INTERNATIONAL STANDARD

ISO/IEC 15946-5

Second edition
2017-08

Information technology — Security techniques — Cryptographic techniques based on elliptic curves —

Part 5:
Elliptic curve generation

Technologies de l'information — Techniques de sécurité — Techniques cryptographiques fondées sur les courbes elliptiques —

Partie 5: Génération de courbes elliptiques

Reference number
ISO/IEC 15946-5:2017(E)

ISO copyright office
Ch. de Blandonnet 8 • CP 401
CH-1214 Vernier, Geneva, Switzerland
Tel. +41 22 749 01 11
Fax +41 22 749 09 47
copyright@iso.org
www.iso.org

Contents

Page

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation on the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see the following URL: www.iso.org/iso/foreword.html.

This document was prepared by Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques*.

This second edition cancels and replaces the first edition (ISO/IEC 15946-5:2009), which has been technically revised.

It also incorporates the Technical Corrigendum ISO/IEC 15946-5:2009/Cor.1:2012.

The main technical changes between the first edition and this second edition are as follows:

- the terms and definitions given in ISO/IEC 15946-1 are used;
- the scope of verifiably pseudo-random elliptic curve generation has been added;
- the numerical examples in C.4.2 and C.4.3 have been modified.

A list of all parts in the ISO/IEC 15946 series can be found on the ISO website.

Introduction

Some of the most interesting alternatives to the RSA and $F(p)$ based systems are cryptosystems based on elliptic curves defined over finite fields. The concept of an elliptic curve based public-key cryptosystem is rather simple.

- Every elliptic curve over a finite field is endowed with an addition operation "+", under which it forms a finite abelian group.
- The group law on elliptic curves extends in a natural way to a "discrete exponentiation" on the point group of the elliptic curve.
- Based on the discrete exponentiation on an elliptic curve, one can easily derive elliptic curve analogues of the well-known public-key schemes of Diffie-Hellman and ElGamal type.

The security of such a public-key system depends on the difficulty of determining discrete logarithms in the group of points of an elliptic curve. This problem is, with current knowledge, much harder than the factorization of integers or the computation of discrete logarithms in a finite field. Indeed, since Miller and Koblitz independently suggested the use of elliptic curves for public-key cryptographic systems in 1985, the elliptic curve discrete logarithm problem has only been shown to be solvable in certain specific and easily recognizable cases. There has been no substantial progress in finding an efficient method for solving the elliptic curve discrete logarithm problem on arbitrary elliptic curves. Thus, it is possible for elliptic curve based public-key systems to use much shorter parameters than the RSA system or the classical discrete logarithm-based systems that make use of the multiplicative group of a finite field. This yields significantly shorter digital signatures and system parameters.

This document describes elliptic curve generation techniques useful for implementing the elliptic curve based mechanisms defined in ISO/IEC 29192-4, ISO/IEC 9796-3, ISO/IEC 11770-3, ISO/IEC 14888-3, and ISO/IEC 18033-2.

It is the purpose of this document to meet the increasing interest in elliptic curve based public-key technology by describing elliptic curve generation methods to support key-exchange, key-transport and digital signatures based on an elliptic curve.

Information technology — Security techniques — Cryptographic techniques based on elliptic curves —

Part 5: Elliptic curve generation

1 Scope

The ISO/IEC 15946 series specifies public-key cryptographic techniques based on elliptic curves described in ISO/IEC 15946-1.

This document defines elliptic curve generation techniques useful for implementing the elliptic curve based mechanisms defined in ISO/IEC 29192-4, ISO/IEC 9796-3, ISO/IEC 11770-3, ISO/IEC 14888-3 and ISO/IEC 18033-2.

This document is applicable to cryptographic techniques based on elliptic curves defined over finite fields of prime power order (including the special cases of prime order and characteristic two). This document is not applicable to the representation of elements of the underlying finite field (i.e. which basis is used).

The ISO/IEC 15946 series does not specify the implementation of the techniques it defines. Interoperability of products complying with the ISO/IEC 15946 series will not be guaranteed.

2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 15946-1, *Information technology — Security techniques — Cryptographic techniques based on elliptic curves — Part 1: General*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 15946-1 and the following apply.

ISO and IEC maintain terminological databases for use in standardization at the following addresses:

- IEC Electropedia: available at http://www.electropedia.org/
- ISO Online browsing platform: available at http://www.iso.org/obp

3.1
definition field of an elliptic curve
field that includes all the coefficients of the formula describing an elliptic curve

3.2
hash-function
function which maps strings of bits of variable (but usually upper bounded) length to fixed-length strings of bits, satisfying the following two properties:

- for a given output, it is computationally infeasible to find an input which maps to this output;

— for a given input, it is computationally infeasible to find a second input which maps to the same output

Note 1 to entry: Computational feasibility depends on the specific security requirements and environment. Refer to ISO/IEC 10118-1:2016, Annex C.

[SOURCE: ISO/IEC 10118-1:2016, 3.4]

3.3
nearly prime number

positive integer, $n = m \cdot r$, where m is a large prime number and r is a small *smooth integer* (3.5)

Note 1 to entry: The meaning of the terms large and small prime numbers is dependent on the application, and is based on bounds determined by the designer.

3.4
order of an elliptic curve

***E*(*F*)**

number of points on an elliptic curve, E, defined over a finite field, F

3.5
smooth integer

integer, r, whose prime factors are all small (i.e. less than some defined bound)

4 Symbols and conversion functions

4.1 Symbols

B embedding degree, the smallest B such that number $\#E[F(q)] \mid q^B-1$

E elliptic curve, given by a formula of the form $Y^2 = X^3 + aX + b$ over the field $F(p^m)$ for $p > 3$, by a formula of the form $Y^2 + XY = X^3 + aX^2 + b$ over the field $F(2^m)$, or by a formula of the form $Y^2 = X^3 + aX^2 + b$ over the field $F(3^m)$, together with an extra point O_E referred to as the point at infinity. The elliptic curve is denoted by $E/F(p^m)$, $E/F(2^m)$, or $E/F(3^m)$ respectively.

NOTE 1 In applications not based on a pairing, $E/F(p)$ or $E/F(2^m)$ is preferable from an efficiency point of view. In applications that use a pairing, $E/F(p)$ or $E/F(3^m)$ is preferable from an efficiency point of view.

NOTE 2 An elliptic curve is not only the set of points on the curve, but also a group under an operation defined on these points.

N number of points on an elliptic curve E over $F(q)$, $\#E[F(q)]$

n prime divisor of $\#E[F(q)]$

O_E elliptic curve point at infinity

p prime number

q prime power, p^m for some prime p and some integer $m \geq 1$

r cofactor, that is $\#E[F(q)] = rn$

$\#E[F(q)]$ order (or cardinality) of $E[F(q)]$

$\lceil x \rceil$ smallest integer greater than or equal to the real number x

$\lfloor x \rfloor$ largest integer smaller than or equal to the real number x

4.2 Conversion functions

BS2IP	bit string to integer conversion primitive
BS2OSP	bit string to octet string conversion primitive
EC2OSP$_E$	elliptic curve point to octet string conversion primitive
FE2IP$_F$	finite field element to integer conversion primitive
FE2OSP$_F$	finite field element to octet string conversion primitive
I2BSP	integer to bit string conversion primitive
I2OSP	integer to octet string conversion primitive
I2ECP	integer to elliptic curve conversion primitive
OS2BSP	octet string to bit string conversion primitive
OS2FEP$_F$	octet string to finite field element conversion primitive
OS2ECP$_E$	octet string to elliptic curve point conversion primitive
OS2IP	octet string to integer conversion primitive

5 Framework for elliptic curve generation

5.1 Types of trusted elliptic curve

There are a number of ways in which a user can obtain trust in the provenance of an elliptic curve, including the following.

— The curve could be obtained from an impartial trusted source (e.g. an international or national standard).

— The curve could be generated and/or verified by a trusted third party.

— The curve could be generated and/or verified by the user.

NOTE 1 Refer to Annex A for background information on elliptic curves.

NOTE 2 Refer to Annex B for background information on elliptic curve cryptosystems.

5.2 Overview of elliptic curve generation

There are three main ways to generate elliptic curves.

— Generate an elliptic curve by applying the order counting algorithms to a (pseudo-)randomly chosen elliptic curve. Such a technique is specified in Clause 6.

— Generate an elliptic curve by applying the complex multiplication method. Such a technique is specified in Clause 7.

— Generate an elliptic curve by lifting an elliptic curve over a small finite field to that over a reasonably large field. Such a technique is specified in Clause 8.

NOTE 1 Refer to Annex A for background information on elliptic curves.

NOTE 2 Refer to Annex B for background information on elliptic curve cryptosystems.

6 Verifiably pseudo-random elliptic curve generation

6.1 General

The generation of verifiably pseudo-random elliptic curves focuses on curves over prime and binary fields (and so, for example, does not deal with curves over fields of characteristic 3).

6.2 Constructing verifiably pseudo-random elliptic curves (prime case)

6.2.1 Construction algorithm

The following algorithm produces a set of elliptic curve parameters over a field $F(p)$ selected (pseudo-) randomly from the curves of appropriate order, along with sufficient information for others to verify that the curve was indeed chosen pseudo-randomly.

NOTE 1 The algorithm is consistent with Reference [9].

NOTE 2 Methods of choosing a prime number p (pseudo) randomly are described in Reference [5].

It is assumed that the following quantities have been chosen:

— a lower bound, $n_{\min}$, for the order of the base point;

— a cryptographic hash function, H, with output length L_{Hash} bits;

— the bit length, L, of inputs to H, satisfying $L \geq L_{\mathrm{Hash}}$.

The following notation is adopted below:

— $v = \lceil \log_2 p \rceil$,

— $s = \lfloor (v - 1)/L_{\mathrm{Hash}} \rfloor$,

— $w = v - sL_{\mathrm{Hash}} - 1$.

Input: a prime number p; lower bound $n_{\min}$ for n; a trial division bound $l_{\max}$.

Output: a bit string X; EC parameters a, b, n, and G.

a) Choose an arbitrary bit string X of bit length L.

b) Compute $h = H(X)$.

c) Let W_0 be the bit string obtained by taking the w rightmost bits of h.

d) Let $Z = \mathrm{BS2IP}(X)$.

e) For i from 1 to s do:

  1) Let $X_i = \mathrm{I2BSP}(Z + i \bmod 2^L)$.

  2) Compute $W_i = H(X_i)$.

f) Let $W = W_0 \,||\, W_1 \,||\, \ldots \,||\, W_s$.

g) Let $c = \mathrm{OS2FEP}\,[\mathrm{BS2OSP}\,(W)]$.

h) If $c = 0_F$ or $4c + 27 = 0_F$, then go to step a).

i) Choose finite field elements a, $b \in F(p)$ such that $b \neq 0_F$ and $cb^2 - a^3 = 0_F$. Choosing $a = b = c$ will guarantee the conditions hold, and this choice is recommended.

NOTE 3 Choosing $a = b = c$ may not be optimal from a performance perspective.

NOTE 4 If the default values are chosen as suggested, the randomness of the generated curve is explicitly guaranteed.

j) Compute the order $\#E[F(p)]$ of the elliptic curve E over $F(p)$ given by $y^2 = x^3 + ax + b$.

k) Test whether $\#E[F(p)]$ is a nearly prime number using the algorithm specified in 6.2.2. If so, the output of the algorithm specified in 6.2.2 consists of integers r, n. If not, then go to step a).

NOTE 5 The necessity of near primality is described in B.2.2

l) Check if $E[F(p)]$ satisfies the MOV-condition specified in B.2.3, that is the smallest integer B such that n divides $q^B - 1$ ensures the desirable security level. If not, then go to step a).

m) If $\#E[F(p)] = p$, then go to step a).

NOTE 6 This check is performed in order to protect against the attack specified in B.2.2.

n) Test whether the prime divisor n satisfies the condition described in B.2.4 for cryptosystems based on ECDLP, ECDHP, or BDHP with auxiliary inputs as in B.1.5. If not, then go to step a).

o) Generate a point G on E of order n using the algorithm specified in 6.2.3.

p) Output X, a, b, n, G.

NOTE 7 Methods to compute the order $\#E[F(p)]$ are described in References [11], [30] and [31].

6.2.2 Test for near primality

Given a lower bound $n_{\min}$ and a trial division bound $l_{\max}$, the following procedures test $N = \#E[F(p)]$ for near primality.

Input: positive integers N, $l_{\max}$, and $n_{\min}$.

Output: if N is nearly prime, output a prime n with $n_{\min} \le n$ and a smooth integer r such that $N = rn$. If N is not nearly prime, output the message “not nearly prime”.

a) Set $n = N$, $r = 1$.

b) For l from 2 to $l_{\max}$ do:

   1) If l is composite, then go to step 3).

   2) While (l divides n)

      i) Set $n = n/l$ and $r = rl$.

      ii) If $n < n_{\min}$, then output “not nearly prime” and stop.

   3) Next l.

c) Test n for primality.

d) If n is prime, then output r and n and stop.

e) Output “not nearly prime”.

NOTE Methods to test for primality are described in References [5] and [10]

6.2.3 Finding a point of large prime order

If the order $\#E[F(q)]$ of an elliptic curve E is nearly prime, the following algorithm efficiently produces a random point in $E[F(q)]$ whose order is the large prime factor n of $\#E[F(q)] = rn$.

Input: an elliptic curve E over the field $F(q)$, a prime n, and a positive integer r not divisible by n.