Elliptic Curves in Crypto part II

1. addition is closed on the set E,
2. addition is commutative,
3. $\mathcal{O}$ is an identity with respect to addition, and
4. every point on E has an inverse with respect to addition.

missing: associativity
(messy proof omitted)

special case

$$(-P) + (P + Q) = Q$$

THEOREM 6.1 *Let E be an elliptic curve defined over $\mathbb{Z}_p$, where p is prime and $p > 3$. Then there exist positive integers n_1 and n_2 such that $(E, +)$ is isomorphic to $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$. Further, $n_2 \mid n_1$ and $n_2 \mid (p-1)$.*

Note that $n_2 = 1$ is allowed in the above theorem. In fact, $n_2 = 1$ if and only if E is a cyclic group. Also, if $\#E$ is a prime, or the product of distinct primes, then E must be a cyclic group.

In any event, if the integers n_1 and n_2 are computed, then we know that $(E, +)$ has a cyclic subgroup isomorphic to $\mathbb{Z}_{n_1}$ that can potentially be used as a setting for an *ElGamal Cryptosystem.*

Z_4 vs $Z_2 \times Z_2$

Properties of Elliptic Curves

Hasse bound

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

↑

Schoof algorithm
$O(\log^8 p)$ - bit ops

THEOREM 6.1 *Let E be an elliptic curve defined over $\mathbb{Z}_p$, where p is prime and $p > 3$. Then there exist positive integers n_1 and n_2 such that $(E, +)$ is isomorphic to $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$. Further, $n_2 \mid n_1$ and $n_2 \mid (p-1)$.*

cyclic subgroup of E
of order 2^{160} is "safe"

$n_2 = 1$ iff E cyclic

Schoof's algorithm

From Wikipedia, the free encyclopedia

Schoof's algorithm, first described by R. Schoof in 1985, allows one to calculate the number of points on an elliptic curve over a finite field and is used mostly in elliptic curve cryptography.

From Hasse's theorem on elliptic curves the number of point on a curve is roughly known:

$$|E(\mathbf{F}_q)| = q + 1 \pm 2\sqrt{q},$$

so to find the exact number it is enough to find it modulo $R > 4\sqrt{q}$. Schoof's algorithm calculates

$$q + 1 - |E(\mathbf{F}_q)| \pmod{r_i}$$

for several small primes r_i, where $\prod r_i = R$, and uses the Chinese remainder theorem to combine the results.

The running time of the original algorithm is proportional to q^8 and with several improvements can be reduced to $O(q^6)$, which is adequate for $q < 256$ on a PC.

The algorithm has been extended by Noam Elkies and A. O. L. Atkin to give the Schoof-Elkies-Atkin algorithm, which has only $O(q^5)$ time complexity and thus is always faster.

References

- R. Schoof, *Elliptic curves over finite fields and the computation of square roots mod p*, Mathematics of Computation, Volume 44, 1985.

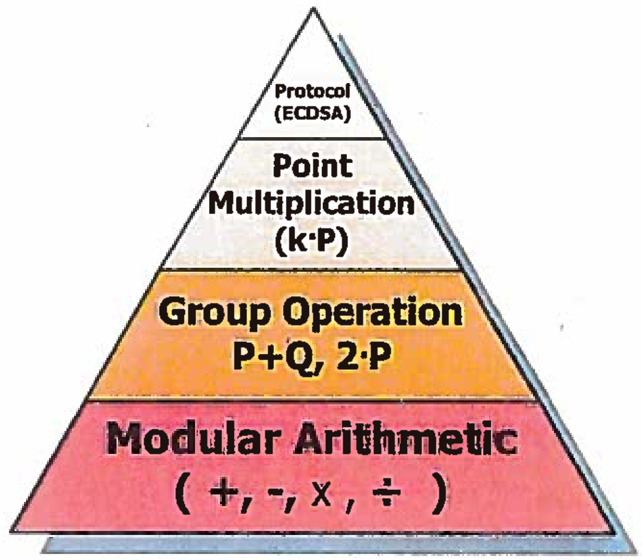
Implementations

Several algorithms were implemented in C++ by Mike Scott and are available with source code (ftp://ftp.compapp.dcu.ie/pub/crypto/) . The implementations are free (no terms, no conditions), but they use MIRACL (http://indigo.ie/~mscott/) library which is only free for non-commercial use. Note that (unmodified) programs may be used to generate curves for commercial use. There are

- Schoof's algorithm implementation (ftp://ftp.compapp.dcu.ie/pub/crypto/schoof.cpp) for $E(\mathbf{F}_p)$ with prime p.
- Schoof's algorithm implementation (ftp://ftp.compapp.dcu.ie/pub/crypto/schoof2.cpp) for $E(\mathbf{F}_{2^m})$.

Implementations in Hardware and Software

- Elliptic curve computations usually regarded as consisting of four layers:
  - Basic modular arithmetic operations are computationally most expensive
  - Group operation implements point doubling and point addition
  - Point multiplication can be implemented using the Double-and-Add method
  - Upper layer protocols like ECDH and ECDSA



- Most efforts should go in optimizations of the modular arithmetic operations, such as
  - Modular addition and subtraction
  - Modular multiplication
  - Modular inversion

Let c be an integer. A *signed binary representation* of c is an equation of the form

$$c = \sum_{i=0}^{\ell-1} c_i 2^i,$$

where $c_i \in \{-1, 0, 1\}$ for all i. In general, there will be more than one signed binary representation of an integer c. For example, we have that

$$11 = 8 + 2 + 1 = 16 - 4 - 1,$$

so

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

are both signed binary representations of 11.

Algorithm 6.5: DOUBLE-AND-(ADD OR SUBTRACT)$(P, (c_{\ell-1}, \ldots, c_0))$

```
Q ← 0
for i ← ℓ − 1 downto 0
     ⎧ Q ← 2Q
     ⎪ if c_i = 1
  do ⎨   then Q ← Q + P
     ⎪   else if c_i = −1
     ⎩   then Q ← Q − P
return (Q)
```

$$2^i + 2^{i-1} + \cdots + 2^j = 2^{i+1} - 2^j,$$

$i > j$

011111 → 10000-1

	1	1	1	1	0	0	1	1	0	1	1	1
						↓						
	1	1	1	1	0	0	1	1	1	0	0	−1
						↓						
	1	1	1	1	0	1	0	0	−1	0	0	−1
						↓						
1	0	0	0	−1	0	1	0	0	−1	0	0	−1

Hence the NAF representation of

$$(1, 1, 1, 1, 0, 0, 1, 1, 0, 1, 1, 1)$$

is

$$(1, 0, 0, 0, -1, 0, 1, 0, 0, -1, 0, 0, -1).$$

Obtaining Non Adjacent Form

unique

binary $l + l/2$

NAF $l + l/3$

doubles adds

ratio $\frac{3/2}{4/3} = \frac{9}{8}$

NIST recommended a certain set of elliptic curves for government use. This set of curves can be divided into two classes: curves over a prime field $GF(p)$ and curves over a binary field $GF(2^m)$. The curves over $GF(p)$ are of the form

$$y^2 = x^3 - 3x + b$$ 192, 224, 256, 384, 521

with b random, while the curves over $GF(2^m)$ are either of the form

$$y^2 + xy = x^3 + x^2 + b$$ 163, 233, 283, 409, 571

with b random or *Koblitz curves*. A Koblitz curve has the form

$$y^2 + xy = x^3 + ax^2 + 1$$

with $a = 0$ or 1.

two NIST Koblitz curves

in binary Galois fields

K163

```
p(t) = t^163 + t^7 + t^6 + t^3 + 1
     = 800000000000000000000000000000000000000C9
a    = 1
G_x  = 2fe13c0537bbc11acaa07d793de4e6d5e5c94eee8
G_y  = 289070fb05d38ff58321f2e800536d538ccdaa3d9
n    = 5846006549323611672814741753598448348329118574063
h    = 2
```

K233

```
p(t) = t^233 + t^74 + 1
     = 20000000000000000000000000000000000000004000000000000000001
a    = 0
G_x  = 17232ba853a7e731af129f22ff4149563a419c26bf50a4c9d6eefad6126
G_y  = 1db537dece819b7f70f555a67c427a8cd9bf18aeb9b56e0c11056fae6a3
n    = 3450873173395281893717377931138512760570940988862252126328087024741
h    = 4
```

STANDARDS FOR EFFICIENT CRYPTOGRAPHY

SEC 2: Recommended Elliptic Curve Domain Parameters

Certicom Research

Contact: Daniel R. L. Brown (dbrown@certicom.com)

January 27, 2010
Version 2.0

E - masking plaintext

Menezes-Vanstone Elliptic Curve Cryptosystem

Let E be an elliptic curve defined over $\mathbb{Z}_p$ ($p > 3$ prime) such that E contains a cyclic subgroup H in which the discrete log problem is intractible.

Let $\mathcal{P} = \mathbb{Z}_p{}^* \times \mathbb{Z}_p{}^*$, $\mathcal{C} = E \times \mathbb{Z}_p{}^* \times \mathbb{Z}_p{}^*$, and define

$$\mathcal{K} = \{(E, \alpha, a, \beta) : \beta = a\alpha\},$$

where $\alpha \in E$. The values α and β are public, and a is secret.

For $K = (E, \alpha, a, \beta)$, for a (secret) random number $k \in \mathbb{Z}_{|H|}$, and for $x = (x_1, x_2) \in \mathbb{Z}_p{}^* \times \mathbb{Z}_p{}^*$, define

$$e_K(x, k) = (y_0, y_1, y_2),$$

where

$$\begin{aligned} y_0 &= k\alpha, \\ (c_1, c_2) &= k\beta, \\ y_1 &= c_1 x_1 \bmod p, \quad \text{and} \\ y_2 &= c_2 x_2 \bmod p. \end{aligned}$$

For a ciphertext $y = (y_0, y_1, y_2)$, define

$$d_K(y) = (y_1 c_1{}^{-1} \bmod p, y_2 c_2{}^{-1} \bmod p),$$

where

$$a y_0 = (c_1, c_2).$$

problems found

EC Integrated encryption scheme

↓

Cryptosystem 6.2: *Simplified ECIES*

Let E be an elliptic curve defined over $\mathbb{Z}_p$ ($p > 3$ prime) such that E contains a cyclic subgroup $H = \langle P \rangle$ of prime order n in which the Discrete Logarithm problem is infeasible.
Let $\mathcal{P} = \mathbb{Z}_p{}^*$, $\mathcal{C} = (\mathbb{Z}_p \times \mathbb{Z}_2) \times \mathbb{Z}_p{}^*$, and define

$$\mathcal{K} = \{(E, P, m, Q, n) : Q = mP\}.$$

The values P, Q and n are the public key, and $m \in \mathbb{Z}_n{}^*$ is the private key.

For $K = (E, P, m, Q, n)$, for a (secret) random number $k \in \mathbb{Z}_n{}^*$, and for $x \in \mathbb{Z}_p{}^*$, define

$$e_K(x, k) = (\text{POINTCOMPRESS}(kP), x x_0 \bmod p),$$

where $kQ = (x_0, y_0)$ and $x_0 \neq 0$.

For a ciphertext $y = (y_1, y_2)$, where $y_1 \in \mathbb{Z}_p \times \mathbb{Z}_2$ and $y_2 \in \mathbb{Z}_p{}^*$, define

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

where

$$(x_0, y_0) = m\,\text{POINTDECOMPRESS}(y_1).$$

Diffie-Hellman key exchange. Users A and B want to share a common key. Using a publicly known curve E and point P they do the following. User A choose a number t_A and sends the point $Q = t_A P$ to user B. User B chooses a number t_B and sends $R = t_B P$ to user A. User A then computes the key $K = t_A R = t_A(t_B P) =$ $(t_A t_B)P$. User B can also compute the key K from $t_B Q = t_B(t_A P) = (t_A t_B)P$.

ECDH

Bit Security of Discrete Logarithms

Problem 6.2:	Discrete Logarithm ith Bit
Instance:	$I = (p, \alpha, \beta, i)$, where p is prime, $\alpha \in \mathbb{Z}_p{}^*$ is a primitive element, $\beta \in \mathbb{Z}_p{}^*$, and i is an integer such that $1 \le i \le \lceil \log_2(p-1) \rceil$.
Question:	Compute $L_i(\beta)$, which (for the specified α and p) denotes the ith least significant bit in the binary representation of $\log_\alpha \beta$.

$$QR(p) = \{x^2 \bmod p : x \in \mathbb{Z}_p{}^*\}.$$

$$|QR(p)| = \frac{p-1}{2}.$$

$$QR(p) = \{\alpha^{2i} \bmod p : 0 \le i \le (p-3)/2\}.$$

$$L_1(\beta) = \begin{cases} 0 & \text{if } \beta^{(p-1)/2} \equiv 1 \pmod{p} \\ 1 & \text{otherwise.} \end{cases}$$

$$\log_\alpha \beta = \sum_{i \geq 0} x_i 2^i .$$

Algorithm 6.6: $\mathrm{L_2ORACLEDISCRETELOGARITHM}(p, \alpha, \beta)$

external $\mathrm{L_1}$, $\mathrm{ORACLEL_2}$
$x_0 \leftarrow \mathrm{L_1}(\beta)$
$\beta \leftarrow \beta/\alpha^{x_0} \bmod p$
$i \leftarrow 1$
while $\beta \neq 1$
do $\left\{ \begin{array}{l} x_i \leftarrow \mathrm{ORACLEL_2}(\beta) \\ \gamma \leftarrow \beta^{(p+1)/4} \bmod p \\ \textbf{if } \mathrm{L_1}(\gamma) = x_i \\ \quad \textbf{then } \beta \leftarrow \gamma \\ \quad \textbf{else } \beta \leftarrow p - \gamma \\ \beta \leftarrow \beta/\alpha^{x_i} \bmod p \\ i \leftarrow i + 1 \end{array} \right.$
return $(x_{i-1}, x_{i-2}, \ldots, x_0)$