www.czasopisma.pan.pl PAN www.journals.pan.pl

jet INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2021, VOL. 67, NO. 2, PP. 207–212
Manuscript received February 9, 2020; revised April, 2021. DOI: 10.24425/ijet.2021.135966

Cryptographically Strong Elliptic Curves of Prime Order

Marcin Barański, Rafał Gliwa, and Janusz Szmidt

***Abstract*—The purpose of this paper is to generate cryptographically strong elliptic curves over prime fields $\mathbb{F}_p$, where p is a Mersenne prime, one of the special primes or a random prime. We search for elliptic curves which orders are also prime numbers. The cryptographically strong elliptic curves are those for which the discrete logarithm problem is computationally hard. The required mathematical conditions are formulated in terms of parameters characterizing the elliptic curves. We present an algorithm to generate such curves. Examples of elliptic curves of prime order are generated with Magma.**

***Keywords*—Mersenne primes, elliptic curves, security requirements, search algorithm, Magma.**

I. Introduction

Information security is of paramount importance to many institutions of our society: governments, military, financial, businesses, etc. Many confidential information about research, products, financial status, customers, or employees, is nowadays processed and stored on computers, or transmitted to other computers. The information security is a very important area of radio communication. TRANSEC and COMSEC mechanisms require symmetric cryptography and public key cryptography.

The public key cryptography was introduced in the seminal papers of Diffie and Hellman [6] and Rivest, Shamir and Adleman [21]. The use of elliptic curves over finite fields was proposed by Miller [18] and Koblitz [17]. The cryptosystems with elliptic curves have advantage over RSA cryptosystem since we obtain the comparable security with much shorter keys, as is shown in Table 1.

The purpose of this paper is to generate cryptographically strong elliptic curves over prime fields $\mathbb{F}_p$, where p is a Mersenne prime $p = 2^{521} - 1$ or $p = 2^{607} - 1$, one of the special primes or a random prime. We search for elliptic curves whose order is a prime number and the order of the twisted curve has a big prime factor. The sizes of the keys in the region from 384 bits up to 521 bits fit in the suites A and B of the NATO standard [27]. Examples of elliptic curves with orders in this region are given in the Standards [26]–[29]. Our purpose is to generate independently elliptic curves with good cryptographic properties. We use the mathematical tool Magma [31] to generate the curves.

Cryptographically strong elliptic curves are those for which the *Elliptic Curve Discrete Logarithm Problem (ECDLP)* is resistant to known attacks. In general, *ECDLP* is computationally hard problem. The corresponding requirements are formulated in terms of parameters characterizing the curves. The elliptic curves with prime order have advantage over non-prime case since each non-neutral element of the curve is a generator of the group of points on the curve. The arithmetic on elliptic curves is presented in papers [5], [12] and over the field $\mathbb{F}_{2^{521}-1}$ in [13]. We check the class number criterion and the twist security for our examples of curves. The results of the numerical experiments are given in the Appendix.

Table 1. The sizes of cryptographic keys.

Symmetric alg.	80	112	128	160	256
ECC order q	160	224	256	320	512
RSA modulus n	1024	2048	3072	7680	15360

II. Basic Notions

Let p be a prime number. The prime field $\mathbb{F}_p$ consists of integers $\{0, 1, \ldots, p-1\}$ with arithmetic operations of addition and multiplication modulo p. For a prime $p > 3$ we define an elliptic curve E over the field $\mathbb{F}_p$ by the equation

$$y^2 = x^3 + ax + b, \tag{1}$$

where $a, b \in \mathbb{F}_p$ and $4a^3 + 27b^2 \neq 0$. We define the set of rational points of the elliptic curve E over the field $\mathbb{F}_p$ as the set $E(\mathbb{F}_p)$ of solutions $(x, y) \in \mathbb{F}_p \times \mathbb{F}_p$ of the equation (1) together with the neutral element $\mathcal{O}$. The set $E(\mathbb{F}_p)$ has the structure of an abelian group with operations of addition and doubling of points defined according to the rules, see [4], [25]. Addition of points:

1) $P + \mathcal{O} = \mathcal{O} + P = P$ for all $P \in E(\mathbb{F}_p)$.
2) If $P = (x, y) \in E(\mathbb{F}_p)$, then $(x, y) + (x, -y) = \mathcal{O}$. The point $(x, -y)$ is denoted $-P$ and it is called the negation of P. Let us note that the point $-P$ is on the curve E.
3) Let $P = (x_1, y_1) \in E(\mathbb{F}_p)$ and $Q = (x_2, y_2) \in E(\mathbb{F}_p)$, where $P \neq \pm Q$. Then $P + Q = (x_3, y_3)$, where

$$x_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)^2 - x_1 - x_2$$

and

$$y_3 = \left(\frac{y_2 - y_1}{x_2 - x_1}\right)(x_1 - x_3) - y_1.$$

Doubling of a point:

M. Barański, R. Gliwa, and J. Szmidt are with Military Communication Institute, National Research Institute, Warszawska 22A, 05-130 Zegrze, Poland (e-mail: m.baranski, r.gliwa, j.szmidt@wil.waw.pl).

Let $P = (x_1, y_1) \in E(\mathbb{F}_p)$, where $P \neq -P$. Then $2P = (x_3, y_3)$, where

$$x_3 = \left(\frac{3x_1^2 + a}{2y_1}\right)^2 - 2x_1$$

and

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

The operations of adding and doubling points on an elliptic curve $E(\mathbb{F}_p)$ require performing arithmetic operations of addition and multiplication in the basic field $\mathbb{F}_p$. For elliptic curves over the real field $\mathbb{R}$ the operations of addition and doubling of points on the curve have an geometric interpretation which is shown on Figures 1 and 2. An example of the elliptic curve over the finite field $\mathbb{F}_{23}$ is depicted in Figure 3.

Let E be an elliptic curve over the field $\mathbb{F}_p$. The order of the curve denoted $q = \#E(\mathbb{F}_p)$ is the order of the group $E(\mathbb{F}_p)$. In this case the Hasse Theorem says that

$$\#E(\mathbb{F}_p) = p + 1 - u, \qquad \text{where} \quad |u| < 2\sqrt{p}.$$

The integer u is called the trace of the curve. The group $E(\mathbb{F}_p)$ has the structure of an abelian group of rank 1 or 2, i.e., it is isomorphic to the group $\mathbb{Z}_{n_1}$ or to the group $\mathbb{Z}_{n_1} \times \mathbb{Z}_{n_2}$, where $n_2 | n_1$. Here $\mathbb{Z}_{n_1}$ and $\mathbb{Z}_{n_2}$ are cyclic groups. If $n_2 = 1$, then it is a point $P \in E(\mathbb{F}_p)$, named the generator of the group, which satisfies

$$E(\mathbb{F}_p) = \{kP : 0 \leqslant k \leqslant n_1 - 1\},$$

where $kP = P + \cdots + P$, k times. In the special case, when the order q is a prime number, the group $E(\mathbb{F}_p)$ is a simple one and each non-neutral element is a generator of this group. It is important in cryptographic applications. In our investigations we consider only prime order elliptic curves over prime fields $\mathbb{F}_p$. In particular, we take primes of the form $p = 2^{p'} - 1$, where p' is another prime which are called Mersenne primes and it has been found together 51 of them up to now [30].

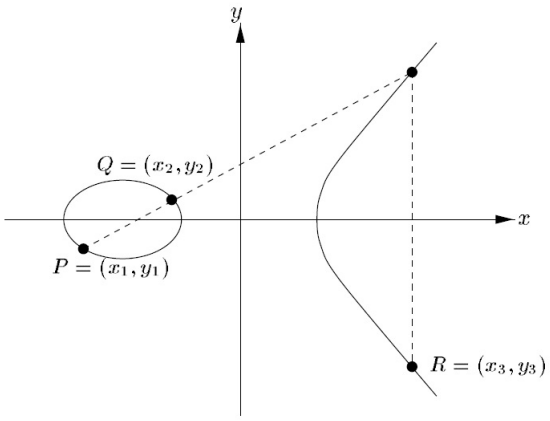

Fig. 1. Adding of points on the elliptic curve.

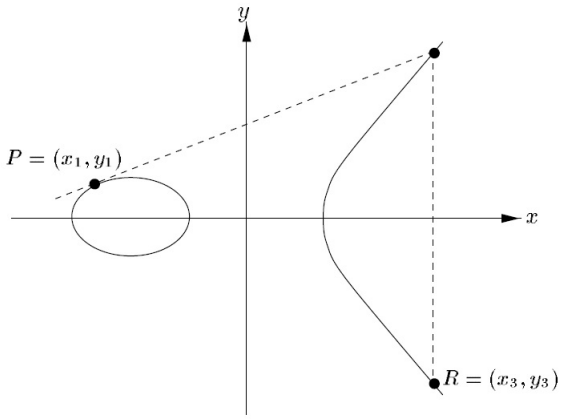

Fig. 2. Doubling of a point on the elliptic curve.

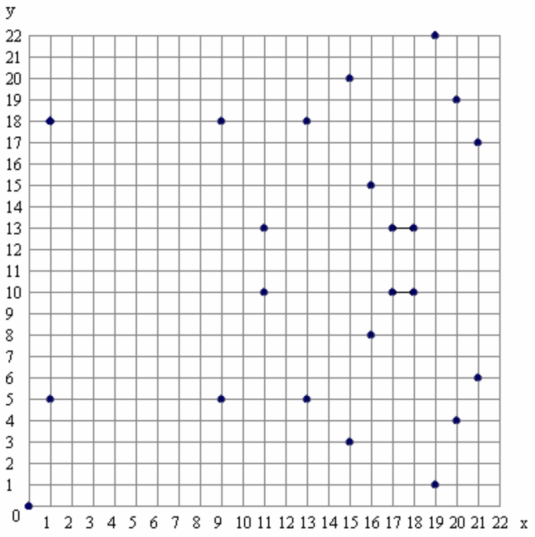

Fig. 3. Elliptic curve $y^2 = x^3 + x$ over prime field $\mathbb{F}_{23}$.

Let E be an elliptic curve over the prime field $\mathbb{F}_p$ and P_0 a point in $E(\mathbb{F}_p)$. We denote $\langle P_0 \rangle$ the subgroup of $E(\mathbb{F}_p)$ generated by the point P_0. Let $Q \in \langle P_0 \rangle$ be an arbitrary point. Then it has the form $Q = kP_0$ for some integer k. We call k the discrete logarithm of Q to the base P_0. In this case the *Elliptic Curve Discrete Logarithm Problem (ECDLP)* is to find the number k. The best known algorithm to solve *ECDLP*, the Pollard *rho* method [20], finds discrete logarithms in time $O(\sqrt{p})$. Hence for large values of the prime p it is infeasible by today's computers. The record of calculating discrete logarithm on elliptic curves over prime fields is for 113-bit prime [10].

III. The Cryptographic Requirements

To resist the known attacks on *ECDLP* the elliptic curves have to satisfy special requirements. We follow partly the ECC Brainpool Standard [26]. The NATO [27] and the BSI [26] requirements for cryptographically strong elliptic curves are in fact the Brainpool conditions. In our process of generation we use the function *CryptographicCurve* from Magma [31].

1) The prime p is of the size at least 2^{256} to resist the attack with Pollard *rho* method [20]. We generate elliptic

curves over prime fields $\mathbb{F}_p$, where p is a prime of the special form or a random prime.

2) Immunity to attacks using the Weil-pairing or Tate-pairing [2]. The attacks allow the embedding of the cyclic subgroup of $E(\mathbb{F}_p)$ into the group of units of a degree-l extension $\mathbb{F}_{p^l}$ of $\mathbb{F}_p$, where subexponential attacks on the *Discrete Logarithm Problem* in finite fields exist [14], [16], [19]. We have $l = min\{t : q \text{ divides } q^t - 1\}$, i.e. l is the *order* of p *mod* q. The requirement is that the quotient $(q-1)/l < 100$.
3) The curves are not trace one curves. Trace one curves (or anomalous curves) are those for which $\#E(\mathbb{F}_p) = p$. According to the works [22]–[24] for anomalous curves there are efficient algorithms to solve *ECDLP*.
4) The group order $q = \#E(\mathbb{F}_p)$ must be a prime number or has a sufficiently big prime factor in order to avoid small group attack [19].
5) The class number of the maximal order of the endomorphism ring of the elliptic curve E is larger than 10000000 [26], [27].

We search for elliptic curves over prime fields whose orders are prime numbers. This is the most consuming time of the search algorithm. In [11] it was stated a conjecture about the probability Q_1 that randomly chosen elliptic curve over a prime field $\mathbb{F}_p$ is a prime. The conjecture states that

$$Q_1 \sim c_p P_1 \sim \frac{c_p}{\log p} \quad \text{as} \quad p \to \infty, \tag{2}$$

where c_p is explicitly computed constant, P_1 is the probability that a randomly chosen integer from the Hasse interval $[p+1-2\sqrt{p}, p+1+2\sqrt{p}]$ is a prime and $\sim$ means asymptotically equal. The numerical values of the constant satisfy $0.44 < c_p < 0.64$ and usually it is closer to 0.44. The conjecture was experimentally confirmed for values of p up to 10^9. The authors of [11] put a similar conjecture for the probability Q_k that a randomly chosen elliptic curve has the order equal to kq, where k is a small integer and q is a prime.

In fact, there are no attacks directly related to the class number criterion. It was stated in view of the future development of the theory of elliptic curves [8], [9]. Let us describe this condition in more details. The ring of endomorphism $End(E)$ is isomorphic to an order in an imaginary quadratic field $K = Q(\sqrt{-d})$ with square-free $d \in \mathbb{N}$. Let $\#E(\mathbb{F}_p) = p+1+u$, then d can be computed as the square-free part of $4p-u^2$ and then the factorization of that number is necessary to calculate the class number of the field K. If $v = \max\{a : a^2 \text{ divides } 4p-u^2\}$ then $d = (4p-u^2)/v^2$. The complexity of the best known algorithm for explicitly determining the class number of K is too high in practice, hence one tries to find elements of the ideal class group of K with a large order, as the class number is not smaller than the order of an element.

We present the methods to compute class numbers and finding elements in the ideal class group with a large order. We follow the descriptions in [26] and [3]. There is a bijection between the class group of binary quadratic forms with discriminant $d_K < 0$ and the ideal class group of the order with discriminant d_K. For this purpose we represent binary quadratic forms $ax^2 + bxy + cy^2$ as triples (a, b, c). Then a triple (a, b, c) of integers is called a *positive definite primitive reduced binary quadratic form of discriminat* d_K if:

- $gcd(a, b, c) = 1$, the form is *primitive*,
- $a > 0$ and $|b| \leqslant a \leqslant c$ and if $a = c$ or $|b| = a$ than also $b \geqslant 0$, the form is *reduced*,
- $b^2 - 4ac = d_K$, the form is *positive definite*.

We have the relation:

$$d_K = \begin{cases} -d & \text{if } -d = 1 \ mod\ 4, \\ -4d & \text{if } -d = 2 \ mod\ 4 \text{ or } -d = 3 \ mod\ 4. \end{cases}$$

The elements of the ideal class group of the number field $K = \sqrt{-d}$ with discriminat d_K correspond bijectively to the primitive reduced quadratic forms of discriminant d_K. The group law in this set is defined as follows:

- For two primitive reduced quadratic forms (a_1, b_1, c_1) and (a_2, b_2, c_2) their composition (a_3, b_3, c_3) can be calculated by Algorithm 5.4.7 in Cohen's book [3];
- The form (a_3, b_3, c_3) is primitive and has discriminant d_K but it is not necessarily reduced. The reduction Algorithm 5.4.2 of [3] applied to (a_3, b_3, c_3) outputs a primitive reduced quadratic form (a, b, c) with discriminant d_K;
- We denote the multiplication of quadratic forms by $\bullet$, i.e.

$$(a, b, c) = (a_1, b_1, c_1) \bullet (a_2, b_2, c_2).$$

The neutral element I is represented by the triple $(1, 0, -d_K/4)$ if $d_K = 0 \ mod\ 4$ and it is represented by $(1, 1, (1-d_K)/4)$ if $d_K = 1 \ mod\ 4$. The Algorithm of [26] determines whether an element of the ideal class group of the number field $K = Q(\sqrt{-d})$ has an order of at least a value *MinClass*.

Algorithm ([26])
Input: A primitive reduced quadratic form (a, b, c) of discriminant d_K.
Output: "*true*" if the order of the corresponding element of the ideal class group is at least *MinClass*; and "*false*" otherwise.

1) Set $t = I$.
2) for i from 1 to *MinClass - 1* do
   Set $t = t \bullet (a, b, c)$.
   If $t = I$ then output "*false*" and stop.
3) Output "*true*".

Code 3 is an implementation of the Algorithm in Magma.

IV. Twist Security

Let E be an elliptic curve $E : y^2 = x^3 + aX + b$ over the base field $\mathbb{F}_p$. The order of the curve is $\#E(\mathbb{F}_p) = p+1+u$. The twisted curves are elements of the $\mathbb{F}_p$-isomorphism class of curves $E^{tw} : x^3 + at^2x + bt^3$ with $\#E^{tw}(\mathbb{F}_p) = p+1-u$, where $t \in \mathbb{F}_p$ is square-free. The curve E is called *twist-secure* ([1], [15]) if both E and E^{tw} are cryptographically strong. As a minimum both $\#E(\mathbb{F}_p)$ and $\#E^{tw}(\mathbb{F}_p)$ have to be almost prime, i.e., have only small prime factors and one big prime factor.

The conjecture formulated in formula (2) implies that the probability Q_1^{tw} that a randomly chosen elliptic curve over a prime field $\mathbb{F}_p$ is both secure and twist-secure seems to satisfy

$$\frac{0.5}{\log^2 p} < Q_1^{tw} < \frac{5}{\log^2 p}.$$

In [7] it was analysed the twist security of elliptic curves. It is believed that the use of twist-security helps to improve security in the following situations:

- If only x-coordinates of points are used than an expanded x-coordinate could lead to a point in $E^{tw}(\mathbb{F}_p)$ instead of $E(\mathbb{F}_p)$. If E^{tw} is not cryptographically strong and if Q has smooth order on E^{tw} an attacker might take advantage of this situation by providing x-coordinates of points that lie on E^{tw}. This method is called '*invalid* x'.
- Let P_0 be a point on E and d an integer. The attack goes in the way that during the computation of dP_0 a fault is introduced that leads to computations on E^{tw} instead of E. Then the same ideas as for '*invalid* x' apply. For simplicity one can assume that the fault is injected immediately at the beginning of the computations.

Authors of [15] and [7] argued that even for twist secure curves a point validation has to be performed. Assume that a cryptographic mechanism computes dP_0 for a secret scalar d. In the applications the computation of dP_0 is performed by using a blinded value $d + r_i q$ with randomly chosen $r_i \in \{1, \ldots, 2^R\}$, where R is a system parameter. Blinding is a widely used counter method to thwart side-channel attacks on implementations that can be accessed by an attacker. The authors of [15] state that the value of R must be sufficiently big and its length depends on the structure of the underlying base field. For random p 64-bit R is sufficient and for special primes p it has to be $\log_2(\sqrt{q})$ bits of length.

V. The Numerical Experiments

In Appendix there are examples of cryptographically strong elliptic curves of prime order over base fields $\mathbb{F}_p$, where p is a special prime or a random prime of the size from 256 bits to 607 bits. The time of finding the curves increases as the size of p grows. Only for the 256-bit prime it was possible to explicitly calculate the class number of the corresponding quadratic number field. In other cases we checked by running Code 3 the class number criterion. For the 607-bit prime it was not possible to check the class number criterion. The time of checking the criterion is random. We have calculated the order $l = Modorder(p, q)$ of the embedding of the field $\mathbb{F}_p$ into the extended field $\mathbb{F}_{p^l}$. The factorization of the orders of the twisted curves and the factorization of $q - 1$ have been done and the sizes of the biggest factors have been given. All calculations have been done on macOS Catalina with 3.6 GHz Intel Core i7 processor. The results of checking the security criteria are shown in Tables 2 and 3.

VI. Conclusions

In classified applications we need elliptic curves with secret parameters which are independent from the examples of the curves given in the commercial standards. We have generated cryptographically strong elliptic curves over the base field $\mathbb{F}_p$, where p is in a region corresponding to requirements of Suite A and Suite B. The purpose was to show how to search for such curves and it has been achieved.

References

[1] Daniel J. Bernstein and Tanja Lange. SafeCurves: choosing safe curves for elliptic curve cryptography, 2015. http://safecurves.cr.yp.to (accessed 27 September 2015).

[2] I. Blake, G. Serroussi, N. Smart. *Elliptic curves in cryptography*. Cambridge University Press, 1999.

[3] H. Cohen. *A course in computational number theory*. Springer 1983.

[4] H. Cohen, G. Frey. *Handbook of Elliptic and Hyperelliptic Curve Cryptography*. Chapman and Hall CRC, 1994.

[5] P. Dąbrowski, R. Gliwa, J. Szmidt, R. Wicik. *Generation and Implementation of Cryptographically Strong Elliptic Curves*. Number-Theoretical Methods in Cryptology. First International Conference, NuTMiC 2017. Warsaw, Poland, 11-13, 2017. Lecture Notes in Computer Sciences, (Eds), Jerzy Kaczorowski, Josef Piprzyk, Jacek Pomykała. Volume 10737, pages 25-36. 2017.

[6] W. Diffie, M. E. Hellman. *New Directions in Cryptography*. IEEE Trans. Information Theory, IT 22(6), pp. 644-654, 1976.

[7] Jean-Pierre Flori, Jerome Plut, Jean-Rene Reinhard. *Diversity and transparency for ECC*. NIST Workshop on ECC Standards, June 11-12, 2015.

[8] Gerhard Frey, private communication, 2015.

[9] G. Frey, H. Rück. *A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves*. Mathematics of Computations, 62 91994), 865-874.

[10] S. D. Galbraith, P. Gaudry. *Recent progress on the elliptic curve discrete logarithm problem*. Cryptology ePrint Archive, 2015/1022.

[11] Steven D. Galbraith and James McKee. *The probability that the number of points on an elliptic curve over a finite field is prime*. J. London Math. Soc. (2), 62(3):671–684, 2000.

[12] R. Gliwa, J. Szmidt, R. Wicik *Searching for cryptographically secure elliptic curves over prime fields* Science and Military, 2016, nr 1, volume 11, pages 10-13, ISSN 1336-8885 (print), ISSN 2453-7632 (on-line).

[13] R. Granger, M. Scott. *Faster ECC over $\mathbb{F}_{2^{521}-1}$*. In: Katz, J. ed., PKC 2015. LNCS, vol. 9020, pp. 539–553.

[14] D. Johnson, A. Menezes. *The Elliptic Curve Digital Signature Algorithm (ECDSA)*. Technical Report CORR 99-34, University of Waterloo, Canada. http://www.math.uwaterloo.ca

[15] Manfred Lochter and Andreas Wiemers. *Twist insecurity*, 2015. iacr. ePrint Archive 577 (2015).

[16] A. Menezes, T. Okamoto, S. Vanstone. *Reducing elliptic curve logarithms to logarithms in a finite field*. IEEE. Transactions on Information Theory, 39 (1993), 1639-1646.

[17] N. Koblitz. *Elliptic curve cryptosystems*. Math. Comp., 48(177), pp. 203-209, 1987.

[18] V. S. Miller. *Use of elliptic curves in cryptography*. In Advances in Cryptology - CRYPTO'85, LNCS vol 218, pp. 417-426, 1985.

[19] P. Pohlig, M. Hellman. *An improved algorithm for computing logarithms over GF(p) and its cryptographic significance*. IEEE Transaction on Information Theory, 24 (1979), 106-110.

[20] J. Pollard. *Monte Carlo methods for index computations mod p^n*. Mathematics of Computations, 32 (1978), 918-924.

[21] R. L. Rivest, A. Shamir, L. Adleman. *A method for obtaining digital signatures and public-key cryptosystems*. Comm. ACM, 21(2), pp. 120-126, 1978.

[22] T. Satoh, K. Araki. *Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves*, Commentarii Mathematici Universitatis Sancti Pauli, 47 (1998), 81-92.

[23] I. Semaev. *Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p*. Mathematics of Computations, 67 (1998), 353-356.

[24] N. Smart. *The discrete logarithm problem on elliptic curves uf trace one*. Journal of Cryptology, 12 (1999), 193-196.

[25] J. H. Silverman. *The arithmetic of elliptic curves*. Springer 1986.

[26] Elliptic Curve Cryptography (ECC) Brainpool Standard. Curves and Curve Generation, v. 1.0. 2005. Request for Comments: 5639, 2010. 7027, 2013. http://www.bsi.bund.de

[27] Technical and Implementation Guidance on Generation and Application of Elliptic Curves for NATO classified, 2010.

[28] US Department of Commerce. N.I.S.T. 2000. Federal Information Processing Standards Publication 186-2. FIPS 186-2. Digital Signature Standard.

[29] Standards for Efficient Cryptography Group. Recommended elliptic curve domain parameters, 2000. www.secg.org/collateral/sec2.pdf
[30] Mersenne prime. en.wikipedia.org
[31] Magma Computational Algebra System. School of Mathematics and Statistics. University of Sydney.

VII. Appendix

All numerical experiments have been done with Magma.

Code 1: searching for cryptographic curves

Input: a prime number or generation of a prime.
Output: a cryptographic elliptic curve and its prime order.

```
p := 2^521 − 1;
p;
K := GF(p);
repeat
E := CryptographicCurve(K);
n := #E;
until IsPrime(n); E;
n;
IsPrime(n);
```

Code 2: The calculation of the class number

Input: the prime p, q - the order of the elliptic curve over the base field GF(p).
Output: the exact value of the class number and the statement that it is greater than 10000000.

```
function ClassNumberField(p, q)
  u := q − p − 1;
  expr := 4 ∗ p − u^2;
  d := SquarefreeFactorization(expr);
  K := QuadraticField(-d);
  cln := ClassNumber(K);
  printf "ClassNumber(Field) = %o \n\n", cln;
        if cln qt 10000000 then
           wsklk := 1;
        else
           wsklk := 0;
        end if;
  return wsklk;
  end function;
```

Code 3: The estimation of the class number

Input: The prime order p of the base field and the prime order q of the elliptic curve.
Output: The assertion that the class number criterion is satisfied.

```
function ClassNumberEstim(p, q)
  u := q − p − 1;
  expr := 4 ∗ p − u^2;
  d := SquarefreeFactorization(expr);
  v := Sqrt(expr div d);
  qfmod := -d  mod  4;
  if qfmod eq 1 then d := d;
  else d := 4*d;
  end if;
  Q := BinaryQuadraticForms(-d);
  I := Q!1;
  t := 1;
  MinClass := 10000000;
  for iqfb in [1..127] do qfb := iqfb;
      left := qfb^2 + d;
  if  ((left  mod  4) eq 0) than
    qfaqfc := left div 4;
    if not IsPrime(qfaqfc) then
       for iqfa in [2..2011] do qfa := iqfa;
          if  ((qfaqfc  mod  qfa eq 0) then
             qfa := qfaqfc div qfa;
             f := Q!(qfa, qfb, qfc);
             fred := Reduction(f);
             wsklk := 0;
             for i in [2..MinClass-1] do
                  t := t*fred;
               if f eq I then wsklk := 0;
                  break;
               end if;
             end for;
             if wsklk eq 1 then break iqfb;
             end if;
          end if;
       end for;
    end if;
  end if;
  end for;
  return wsklk;
end function;
```

Examples of curves

1. The prime $p = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1$.
Elliptic curve given by the equation (1):
a = 4768150577662800202698453809742175216678268262733178958907974637140862848993 7.
b = 1131166390289526225695301715009927068675675716865870773924486995458890409536 20.

2. The pseudo Mersenne prime: $p = 2^{384} - 317$.
Elliptic curve given by the equation (1):
a = 2965982059611140164585089344874519177962181369481991315132530837109537866295102170802876845297667530314496999891029 7.
b = 5465338922819396922164836973361159430870073578964638778711604720795990218131993476047481072724415943710249970348068.

3. The prime $p = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1$.
Elliptic curve given by the equation (1):
a = 8735070975157951234395606249233766482461152126616266962191813350676349191319319883281968820373517003946662752511691 6.
b = 1831389986083361655499991556610300645390555071040449587450547028011912149864799070978966702680949683434376045781256 8.

4. The pseudo Mersenne prime $p = 2^{512} - 569$.
Elliptic curve E given by the equation (1):
a = 8558853336493438020889377784166399308051413979854505157907881866170672149628131299906277478550402619039743549249977003492712085552439607269155230528747523.

b = 24631984075740246707661438099554333658884733027375907913895074592916382420409687485528740492793786514814411484488929794321354296655834620857383922095941141.

5. The 513-bit random prime:
p = 25594975191524057596686236219057348657910099331523732916082126454085175140683787546459442936770729331018863375179483052900478078771405745122853613344030483.
Elliptic curve given by the equation (1):
a = 86170090287902258558784762993814650290752778847275936909452193386960014424442946273924909219060581332673755953096815411424787959679612842191092869548764 73.
b = 20060163867953952778780809742625076538413176025444795239813351995784590897749493191531274349838167690134908585839679079431541359098321038472216320805180061.

6. The Mersenne prime $p = 2^{521} - 1$.
Elliptic curve given by the equation (1):
a = 26782334858715756244216640798274018517922031219904580576503743268353733071850516395699759892781004324741648715400718429412033586571500320535025320985617197 15.
b = 16404287061903992302811204876815689866087602405133922926790815725246003981834404144401213802827777697335821731796031360927800829783060619035796253240425855 09.

7. The 522-bit random prime:
p = 88513391879763220844904307013057547019597847943336084780044038690768313937873845436700961624789657710065734824182114997169325869169728139759920427255044730 29.
The elliptic curve given by the equation (1):
a = 71145593715887937911003536673585139496317294583018845565720769881264005697069305981874657265322976276356692655238797662449207803664699068242816139532948221 23.
b = 22050386026573591928149963251170049724886710773379224217283018631329220576697211692460937446402220022988077712930507486377542061213793264470727322022200444436.

8. The Mersenne prime $2^{607} - 1$.
The elliptic curve given by the equation (1):
a = 14009390140185218866235593431872186705960339098167377901223402590006396429124623345471332640991683310368924008072222420284331556646232476295017778572299430054673992579332943679995534.
b = 12782482474237879787621910894206732592068546834433768740057542764175016483490329276425978667482873954065514289092296284383259042365027505518819894984268996866124888247396374031704290 1.

Table 2. Properties of generated elliptic curves.

Property	Curve 1	Curve 2	Curve 3	Curve 4
Search time	10 hours	5 hours	2 hours	27 hours
Order q	prime	prime	prime	prime
p	256 bits	384 bits	384 bits	512 bits
q	256 bits	384 bits	385 bits	512 bits
Class #	yes	yes	yes	yes
Twist factor	237 bits	272 bits	195 bits	506 bits
$(q-1)/l$	1	2	3	1
Factor q-1	106 bits	272 bits	382 bits	213 bits

Table 3. Properties of generated elliptic curves.

Property	Curve 5	Curve 6	Curve 7	Curve 8
Search time	13 hours	17 hours	19 hours	3 days
Order q	prime	prime	prime	prime
p	513 bits	521 bits	522 bits	607 bits
q	519 bits	521 bits	522 bits	607 bits
Class #	yes	yes	yes	?
Twist factor	478 bits	236 bits	362 bits	249 bits
$(q-1)/l$	6	8	1	2
Factor q-1	427 bits	249 bits	400 bits	534 bits

Abbreviations:

- Search time: the time to find the elliptic curve.
- Order q: the confirmation that the order of the curve is a prime number.
- The length of the order of the base field.
- The length of the order of the elliptic curve.
- The confirmation that the class number criterion is satisfied, i.e., the class number is > 10000000.
- The length of the biggest factor in the factorization of the order of the twisted elliptic curve.
- The value of $(q-1)/l$, where $l = Modorder(p, q)$. The confirmation that the corresponding criterion is satisfied.
- The length of the biggest factor in the factorization of $q - 1$.