Generation of cryptographically secure elliptic curves over prime fields

Rafał Gliwa, Janusz Szmidt, Robert Wicik
Military Communication Institute,
05-130 Zegrze Poludniowe, Poland,
{r.gliwa, j.szmidt, r.wicik}@wil.waw.pl

Abstract. Elliptic curves over finite fields are applied to construct public key cryptosystems and to realize a digital signature. The security of these systems is based on computational intractability of the discrete logarithm problem in the group of points on an elliptic curve over a finite field. Elliptic curve cryptosystems provide security comparable to that of the RSA cryptosystem but with cryptographic keys of smaller size. This note presents conditions which cryptographically secure elliptic curves over prime fields have to satisfy and methods to generate such curves.

Keywords: Public key cryptography, elliptic curves over prime fields, security conditions, generation of elliptic curves, probabilistic analysis.

1. Introduction

Elliptic curves over finite fields have applications in public key cryptography for key agreement, encryption, digital signatures and pseudo-random generators. The security of the corresponding cryptosystems is based on intractability of the elliptic curve discrete logarithm problem (ECDLP) in the group of points on an elliptic curve over a finite field. One of the main benefits of Elliptic Curve Cryptography (ECC) in comparison with non-ECC cryptography is the same level of security provided by keys of smaller size. For example, a 256-bit ECC public key provides comparable security to a 3072-bit RSA public key. The U.S. National Institute of Standards and Technology (NIST) has endorsed ECC in the set of recommended algorithms, specifically Elliptic Curve Diffie-Hellman (ECDH) algorithm for key exchange and Elliptic Curve Digital Signature Algorithm (ECDSA) for signatures. The U.S. National Security Agency (NSA) allows their use for protecting information classified up to secret with 384-bit EC keys.

This note presents the results of searching for elliptic curves over finite prime fields F_p satisfying the security conditions of the Brainpool Standard [3]. We have conducted numerical experiments showing how many cryptographically secure curves one can choose from a given set of random elliptic curves over a fixed prime field F_p. Two sets of curves are considered: these generated from seeds coming from expansion of π and e, as in the Brainpool Standard, and those generated from random seeds obtained by our random number generator [5]. We also provide one 384-bit elliptic curve generated from a random seed which satisfies all security and implementation requirements of the Brainpool Standard [3].

2. Basic definitions

Let $p > 3$ be a prime number and let $\mathrm{F_p}$ or *GF(p)* denote the finite field of p elements

$$\mathrm{F_p} = \{0, 1, \dots, \mathrm{p} - 1\}$$

with addition and multiplication modulo p. An elliptic curve over the field $\mathrm{F_p}$ is the set of solutions of an equation

$$E: \; y^2 = x^3 + Ax + B \; mod \; p \qquad (1)$$

together with a "point at infinity" O, where the coefficients $A, B \in F_p$ satisfy

$$\Delta = 4A^3 + 27B^2 \neq 0 \; mod \; p \, . \qquad (2)$$

This set forms an abelian group with neutral element O and the addition law given for example in [4]. The group $E(F_p)$ of points of the elliptic curve (1) defined over the finite field F_p has order $\#E\{F_p\}$ which satisfies the Hasse inequality

$$p + 1 - 2\sqrt{p} \; \leq \#E(F_p) \leq \; p + 1 + 2\sqrt{p} \, .$$

The exact value of $\#E(F_p)$ can be calculated using the SEA–algorithm whose optimized implementation is available in the package Magma [6].

3. Criteria for elliptic curves over prime fields

We consider here only elliptic curves over finite fields $\mathrm{F_p}$, where *p* is a prime number of suitable size in bits. These curves must satisfy the suitable conditions to ensure resistance against known attacks on ECDLP which are the main security requirements. We have searched for cryptographically secure elliptic curves over prime fields F_p according to the ECC Brainpool Standard [3]. We have considered the bit lengths 160, 192, 224, 256 and 384 of the basic primes p which were generated according to the algorithm given in [3] from the Brainpool seeds which were chunks of the hexadecimal representation of the number π^{1120} and from our seeds which were produced by a random number generator. The elliptic curves were chosen according to the following conditions.

C1. The group order $q = \#E(F_p)$ of an elliptic curve is a prime number in order to prevent a small-subgroup attack [4]. Every non-identity point on such a curve is a generator of the group of points on this curve. The curves with prime group order have no points of order 2 and therefore no points with *y*-coordinate 0.

C2. Assuming the inequality $q < p$ one avoids overruns in implementation since in some cases even the bit-length of q can exceed the bit-length of p. Elliptic curves with $q = p$ are called trace one curves or anomalous curves. Satoh and Araki [6] proposed an efficient solution to the ECDLP on trace one curves.

C3. Immunity to attacks using the Weilpairing or Tatepairing. These attacks allow the embedding of the elliptic group $E(F_p)$ into the group of units of an extension $GF(p^l)$ of degree l of the field F_p, where subexponential attacks on DLP exist. Here we have $l = \min\{t: q \mid p^t - 1\}$, i.e. l is the order of p modulo q. The requirement is that *(q-1)/l < 100;* this means that l is close to the maximal possible value. This requirement also excludes supersingular curves.

C4. We define the number u from the equation $q = p + 1 + u$ and the number $d = (4p - u^2)/v^2$, where $v = \max\{a: a^2 \mid 4p - u^2\}$, i.e., d is the square-free part of $4p - u^2$. Let $K = \sqrt{-d}$ be an imaginary quadratic number field. One of the

requirements put in [1] is that $d > 2^{100}$ which means that the discriminant of the field K is sufficiently large.

C5. The ring of isogenies of an elliptic curve over the finite field (an isogeny is a transformation between elliptic curves which preserves the orders of groups of points on the curves) is isomorphic to a lattice in the ring of integers in the quadratic field $K = \sqrt{-d}$. The requirement formulated in the Brainpool Standard [3] is that the class number of the field K is greater than 10 000 000. It is time consuming to calculate the exact value of this class number, so the Standard [3] proposes an algorithm to assert the required inequality.

The requirements C4 and C5 are sophisticated, they are related to possible improvements of the algorithms solving ECDLP.

4. Experimental results

The following experiments have been carried out. We have generated random elliptic curves for fixed length of the base prime p according to the algorithm of Brainpool Standard [3] and checked first condition C1 when collecting 100 or 1000 such curves. The second columns of the tables give the total numbers of the curves checked. The next columns of the tables give the numbers of curves satisfying the indicated conditions or times of calculation. Tables 1÷4 concern the elliptic curves generated from the Brainpool seeds which were the numbers π and e, and Tables 5÷8 the curves obtained from seeds generated by a random number generator [6]. All computations have been done in Magma [7].

Table 1. The numbers of elliptic curves satisfying the security conditions (the curves generated from Brainpool seeds).

	All curves	C1	C1+C2	C1+C2+C3 +C4 +C5
160 bit	19604	100	51	51
192 bit	24432	100	57	57
224 bit	22227	100	51	51
256 bit	36349	100	45	44

Table 2. The times of generation (in minutes) of the curves satisfying the given conditions.

	All curves	C1	C1+C2+C3 +C4 +C5
160 bit	19604	29	66
192 bit	24432	64	109
224 bit	22227	120	173
256 bit	36349	246	297

Table 3. The numbers of elliptic curves satisfying the security conditions (the curves generated from Brainpool seeds).

	All curves	C1	C1+C2	C1+C2+C3+C4+C5
160 bit	206021	1000	502	498
192 bit	229840	1000	501	493
224 bit	264196	1000	487	479
256 bit	356036	1000	524	514

Table 4. The times of generation (in hours) of the curves satisfying the given conditions.

	All curves	C1	C1+C2+C3+C4+C5
160 bit	206021	5.46	9.12
192 bit	229840	11.68	14.00
224 bit	264196	25.47	27.84
256 bit	356036	39.81	47.75

Table 5. The numbers of elliptic curves satisfying the security conditions (the curves generated from random seeds).

	All curves	C1	C1+C2	C1+C2+C3+C4+C5
160 bit	22693	100	52	52
192 bit	32851	100	46	46
224 bit	25684	100	57	56
256 bit	42211	100	47	47

Table 6. The times of generation (in minutes) of the curves satisfying the given conditions.

	All curves	C1	C1+C2+C3+C4+C5
160 bit	22693	35	63
192 bit	32851	77	96
224 bit	25684	127	151
256 bit	42211	255	327

Table 7. The numbers of elliptic curves satisfying the security conditions (the curves generated from random seeds).

	All curves	C1	C1+C2	C1+C2+C3+C4+C5
160 bit	253914	1000	512	503
192 bit	310477	1000	503	500
224 bit	290629	1000	528	523
256 bit	396279	1000	503	496

Table 8. The times of generation (in hours) of the curves satisfying the given conditions.

	All curves	C1	C1+C2+C3+C4+C5
160 bit	253914	5.5	9.45
192 bit	310477	9.87	14
224 bit	290629	18	23.5
256 bit	396279	32.64	44.69

To end this note we present an elliptic curve over the prime field F_p, where p is a randomly generated 384-bit prime number, A, B are coefficients of the curve, $P = (x\_0, y\_0)$ is a randomly chosen point on this curve, q the order of the group of points on the curve, l the value from the condition C4 and d the value from condition C5. This 384-bit elliptic curve satisfies all security conditions of the Brainpool Standard [3].

```
p = 0x85552AE413E218FE96407A08D375AB7122EFE40643672D7803BB9E
729E6C9F117815B2B0CC058F986CB31DACB9144FEB,
seed_random = 0x629991099D8BA5241BF1601E3A7EFA3F16992B48,
A = 0x7E650093A9E415324F3879F5EC1F9A89C21F89701B9F117C4D33FB5C2A
50D2EC647EE715E5BC0C63C5FEBF84DC7F8AC,
B = 0x3C5FEBF84DC7F8ACC5B466EF0C1E9C2F135E17980B3BEEDABAB4A7550D
0DDD684928AD038BEEEC841CA26F18727E243F,
x_0 = 0x8405BD74DAB1F3B41658E114F29F78B28E3EF60AAAD118D4A534
5BC3320209945F5D23049BB570F4EF053F07FBBD5287,
y_0 = 0x35F0E509942B5D426682BDC7ABE0EB48CADBD544ECCA71F0C40C1
85955CBFE62D5F0EC84862D062695FD2AE0B0B8793,
q = 0x85552AE413E218FE96407A08D375AB7122EFE40643672D77E0FDE8B5
BF6D64F7C0E58B569CD741BFCE2AB46E804B51BF,
l = 20521778549493474636638036249022264275456139090355823129913
34330680481753266456663261713331143443935256076486796456 30,
(q-1)/l = 1,
d = 36160652838424930600992835554615416999016999837769459449443
97066220338729016232123870880497260049054870374085051 23.
Time of generation: 5343,6 seconds.
```

BIBLIOGRAPHY

1. D. J. Bernstein, T. Lange, SafeCurves: choosing safe curves for elliptic-curve cryptography, http://safecurves.cr.yp.to
2. D.J. Bernstein, Tung Chou, Ch. Chuengsatiansup, A. Huelsing, T. Lange, R. Niederhagen, Ch. Van Vredendaal. How to manipulate curve standards: a white paper for the black hat. Cryptology ePrint Archive, 2014/571, www.iacr.org
3. ECC Brainpool. ECC Brainpool Standard Curves and Curve generation, 2005. http://www.ecc-brainpool.org/download/Domain-parameters.pdf
4. D. Hankerson, A. Menezes, S. Vanstone, Guide to Elliptic Curve Cryptography. Springer 2004, ISBN 0-387-95273-X.
5. M. Leśniewicz. Sprzętowa generacja losowych ciągów binarnych. Wydawca: Wojskowa Akademia Techniczna, Warszawa 2009, ISBN 978-83-61486-31-2.
6. T. Satoh, K. Araki. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. Comm. Math. Univ. Sancti Pauli, 47, pp. 81-92, 1998.
7. Magma Computational Algebra System, www.magma.math.usyd.edu.au