



现代密码学

Modern Cryptography

张方国

中山大学计算机学院

Office: Room 305, IM School Building

E-mail: isszhfg@mail.sysu.edu.cn

HomePage: <https://cse.sysu.edu.cn/content/2460>





第十九讲 DLP (二)

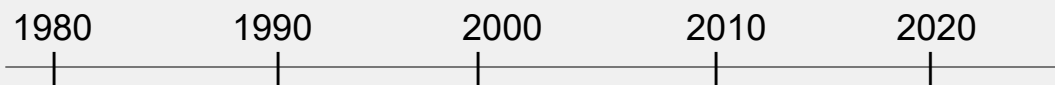
第6章 公钥密码学和离散对数

- 椭圆曲线密码体制







公钥密码学的研究热点




RSA (整数分解**问题**)




ECC (短的密钥, 离散对数问题困难)



基于配对的密码体制(IBE)



后量子密码体制(格, 纠错码, **同源**等)





ECC

- 传统的ECC

基于ECDLP的密码体制，如ECDSA, ECIES, SM2等

- 广义的ECC

传统ECC，

基于双线性对的密码体制

基于椭圆曲线同源的密码体制





椭圆曲线及其相关问题

- 什么是椭圆曲线？

椭圆曲线(**Elliptic Curve**) 是由如下形式的方程给出的一条曲线

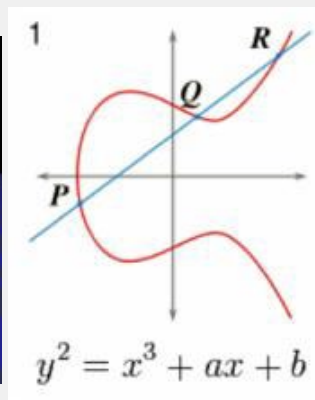
$$y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6;$$

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

$$-(4a^3 + 27b^2) \neq 0$$

在 \mathbf{C} 上, E 同构于 \mathbf{C}/Λ , Λ 是一个格.

密码上: 主要研究曲线在 $\mathbf{F}_p = \mathbf{Z}/p\mathbf{Z}$
或更一般的 \mathbf{F}_q 上的解





Number Theory, Physics

- 椭圆曲线与椭圆函数(复分析理论)
- 椭圆曲线与模形式
- 椭圆曲线与费马大定理Fermat's last theorem
- **椭圆曲线与密码学**
- 椭圆曲线与弦理论(String Theory)
- 椭圆曲线与纠错码—AG code

张方国, [椭圆曲线在密码中的应用:过去,现在,将来...](#) . 山东大学学报 (理学版), 2013 Vol. 48 (05): 1-13





DLP 密码体制

当一个群 G 满足如下性质时，可用于构造密码体制：

- 1，群元素可以（用计算机）紧致的表示；
- 2，群运算可以有效的执行；
- 3，DLP(给定 $g, h=g^a$, 计算 a)是困难的。





Koblitz, N.: Elliptic curve cryptosystems. Math. Comput. 48, 203–209 (1987).

Miller, V.: Use of elliptic curves in cryptography. CRYPTO'85, LNCS 218, pp. 417–426

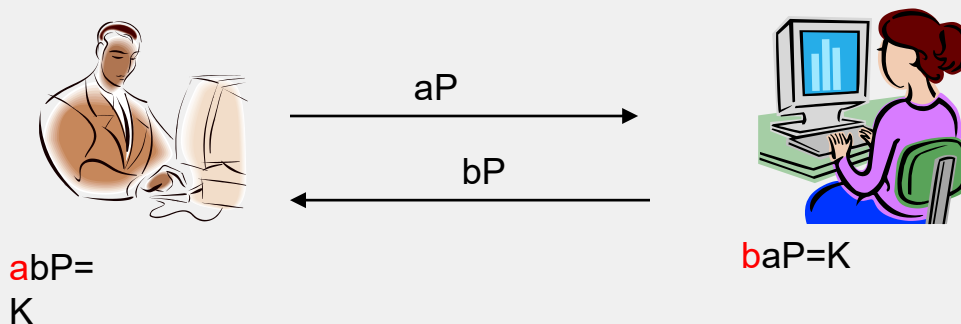


- **ECC参数:**

有限域 F_q 上的椭圆曲线 E , $\#E(F_q)=nh$, n 是一个素数, P 是群 $E(F_q)$ 的一个 n 阶生成元。从 $[0, n-1]$ 中随机选 d 。

公钥是 $Q=dP$; **私钥**是 d 。

- 基于一般离散对数的方案可以转化成ECC的, 例如Diffie-Hellman密钥协商 — ECDH





ECC是当前主流的公钥密

- 比特币与ECC 码体制!

$$E : y^2 = x^3 + 7 \text{ over } F_p, \quad p = 2^{256} - 2^{32} - 977$$



- SSH: ECDSA, ECDH
- TLS
- Austrian e-ID Card奥地利电子身份证
- 伪随机数产生器
- 中国商密标准及相关产品:
SM2

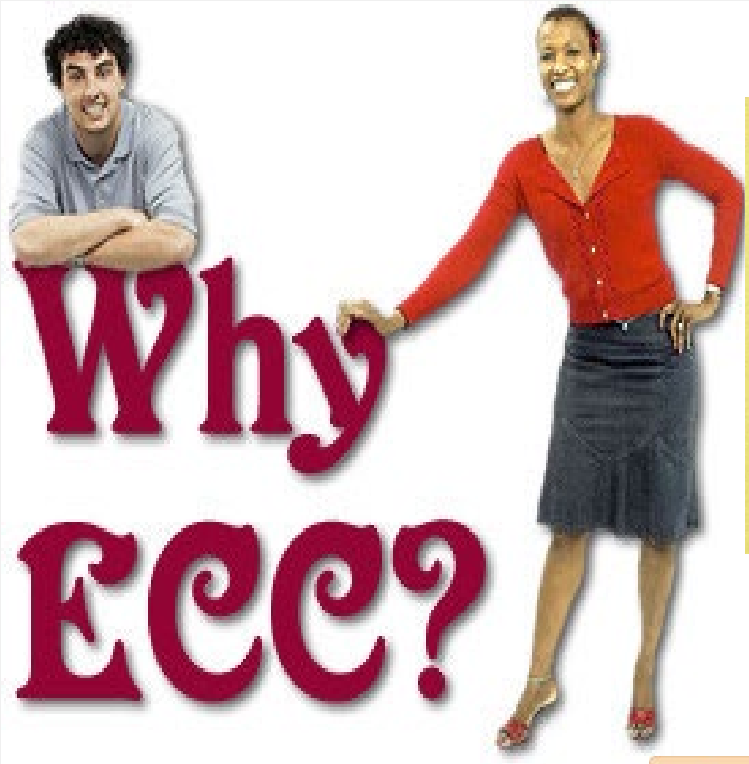




当前ECC标准

- ANSI X9: 62, 63, 92, ...
- IEEE: 1363-2000, P1363a, P1363.2, P802.15.3/4, ...
- ISO: 14888-3, 9496, 15496, 18033-2, ...
- FIPS: 186-2, 2XX, ...
- NESSIE, IPA Cryptrec, ...
- SECG: SEC1, SEC2, ...
- IETF: PKIX, IPsec, SMIME, TLS, ...
- SET, MediaPlayer, 5C, WAP, ...
- China: SM2





Why ECC?

ECC key (bits)	RSA key (bits)	Ratio
163	1024	1:6
256	3072	1:12
384	7680	1:20
512	15360	1:30

我们希望更短的秘钥:





实数域上的椭圆曲线

定义6.3 设 $a, b \in R$ 是满足 $4a^3 + 27b^2 \neq 0$ 的常实数。方程

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

的所有解 $(x, y) \in R \times R$ 连同同一个无穷远点 \mathcal{O} 组成的集合 E 称为一个非奇异椭圆曲线。

条件 $4a^3 + 27b^2 \neq 0$ 是保证方程 $y^2 = x^3 + ax + b$ 有3个不同解的充要条件。如果 $4a^3 + 27b^2 = 0$ ，则对应的椭圆曲线称为奇异椭圆曲线。

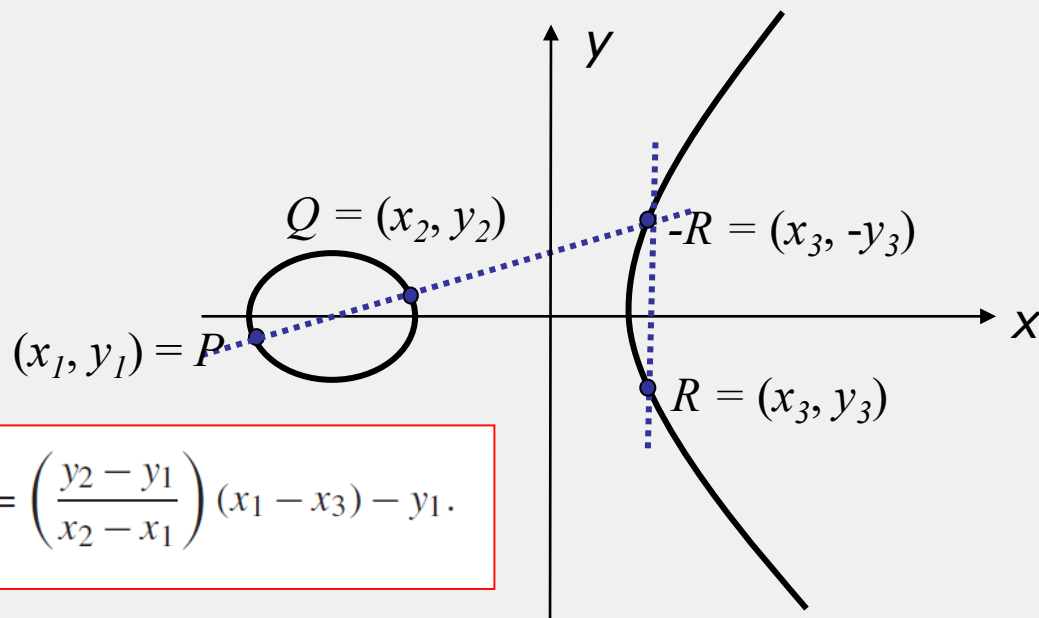




椭圆曲线群运算：切割线法则

- 点加

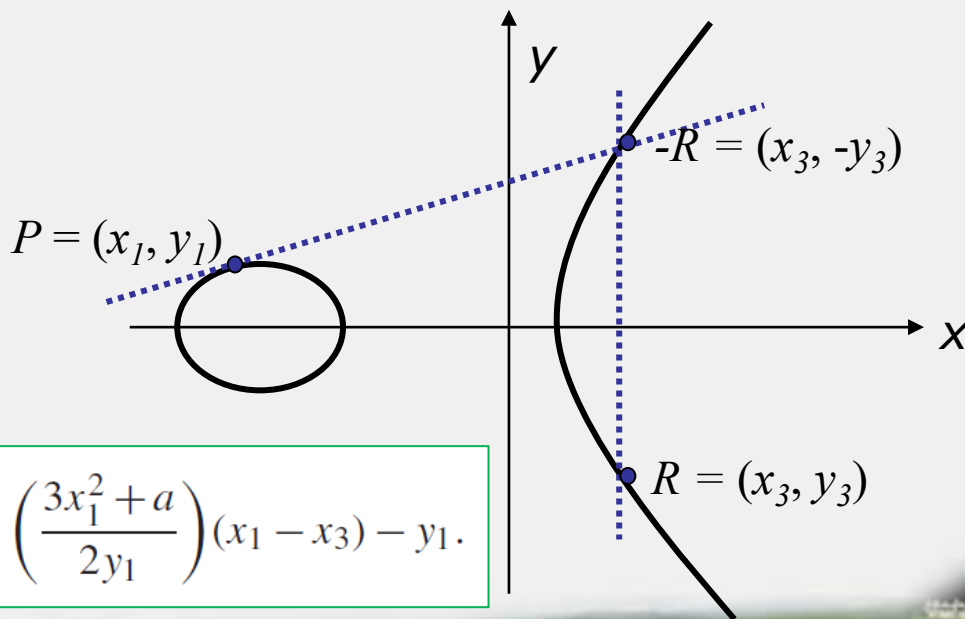
$$P + Q = R, P \neq Q$$



$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1} \right)^2 - x_1 - x_2 \quad \text{and} \quad y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1} \right) (x_1 - x_3) - y_1.$$

- 倍点

$$P + P = 2P = R$$



$$x_3 = \left(\frac{3x_1^2 + a}{2y_1} \right)^2 - 2x_1 \quad \text{and} \quad y_3 = \left(\frac{3x_1^2 + a}{2y_1} \right) (x_1 - x_3) - y_1.$$



椭圆曲线群运算

假定 E 是一个非奇异椭圆曲线。在 E 上定义一个二元运算，使其成为一个阿贝尔群。这个二元运算通常用加法表示。无穷远点 \mathcal{O} 将是单位元。因此，有 $P + \mathcal{O} = \mathcal{O} + P = P$ 对于所有的 $P \in E$ 。

假设 $P, Q \in E$ ，其中 $P = (x_1, y_1), Q = (x_2, y_2)$ 。

- 如果 $x_1 \neq x_2$ ，则 $(x_1, y_1) + (x_2, y_2) = (x_3, y_3)$ ，其中 $x_3 = \lambda^2 - x_1 - x_2$ ， $y_3 = \lambda(x_1 - x_3) - y_1$ ， $\lambda = \frac{y_2 - y_1}{x_2 - x_1}$ ；
- 如果 $x_1 = x_2$ 和 $y_1 = -y_2$ ，则 $(x, y) + (x, -y) = \mathcal{O}$ 。因此， (x, y) 和 $(x, -y)$ 是关于椭圆曲线加法运算互逆的；
- 如果 $x_1 = x_2$ 和 $y_1 = y_2$ ，则 $x_3 = \lambda^2 - x_1 - x_2$ ， $y_3 = \lambda(x_1 - x_3) - y_1$ ， $\lambda = \frac{3x_1^2 + a}{2y_1}$

群运算满足封闭性，结合律，单位元，逆元，交换律成立

因此， $(E, +)$ 是一个阿贝尔群。





模素数的椭圆曲线

定义在有限域 $GF(p)$ 上的椭圆曲线

设 $p > 3$ 是素数。 Z_p 上的椭圆曲线可以与实数域上的一样定义，只是 R 上的运算用 Z_p 上的类似运算代替。

定义6.4 $p > 3$ 是素数。 Z_p 上的同余方程

$$y^2 \equiv x^3 + ax + b \pmod{p} \quad (6.6)$$

的所有解 $(x, y) \in Z_p \times Z_p$ ，连同一个特殊的点 \mathcal{O} ，即无穷远点。共同构成 Z_p 上的椭圆曲线 $y^2 = x^3 + ax + b$ 。其中 $a, b \in Z_p$ 是满足 $4a^3 + 27b^2 \neq 0$ 的常量。

按照实数上的椭圆曲线定义群的方式，我们得到一个完全类似的阿贝尔群。

有限交换群！





椭圆曲线的性质

定义在 \mathbb{Z}_p (p 是素数), $p > 3$ 上的椭圆曲线 E 大致有 p 个元素。

Hasse定理 E 上的点数如果记为 $\#E$, 那么它满足下面的不等式

$$p + 1 - 2\sqrt{p} \leq \#E \leq p + 1 + 2\sqrt{p}$$

计算 $\#E$ 的准确值比较困难, 但有一个有效的算法计算它, 这个算法由Schoof发现。

如果可以计算 $\#E$, 进一步要找到 E 的一个循环子群, 在其中离散对数问题是难解的。所以要对群 E 的结构有所了解。

计算点数的SEA算法! 从 $O(\log^8 p)$ 到 $O(\log^6 p)$ 。





椭圆曲线的性质

定理6.1 E 是定义在 \mathbb{Z}_p 上的一个椭圆曲线，其中 p 是素数， $p > 3$ 。则存在正整数 n_1 和 n_2 ，使得 $(E, +)$ 同构于 $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$ ，并且有 $n_2 | n_1$ 和 $n_2 | p - 1$ 。

$n_2 = 1$ 是可能的。事实上， $n_2 = 1$ 当且仅当 E 是循环群。同时，如果 $p \nmid n_1$ 或者是一个素数，或者是两个不同素数的乘积，那么 E 也必定是循环群。

一旦整数 n_1 和 n_2 确定，则 $(E, +)$ 具有一个循环子群同构于 \mathbb{Z}_{n_1} ，这意味着它可以用于 $ElGamal$ 密码体制的情形。

通用算法适用于椭圆曲线的离散对数问题，而指数演算法对椭圆曲线情形的适用性不得而知。

有一个方法揭示了椭圆曲线与有限域的同构，这对某些类型的椭圆曲线给出了一个有效的算法。

这个技巧属于Menezes, Okamoto和Vanstone，它适用于所谓超奇异曲线，这是一类特殊椭圆曲线，它们被建议用于密码体制。

另外一类弱椭圆曲线，就是所谓的“迹一”的曲线。这些曲线定义在 \mathbb{Z}_p 上，具有 p 个元素。这些曲线上的离散对数问题是易解的。



点压缩与ECIES

在实际中，实现椭圆曲线上的ElGamal密码体制存在着一些困难。

在 \mathbb{Z}_p 中实现ElGamal密码体制有一个二倍的消息扩张因子。椭圆曲线的实现有一个大约四倍的消息扩张因子。之所以如此，因为大致有 p 个明文，但每个密文有四个域元素组成。

更为严重的是，密文空间由 E 上的点组成，没有一个方便的方法能够确定地生成 E 上的点。

一个较为有效的ElGamal型体制用在所谓的ECIES(椭圆曲线集成加密方案)。

ECIES不仅结合了消息认证码，而且结合了对称密钥加密，描述十分复杂。我们用一个简单的版本来描述，主要实现用于ECIES的基于ElGamal 公钥加密方案的椭圆曲线。





点压缩与ECIES

点压缩,可以降低椭圆曲线上点的存储空间。椭圆曲线 E 上的一个(非无穷)点是一个对 (x,y) 。给定一个 x 的值, y 有两个可能的值。这两个可能的 y 值模 p 是互为相反数。因为 p 是奇数,两个可能的 $y \bmod p$ 中,一个是奇数,一个是偶数。因此可以通过指定 x 的值,连同 $y \bmod 2$ 一个比特确定 E 上唯一点 $P = (x,y)$ 。这减少了大约50%的存储空间,代价是需要额外的计算来重构 P 点的 y 坐标。

点压缩运算可以表示为函数

$$\text{Point-Compress} : E \setminus \{O\} \rightarrow Z_p \times Z_2$$

具体定义为:

$$\text{Point-Compress}(P) = (x, y \bmod 2)$$

其中 $P = (x,y) \in E$ 。逆运算Point-Decompress从 $(x, y \bmod 2)$ 重构 E 上的点 $P = (x,y)$ 。



密码体制6.2 简化的ECIES

令 E 是定义在 Z_p ($p > 3$ 是素数)上的一个椭圆曲线, E 包含一个素数阶 n 的循环子群 $H = \langle P \rangle$, 其上的离散对数问题是难解的。

设 $P \in Z_p^*$, $C = (Z_p \times Z_2) \times Z_p^*$, 定义

$$K = \{(E, P, m, Q, n) : Q = mp\}$$

值 P, Q 和 n 是公钥, $m \in Z_n^*$ 是私钥。

对 $K = (E, P, m, Q, n)$, 一个(秘密)随机数 $k \in Z_n^*$, 以及 $x \in Z_p^*$, 定义

$$e_K(x, k) = (\text{Point-Compress}(kP), xx_0 \mod p)$$

其中 $kQ = (x_0, y_0)$ 和 $x_0 \neq 0$ 。

对密文 $y = (y_1, y_2)$, 这里 $y_1 \in Z_p \times Z_2$ 和 $y_2 \in Z_p^*$, 定义

$$d_K(y) = y_2(x_0)^{-1} \mod p$$

其中

$$(x_0, y_0) = m\text{Point-Decompress}(y_1)$$





计算椭圆曲线的乘积（标量乘）

可以利用平方-乘算法在乘法群中有效地计算幂 α^a 。

类似，我们可以使用一个“倍数-和”算法计算椭圆曲线点 P 的倍数 aP 。

设 c 是一个整数， c 的一个带符号的二进制表示可以如下：

$$(c_4, c_3, c_2, c_1, c_0) = (0, 1, 0, 1, 1) \text{ or } (1, 0, -1, 0, -1)$$

其中 $c = 11$ 。

假定 $c = (c_{l-1}, \dots, c_0)$ ，则运用下列算法可以计算椭圆曲线上的点集 cP 。





计算椭圆曲线的乘积（标量乘）

算法**6.5** 倍数-和差算法($P, (c_{l-1}, \dots, c_0)$)

$Q \leftarrow \mathcal{O}$

for $i \leftarrow l - 1$ downto 0

do

$$\left\{ \begin{array}{l} Q \leftarrow 2Q \\ \text{if } c_i = 1 \\ \text{then } Q \leftarrow Q + P \\ \text{else if } c_i = -1 \\ \text{then } Q \leftarrow Q - P \end{array} \right.$$

return(Q)

