Advanced Cryptography

Elliptic Curve Cryptography

Christophe Petit

University of Oxford

Discrete Logarithm Problem (1977)

The new technique makes use of the apparent difficulty of computing logarithms over a finite field $GF(q)$ with a prime number q of elements. Let

$$Y = \alpha^X \bmod q, \qquad \text{for } 1 \le X \le q-1, \qquad (4)$$

where α is a fixed primitive element of $GF(q)$, then X is referred to as the logarithm of Y to the base α, mod q:

$$X = \log_\alpha Y \bmod q, \qquad \text{for } 1 \le Y \le q-1. \qquad (5)$$

Calculation of Y from X is easy, taking at most $2 \times \log_2 q$ multiplications [6, pp. 398–422]. For example, for $X = 18$,

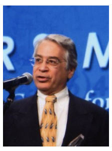
$$Y = \alpha^{18} = (((\alpha^2)^2)^2)^2 \times \alpha^2. \qquad (6)$$

Computing X from Y, on the other hand can be much more difficult and, for certain carefully chosen values of q, requires on the order of $q^{1/2}$ operations, using the best known algorithm [7, pp. 9, 575–576], [8].

Source : Whitfield Diffie, Martin Hellman, *New directions in Cryptography*

Generalization to other Groups

- Given a cyclic group $(G, \circ)$ (written multiplicatively), a generator g of G and a second element $h \in G$, compute k such that $g^k = h$



- 1985 : Koblitz and Miller independently propose to use elliptic curves in cryptography

ECRYPT II key length recommendations (2012)

Level	Protection	Symmetric	Factoring Modulus	Discrete Logarithm Key	Discrete Logarithm Group	Elliptic Curve	Hash
1	Attacks in "real-time" by individuals *Only acceptable for authentication tag size*	32	-	-	-	-	-
2	Very short-term protection against small organizations *Should not be used for confidentiality in new systems*	64	816	128	816	128	128
3	Short-term protection against medium organizations, medium-term protection against small organizations	72	1008	144	1008	144	144
4	Very short-term protection against agencies, long-term protection against small organizations *Smallest general-purpose level, 2-key 3DES restricted to 2^{40} plaintext/ciphertexts, protection from 2015 to 2015*	80	1248	160	1248	160	160
5	Legacy standard level *2-key 3DES restricted to 10^6 plaintext/ciphertexts, protection from 2015 to 2020*	96	1776	192	1776	192	192
6	Medium-term protection *3-key 3DES, protection from 2015 to 2030*	112	2432	224	2432	224	224
7	Long-term protection *Generic application-independent recommendation, protection from 2015 to 2040*	128	3248	256	3248	256	256
8	"Foreseeable future" *Good protection against quantum computers, unless Shor's algorithm applies*	256	15424	512	15424	512	512

All key sizes are provided in bits. These are the minimal sizes for security.

Source : `www.keylength.com`

Outline

Elliptic Curves

Elliptic Curve Discrete Logarithm Problem

Algorithmic Aspects

Factorization and Primality testing

Pairings

Main references

- Silverman, *The Arithmetic of Elliptic Curves*
- Blake-Seroussi-Smart, *Elliptic curve cryptography*
- Blake-Seroussi-Smart, *Advances in Elliptic curve cryptography*

Outline

Elliptic Curves
- Weierstrass equations
- The group law
- Rational maps between elliptic curves

Elliptic Curve Discrete Logarithm Problem

Algorithmic Aspects

Factorization and Primality testing

Pairings

Outline

Elliptic Curves
- Weierstrass equations
- The group law
- Rational maps between elliptic curves

Elliptic Curve Discrete Logarithm Problem

Algorithmic Aspects

Factorization and Primality testing

Pairings

Elliptic Curve

- Smooth, projective algebraic curve of genus one, with a specified point O
- O is the"point at infinity" in the projective plane
- Abelian variety : forms a commutative group defined by algebraic fomulae, with O as the identity element
- In this course
  - Curves over finite fields
  - Concrete models

Weierstrass equation

- Let K be a (finite) field and $a_i \in K$
- Weierstrass equation

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

- $E(K) = \{(x, y) \in K^2 \text{ satisfying the equation}\} \cup \{O\}$
- O is a special point, called *point at infinity*

Projective coordinates

$$Y^2Z + a_1XYZ + a_3YZ^2 = X^3 + a_2X^2Z + a_4XZ + a_6Z^3.$$

- *Homogeneous* or *projective* Weierstrass equation
- Point at infinity $O = [0 : 1 : 0]$

Parameters of Weierstrass curves

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

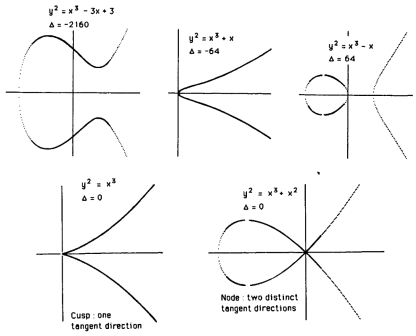
- Define $b_2 = a_1^2 + 4a_2$, $b_4 = 2a_4 + a_1a_3$, $b_6 = a_3^3 + 4a_6$, $b_8 = a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2$,
- Define $c_4 = b_2^2 - 24b_4$, $c_6 = b_2^3 + 36b_2b_4 - 216b_6$
- Define the *discriminant* $\Delta = -b_2^2b_8 - 8b_4^3 - 27b_6^2 + 9b_2b_4b_6$
- Define the *j-invariant* $j = \frac{c_4^3}{\Delta}$

Regular curves and Discriminant

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

- Let $f(x, y) = y^2 + a_1xy + a_3y - x^3 - a_2x^2 - a_4x - a_6$
- The curve is *regular* iff
  $f(x_0, y_0) = 0 \Rightarrow (\delta f/\delta x, \delta f/\delta y)_{|(x_0,y_0)} \neq (0, 0)$
  Otherwise, the curve is *singular*
- E regular $\Leftrightarrow \Delta(E) \neq 0$ (Proof : Silverman Prop. III.1.4)
- This course : elliptic curves are smooth/regular curves

Regular vs Singular Curves (over rational field)



Pictures source : Silverman, the arithmetic of elliptic curves

Reduced Weierstrass equation

$$y^2 = x^3 + Ax + B.$$

- Suppose $p \neq 2, 3$. Any Weierstrass equation can be reduced to this form using a *linear change of variables*
- We have $\Delta = -16(4A^3 + 27B^2)$
- We have $j = \frac{1728(4A)^3}{\Delta}$

Weierstrass in characteristic 2 and 3

- If $p = 3$: reduced form either
  $y^2 = x^3 + a_2x^2 + a_6$, $\quad \Delta = -a_2^3a_6$, $j = -a_2^3/a_6$
  or
  $y^2 = x^3 + a_4x + a_6$, $\quad \Delta = -a_4^3$, $j = 0$
- If $p = 2$: reduced form either
  $y^2 + xy = x^3 + a_2x^2 + a_6$, $\quad \Delta = a_6$, $j = 1/a_6$
  or
  $y^2 + a_3y = x^3 + a_4x + a_6$, $\quad \Delta = a_3^4$, $j = 0$

Isomorphisms of curves

- What are the rational maps that preserve Weierstrass form and point at infinity, and are isomorphisms ?
  Only some linear transformations
  $$(X, Y) = (u^2x + r, u^3y + u^2sx + t)$$
  for any $u \neq 0$ and any r, s, t
- $u^{12}\Delta(E_2) = \Delta(E_1)$ and $j(E_2) = j(E_1)$

Isomorphisms and j-invariants

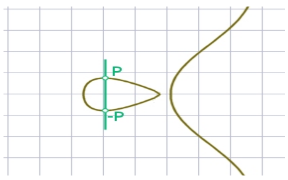
- j is invariant under isomorphisms
- The curve $y^2 = x^3 + 1$ has $j = 0$
  The curve $y^2 = x^3 + x$ has $j = 1728$
  The curve $y^2 = x^3 + \frac{3j}{1728-j}x + \frac{2j}{1728-j}$ has j-invariant j ($j \neq 0, 1728$)

Outline

"Inverse" of a point

$$y^2 = x^3 + Ax + B.$$

- Let $P := (x, y)$ be a point of a curve
- Define $-P$ as the symmetric of P by the x-axis, that is $-P := (x, -y)$



More generally

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

- Let $P = (x, y) \in E(K)$.
- Define $-P$ as the other point on the curve with the same x-coordinate

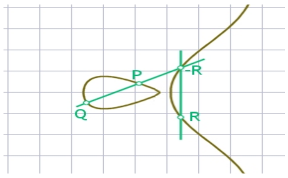
$$-P := (x, -y - a_1x - a_3)$$

- Define $P + (-P) = O$ the point at infinity
- Define $P + O = P = O + P$

Adding two distinct points

$$y^2 = x^3 + Ax + B.$$

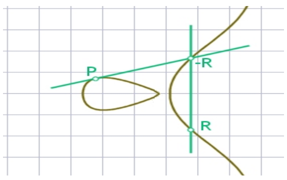
- Let $P := (x_1, y_1)$ and $Q := (x_2, y_2)$ where $x_1 \neq x_2$
- Draw the line through P and Q
- Call $-R$ the third intersection of this line with the curve
- Define $P + Q$ as the symmetric of $-R$ by the x-axis



Doubling a point

$$y^2 = x^3 + Ax + B.$$

- Let $P := (x, y)$
- Draw the tangent line through P
- Call $-R$ the second intersection of this line with the curve
- Define $P + P$ as the symmetric of $-R$ by the x-axis



Secant and tangent rules

- Similar definitions for more general equations :
  - Draw tangent or secant
  - Intersect with the curve to get $-(P + Q)$
  - Take second point on the curve with same x coordinate
- Any non vertical line intersects the curve at exactly three points (counted with multiplicities)

  A tangent point is counted twice

  The *point at infinity* O intersects every vertical line

Adding two distinct points

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

- Suppose $p \neq 0, 2, 3$.
- Let $P_1 = (x_1, y_1), \quad P_2 = (x_2, y_2) \in E(K)$
- Let $\lambda := \frac{y_2 - y_1}{x_2 - x_1}$
- Let $\nu := \frac{y_1x_2 - y_2x_1}{x_2 - x_1}$
- Let $P_1 + P_2 := (x_3, y_3)$ where $x_3 := \lambda^2 + a_1\lambda - a_2 - x_1 - x_2$ and $y_3 := -(\lambda + a_1)x_3 - \nu - a_3$
- $y = \lambda x + \nu$ is the line through P_1 and P_2

Doubling a point

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

- Suppose $p \neq 0, 2, 3$.
- Let $P = (x, y) \in E(K)$
- Let $\lambda := \frac{3x^2 + 2a_2x + a_4 - a_1y}{2y + a_1x + a_3}$
- Let $\nu := \frac{-x^3 + a_4x + 2a_6 - a_3y}{2y + a_1x + a_3}$
- Let $P + P := (x_3, y_3)$ where $x_3 := \lambda^2 + a_1\lambda - a_2 - 2x$ and $y_3 := -(\lambda + a_1)x_3 - \nu - a_3$
- $y = \lambda x + \nu$ is the tangent line at P

A group law ?

- The sum of two points of the curve is a point of the curve (including the point at infinity)
- The point at infinity is the neutral element
- Every element has a unique inverse
- Associativity ? $(P + Q) + R = P + (Q + R)$
  Consider 6 homogeneous lines defined by the above operations, and the 8 points $O, P, Q, R, \pm(P + R), \pm(P + Q)$, plus 2 points S, T that we must show equal. Suppose they are distinct. Multiplying the line equations you get two homogeneous cubics. The space of homogeneous cubics vanishing at 8 given points has dimension 2, so the curve equation must be linear combination of the two cubics. Evaluation at S, T shows the curve equation is identically 0.

Scalar multiplication

$$y^2 = x^3 + Ax + B.$$

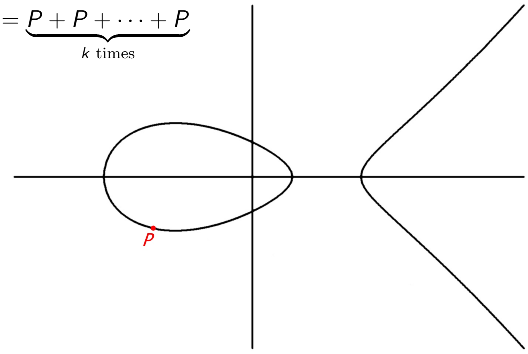
- For $k \in \mathbb{Z}$, define

$$[k](P) := \underbrace{P + P + \ldots + P}_{k \text{ times}}$$

- If K finite, then for any $P \in E(K)$, there is $m \in \mathbb{Z}$ such that $[m](P) = O \qquad$ (m is called *the order* of P)

Scalar multiplication

$$kP = \underbrace{P + P + \cdots + P}_{k \text{ times}}$$



Slide credit : Philippe Bulens

Scalar multiplication

- In reduced Weierstrass form there exist polynomial maps u_k, v_k, s_k, t_k such that

$$[k](x, y) = \left(\frac{u_k(x)}{v_k(x)}, y\frac{s_k(x)}{t_k(x)}\right)$$

- Proof :
  - Start with arbitrary rational maps in x, y
  - Use curve equation to replace any non linear term in y
  - Complete the squares in the denominators
  - Use $-[k]P = [k](-P)$ to deduce $[k](P) = \left(\frac{u(x)}{v(x)}, y\frac{s(x)}{t(x)}\right)$
- Replace in equation $y^2 = f(x)$ to deduce $v^3|t^2$ and $t^2|fv^3$

Torsion points

- Let $N \in \mathbb{Z}^*$. The N-torsion $E_K[N]$ over K is

$$E_K[N] := \{P \in E(K) \mid [N](P) = O\}$$

- $E_K[N]$ is a subgroup of $E(K)$
- Example for $y^2 = x^3 + Ax + B$, the 2-torsion is

$$E_K[2] = \{(x, 0) \mid x^3 + Ax + B = 0\}$$

Division polynomials

- Division polynomials $\Psi_N(x, y)$: polynomial with degree at most 1 in y and minimal degree in x such that

$$\psi_N(x, y) = 0 \;\Leftrightarrow\; \exists(x, y) \in E_K[N] \setminus \{O\}$$

- Recursive formulae

$$\begin{gathered}
\psi_0 = 0,\; \psi_1 = 1,\; \psi_2 = 2y,\\
\psi_3 = 3x^4 + 6ax^2 + 12bx - a^2,\\
\psi_4 = 4y(x^6 + 5ax^4 + 20bx^3 - 5a^2x^2 - 4abx - 8b^2 - a^3),\\
\psi_{2m+1} = \psi_{m+2}\psi_m^3 - \psi_{m-1}\psi_{m+1}^3,\\
\psi_{2m} = (\psi_{m+2}\psi_{m-1}^2 - \psi_{m-2}\psi_{m+1}^2)\psi_m/2y
\end{gathered}$$

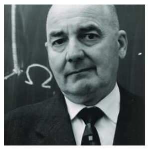
Division polynomials (2)

- Can prove
$$[k](x,y) = \left(\frac{\phi_k(x)}{\psi_k^2(x)}, \frac{\omega_k(x,y)}{\psi_k^3(x,y)}\right) = \left(x - \frac{\psi_{k-1}\psi_{k+1}}{\psi_k^2(x)}, \frac{\psi_{2k}(x,y)}{2\psi_k^4(x)}\right)$$
where $\phi_k = x\psi_k^2 - \psi_{k+1}\psi_{k-1}$, $\omega_k = \frac{\psi_{k+2}\psi_{k-1}^2 - \psi_{k-2}\psi_{k+1}^2}{4y}$
- Over algebraic closure of K we have
$$E_{\bar{K}}[N] \approx (\mathbb{Z}/N\mathbb{Z}) \times (\mathbb{Z}/N\mathbb{Z})$$

Number of rational points

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

- Pick a random $x \in \mathbb{F}_{p^n}$.
  2 solutions for y with probability $\approx 1/2$
  0 solution for y with probability $\approx 1/2$
- In fact : Hasse's theorem



$$|\#E(\mathbb{F}_{p^n}) - (p^n+1)| \leq 2\sqrt{p^n}$$

proof : see Silverman, Chapter V.

Number of rational points

- Weil-Deligne theorem (Riemann hypothesis for EC) implies
$$\#E(\mathbb{F}_{p^n}) = 1 - \pi^n - \bar{\pi}^n + p^n$$
where $\pi^2 - \pi t + p = 0$ and $t \in \mathbb{Z}$
($\bar{\pi}$ and π complex conjugates) (proof : see Silverman, Chapter V)
- Value $t_n := \pi^n + \bar{\pi}^n \leq 2\sqrt{p^n}$ called *trace* of $E(\mathbb{F}_{p^n})$

Group structure over finite fields

- Finite, Abelian group of rank at most 2
- We have
$$E(\mathbb{F}_q) \approx \mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$$
where $n_1|n_2$ and $n_1|q-1$

Outline

Isogeny

- Group homomorphism defined by a rational map
$$\Psi : E_1 \to E_2$$
- Example : scalar multiplications when $E_1 = E_2$
- In reduced Weierstrass form there exist polynomial maps u_k, v_k, s_k, t_k such that
$$\Psi(x, y) = \left(\frac{u(x)}{v(x)}, y\frac{s(x)}{t(x)}\right)$$
and moreover $v^3|t^2$ and $t^2|fv^3$
- Proof identical as for scalar multiplications
- Isogeny is *separable* if $(u/v)' \neq 0$

Isogeny kernel

- Kernel $\ker \Psi = \{P \in E_1 \mid \Psi(P) = O\}$
- Degree $\Psi = \max(\deg u, \deg v)$
- For example $\ker[k] = E[k]$ and $\deg[k] = k^2$
- Degree $\deg \Psi = \# \ker \Psi$ when Ψ separable
- $\ker \Psi$ is a subgroup of order $\deg \Psi$ in $E(\bar{K})$
- For any curve E there are $\ell + 1$ isogenies of degree ℓ (defined over $\bar{K}$) from this curve, and each one defines a quotient curve E/Ψ

Dual isogeny

- For any isogeny $\Psi : E_1 \to E_2$, there exists a *dual isogeny* $\hat{\Psi} : E_2 \to E_1$ such that
$$\hat{\Psi} \circ \Psi = [\deg \Psi]$$
- $\deg \hat{\Psi} = \deg \Psi$
- $\Psi(E[\deg \Psi]) = \ker \hat{\Psi}$
- $\Psi \circ \hat{\Psi} = [\deg \Psi]$ on E_2

Endomorphism

- An endomorphism of E is an isogeny $E \to E$
- Example : scalar multiplications

Frobenius endomorphism

- Let $A, B \in K := \mathbb{F}_q$, for $q := p^m$
  Let $E : y^2 = x^3 + Ax + B$
- The map
  $$[\pi] : (x, y) \to (x^q, y^q)$$
  is an endomorphism
- Called the *Frobenius endomorphism*
- Commutes with scalars : $[k]([\pi](P)) = [\pi]([k](P))$
- Unseparable isogeny

The endomorphism ring

- If ϕ_1 and ϕ_2 are endomorphisms, then
  $\phi_1 + \phi_2$ is an endomorphim ($+$ is addition on E)
  $\phi_1 \circ \phi_2$ is an endomorphism ($\circ$ is composition)
- Therefore $\forall a, b \in \mathbb{Z}$, the map
  $$[a + b\pi] : (x, y) \to [a](x, y) + [b]([\pi](x, y))$$
  is an endomorphim
- Addition and composition define a ring structure

Trace of the Frobenius

- Frobenius endomorphism for a curve defined over $\mathbb{F}_q$
  $$\begin{array}{rcl} [\pi] : E(\mathbb{F}_{q^n}) & \to & E(\mathbb{F}_{q^n}) \\ (x, y) & \to & (x^q, y^q) \end{array}$$
- Satisfies a quadratic equation
  $$[\pi^2 - t\pi + q] = [0]$$
  meaning $\forall (x, y) \in E(K)$, $(x^{q^2}, y^{q^2}) - [t](x^q, y^q) + [q](x, y) = O$
- Note $E(\mathbb{F}_q) = \ker(1 - \pi)$
- Remember $t = q + 1 - \#E(\mathbb{F}_q)$ with $|t| \leq 2\sqrt{q}$

Structure of the endomorphism ring

- Scalar multiplications $\{[k] : k \in \mathbb{Z}\} = \mathbb{Z}$
- The endomorphism ring of an elliptic curve is either $\mathbb{Z}$, an order in a quadratic imaginary field, or an order in a quaternion algebra [Silverman III.9.4]
- Over finite fields it is always bigger than $\mathbb{Z}$
- Ordinary curves : order in a quadratic imaginary field $End(E) \subset \mathbb{Z} \times \pi\mathbb{Z}$ with $\pi^2 - t\pi + p = 0$
- Supersingular curves : order in a quaternion algebra $\exists \pi, \phi : E \to E$ s.t. $\pi\phi \neq \phi\pi$ and $End(E) \subset \mathbb{Z} \times \pi\mathbb{Z} \times \phi\mathbb{Z} \times \pi\phi\mathbb{Z}$

Outline

Outline

Elliptic curve discrete logarithm problem (ECDLP)

- Let K be a finite field, let E be a curve over K, let $P \in E(K)$ and let $Q \in \langle P \rangle$. Find k such that $Q = [k]P$.
- Fields used in cryptography are prime fields (mostly) and binary fields with a prime degree extension (for efficiency, particularly in hardware devices)
- Typically the order of P is a large prime, at least 160 bits

Elliptic Curve Diffie-Helman problem (ECDHP)

- Given P and $[a]P$ and $[b]P$, compute $[ab]P$
- Also believed to be very hard
- Can solve ECDHP if can solve ECDLP, but other way around not known in general

Elliptic curve Diffie-Helman

- Goal : key agreement
  Two parties want to build a common secret key
- Alice chooses random r_a. She sends $Q_a := [r_a](P)$ to Bob
- Bob chooses random r_b. He sends $Q_b := [r_b](P)$ to Alice
- Alice computes $K_a := [r_a](Q_b)$
- Bob computes $K_b := [r_b](Q_a)$
- We have $K_a = [r_a r_b](P) = K_b$

Elliptic curve El Gamal

- Goal : public key encryption
- Key generation : choose K, E and $P \in E(K)$.
  Choose secret key x. Reveal public key $E, P, Q = [x](P)$
  If ECDLP hard, x cannot be recovered from Q.
- Encryption : to encrypt $M \in E(K)$, choose random r.
  Compute $C_1 = [r](P)$ and $C_2 = M + [r](Q)$
  Both C_1 and C_2 are random points on the curve.
- Decryption : compute
  $C_2 - [x](C_1) = M + [r]([x](P)) - [x]([r](P)) = M$

Elliptic curve El Gamal (2)

- Elliptic curve decisional Diffie-Hellman :
  Given P and $[a]P$ and $[b]P$ and $[c]P$,
  decide whether c is ab or random
- Theorem : elliptic curve El Gamal is IND-CPA secure under ECDDH assumption
  (proof is the same as over finite fields)

ECDSA

- Public key signature standard
- Parameters defined by
  - H a hash function
  - K a finite field
  - q a prime
  - E an elliptic curve over K with qh points, $h \leq 4$
  - P a point of order q on E
- Key generation
  - Choose secret key x randomly in $\{1, \ldots, q-1\}$
  - Set public key $Q = xP$

ECDSA : signature

- Let $f : E \to \mathbb{F}_q : P = (x, y) \to x \bmod q$ where x is some well-defined integer representation of the x-coordinate of P
- To sign a message m :
  1. Choose k randomly in $\{1, \ldots, q-1\}$
  2. Let $T = kP$
  3. Let $r = f(T)$. If $r = 0$ start again
  4. Let $e = H(m)$
  5. Let $s = (e + xr)/k \bmod q$. If $s = 0$ start again
  6. Return (r, s)

ECDSA : verification

- To verify signature (r, s) on a message m
  - Reject if $r, s \notin \{1, \ldots, q-1\}$
  - Let $e = H(m)$
  - Let $u_1 = e/s \bmod q$ and $u_2 = r/s \bmod q$
  - Let $T = u_1 P + u_2 Q$
  - Accept iff $r = f(T)$
- Correctness : $u_1 + xu_2 = (e + rx)/s = k \bmod q$ hence $u_1 P + u_2 Q = T$

ECDSA : security

- Existential unforgeability, if we replace the hash function by a random function and the group by a generic group, and suppose f is almost invertible
- Essential that k does not repeat as otherwise two signatures (r, s) and (r', s') give

$$x = \frac{se' - s'e}{r(s' - s)} \bmod q$$

  (used to recover the secret key of Sony PS3)
- More attacks if some bits of k leak or repeat
  (see later for lattice-based attacks)

Outline

ECDLP cryptanalysis : state-of-the-art

- Generic attacks : Pollard's rho, Pohligh-Hellman,...
- Anomalous attack if $|E(\mathbb{F}_p)| = p$ (see later in these slides)
- MOV attack if efficient pairings (see later in these slides)
- Weil descent reduction to hyperelliptic curve discrete logarithms (not covered in this course)
- Index calculus attacks being developed, perhaps subexponential but worse than generic ones in practice (see bonus slides ; ask me for a dissertation project)
- Best attacks are generic ones for well-chosen parameters 160-bit ECDLP $\sim$ 1024-bit RSA

NIST curves

- 15 curves and base points on these curves, to be used by US federal government
  - 5 curves over prime fields, with pseudo-Mersenne primes for efficiency
  - 5 curves with parameters defined over $\mathbb{F}_2$
  - 5 additional curves in characteristic 2
- Curve parameters are derived from a given seed, using SHA-1, until the curve has prime order
- See `http://csrc.nist.gov/groups/ST/toolkit/documents/dss/NISTReCur.pdf`

NIST curves (2)

- De facto world standard
- Ongoing revision, likely to drop most curves including all binary curves, and to include a few other popular curves
- Seed claimed to derive from some passphrase
- Suspicion seed chosen to allow trapdoor attacks

Other commonly used curves

- See `http://safecurves.cr.yp.to/` for a classification and analysis of curves recommended by ANSI, IEEE, BSI,... and the one used in Bitcoin

Outline

Anomalous attack

- Let E be an elliptic curve over $\mathbb{F}_p$, with p prime
- Assume $\#E(\mathbb{F}_p) = p$
- In that case, we can lift the discrete logarithm problem over the p-adic numbers, where it turns out to be easy
- Remark : condition $\#E(\mathbb{F}_p) = p$ can be checked easily, and very unlikely to hold for random curves

p-adic numbers (informally)

- Let p be a prime
- Let $r \in \mathbb{Q}$
  - We can write $r = \frac{a}{b}p^i$ with a, b, p coprime
  - p-valuation of r defined by $|r|_p = p^{-i}$
  - $|.|_p$ defines a norm hence a distance $d(r_1, r_2) = |r_1 - r_2|_p$
- Considering all limits of Cauchy sequences of rationals under $|.|_p$ norm, we get the p-adic numbers $\mathbb{Q}_p$ (the same way we get $\mathbb{R}$ using usual $|.|$ norm)
- Any p-adic number can be uniquely written as $\sum_{i=k}^{\infty} a_i p^i$
- Practical computations up to some precision (modulo p^ℓ)

Lifting from $\mathbb{F}_p$ to $\mathbb{Q}_p$

- Let $y^2 = x^3 + Ax + B$ defining an elliptic curve E over $\mathbb{F}_p$
- See E as an elliptic curve $\bar{E}$ defined over $\mathbb{Q}_p$
- Lift P and Q to points $\bar{P}$ and $\bar{Q}$ over $\bar{E}$
  To lift a point (x, y) keep x and solve $y^2 = x^3 + Ax + B$ up to some precision over $\mathbb{Q}_p$ using Hensel's lemma (solve the equation modulo p^2, then p^3, etc)
- Let $E_1(\mathbb{Q}_p) = \{P \in E(\mathbb{Q}_p) \mid P \bmod p = O\}$ subgroup of $E(\mathbb{Q}_p)$
- We have $E(\mathbb{Q}_p)/E_1(\mathbb{Q}_p) = E(\mathbb{F}_p)$

A group homomorphism

- Let $y^2 = x^3 + Ax + B$ defining an elliptic curve over $\mathbb{Q}_q$
- Let $\omega(z)$ be the power series solution to the equation

$$\omega = z^3 + Az\omega^2 + B\omega^3$$

- There is a **group homomorphism** $\theta_q : E_1(\mathbb{Q}_q) \to \hat{G}_q$ defined by

$$\theta_q(P) = \left(\int \omega(z)dz \right)(z_q) = z_q + \frac{d_1}{2}z_q^2 + \frac{d_2}{3}z_q^3 + \dots$$

where d_i are polynomials in A, B and $z_q = -x(P)/y(P)$

Anomalous attack

- Assumption : $p = \#E(\mathbb{F}_p)$
- Lift P and Q to points $\bar{P}$ and $\bar{Q}$ over $\bar{E}$
- Compute $\theta_p(p\bar{P})$ and $\theta_p(p\bar{Q})$ up to $O(p^2)$ terms
- The equation $Q - kP = 0$ implies

$$\theta_p(p\bar{Q}) - k\theta_p(p\bar{P}) = 0 \bmod p^2$$

- Compute

$$k = \frac{\theta_p(p\bar{Q})}{\theta_p(p\bar{P})} \bmod p$$

Outline

Outline

Elliptic Curves

Elliptic Curve Discrete Logarithm Problem

Algorithmic Aspects
- Faster arithmetic
- Alternative Models
- Generating good curves

Factorization and Primality testing

Pairings

Point addition and point doubling

- Let $E : y^2 = x^3 + a_4x + a_6$ and let $P_i = (x_i, y_i) \in E$
- If $(x_2, y_2) = (x_1, -y_1)$ then $P_1 + P_2 = O$
- Otherwise $P_1 + P_2 = (x_3, y_3)$ with

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= -\lambda x_3 - \nu \end{aligned}$$

where

$$\begin{cases} \lambda = \frac{y_2 - y_1}{x_2 - x_1} \text{ and } \nu = \frac{y_1x_2 - y_2x_1}{x_2 - x_1} & \text{if } P_1 \neq P_2 \\ \lambda = \frac{3x^2 + a_4}{2y} \text{ and } \nu = \frac{-x^3 + a_4x + 2a_6}{2y} & \text{if } P_1 = P_2 \end{cases}$$

- How to compute $[k]P$?

Double and add algorithm

- Scalar multiplication : given k and P, return $[k]P$

  1: Let $k = \sum_{i=0}^{n} k_i 2^i$
  2: $P' \leftarrow P$; $Q \leftarrow [k_0]P$
  3: **for** i=1 **to** n **do**
  4: $\quad P' \leftarrow [2]P$
  5: $\quad Q \leftarrow Q + [k_i]P'$
  6: **end for**
  7: **return** Q
- Let $n = \log k$. Scalar multiplication requires
  - n point doublings
  - $n/2$ point additions on average

Double and add algorithm ("left-to-right")

- Scalar multiplication : given k and P, return $[k]P$

  1: Let $k = 2^{n+1} + \sum_{i=0}^{n} k_i 2^i$
  2: $Q \leftarrow P$
  3: **for** i=n **to** 1 **do**
  4: $\quad Q \leftarrow [2]Q$
  5: $\quad Q \leftarrow Q + [k_{i-1}]P$
  6: **end for**
  7: **return** Q
- Same cost as the "right-to-left" version

Projective coordinates

- Goal : avoid divisions in addition/doubling formulae (more expensive than additions and multiplications)
- A point $P = (x, y)$ is now represented as $P = (X, Y, Z)$ where $x = X/Z$ and $y = Y/Z$
- $P_2 = \pm P_1$ tested as $x_1 z_2 = x_2 z_1$ and $y_1 z_2 = \pm y_2 z_1$
- Doubling and addition re-written without division
- Double-and-add algorithm performed without division
- If needed, a single division $1/Z$ is performed at the end to recover affine coordinates

Window methods

- Goal : reduce the number of additions with precomputation
- Simplest variant
  - Precompute $P_i = [i]P$ for $i = 0, \ldots, 2^k - 1$
  - In left-to-right version, perform at least k doublings before each addition
- Cost is now n doublings, $\frac{2^k-1}{2^k} \cdot \frac{n}{k}$ additions, plus a precomputation of $2^k - 2$ additions

Non adjacent form representations

- Goal : reduce the number of additions
- Observe addition and substraction have the same cost
- Use signed representations with $k_i \in \{-1, 0, 1\}$ to ensure that 1 or -1 is always followed by 0
- Algorithm to compute signed representations :

```
while  k > 0  do
    if k odd then k_i = 2 − (k mod 4)
    if k even then k_i = 0
    k ← (k − k_i)/2
    i ← i + 1
end while
```


Cost of NAF representation

- Cost is n doublings and αn additions/substractions
- What is α ?
  - Consider a state automaton with two states 0 and ± 1
  - State goes from ± 1 to 0 with probability 1
  - State goes from 0 to ± 1 with probability $1/2$
  - State goes from 0 to 0 with probability $1/2$
  - At equilibrium we have $\alpha = (1 - \alpha)\frac{1}{2}$ hence $\alpha = \frac{1}{3}$
- Cost is n doublings and $n/3$ additions or substractions
- Better addition/substraction chains may exist, but finding optimal one is NP-hard
- Can be combined with window methods

Multi-exponentiations

- Goal : compute $\sum [k_i]P_i$ faster than by serial computation
- Idea : do the doublings only once
- Algorithm

  Let $k_j = \sum_{i=0}^{n} k_{j,i}2^i$
  $Q \leftarrow \sum_j P_{j,n}$
  for i=n **to** 1 **do**
      $Q \leftarrow [2]Q$
      $Q \leftarrow Q + \sum_j [k_{j,i-1}]P_j$
  end for
  return Q

- Can be combined with other tricks

Frobenius expansions

- Goal : for elliptic curves defined over $\mathbb{F}_2$, replace double-and-add by Frobenius-and-add
- Note that Frobenius map $(x, y) \to (x^2, y^2)$ is a very efficient operation, especially if elements of $\mathbb{F}_{2^n}$ are represented using a normal basis $\{\alpha^{2^i}, i = 0, \ldots, n-1\}$
- There is efficient algorithm writing $k = \sum_i k_i \varphi^i$
  (use $\varphi^2 - t\varphi + 2 = 0$) (see BSS IV.3 and references therein)

Gallant-Lambert Vanstone

- Goal : exploit efficient endomorphism when we have one
- Example : if $\iota^2 = -1$ then $\phi : (x, y) \to (-x, \iota y)$ is an endomorphism of $E : y^2 = x^3 + ax$, with $\phi^2 = [-1]$
- Let $f(x) = x^2 + t_\phi x + n_\phi$ characteristic polynomial of ϕ
- Let N order of P
- Assume there is $\lambda \in \mathbb{Z}/N\mathbb{Z}$ such that $f(\lambda) = 0 \bmod N$
- Then $\phi = [\lambda]$ or $[t_\phi - \lambda]$ on $\langle P \rangle$ (assume first case)
- Idea : write $k = a\lambda + b \bmod N$ with a, b about $n/2$ bits then compute $a\phi(P) + bP$ using multi-exponentiation

Euclide algorithm

- Goal : given integers a and b, find $d = \gcd(a, b)$
- $d|a$, $d|b$ imply $d|(a + kb)$ for any integer k

  Require: $a \geq b$
  Ensure: $\gcd(a, b)$
  if $b|a$ **then**
      return b
  else
      Compute q such that $0 < a - qb < b$
      return $\gcd(b, a - qb)$
  end if

- Complexity $O(|a|^2)$; best algorithms achieve $O(|a| \log |a|)$

Extended Euclide algorithm

- Goal : compute r and s such that $ra + sb = \gcd(a, b)$

Require: $a \geq b$
Ensure: $d = \gcd(a, b)$ **and** r, s, such that $ar + bs = d$

if $b|a$ **then**
  return $a, 0, 1$
else
  Compute q such that $0 < a - qb < b$
  $d, r, s \leftarrow \gcd(b, a - qb)$
  return $d, s, r - qs$
end if

- Indeed if $rb + s(a - qb) = d$ then $sa + (r - qs)b = d$
- Complexity $O(|a|^2)$; best algorithms achieve $O(|a| \log |a|)$

Gallant-Lambert Vanstone

- Goal : exploit efficient endomorphism when we have one
- Idea : write $k = a\lambda + b \bmod N$ with a, b about $n/2$ bits then compute $a\phi(P) + bP$ using multi-exponentiation
- To find a, b
  - Run extended Euclide algorithm with inputs N, λ and stop it at the middle, when all coefficients are $\approx \sqrt{N}$
  - Deduce u_i, v_i of size about $\sqrt{N}$ s.t. $u_i\lambda + v_i = 0 \bmod N$
  - Compute $C_1, C_2 \in \mathbb{R}$ s.t. $(0, k) = C_1(u_1, v_1) + C_2(u_2, v_2)$
  - Round C_1, C_2 to nearest integers c_1, c_2
  - Return $(a, b) = (C_1 - c_1)(u_1, v_1) + (C_2 - c_2)(u_2, v_2)$
- Equivalently, apply Babai's nearest plane algorithm to a well-chosen lattice (see later)

Point compression

- Goal : reduce both memory and scalar multiplication costs by projecting points on x coordinates
- Observation :
  - x coordinate determines an elliptic curve point up to sign
  - $x([k]P)$ only depends on $x(P)$
- Some existing tricks :
  - Represent (x, y) by x and an additional bit for y
  - Only use x in signature schemes

Outline

Birrational equivalence

- Informal definition : two curves E and E' defined by the equations $F(x,y)=0$ and $F'(X,Y)=0$ are birrationally equivalent if there exist rational maps

$$\varphi : E \to E' \quad \text{and} \quad \phi : E' \to E$$

such that $\phi \circ \varphi$ is the identity map on E for all but a few exceptional points

(definition makes sense over characteristic 0 fields)

Why other models ?

- Any elliptic curve can be written in (short) Weierstrass form, but other forms may be better in practice
- Faster arithmetic, in particular decreasing the number of field multiplications for one scalar multiplication
- Complete addition formula ("no special case")
  - Preventing implementation bugs
  - Side-channel resistance
- Special curves with "useful" properties (but beware cryptanalysis)

Elliptic curve formula database

Genus-1 curves over large-characteristic fields

Doubling-oriented Doche–Icart–Kohel curves: $y^2=x^3+a*x^2+16*a*x$
Tripling-oriented Doche–Icart–Kohel curves: $y^2=x^3+3*a*(x+1)^2$
Edwards curves: $x^2+y^2=c^2*(1+d*x^2*y^2)$
Hessian curves: $x^3+y^3+1=3*d*x*y$
Jacobi intersections: $s^2+c^2=1$, $a*s^2+d^2=1$
Jacobi quartics: $y^2=x^4+2*a*x^2+1$
Montgomery curves: $b*y^2=x^3+a*x^2+x$
Short Weierstrass curves: $y^2=x^3+a*x+b$
Twisted Edwards curves: $a*x^2+y^2=1+d*x^2*y^2$
Twisted Hessian curves: $a*x^3+y^3+1=d*x*y$

Ordinary genus-1 curves over binary fields

Binary Edwards curves: $d1*(x+y)+d2*(x^2+y^2)=(x+x^2)*(y+y^2)$
Hessian curves: $x^3+y^3+1=3*d*x*y$
Short Weierstrass curves: $y^2+x*y=x^3+a2*x^2+a6$

Source : www.hyperelliptic.org/EFD/

Edwards curves

- Defined by an equation (assume characteristic not 2)

$$x^2+y^2=c^2(1+dx^2y^2)$$

- Addition of two points (x_1,y_1) and (x_2,y_2) defined by

$$\left(\frac{x_1y_2+x_2y_1}{c(1+dx_1x_2y_1y_2)}, \frac{y_1y_2-x_1x_2}{c(1-dx_1x_2y_1y_2)}\right)$$

- Neutral element is $(0,c)$ and $-(x,y)=(-x,y)$
- $(0,-c)$ has order 2 and $(-c,0)$ has order 4

Edwards curves (2)

- Only elliptic curves with order divisible by 4 can be written in this form
- Complete addition law : no special case
- Efficient arithmetic
- See Bernstein-Lange, *Faster addition and doubling on elliptic curves*

Outline

Generating good curves

- In cryptography we need curves defined over $\mathbb{F}_p$ with a large subgroup of prime order q
- Two methods
  - Random generation : generate random curves and compute their orders
  - Complex multiplication method : choose suitable p and q then compute a corresponding curve
- The second method is faster but it produces special curves

Random Curves

- Idea :
  - Generate random $A, B \in K$ with $4A^3 + 27B^2 \neq 0$
  - Compute the order of $E : y^2 = x^3 + Ax + B$
  - Repeat until the order is good
- We need an efficient algorithm to compute the order

Remember : CRT

- Chinese Remainder Theorem
- Let $p_i = 2, 3, 5, ...$ be prime numbers, let e_i be integers let $t_i \in \{0, 1, \ldots, p_i^{e_i} - 1\}$
- The congruence system $t = t_i \bmod p_i^{e_i}, \forall i$ has a unique solution modulo $\prod p_i^{e_i}$
- This solution can be computed efficiently
- Example : $t = 3 \bmod 2^2, t = 2 \bmod 5 \Rightarrow t = 7 \bmod 20$

Point counting

- Goal : given E defined over $\mathbb{F}_q$, return $\#E(\mathbb{F}_{q^n})$
- By Hasse's theorem, we have lower and upper bounds
- By Weil-Deligne's theorem, sufficient to compute $\#E(\mathbb{F}_q)$
- We know that $[\pi^2 - t\pi + q] = [0]$
- On the m-torsion, we have $[\pi^2 - t\pi + q] = [\pi^2 - (t \bmod m)\pi + (q \bmod m)]$
- If $[m](P) = O$ and $[\pi^2 - t_m\pi + q](P) = O$ then $t = t_m \bmod m$

Schoof's algorithm

1. Use Hasse's theorem to bound $\#E(\mathbb{F}_q)$
2. Find primes p_i s.t. $\prod p_i \geq 2\sqrt{q}$
3. For each p_i
   - 3.1 Find $P_i \in E[p_i]$, $P_i \neq O$
   - 3.2 For $t_i = 0, \ldots, p_i - 1$, compute $[\pi^2 - t_i\pi + q](P_i)$ until we get O
   - 3.3 Deduce $t = t_i \bmod p_i$
4. Use CRT to recover t. Deduce $\#E(\mathbb{F}_p)$.

Further on point counting

- Can also use powers of primes
- Improvements by Elkies, Atkin,...
- p-adic methods

Complex multiplication method

- Two steps :
  - Choose suitable p and N
  - Compute the curve using complex multiplication theory
- Faster than random curves/ point counting
  but it produces special curves

Complex multiplication

- Remember

$$\pi^2 - t\pi + p = 0 \quad \text{and} \quad N = \#E(\mathbb{F}_p) = p + 1 - t$$

- By Hasse's theorem,

$$\Delta = t^2 - 4p \leq 0$$

Careful : this Δ is not the discriminant defined before

- $End(E)$ is an order in an imaginary quadratic field
  (we say E has complex multiplication by this order)

Deuring's lifting theorem

- For any $E/\mathbb{F}_p$, there exists $\tilde{E}/K$ such that $\tilde{E}$ mod $p = E$
  (where K is some number field such that p is split in K)
  and moreover any $\phi \in End(E)$ arises as $\tilde{\phi}$ mod p
  where $\tilde{\phi} \in End(\tilde{E})$
- Efficient to compute elliptic curves over $\mathbb{C}$
  with complex multiplication by a given order,
  as long as the discriminant is small enough

Computing CM j-invariants over $\mathbb{C}$

- Let p and N fixed, hence fixing Δ as well
- j-invariants with complex multiplication by a given order
  with discriminant Δ satisfy some symmetry property
- This symmetry leads to an equation in j that can be
  solved numerically with arbitrary precision
- Non trivial fact : all such j are roots of a polynomial H_D
  with integer coefficients
  - H_Δ is called the Hilbert class polynomial
  - $\deg H_\Delta =$ the class number of K

Reduction modulo p

- H_Δ can be computed exactly using finite precision for j since its coefficients are integer
- The j-invariants over $\mathbb{F}_p$ with complex multiplication by an order of discriminant Δ are the roots of H_Δ mod p
- Coefficients of H_Δ are huge, so only practical for small Δ
- In fact if $\Delta = Df^2$ then H_Δ mod $p = H_D$ mod p so it is enough to compute H_D

Curve equation from j-invariant

- $E : y^2 = x^3 + 1$ has $j = 0$
- $E : y^2 = x^3 + x$ has $j = 1728$
- $E : y^2 = x^3 + ax - a$ with $a = \frac{27j}{4(1728-j)}$ has j-invariant $j \neq 0, 1728$
- $E : y^2 = x^3 + Ax + B$ and $E^d : y^2 = x^3 + Ad^2x + Bd^3$ have the same j invariant
- If d is a quadratic non residue then E^d is called the **quadratic twist** of E and $t(E) = -t(E^d)$
- If $\#E = p + 1 + t$ then $\#E^d = p + 1 - t$

CM method

- Choose a small D and a suitable p (see next slide)
- Compute the *Hilbert polynomial* H_D (computation complexity is quasi-linear in D)
- Compute j as a root of H_D modulo p
- Compute $E(j)$
- Return $E(j)$ if $\#E(j) = N$
- Otherwise return its quadratic twist

Finding a suitable p

- There exists an elliptic curve E with discriminant $D < 0$ reducing modulo p to a curve of order N if and only if

  $$Df^2 = t^2 - 4p = (p + 1 - N)^2 - 4p = (N + 1 - p)^2 - 4N$$

  for some integer f
- Simple algorithm
  1. Choose a random prime p
  2. Use Cornacchia's algorithm to solve $p = u^2 - Dv^2$
  3. If there is a solution deduce $N_\pm = p + 1 \pm 2u$
  4. Repeat until either N_+ or N_- is "good"

Best rational approximations

- Euclide algorithm can also be used to compute **best rational approximations** : given $\chi \in \mathbb{R}$ (not necessarily rational) find p, q such that p/q is closer to χ than any fraction with a smaller or equal denominator
- Roughly : run Euclide algorithm with inputs χ, 1

  $$\chi = s_0 \cdot 1 + r_1, \quad 1 = s_1 \cdot r_1 + r_2, \quad r_1 = s_2 r_2 + r_3, \ldots$$

  simultaneously compute p_n, q_n such that $r_n = -p_n + q_n\chi$
- Can prove that $\left|\chi - \frac{p_n}{q_n}\right| = \left|\frac{r_n}{q_n}\right| \leq \frac{1}{q_n q_{n+1}}$

Cornacchia's algorithm

Require: Squarefree $d > 0$ and prime p
Ensure: A solution to $p = u^2 + dv^2$, if one exists

1: Compute r such that $r^2 = -d \bmod p$
2: Compute continued fraction approximations a_n/b_n to r/p, until $b_n < \sqrt{p} < b_{n+1}$
3: Set $v = b_n$ and compute $u = \sqrt{\frac{p-v^2}{d}}$
4: If u is an integer, return u, v

- Remark : when there is a solution we have $u = rb_n - a_n p$
- Proof when $d = 1$: show that $u^2 + v^2 = 0 \bmod p$, and $u^2 + v^2 < 2p$ using continued fraction properties

Generating good curves

- In cryptography we need curves defined over $\mathbb{F}_p$ with a large subgroup of prime order q
- Two methods
  - Random generation : generate random curves and compute their orders
  - Complex multiplication method : choose suitable p and q then compute a corresponding curve
- The second method is faster but it produces special curves with small discriminant

Outline

Outline

Pollard's $p-1$ factorization method

- Goal : factor $n = pq$ assuming $p-1$ is B-powersmooth (recall $x = \prod p_i^{e_i}$ is B-powersmooth if $p_i^{e_i} < B$)
- Let s be the product of all $p_i^{e_i} < B$
- By assumption $(p-1)|s$, hence $g^s = 1 \bmod p$
- We deduce $\gcd(g^s - 1, n) = p$
- Only works if some factor p such that $p-1$ smooth !

Elliptic curve factorization method

- Idea : generalize previous method when neither $p-1$ nor $q-1$ are smooth
- The group order $\#E(\mathbb{F}_p)$ of an elliptic curve can be smooth even when $p-1$ is not !

Elliptic curve addition law

- Let $E : y^2 = x^3 + Ax + B$
- Let $P_1 = (x_1, y_1)$, $P_2 = (x_2, y_2)$ two points on the curve
- The chord-and-tangent rules lead to addition formulae : for example we have $P_1 + P_2 = (x_3, y_3)$ where
  $\lambda = \frac{y_2 - y_1}{x_2 - x_1}, \quad \nu = \frac{y_1 x_2 - y_2 x_1}{x_2 - x_1},$
  $x_3 = \lambda^2 - x_1 - x_2, \qquad y_3 = -\lambda x_3 - \nu$
- These formulae involve divisions
- Over $\mathbb{F}_p$, a division by 0 means P_3 is point at infinity
- Over $\mathbb{Z}_n$, a division fails if $(x_2 - x_1)$ is not invertible
- A failure reveals a factor of n !

Elliptic curve factorization method

1. Choose E and $P = (x, y) \in E(\mathbb{Z}_n)$
2. Let B be a smoothness bound on $\#E(\mathbb{Z}_p)$ for $p|n$
3. Compute $s = \prod p_i^{e_i}$ where $p_i^{e_i} \leq B$
4. We have $[s]P = O$ in $E(\mathbb{Z}_p)$
5. Try to compute $[s](P)$ in $E(\mathbb{Z}_n)$: division by p occurs and produces an error
6. When a division by some d fails, compute

$$\gcd(d, n) \neq 1$$

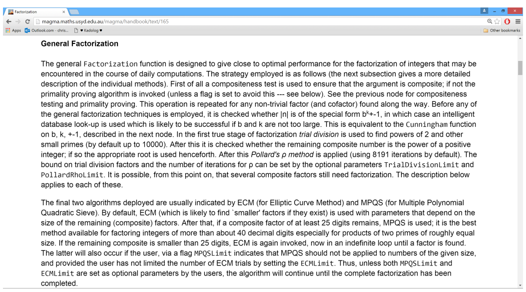
Elliptic curve factorization method

- For a random curve, we expect $\#E(\mathbb{F}_p)$ to be roughly uniformly distributed in

$$\#E(\mathbb{F}_p) \in [(p+1) - 2\sqrt{p}, (p+1) + 2\sqrt{p}]$$

- Let $B \approx L_p(1/2)$
- Probability to be B-smooth is about $(L_p(1/2))^{-1} = \exp(-c(\log p)^{1/2}(\log \log p)^{1/2})$
- Repeat with random curves until you get a factor
- Remark : runtime depends on the smallest factor
- In practice, the method is used as subroutine to factor middle-size integers when $\log_2 n \approx 60 - 80$ bits

Factorization in practice : Magma



Factorization

magma.maths.usyd.edu.au/magma/handbook/text/165

General Factorization

The general `Factorization` function is designed to give close to optimal performance for the factorization of integers that may be encountered in the course of daily computations. The strategy employed is as follows (the next subsection gives a more detailed description of the individual methods). First of all a compositeness test is used to ensure that the argument is composite; if not the primality proving algorithm is invoked (unless a flag is set to avoid this --- see below). See the previous node for compositeness testing and primality proving. This operation is repeated for any non-trivial factor (and cofactor) found along the way. Before any of the general factorization techniques is employed, it is checked whether |n| is of the special form b^k+-1, in which case an intelligent database look-up is used which is likely to be successful if b and k are not too large. This is equivalent to the `Cunningham` function on b, k, +-1, described in the next node. In the first true stage of factorization *trial division* is used to find powers of 2 and other small primes (by default up to 10000). After this it is checked whether the remaining composite number is the power of a positive integer; if so the appropriate root is used henceforth. After this *Pollard's ρ method* is applied (using 8191 iterations by default). The bound on trial division factors and the number of iterations for ρ can be set by the optional parameters `TrialDivisionLimit` and `PollardRhoLimit`. It is possible, from this point on, that several composite factors still need factorization. The description below applies to each of these.

The final two algorithms deployed are usually indicated by ECM (for Elliptic Curve Method) and MPQS (for Multiple Polynomial Quadratic Sieve). By default, ECM (which is likely to find `smaller' factors if they exist) is used with parameters that depend on the size of the remaining (composite) factors. After that, if a composite factor of at least 25 digits remains, MPQS is used; it is the best method available for factoring integers of more than about 40 decimal digits especially for products of two primes of roughly equal size. If the remaining composite is smaller than 25 digits, ECM is again invoked, now in an indefinite loop until a factor is found. The latter will also occur if the user, via a flag `MPQSLimit` indicates that MPQS should not be applied to numbers of the given size, and provided the user has not limited the number of ECM trials by setting the `ECMLimit`. Thus, unless both `MPQSLimit` and `ECMLimit` are set as optional parameters by the users, the algorithm will continue until the complete factorization has been completed.

Outline

Elliptic Curves

Elliptic Curve Discrete Logarithm Problem

Algorithmic Aspects

Factorization and Primality testing
- Elliptic Curve Factorization Method
- Elliptic Curve Primality Proving

Pairings

Primality test vs Primality Proof

- Given $n \in \mathbb{Z}$, is n prime ?
- Fermat and Miller-Rabin algorithms are primality tests : they return a definitive no or a plausible yes
- Goldwasser-Killian algorithm aims at primality **proving** : returns a short proof that a number is prime

Main idea

- Let E be an elliptic curve over $\mathbb{Z}_n$, let $P \neq O \in E$
- When $p|n$ can consider E and P "modulo p"
- If $\text{ord}(P)$ is prime then $\text{ord}(P \bmod p) = \text{ord}(P)$
- Let $p|n$ with $p < \sqrt{n}$
- $\text{ord}(P \bmod p)$ is bounded above by Hasse's theorem $\text{ord}(P \bmod p) \leq p + 1 + 2\sqrt{p} = (\sqrt{p}+1)^2 < (n^{1/4}+1)^2$
- So if $\text{ord}(P)$ is prime and $\geq (n^{1/4}+1)^2$ then n is prime

Goldwasser-Killian Algorithm

1. Randomly choose A, B such that $\gcd(4A^3 + 27B^2, n) = 1$ and $E : y^2 = x^3 + Ax + B$ has order $2q$ with q prime
   - GCD condition ensures E regular modulo any divisor of n
   - Order of E computed with Schoof's algorithm, assuming n is prime : if algorithm fails then n is composite
   - Primality of q tested with Miller-Rabin algorithm
2. Find P of order q in E
   - Select random x until $x^3 + Ax + B$ is square
   - Compute corresponding y assuming n is prime : if algorithm fails then n composite
3. Recursively prove that q is prime : if not then restart
4. Return A, B, P, q and a proof that q is prime

Remarks

- Heuristically expect $O(\log q)$ trials until q is prime, and polytime algorithm
- Atkin-Morain : avoid point counting and use complex multiplication instead

Outline

Outline

Cryptographic pairings

- Pairings are non-degenerate, bilinear maps
$$e : G_1 \times G_2 \to G_3$$
where G_i are all cyclic groups of the same order r (usually consider G_1, G_2 additive and G_3 multiplicative)
- Bilinear :
$$\begin{aligned} e(P_1 + Q_1, P_2) &= e(P_1, P_2)e(Q_1, P_2) \\ e(P_1, P_2 + Q_2) &= e(P_1, P_2)e(P_1, Q_2) \end{aligned}$$
- Non-degenerate : for all $P_1 \in G_1$, $P_1 \neq O$, there exists $P_2 \in G_2$ such that $e(P_1, P_2) \neq 1$ (and vice-versa)

Properties

- $e(P, O) = e(O, P) = 1$
- $e(-P, Q) = e(P, Q)^{-1} = e(P, -Q)$
- $e(jP, Q) = e(P, Q)^j = e(P, jQ)$ for any $j \in \mathbb{Z}$
- We say the pairing is symmetric if $G_1 = G_2$
- For symmetric pairings
$$e(P_1, P_2) = e(P_1, kP_1) = e(kP_1, P_1) = e(P_2, P_1)$$

Pairings : applications

- Tripartite Diffie-Hellman
- Identity-based encryption
- Short signatures
- Groth-Sahai zero-knowledge proofs of knowledge
- ...
- Building new primitives
- Improving efficiency
- Removing random oracles in security proofs (replacing them with new pairing computational assumptions)

Remember : (Elliptic curve) Diffie-Helman

- Goal : key agreement
  Two parties want to build a common secret key
- Alice chooses random r_a. She sends $Q_a := [r_a](P)$ to Bob
- Bob chooses random r_b. He sends $Q_b := [r_b](P)$ to Alice
- Alice computes $K_a := [r_a](Q_b)$
- Bob computes $K_b := [r_b](Q_a)$
- We have $K_a = [r_a r_b](P) = K_b$

3-partite Diffie-Hellman

- Goal : key agreement
  Three parties want to build a common secret key
- Public parameters
  - Symmetric pairing $e : G_1 \times G_1 \to G_3$
  - A generator P in G_1
- Party i chooses random r_i and sends $Q_i = [r_i]P_i$ to the other parties
- On receiving Q_i and Q_j, party k computes the common secret key

$$e(Q_i, Q_j)^{r_k} = e(P, P)^{r_i r_j r_k}$$

3-partite Diffie-Hellman (2)

- Originally described for asymmetric pairings, in which case two elements are sent instead of one in first round
- **Bilinear Diffie-Hellman problem** (BDH) :
  given P, $P_i = [r_i]P$ for random r_i, compute $e(P, P)^{r_1 r_2 r_3}$
- BDH must be hard for secure 3-partite Diffie-Hellman
- BDH hard implies DH (hence DLP) hard in G_1 and G_3

Identity-based cryptography

- Identity (ID)-based cryptography
  - Public key is (some hash of) identity
  - No need for certificates
  - Trusted Authority (TA) generates private keys
- Simplifies public key infrastructure at the price of key escrow : Trusted Authority knows all secret keys
- Idea of ID-based encryption suggested by Shamir ; solution using bilinear pairings by Boneh-Franklin

Boneh-Franklin

- Identity-based encryption scheme
- Public parameters
  - Pairing $e : G_1 \times G_1 \to G_3$
  - Generator $P \in G_1$
  - Hash function $H_1 : \{0,1\}^* \to G_1$
  - Hash function $H_3 : G_3 \to \{0,1\}^n$
- TA chooses a master secret key s randomly and publishes master public key $Q = [s]P$

Boneh-Franklin (2)

- Public key of user i computed as $Q_i = H_1(ID_i)$
- Private key of user i computed as $S_i = [s]Q_i$ by TA, and sent to user i using a secure channel
- Encryption of n-bit message M for party i is

$$C = (C_1, C_2) = ([t]P, M \oplus H_3(e(Q, Q_i)^t))$$

  for a randomly chosen t
- Party i uses its private key S_i to decrypt as

$$M = C_2 \oplus H_3(e(C_1, S_i))$$

Security notions

- IND-ID-CPA security
  - Adversary can get encryptions on messages of his choice
  - Adversary can also get secret keys on identities of his choice
  - Adversary chooses two messages M_1, M_2
  - Adversary must distinguish encryptions of M_1 and M_2
- IND-ID-CCA2 security
  - After receiving an encryption of either M_1 or M_2, adversary can additionally make decryption queries on other ciphertexts of his choice

Boneh-Franklin security

- IND-ID-CPA secure if BDH is hard & random oracles
  Proof : H_3 is random oracle so only way to distinguish is to guess pairing value.
  Guessing the pairing value corresponds to BDH. Indeed, let r such that $Q_i = rP$ then adversary sees P, rP, sP, tP and must produce $e(P,P)^{rst}$
- Not IND-CCA2
  Given $C = (C_1, C_2)$, ask for a decryption of $C' = (C_1, C_2 \oplus R)$
- Can be extended into an IND-CCA2 secure version
  See paper for details.

Boneh-Lynn-Sacham (BLS) signatures

- Public parameters
  - Pairing $e : G_1 \times G_1 \to G_3$
  - Generator $P \in G_1$
  - Hash function $H : \{0,1\}^* \to G_1$
- User public key is $Q = [s]P$, for randomly chosen s
  User secret key is s
- Signature on $M \in \{0,1\}^*$ is $\sigma = [s]H(M)$
- Signature σ on M is verified by

$$e(P, \sigma) = e(Q, H(M))$$

BLS signatures (2)

- Existentially unforgeable under chosen message attacks if CDH is hard in G_1 & random oracles
  Intuition : as H is random oracle $R = H(m)$ cannot be manipulated, hence adversary left with computing $[s]R$ from R, P and $[s]P$
- Very short signatures : one element in G_1
  Paper suggests elliptic curve pairing parameters such that security 168-bit BLS signatures $\sim$ 1024-bit RSA at the time (needs revision)

Outline

Divisors

- Let E be an elliptic curve over a finite field $K = \mathbb{F}_q$
- A divisor D on E is a formal sum of points
$$D = \sum_{P \in E(\bar{K})} n_P(P)$$
where $n_P \in \mathbb{Z}$ and all but a finite number of them are 0
- Support of D is the set of all points with $n_P \neq 0$
- Degree of D is $\sum n_P$
- Natural group structure, with neutral element written 0

Divisor of a function

- Let f be a function on E, and P a point on E
- Write $\mathrm{ord}_P f$ for the order of P at f
sign depends on whether P is a zero or a pole of f
- Define the divisor of f as $(f) = \sum_{P \in E(\bar{K})} \mathrm{ord}_P f(P)$
- $(fg) = (f) + (g)$
- $(f) = 0 \Leftrightarrow f$ is constant
- (f) defines f up to a constant factor
- D is called a principal divisor if $D = (f)$ for some f
- Define equivalence relation $D_1 \sim D_2$ if $D_1 - D_2$ principal

Addition law

- Any vertical line corresponds to a linear function
$v(x, y) = x - \hat{x}$ with divisor $(P) + (-P) - 2(O)$
- Any non vertical line corresponds to a linear function
$\ell(x, y)$ with divisor $(P_1) + (P_2) + (-P_1 - P_2) - 3(O)$
- Equation $P_1 + P_2 = P_3$ equivalent to divisor equality
$(P_3) - (O) = (P_1) - (O) + (P_2) - (O) - (\ell/v)$
- Group homomorphism $P \to (P) - (0)$
up to principal divisors
- Let $D = \sum_P n_P(P)$ be a degree 0 divisor on E.
Then $D \sim 0$ if and only if $\sum_P [n_P]P = O$.

Weil reciprocity

- Let $D = \sum_P n_P(P)$ a divisor and f a function on E
- Define
$$f(D) = \prod_P f(P)^{n_P}$$
- If $\deg D = 0$ and $g = cf$ for a constant c
then $f(D) = g(D)$
- **Weil reciprocity** : if the support of (f) and (g) are disjoint then
$$f((g)) = g((f))$$

See [BSS], chapter 9 for a proof.

Tate pairing

- Let E defined over a finite field $K_0 = \mathbb{F}_q$, and $n \in \mathbb{Z}_+$
- Let $K = \mathbb{F}_{q^k}$ be the extension of K_0 containing all the nth roots of unity
- Let $E(K)[n] = \{P \in E(K) \mid [n]P = 0\}$
- Let $nE(K) = \{[n]P \mid P \in E(K)\}$
- Let $(K^*)^n = \{u^n \mid u \in K^*\}$
- Tate pairing $e : E(K)[n] \times E(K)/nE(K) \to K^*/(K^*)^n$ defined by
$$e(P, Q) = f(D)$$
where $(f) = n(P) - n(0)$ and $D \sim (Q) - (0)$ have disjoint supports

Properties of the Tate pairing

- Result independent of the choice of $D \sim (Q) - (0)$: $D' \sim D$ implies $f(D')/f(D) \in (K^*)^n$
  Proof : write $D' = D + (g)$. Then $f(D') = f(D)f((g))$ and $f((g)) = g((f)) = g(n(P) - n(0)) = (g(P)/g(0))^n$
- Result independent of representative of $Q \in E(K)/nE(K)$ $Q' = Q + [n]R$ implies $f(D')/f(D) \in (K^*)^n$
  Proof : we have $D' \sim D + n(R) - n(0)$ hence up to nth powers $f(D') = f(D + n(R) - n(0)) = f(D)(f((R) - (0)))^n = f(D)$
- Can replace $(f) = n(P) - n(0)$ by $(f') = n(P + R) - n(R)$
  Proof : let h so that $(h) = (P + R) - (P) - (R) + (0)$. Let $f' = fh^n$ so $f'(D)/f(D) \in (K^*)^n$ and $(f') = (f) + n(h) = n(P + R) - n(R)$

Properties of the Tate pairing(2)

- Linearity wrt first term
  Proof : let $P_1 + P_2 = P_3$ and g a function such that $(P_3) - (0) = (P_1) - (0) + (P_2 - 0) + (g)$.
  Let f_i such that $(f_i) = n(P_i) - n(0)$ for $i = 1, 2$.
  Then $f_3 = f_1 f_2 g^n$ satisfies $(f_3) = n(P_3) - n(0)$.
- Linearity wrt second term
  Proof : let $Q_1 + Q_2 = Q_3$. Let $D_i \sim (Q_i) - (0)$ for $i = 1, 2, 3$.
  Then $D_3 \sim D_1 + D_2$ hence $f(D_3) = f(D_1)f(D_2)$.
- Non-degenerate

Reduced Tate pairing

- The output of the Tate pairing is an element in $K^*/(K^*)^n$
- Representation as an element of K^* not unique
- Let $\mu_n = \{u \mid u^n = 1\} \subseteq K^*$ be the nth roots of unity
- Reduced Tate pairing
$$e : E(K)[n] \times E(K)/nE(K) \to \mu_n$$
defined by
$$e(P, Q) = e'(P, Q)^{(q^k - 1)/n}$$
where e' is the Tate pairing

Embedding degree

- To define the Tate pairing we need an extension $\mathbb{F}_{q^k}$ of $\mathbb{F}_q$, such that n divides $q^k - 1$
- The smallest such k is called the embedding degree
- k is the order of q modulo n i.e. k divides $\varphi(n)$
- Construction only practical for small embedding degrees
- Given any reasonably large n dividing $\#E(\mathbb{F}_q)$, unlikely that n divides $q^k - 1$ for small k
- We will need special curves to implement pairings

Weil pairing

- Let $\mu_n = \{u \mid u^n = 1\} \subseteq K^*$ be the nth roots of unity
- Weil pairing is a map

$$e_n : E(K)[n] \times E(K)[n] \to \mu_n$$

defined by

$$e_n(P_1, P_2) = f_1(D_2)/f_2(D_1)$$

where $D_i \sim (P_i) - (0)$ and $(f_i) \sim nD_i$

Properties of Weil pairing

- Bilinearity
- Alternating : $e_n(P, P) = 1$ so $e_n(P, Q) = e_n(Q, P)^{-1}$
- Non-degenerate : if $e_n(P, Q) = 1$ for all $Q \in E[n]$ then $P = 0$
- Let n prime and $P \neq 0$ in $E[n]$. Then $e_n(P, Q) = 1$ if and only if $Q \in \langle P \rangle$.
- Often for cryptographic parameters

$$e_n(P, Q) = e'(P, Q)/e'(Q, P)$$

where e' is the Tate pairing

FR / MOV reduction

- Let E an elliptic curve over $\mathbb{F}_q$, let $P \in E(\mathbb{F}_q)$ a point of prime order r, and let $Q \in \langle P \rangle$
- Let e a bilinear pairing on E with images in $\mathbb{F}_{q^k}^*$
- Frey-Rück / Menezes-Okamoto-Vanstone reduction
  - Find S such that $e(P, S) \neq 1$
  - Let $g = e(P, S)$ and $h = e(Q, S)$
  - Find x such that $h = g^x$
- The last step is subexponential in q^k, hence subexponential in q when k is not too large

Symmetric pairings

- In many protocols we need symmetric pairings $e : G_1 \times G_1 \to G_3$
- Tate and Weil pairings are such that $e(P, P) = 1$
- Idea to build symmetric pairings
  - Find an endomorphism $\phi : E \to E$ sending some $G_1 = \langle P \rangle$ of prime order to $G_2 \neq G_1$
  - Define a modified Weil/Tate pairing as

$$\hat{e}_n(P, P) = e_n(P, \phi(P))$$

Distortion maps

- Let $P \in E(\mathbb{F}_q)$ with prime order r, and suppose $k > 1$
- A distortion map is an endomorphism $\phi : E \to E$, defined over $\mathbb{F}_{p^k}$, sending $P \in E(\mathbb{F}_q)[r]$ to $\phi(P) \notin E(\mathbb{F}_q)[r]$
- Example :
  - Let $p = 2 \mod 3$
  - Let $E : y^2 = x^3 + a$ with $a \in \mathbb{F}_{p^2}$ square but not cube
  - Let $u \in \mathbb{F}_{p^6} \setminus \mathbb{F}_{p^2}$ such that $u^6 = a/a^p$
  - Define $\phi(x, y) = (u^2 x^p, u^3 y^p)$
- If E has a distortion map then E is supersingular

  Proof : let π be the Frobenius. For $P \in E(\mathbb{F}_q)$ we have $\pi(\phi(P)) \neq \phi(P) = \phi(\pi(P))$ hence $End(E)$ is non-Abelian.

Trace map

- Let E defined over $\mathbb{F}_q$
- Let P be a point of order r in $E(\mathbb{F}_{q^k})$
- Trace map defined as

$$\mathrm{Tr}(x, y) = \sum_{i=0}^{k-1} (x^{q^i}, y^{q^i})$$

- The map $\phi : P \to [k]P - \mathrm{Tr}(P)$ is a homomorphism, with the trace zero subgroup as image
- Can ensure $e(P, \phi(P)) \neq 1$

Miller's algorithm

- Both Tate and Weil pairing require the evaluation $f(D)$ where $(f) = n(P) - n(0)$ and $D \sim (Q) - (0)$
- As n is of cryptographic size, even storing f could take prohibitive time and space
- Miller's algorithm uses a variant of square-and-multiply algorithm to successively compute $f_i(D)$ where

$$(f_i) \sim i(P) - ([i]P) - (i - 1)(0)$$

- The algorithm uses recursive formula

$$\begin{aligned} (f_{i+j}) &= (f_i) + (f_j) + ([i]P) + ([j]P) - ([i + j]P) - (0) \\ &= (f_i) + (f_j) + (\ell_{i+j}/v_{i+j}) \end{aligned}$$

Pairing-friendly elliptic curves

- Efficient arithmetic on the elliptic curve (q not too big)
- Efficient pairing computation (q^k not too big)
- Elliptic curve discrete logarithm hard (elliptic curve subgroup of size r at least 160 bits)
- Discrete logarithm problem hard in image group (q^k at least 1000-2000 bits, sometimes bigger)
- We want $\rho = \log q / \log r$ as small as possible

Supersingular curves

- Embedding degree is at most 6

k	Elliptic curve data
2	$E : y^2 = x^3 + a$ over $\mathbb{F}_p$, where $p \equiv 2 \pmod 3$ $\#E(\mathbb{F}_p) = p + 1$ Distortion map $(x, y) \mapsto (\zeta_3 x, y)$, where $\zeta_3^3 = 1$.
2	$y^2 = x^3 + x$ over $\mathbb{F}_p$, where $p \equiv 3 \pmod 4$ $\#E(\mathbb{F}_p) = p + 1$. Distortion map $(x, y) \mapsto (-x, iy)$, where $i^2 = -1$.
3	$E : y^2 = x^3 + a$ over $\mathbb{F}_{p^2}$, where $p \equiv 5 \pmod 6$ and $a \in \mathbb{F}_{p^2}$, $a \notin \mathbb{F}_p$ is a square which is not a cube. $\#E(\mathbb{F}_{p^2}) = p^2 - p + 1$. Distortion map $(x, y) \mapsto (x^p/(\gamma a^{(p-2)/3}), y^p/a^{(p-1)/2})$, where $\gamma \in \mathbb{F}_{p^6}$ satisfies $\gamma^3 = a$.

Picture source : Advances in ECC, p204.

- Small characteristic curves ($k = 4, 6$) are now broken by MOV + quasipolynomial time algorithm in image field

Miyaji-Nakabayashi-Takano (MNT) curves

- Suppose $\#E(\mathbb{F}_q) = q + 1 - t$ prime and $k \in \{3, 4, 6\}$. Then q and t must satisfy the following relations

k	q	t
3	$12\ell^2 - 1$	$-1 \pm 6\ell$
4	$\ell^2 + \ell + 1$	$-\ell$ or $\ell + 1$
6	$4\ell^2 + 1$	$1 \pm 2\ell$

- Fix small D and search for solutions of $t^2 - 4q = -Dy^2$ with $q + 1 - t$ prime (quadratic diophantine equation in y and ℓ solved with Euclidean algorithm, & primality test)
- Use CM method to generate curves

Other methods

- Cocks-Pinch method
- Barreto-Naehrig curves
- Dupont-Enge-Morain method
- Brezing-Weng curves
- Barreto-Lynn-Scott curves
- ...
- See Freeman-Scott-Teske, *A taxonomy of pairing-friendly elliptic curves*

Security considerations

- Hardness of pairing problems typically implies but is not implied by ECDLP/ECDHP hardness
- Besides BDHP, many other assumptions suggested, but they have not been well-studied (see Koblitz-Menezes, *The brave new world of bodacious assumptions in cryptography*)
- Even the "purest" pairing problems probably need more study (ask me for a research project in that direction)

Outline

Conclusion on Elliptic Curve Cryptography

- ECDLP is typically much harder to solve than DLP, hence allowing smaller keys
- Assumptions related to the Weil and Tate pairings have allowed to build many schemes with interesting properties
- Extra structure (endomorphisms, isogenies) used to improve efficiency, without affecting security so far