INTERNATIONAL STANDARD

ISO/IEC 14888-3

Second edition
2006-11-15

Information technology — Security techniques — Digital signatures with appendix —

Part 3: Discrete logarithm based mechanisms

Technologies de l'information — Techniques de sécurité — Signatures numériques avec appendice —

Partie 3: Mécanismes basés sur un logarithme discret

Reference number
ISO/IEC 14888-3:2006(E)

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

ISO/IEC 14888-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, Subcommittee SC 27, *IT Security techniques*.

This second edition cancels and replaces the first edition (ISO/IEC 14888-3:1998), which has been technically revised. It also incorporates Technical Corrigendum ISO/IEC 14888-3:1998/Cor.1:2001. New mechanisms and object identifiers have been specified.

ISO/IEC 14888 consists of the following parts, under the general title *Information technology — Security techniques — Digital signatures with appendix*:

— *Part 1: General*

— *Part 2: Integer factorization based mechanisms*

— *Part 3: Discrete logarithm based mechanisms*

Introduction

Digital signature mechanisms can be used to provide services such as entity authentication, data origin authentication, non-repudiation, and data integrity. A digital signature mechanism satisfies the following requirements.

- Given either or both of the following two things:
  - the verification key but not the signature key,
  - a set of signatures on a sequence of messages that an attacker has adaptively chosen,

  it should be computationally infeasible for the attacker:
  - to produce a valid signature on a new message,
  - to produce a new signature on a previously signed message, or
  - to recover the signature key.
- It should be computationally infeasible, even for the signer, to find two different messages with the same signature.

NOTE – Computational feasibility depends on the specific security requirements and environment.

Digital signature mechanisms are based on asymmetric cryptographic techniques and involve three basic operations.

- A process for generating pairs of keys, where each pair consists of a private signature key and the corresponding public verification key.
- A process that uses the signature key, called the signature process.
- A process that uses the verification key, called the verification process.

There are two types of digital signature mechanisms.

- When, for a given signature key, any two signatures produced for the same message are always identical, the mechanism is said to be non-randomized (or deterministic); see ISO/IEC 14888-1.
- When, for a given message and signature key, each application of the signature process produces a different signature, the mechanism is said to be randomized.

The eight mechanisms specified in this part of ISO/IEC 14888 are all randomized.

Digital signature mechanisms can also be divided into the following two categories.

- When the whole message has to be stored and/or transmitted along with the signature, the mechanism is termed a "signature mechanism with appendix" (which is the subject of ISO/IEC 14888).
- When the whole message, or part of it, can be recovered from the signature, the mechanism is termed a "signature mechanism giving message recovery" (see ISO/IEC 9796).

Security of the digital signature mechanisms is based on unsolvable problems, i.e. problems for which, given current knowledge, finding a solution is computationally infeasible, such as the factorization problem and the discrete logarithm problem. ISO/IEC 14888-3 specifies digital signature mechanisms with appendix based on the discrete logarithm problem, and ISO/IEC 14888-2 specifies digital signature mechanisms with appendix based on the factorization problem.

NOTE – The previous version of ISO/IEC 14888 grouped identity-based mechanisms into Part 2 and certificate-based mechanisms into Part 3, with each of the two parts covering mechanisms based on both the discrete logarithm and the factorisation problems. This revision re-organizes the grouping, so that Part 2 contains integer factoring based mechanisms and Part 3 discrete logarithm based mechanisms.

This part of ISO/IEC 14888 includes eight mechanisms, two of which were in ISO/IEC 14888-3:1998, and three of which are in ISO/IEC 15946-2:2002. The Korean Certificate-based Digital Signature Algorithm (KCDSA) and two mechanisms based on pairing technology are newly added.

The mechanisms specified in this part of ISO/IEC 14888 use a collision resistant hash-function for hashing the entire message (possibly in more than one part). ISO/IEC 10118 specifies hash-functions.

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this document may involve the use of patents.

The ISO and IEC take no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured the ISO and IEC that he is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with the ISO and IEC. Information may be obtained from:

ISO/IEC JTC 1/SC 27 Standing Document 8 (SD 8) "Patent Information". SD 8 is publicly available at: http://www.ni.din.de/sc27

Further information is available from the identified patent-holders.

Area	Inventors	Patent	Issue date	Contact address
DSA	Kravitz	US 5 231 668	1993-07-27	[no licence required]

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

Information technology — Security techniques — Digital signatures with appendix —

Part 3: Discrete logarithm based mechanisms

1 Scope

This part of ISO/IEC 14888 specifies digital signature mechanisms with appendix whose security is based on the discrete logarithm problem. This part of ISO/IEC 14888 provides

- a general description of a digital signature with appendix mechanism;
- a variety of mechanisms that provide digital signatures with appendix.

For each mechanism, this part of ISO/IEC 14888 specifies

- the process of generating a pair of keys;
- the process of producing signatures;
- the process of verifying signatures.

The verification of a digital signature requires the signing entity's verification key. It is thus essential for a verifier to be able to associate the correct verification key with the signing entity, or more precisely, with (parts of) the signing entity's identification data. This association between the signer's identification data and the signer's public verification key can either be guaranteed by an outside entity or mechanism, or the association can be somehow inherent in the verification key itself. In the former case, the scheme is said to be "certificate-based." In the latter case, the scheme is said to be "identity based." Typically, in an identity-based scheme, the verifier can derive the signer's public verification key from the signer's identification data. The digital signature mechanisms specified in this part of ISO/IEC 14888 are classified into certificate-based and identity-based mechanisms.

NOTE – For certificate-based mechanisms, various PKI standards can be used for key management. For further information, see ISO/IEC 11770-3, ISO/IEC 9594-8 (also known as X.509) and ISO/IEC 15945.

2 Normative references

The following referenced documents are indispensable for the application 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 10118 (all parts), *Information technology — Security techniques — Hash-functions*

ISO/IEC 14888-1:1998, *Information technology — Security techniques — Digital signatures with appendix — Part 1: General*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 14888-1 and the following apply.

3.1
finite commutative group
finite set $\boldsymbol{E}$ with the binary operation "$*$" such that

- for all $a, b, c \in \boldsymbol{E}$, $(a * b) * c = a * (b * c)$;
- there exists $e \in \boldsymbol{E}$ with $e * a = a$ for all $a \in \boldsymbol{E}$;
- for all $a \in \boldsymbol{E}$ there exists $b \in \boldsymbol{E}$ with $b * a = e$;
- for all $a, b \in \boldsymbol{E}$, $a * b = b * a$;
- for all $a, b \in \boldsymbol{E}$, $a * b \in \boldsymbol{E}$.

NOTE 1 – If $a^0 = e$, and $a^{n+1} = a * a^n$ (for $n \geq 0$) is defined recursively, the order of $a \in \boldsymbol{E}$ is the least positive integer n such that $a^n = e$.

NOTE 2 – In some cases, such as when $\boldsymbol{E}$ is the set of points on an elliptic curve, arithmetic in the finite set $\boldsymbol{E}$ is described with additive notation.

3.2
cyclic group
group $\boldsymbol{E}$ of n elements that contains an element $a \in \boldsymbol{E}$, called the generator, of order n

3.3
pairing
function which takes two elements, P and Q, from an elliptic curve cyclic group over a finite field, G_1, as input, and produces an element from another cyclic group over a finite field, G_2, as output, and which has the following two properties (where we assume that the cyclic groups G_1 and G_2 have order q, for some prime q, and for any two elements P, Q, the output of the pairing function is written as $<P, Q>$).

- Bilinearity: If P, P_1, P_2, Q, Q_1, Q_2 are elements of G_1 and a is an integer satisfying $1 \leq a \leq q - 1$, then
  $<P_1 + P_2, Q> = <P_1, Q> * <P_2, Q>$,
  $<P, Q_1 + Q_2> = <P, Q_1> * <P, Q_2>$, and
  $<[a]P, Q> = <P, [a]Q> = <P, Q>^a$.
- Non-degeneracy: If P is a non-identity element of G_1, $<P, P> \neq 1$.

3.4
Trusted Key Generation Centre
KGC
trusted third party, which, in an identity-based signature mechanism, generates a private signature key for each signing entity

4 Symbols

$a\|\|b$	concatenation of a and b, in the order specified
$a \oplus b$	bitwise exclusive OR of a and b
a_1, a_2	elliptic curve coefficients
(A, B, C)	a permutation of (S, T_1, T_2), which specifies the functionality of the signature mechanisms
D	a parameter which specifies the relationship between the signature key and the verification key
E	an elliptic curve defined by two elliptic curve coefficients, a_1 and a_2

$\mathbb{E}$	a finite commutative group; in the mechanisms based on a multiplicative group, elements of $\mathbb{E}$ are in $\mathbf{Z}_p^*$; in the mechanisms based on an additive group of elliptic curve points, elements of $\mathbb{E}$ are the points on the elliptic curve E over $GF(r)$
$\#\mathbb{E}$	the cardinality of $\mathbb{E}$; in the mechanisms based on a multiplicative group, $\#\mathbb{E}$ is p - 1; in the mechanisms based on an additive group of elliptic curve points, $\#\mathbb{E}$ is the number of points on the elliptic curve E over $GF(r)$
$gcd(N_1, N_2)$	the greatest common divisor of integers N_1 and N_2
G	an element of order q in $\mathbb{E}$
$GF(r)$	the Galois field of cardinality r
G_1	a cyclic group of prime order q; elements of G_1 are points on an elliptic curve over $GF(r)$
G_2	a cyclic group of prime order q; elements of G_2 are elements of a finite field $GF(r)$
H_1	a hash-function that converts a data string into an element in G_1 (first express the data string as an integer and then convert the integer into a point on E over $GF(r)$ by using the *I2P* function, see Annex C for details)
h, H_2	hash-functions, i.e. one of the mechanisms specified in ISO/IEC 10118
ID	a data string containing an identifier of the signer, used in Mechanisms IBS-1 and IBS-2
m	an embedding degree (or extension degree)
$[n]P$	multiplication operation that takes a positive integer n and a point P on the curve E as input and produces as output another point Q on the curve E, where $Q = [n]P = P + P + \ldots + P$ added n -1 times. The operation satisfies $[0]P = 0_E$ (the point at infinity), and $[-n]P = [n](-P)$
P	a generator of G_1 which is used in Mechanisms IBS-1 and IBS-2
p	a prime number or a prime power
q	a divisor of $\#\mathbb{E}$ and the order of G_1 and G_2, which is a prime number
r	the size of $GF(r)$; in the mechanisms based on an additive group of elliptic curve points, r is a prime power, p^m, for some prime $p \geq 2$ and integer $m \geq 1$.
T	the assignment
T_1	the first part of the assignment T
T_2	the second part of the assignment T
U	KGC's master private key, which is a randomly chosen integer used in Mechanisms IBS-1 and IBS-2
V	KGC's master public key, which is an element of G_1 used in Mechanisms IBS-1 and IBS-2
$\mathbf{Z}_N^*$	the set of integers U with $0 < U < N$ and gcd $(U, N) = 1$, with arithmetic defined modulo N
α	the bit-length of a prime number (or prime power) p
β	the bit-length of a prime number q
γ	the output bit-length of hash-functions h and H_2
Π_x	x-coordinate of Π
0_E	the point at infinity on the elliptic curve E
< >	a bilinear and non-degenerate pairing

5 General model

5.1 Parameter generation process

5.1.1 Certificate-based mechanisms

5.1.1.1 Generation of domain parameters

For digital signature mechanisms based on discrete logarithms, the set of domain parameters includes the following parameters:

- $\boldsymbol{E}$, a finite commutative group;
- q, a prime divisor of $\#\boldsymbol{E}$;
- G, an element of order q in $\boldsymbol{E}$

In the group $\boldsymbol{E}$, multiplicative notation is used. It is worthwhile to note that the particular signature mechanism chosen may place additional constraints on the choice of $\boldsymbol{E}$, q, and G.

5.1.1.2 Generation of signature key and verification key

A signature key of a signing entity is a secretly generated random or pseudo-random integer X such that $0 < X < q$. The corresponding public verification key Y is an element of $\boldsymbol{E}$ and is computed as

$$Y = G^{X^D},$$

where D is a parameter defined by the mechanism to be used. The value of D is one of two values, -1 and 1.

NOTE – An implementation is still considered compliant if it excludes a few integers from consideration as possible X values. For example, the value 1 can be excluded because this value results in the user's verification key being the generator, G, which is easily detectable.

5.1.2 Identity-based mechanisms

5.1.2.1 Notation

The two identity-based mechanisms specified in clause 7 are both based on the use of pairings over elliptic curve groups. As a result we use additive group notation here and throughout clause 7.

5.1.2.2 Generation of domain parameters

For the identity-based digital signature mechanisms based on discrete logarithms, the set of domain parameters includes the following parameters:

- G_1, a cyclic group of prime order q;
- G_2, a cyclic group of prime order q;
- P, a generator of G_1;
- q, a prime number - the cardinality of G_1 and G_2

5.1.2.3 Generation of master key

A master private key of a KGC is a secretly generated random or pseudo-random integer U such that $0 < U < q$. The corresponding master public key V is an element of G_1 and is computed as

$$V = [U]P.$$

5.1.2.4 Generation of signature key and verification key

A signature key of a signing entity is an element of G_1 and is computed by the KGC as

$X = [U]Y$,

where U is the KGC's master private key and Y is the public verification key generated from an identity string ID and a hash-function H_1, i.e., $Y = H_1(ID)$.

5.1.3 Parameter selection

5.1.3.1 Selecting parameter size

The bit-lengths of parameters for typical security levels are shown in Table 1. The minimum recommended security level is 2^{80}.

NOTE – Security level means the number of steps in the best known attack on a cryptographic primitive. If 2^{80} steps are required in the best known attack on a hash-function, the security level of the hash-function is 2^{80}. For a comprehensive analysis of parameter sizes, see Silverman [27], and Lenstra and Verheul [22].

It is not necessary to select α, β, and γ having same security level; the security level of an implemented signature scheme is the lowest among security levels of parameters.

Table 1 — Parameter sizes according to the security level

Security level	2^{80}	2^{112}	2^{128}	2^{192}	2^{256}
α	1024	2048	3072	7680	15360
β	160	224	256	384	512
γ	160	224	256	384	512

5.1.3.2 Selecting hash-function

Selection of hash-functions should be based on those standardized in ISO/IEC 10118-3. That is, h and H_2 are one of the mechanisms specified in ISO/IEC 10118-3 and H_1 converts a data string obtained by using one of the mechanisms specified in ISO/IEC 10118-3 into an element in G_1 (see Annex B for details).

NOTE 1 – The hash-functions used in this part of ISO/IEC 14888 should be collision-resistant.

NOTE 2 – The security strength for the selected hash-function must meet or exceed the security strength of the parameters in key generation. The relation between the security levels of a hash-function and parameters is shown in the above clause.

Furthermore, implementations that verify digital signatures shall have a way of securely determining which hash-function was used by the signer. Otherwise an attacker might be able to convince a verifier to use a different weaker, hash-function and thus bypass the intended security level.

5.1.4 Validity of domain parameters and verification key

The signature verifier may require assurance that the domain parameters and public verification key are valid, otherwise there is no assurance of meeting the intended security even if the signature verifies, and an adversary may be able to generate signatures that verify.

Assurance of the validity of domain parameters can be provided by one of the following:

- Selection of valid domain parameters from a trusted published source, such as a standard;
- Generation of valid domain parameters by a trusted third party, such as a CA or a KGC;
- Validation of candidate domain parameters by a trusted third party, such as a CA or a KGC;
- For the signer, generation of valid domain parameters by the signer using a trusted system; and
- Validation of candidate domain parameters by the user (i.e., the signer or verifier).

Assurance of validity of a public verification key can be provided by one of the following:

- For the signer, generation of the public verification/private signature key pair using a trusted system;
- For the signer or verifier, validation of the public verification key by a trusted third party, such as a CA or a KGC; and
- Validation of the public verification key by the user (i.e., the signer or verifier).

NOTE 1 – Validation of domain parameters and keys is required. However, how to achieve this is outside the scope of this document.

NOTE 2 – The method of authenticating the signer is dependent on the real applications, which is out of the scope of this document.

5.2 Signature process

All of the signature mechanisms in this part of ISO/IEC 14888 make use of a randomizing value K, which is used (along with the message) to produce a witness R (the first part of the signature) and an assignment (T_1, T_2). The signature for the message is the pair (R, S) where S (the second part of the signature) is computed as the solution of a signature equation.

In the certificate-based mechanisms, specified in Clause 6, the signature equation is

$$AK + BX^D + C \equiv 0 \pmod{q},$$

given that (A, B, C) is a permutation of (S, T_1, T_2), X is the private signature key and D is a parameter depending on the particular mechanism.

In the identity-based mechanisms, specified in Clause 7, the signature equation is

$$[K]A + [U^D]B + C \equiv 0_E \text{ (in } G_1).$$

Given that (A, B, C) is a permutation of (S, T_1, T_2), U is the master private key and D is a parameter depending on the particular mechanism.

The permutation will be specified or agreed upon when setting up the signature system.

The signature process and the formation of a signed message consist of eight stages (See Figure 1):

- Producing the randomizer
- Producing the pre-signature
- Preparing the message for signing
- Computing the witness

- Computing the assignment (it is not necessary to compute the assignment in the identity-based mechanisms)
- Computing the second part of the signature
- Constructing the appendix
- Constructing the signed message.

In this process, the signing entity makes use of its private signature key, its public verification key (optional) and the domain parameters.

5.2.1 Producing the randomizer

For each signature, the signing entity freshly generates a secret randomizer which is an integer K with $0 < K < q$. The output of this stage is K, which shall be kept secret and destroyed safely after use.

NOTE 1 – The randomizer K can be considered as an ephemeral key.

NOTE 2 – For the same rationale of 5.1.1.2, an implementation is still considered compliant if it excludes a few integers from consideration as possible K values.

5.2.2 Producing the pre-signature

The inputs to this stage are the randomizer K and optionally signature key X, with which the signing entity computes the pre-signature, Π, by using K and public parameters as input. In the certificate-based mechanisms specified in Clause 6, it is computed as

$$\Pi = G^K$$

in $\mathbf{E}$. In the identity-based mechanisms specified in Clause 7, it is individually specified in the mechanisms. The output of this stage is the pre-signature, Π.

5.2.3 Preparing the message for signing

In the process of preparing the message, one of M_1 and M_2 becomes message M, the other becomes empty.

5.2.4 Computing the witness (the first part of the signature)

The variables to this stage are the pre-signature Π from 5.2.2 and M_1 from 5.2.3. The values of these variables are taken as inputs to the witness function. The output of the witness function is the witness R. The witness function is specified in the mechanisms.

5.2.5 Computing the assignment

The inputs to the assignment function are the first part of the signature, which is the witness R from 5.2.4, M_2 from 5.2.3, and, optionally, the verification key Y. The output of the assignment function is assignment $T = (T_1, T_2)$. In the certificate-based mechanisms specified in Clause 6, T_1 and T_2 are integers such that

$$0 < |T_1| < q, 0 < |T_2| < q.$$

In the identity-based mechanisms specified in Clause 7, T_1 and T_2 are elements of G_1. It is not necessary to compute T in the identity-based mechanisms.