INTERNATIONAL STANDARD

ISO/IEC 18033-6

First edition
2019-05

IT Security techniques — Encryption algorithms —

Part 6: Homomorphic encryption

Techniques de sécurité IT — Algorithmes de chiffrement —

Partie 6: Chiffrement homomorphe

Reference number
ISO/IEC 18033-6:2019(E)

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.

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) or the IEC list of patent declarations received (see http://patents.iec.ch).

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

For an explanation of 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 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*.

A list of all parts in the ISO/IEC 18033 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user's national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

Introduction

Homomorphic Encryption is a type of symmetric or asymmetric encryption that allows third parties (i.e. parties that are neither the encryptor nor the decryptor) to perform operations on plaintext data while keeping the data in encrypted form. The primary purpose of homomorphic encryption is to allow third parties to perform such computations on data while simultaneously ensuring that the confidentiality of the plaintext data is preserved. It is typically the case that homomorphic encryption schemes require the plaintext to be represented in the form of elements of a group, rather than strings of bits or bytes as is the case with most conventional methods of encryption.

Homomorphic encryption mechanisms can be categorized by the nature of the operation(s) on the plaintext that they can support. This document considers homomorphic encryption mechanisms where the plaintext operation is typically addition and/or multiplication in a prescribed group.

IT Security techniques — Encryption algorithms —

Part 6:
Homomorphic encryption

1 Scope

This document specifies the following mechanisms for homomorphic encryption.

— Exponential ElGamal encryption;

— Paillier encryption.

For each mechanism, this document specifies the process for:

— generating parameters and the keys of the involved entities;

— encrypting data;

— decrypting encrypted data; and

— homomorphically operating on encrypted data.

Annex A defines the object identifiers assigned to the mechanisms specified in this document. Annex B provides numerical examples.

2 Normative references

There are no normative references in this document.

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

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

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at http://www.electropedia.org/

3.1
ciphertext
data which has been transformed to hide its information content

[SOURCE: ISO/IEC 18033-1:2015, 2.11]

3.2
decryption
reversal of a corresponding *encryption* (3.6)

[SOURCE: ISO/IEC 10116:2017, 3.5]

3.3
decryption algorithm
process which transforms *ciphertext* (3.1) into *plaintext* (3.14)

[SOURCE: ISO/IEC 18033-1:2015, 2.17]

3.4
decryptor
entity which decrypts *ciphertexts* (3.1)

[SOURCE: ISO/IEC 18033-5:2015, 3.1]

3.5
deterministic
<algorithm> characteristic of an algorithm that states that given the same input, the same output is always produced

[SOURCE: ISO/IEC 18031:2011, 3.9, modified — "algorithm" has been removed from the term and added as the domain.]

3.6
encryption
(reversible) transformation of data by a cryptographic algorithm to produce *ciphertext* (3.1), i.e. to hide the information content of the data

[SOURCE: ISO/IEC 18033-1:2015, 2.21]

3.7
encryption algorithm
process which transforms *plaintext* (3.14) into *ciphertext* (3.1)

[SOURCE: ISO/IEC 18033-1:2015, 2.22]

3.8
encryptor
entity which encrypts *plaintexts* (3.14)

[SOURCE: ISO/IEC 18033-5:2015, 3.2]

3.9
group
set of elements S and an operation * defined on the set of elements such that (i) $a*(b*c) = (a*b)*c$ for every a, b and c in S, (ii) there exists an identity element e in S such that $a*e = e*a = a$ for every a in S, and (iii) for every a in S there exists an inverse element a^{-1} in S such that $a*a^{-1} = a^{-1}*a = e$

[SOURCE: ISO/IEC 15946-1:2016, 3.6]

3.10
homomorphic map
map from one *group* (3.9) to another that preserves their respective group operations

Note 1 to entry: A definition of homomorphic map is provided by Cohen et al. in [13].

3.11
key
sequence of symbols that controls the operation of a cryptographic transformation

Note 1 to entry: Examples are *encryption* (3.6), *decryption* (3.2), cryptographic check function computation, signature generation, or signature verification.

[SOURCE: ISO/IEC 9798-1:2010, 3.16]

3.12
key generation
process of generating a *key* (3.11)

[SOURCE: ISO/IEC 11770-1:2010, 2.24]

3.13
key generation algorithm
method for generating asymmetric *key* (3.11) pairs

[SOURCE: ISO/IEC 18033-2:2006, 3.27]

3.14
plaintext
unencrypted information

[SOURCE: ISO/IEC 18033-1:2015, 2.30]

3.15
probabilistic
<algorithm> characteristic of an algorithm that states that given the same input, the output could take different values

3.16
security parameter
variables that determine the security strength of a mechanism

[SOURCE: ISO/IEC 20008-2:2013, 3.5]

4 Symbols and abbreviations

$a \in S$	Element a of the set S
sec.key	Private key (secret key)
pub.key	Public key
F_p	Finite field with p elements for a prime p
g	Element in F_p
k	Security parameter
p	Prime number
parameters	Public parameters necessary for encryption, decryption or the group operation on ciphertexts
q	Prime order of g
Z_q^* or Z_n^*	Unit group of Z_q or Z_n, respectively
Z_q or Z_n	Residue ring modulo q or n, respectively
(mod p)	Modulo p
$\bullet$	Operation on the plaintext group
$\odot$	Operation on the ciphertext group
$\langle g \rangle$	Group generated by g

5 General model for homomorphic encryption

5.1 Entities

There are three entities as follows.

— encryptor: an entity that performs homomorphic encryption using a public key;

— decryptor: an entity that performs homomorphic decryption using a private key;

— operator: an entity that performs homomorphic operations on ciphertexts.

5.2 Key roles

The private key *sec.key* shall be kept secret by the decryptor.

The public key *pub.key* shall be public to the encryptor or operator.

The parameters *parameters* are public.

5.3 Algorithms

A homomorphic encryption mechanism is composed of the following three algorithms.

— KeyGen(k). Given a security parameter k, produce a tuple (*pub.key*, *sec.key, parameters*) where *pub.key* denotes the public key, *sec.key* denotes the private key and *parameters* denotes the parameters.

— Encrypt(m, *pub.key, parameters*). Given a public key *pub.key*, parameters *parameters* and a plaintext m in the plaintext group, perform encryption and produce a ciphertext c.

— Decrypt(c, *sec.key, parameters*). Given a private key *sec.key*, parameters *parameters* and a ciphertext c in the ciphertext group, perform decryption and produce a plaintext m.

5.4 Functional requirements

Given any tuple (*pub.key*, *sec.key, parameters*) produced by KeyGen(k), the following two properties are required.

Correctness. For any plaintext m,

Decrypt(Encrypt(m, *pub.key, parameters*), *sec.key, parameters*) = m .

Homomorphic property. The encryption is a homomorphic map from the plaintext group to the ciphertext group. More specifically, for any two plaintexts m_1 and m_2 in the plaintext group, and letting

$$c_1 = \text{Encrypt}(m_1, \textit{pub.key, parameters})$$

$$c_2 = \text{Encrypt}(m_2, \textit{pub.key, parameters}),$$

it is required that

$$\text{Decrypt}(c_1 \odot c_2, \textit{sec.key, parameters}) = m_1 \bullet m_2.$$

In all the mechanisms specified in this document, the key generation and encryption algorithms are probabilistic, while the decryption is a deterministic algorithm.

