The State of the Art of

Elliptic Curve Cryptography

Ernst Kani

Department of Mathematics and Statistics

Queen's University

Kingston, Ontario

Elliptic Curve Cryptography

Outline

1. ECC: Advantages and Disadvantages
2. Discrete Logarithm (DL) Cyptosystems
3. Elliptic Curves (EC)
4. A Small Example
5. Attacks and their consquences
6. ECC System Setup
7. Elliptic Curves: Construction Methods

ECC: Advantages/Disadvantages

Advantages:

- greater flexibility in choosing cryptographic system
- no known subexponential time algorithm for ECDLP $\Rightarrow$ smaller key sizes (with the same security).
  Current recommendation (according to A.K. Lenstra, E.R. Verheul): the minimum key size for ECC should be 132 bits vs. 952 bits for RSA.
- As a result: greater speed, less storage $\Rightarrow$ ECC can be used in smart cards, cellular phones, pagers etc.

Disadvantages:

- Hyperelliptic cryptosystems offer even smaller key sizes.
- ECC is mathematically more subtle than RSA or SDL $\Rightarrow$ difficult to explain/justify to the client.

Main uses of ECC: key exchange, digital signature, authentication, (limited) message transmission, etc.

DL - Cryptosystems

Basic Problem: Let G be an abstract (multiplicative) group (= a set with a multiplication operation $\cdot$). Find a computer realization of G such that:

1) The operation "exponentiation" $a \to b := a^n$ can be implemented as a quick, efficient algorithm;

2) The inverse operation ("discrete logarithm"), i.e., (DLP) Given a and $b \in G$, find n ("$= \log_a(b)$") such that
$$b = a^n = \underbrace{a \cdot a \cdots a}_{n\ times},$$
is technically much harder and hence cannot be carried out in a reasonable amount of time.

Classical Examples:

1) Let $G = \mathbb{F}_p^\times = \{1, \ldots p-1\}$, where p is a prime, with multiplication $a \cdot b = \text{rem}(ab, p)$.

2) (SDL) Take $G = \{1, g, g^2, \ldots g^{q-1}\} \subset \mathbb{F}_p^\times$, a cyclic subgroup of order q.

3) More generally, let $G \subset \mathbb{F}_p^\times$, where $\mathbb{F}_p$ is any finite field; here p is a power of a prime.

A Sample Protocol:

The Diffie-Hellman Key Exchange (1976):

Public information: An element $g \in \mathbb{F}_p^\times$ of large order q.

User A (Alice) picks a random integer a, sends B the number (public key) $P_A = g^a$.

User B (Bob) picks a random integer b, sends A the number (public key) $P_B = g^b$.

Then: A and B can both compute and use the common (secret) key $S_{AB} = (P_A)^b = (P_B)^a = g^{ab}$.

The Diffie-Hellman Problem (DHP): compute the secret key S_{AB} from the data g, P_A, P_B.

The Diffie-Hellman Assumption: a spy (Eve) cannot solve (DHP) in a reasonable amount of time.

Remarks: 1) (DLP) $\Rightarrow$ (DHP).

2) "There is strong evidence" (Lenstra/Verheul) that the (DHP) is equivalent to the (DLP). In fact: this is true for many orders q (Maurer/Wolf/Boneh, 1996).

Elliptic Curves

Elliptic curves: Let $a, b \in \mathbb{F}_p$ and consider

$$G = E_{a,b}(\mathbb{F}_p) := \{(x,y) \in \mathbb{F}_p \times \mathbb{F}_p : y^2 = x^3 + ax + b\} \cup \{P_\infty\}$$

The group law: The multiplication in G is given by

$$(x_1, y_1) * (x_2, y_2) = (x_3, y_3)$$

where

$$\left.\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= \lambda(x_1 - x_3) - y_1 \end{aligned}\right\} \text{ with } \lambda = \frac{y_2 - y_1}{x_2 - x_1} \text{ (or } \underbrace{\lambda}_{\text{if } x_1 = x_2} = \frac{3x_1^2 + a}{2y_1}\text{).}$$

Notes: 1) We usually write the group law additively, i.e. we write $P+Q$ in place of $P*Q$ and hence kP in place of $\underbrace{P * P * \cdots * P}_{k}$.

2) The extra point P_∞ serves as the identity of the group law:

$$P_\infty + P = P + P_\infty = P, \quad \text{for all } P \in E(\mathbb{F}_p).$$

3) V. Miller and N. Koblitz (independently) first proposed in 1986 the use of elliptic curves for cryptography.

Problem: How can we find an elliptic curve $E/\mathbb{F}_p$ which is suitable for implementing a DL-cryptosystem?

Questions: 1) How can we estimate/calculate $\#G$?

2) How can we find a point of large order in G?

Theorem (Hasse, 1933): $\#E(\mathbb{F}_p) \approx p$; more precisely,

$$|\underbrace{p + 1 - \#E(\mathbb{F}_p)}_{tr_E}| \le 2\sqrt{p}.$$

A Small Example

Consider the following elliptic curve $E_{2,1}$ over $\mathbb{F}_5$:

$$E = E_{2,1}: \qquad y^2 = x^3 + 2x + 1.$$

Then a quick calculation (exhaustive search) shows that

$$E(\mathbb{F}_5) = \{\underbrace{P_\infty}_{P_0}, \underbrace{(0,1)}_{P_1}, \underbrace{(1,3)}_{P_2}, \underbrace{(3,3)}_{P_3}, \underbrace{(3,2)}_{P_4}, \underbrace{(1,2)}_{P_5}, \underbrace{(0,4)}_{P_6}\}.$$

For example, $P_4 = (3,2) \in E(\mathbb{F}_5)$ because

$$3^3 + 2 \cdot 3 + 1 = 34 \equiv 4 \equiv 2^2 \pmod 5.$$

The above points P_i have been numbered in such way that

$$P_i + P_j = P_{i+j} \quad (\text{indices mod } 7).$$

Thus, $\#E(F_5) = 7$, which satisfies the Hasse bound since

$$|(5+1) - 7| = 1 \le 2\sqrt{5} \doteq 4.47.$$

Diffie-Hellman Key Exchange:

A chooses the secret key $a = 2$ and computes her public key

$$P_A = 2P_1 = P_1 + P_1 = P_2 = (1,3).$$

B chooses the secret key $b = 3$ and computes his public key

$$P_B = 3P_1 = P_1 + P_1 + P_1 = P_3 = (3,3).$$

Thus, their common secret key is:

$$S_{AB} = P_6 = (0,4) \begin{cases} 2P_B = 2P_3 = P_6 & (\text{as computed by } A) \\ 3P_A = 3P_2 = P_6 & (\text{as computed by } B) \end{cases}$$

Attacks and their Consequences

General DL-Attacks:

SPH - due to Silver, Pohlig, Hellman (1978):
the DLP for a group G of order n can be reduced to solving the DLP for its subgroups of prime order $p|n$.

Pollard's ρ and λ (or Kangaroo) - due to Pollard(1978):
each solves DLP in $O(\sqrt{n})$ steps. (Parallizable!)

Consequence: work with a group G of sufficiently large prime order $q = \#G$.

Elliptic Curve Attacks:

MOV - due to Menezes, Okamoto, Vanstone (1993); cf. also Frey, Rück (1994): if

$$p^r \equiv 1 \pmod{q}, \qquad (1)$$

then the ECDLP can be reduced to the DLP in $\mathbb{F}_{p^r}$
$\Rightarrow$ we can solve the ECDLP by using the (subexponential) Index Calculus in $\mathbb{F}_{p^r}$.

Anomalous - due to Samaev (1998); Satoh, Araki (1998), Smart (1999): the ECDLP can be solved (using p-adic numbers) for anomalous curves, i.e. those with $\#E(\mathbb{F}_p) = p$ ($\Leftrightarrow tr_E = 1$).

Consequence: For ECC, avoid:

1) anomalous curves;

2) primes q which satisfy (1) for small r; i.e. $r << k^2/\log_2 k$, where $k = \log_2 p$.

Explicit Attacks:

Hardware attack estimate: In 1996, an attack against a 120-bit EC sytem was proposed - using a machine running 75 independent Pollard ρ processors. (Estimated cost: \$10 million, running time: 32 days.)

Software attack estimate (A.Lenstra, E.Verheul, 1999): On a 109-bit EC system ($p \approx 2^{109}$), the ECDLP should take 18,000 years on a current PC (or 1 year on 18,000 PC's $\longrightarrow$ "Power of the Internet") by using Pollard's ρ method. (PC = 450MHz Pentium II processor).

RSA155 (= 512 bit RSA) was factored in August 1999 using the NFS (Number Field Sieve). Run Time: 20 years on 1 PC (64Mb memory) = 1 day on 7500 PC's.

Note: RSA155 is still used on the Web (e.g. in the Secure Socket Layer(SSL)), but cannot be considered to be secure.

Consequences: A.K.Lenstra, E.R.Verheul (Sep. 1999) propose the following minimum key sizes (in bits):

Year	RSA	SDL		EC
		q	p	wo (w)*
2000	952	123	952	132 (132)
2005	1149	131	1149	139 (147)
2025	2174	158	2174	169 (202)
2050	4047	193	1447	206 (272)

*without (with) cryptoanalytic progress

ECC Sytem Setup

System Setup: There are several choices to be made:

- Selecting a finite field $\mathbb{F}_p$ (and a field representation)
  – e.g. $p = prime$ or $p = 2^k$.
- Selecting an elliptic curve $E/\mathbb{F}_p$ (+ a point P of order q)
  – random curve vs. a special curve etc. (see next page)
- Selecting the elliptic curve representation
  – affine vs. projective coordinates, etc.)
- Selecting a protocol (e.g. Diffie-Hellman, ECDSA) for task
  Note: Some protocols require additional steps; e.g. ElGamal and others require message embedding $m \to P_m \in E(\mathbb{F}_q)$.

Selection criteria:

- Security: of ECDLP, of protocol.
- Implementation requirements:
  – speed, storage, power consumption
  $\to$ optimization of field/EC operations, of protocol.
- Platform dependence: speed and performance of primitives
  – e.g. on a Pentium PC, the time for multiplication is only a small multiple of that for addition.
- Standards Compatibility: Public Key Infrastructure, Wassenaar Arrangement (Export).

Remark: S. Vanstone (Field Institute Conference, 1999) emphasizes that all these selection criteria must be considered simultaneously.

EC Construction Methods

Method 1: Random elliptic curves $E/\mathbb{F}_p$:

– first choose a field $\mathbb{F}_p$, then a point $P = (x, y) \in \mathbb{F}_p \times \mathbb{F}_p$,
– then choose (by varying the paramater a) an elliptic curve $E_{a,b}$ such that P is a point of suitable order q on $E_{a,b}(\mathbb{F}_p)$.

Note: in order to find $ord(P)$, first calculate $\#E(\mathbb{F}_p)$ using the Schoof-Elkies-Atkin (SEA) algorithm.

Method 2: CM elliptic curves $E/\mathbb{Q}$:

– pick a CM elliptic curve $E/\mathbb{Q}$ (and a point $P \in E(\mathbb{Q})$),
– then look for a prime p such that $(E, P) \pmod p$ has the right cryptographic properties.

Advantage: there is a "formula" for $\#E(\mathbb{F}_p)$.

Example: $E : y^2 = x^3 - n^2x$ is a CM-curve with point $P = (Z^2/4, (Y^2 - X^2)Z/8)$, provided that (X, Y, Z) are the sides of a right-angled triangle with area n. By Gauss (1777-1855):

–if $p \equiv 3 \pmod 4$, then $\#E(\mathbb{F}_p) = p + 1$ (do not use for ECC!)

–if $p \equiv 1 \pmod 4$, then there is an explicit formula.

Method 3: Koblitz (subfield) curves:

– take $p = p_0^r$ (p_0 small), choose E/F_{p_0} and view E over $\mathbb{F}_p$.

Advantage: there is a simple formula (Artin, 1926) for $\#E(F_p)$ in terms of $\#E(F_{p_0})$ (which can be calculated quickly).

Method 4: Arbitrary elliptic curves $E/\mathbb{Q}$ (?):

– similar to method 2 (formula for $\#E(\mathbb{F}_p)$ via modular forms?)