Springer-Verlag, 1994, LNCS877, 60–70.

THE SELECTION OF SECURE ELLIPTIC CURVES AND THEIR BASE POINTS OVER $GF(p)$

Zhang Fangguo Wang Changjie Wang Yumin

(Key Laboratory on ISN, Xidian University, Xi'an 710071, China)

Abstract The study and realization of elliptic curve cryptography have become to be the mainstream of the study on public cryptography, and the selection of secure elliptic curves in cipher and their base points is the basis of the implementing of the elliptic curves cryptography. In this paper, the selection algorithm of the secure elliptic curves and their base point over large prime number is discussed, further more, the algorithm is implemented successfully in C language using the MIRACL system.

Key words Elliptic curve cryptosystem, Secure curve, Base points, Implementation

张方国: 男, 1972 年生, 博士生, 研究兴趣是电子商务, 椭圆曲线密码和超椭圆曲线密码.

王常杰: 男, 1974 年生, 博士生, 研究方向为网络安全及电子商务.

王育民: 男, 1936 年生, 教授, 博士生导师, 主要研究方向为通信网的安全及电子商务.