INTERNATIONAL STANDARD

ISO/IEC 10116

Fourth edition
2017-07

Information technology — Security techniques — Modes of operation for an *n*-bit block cipher

Technologies de l'information — Techniques de sécurité — Modes opératoires pour un chiffrement par blocs de n *bits*

Reference number
ISO/IEC 10116:2017(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

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*, SC 27, *IT Security techniques*.

This fourth edition cancels and replaces the third edition (ISO/IEC 10116:2006) and ISO/IEC 10116:2006/Cor1:2008, which have been technically revised.

The main technical changes between the third edition and this fourth edition are as follows:

a) the inclusion of padding within the normative scope of ISO/IEC 10116;

b) the inclusion of methods for avoiding ciphertext expansion for CBC, CFB, OFB and CTR modes.

Introduction

This document specifies modes of operation for an *n*-bit block cipher. These modes provide methods for encrypting and decrypting data using a block cipher.

This fourth edition of ISO/IEC 10116 specifies five modes of operation:

a) Electronic Codebook (ECB);

b) Cipher Block Chaining (CBC);

c) Cipher Feedback (CFB);

d) Output Feedback (OFB);

e) Counter (CTR).

NOTE Annex C presents figures describing the modes of operation. Annex D provides numerical examples of the modes of operation.

Information technology — Security techniques — Modes of operation for an *n*-bit block cipher

1 Scope

This document establishes five modes of operation for applications of an *n*-bit block cipher (e.g. protection of data during transmission or in storage). The defined modes only provide protection of data confidentiality. Protection of data integrity is not within the scope of this document. Also, most modes do not protect the confidentiality of message length information.

NOTE 1 Methods for protecting the integrity of data using a block cipher are provided in ISO/IEC 9797-1.

NOTE 2 Methods for simultaneously protecting the confidentiality and integrity of data are provided in ISO/IEC 19772.

This document specifies the modes of operation and gives recommendations for choosing values of parameters (as appropriate).

NOTE 3 The modes of operation specified in this document have been assigned object identifiers in accordance with ISO/IEC 9834. The list of assigned object identifiers is given in Annex A. In applications in which object identifiers are used, the object identifiers specified in Annex A are to be used in preference to any other object identifiers that can exist for the mode concerned.

NOTE 4 Annex B contains comments on the properties of each mode and important security guidance.

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 18033-3, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers*.

ISO/IEC 29192-2, *Information technology — Security techniques — Lightweight cryptography — Part 2: Block ciphers*.

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:

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

3.1
block cipher
symmetric encipherment system with the property that the encryption algorithm operates on a block of plaintext, i.e. a string of bits of a defined length, to yield a block of ciphertext

[SOURCE: ISO/IEC 18033-1:2015, 2.9]

3.2
ciphertext
data which has been transformed to hide its information content

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

3.3
counter
bit array of length n bits (where n is the size of the underlying block cipher) which is used in the Counter mode

Note 1 to entry: The value when considered as the binary representation of an integer increases by one (modulo 2^n) after each block of plaintext is processed.

3.4
cryptographic synchronization
co-ordination of the encryption and decryption processes

3.5
decryption
reversal of a corresponding encryption

[SOURCE: ISO/IEC 18033-1:2015, 2.16, modified]

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

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

3.7
feedback buffer
FB
variable used to store input data for the encryption process

Note 1 to entry: At the starting point, *FB* has the value of *SV*.

3.8
key
sequence of symbols that controls the operation of a cryptographic transformation (e.g. encryption and decryption)

[SOURCE: ISO/IEC 18033-1:2015, 2.27, modified]

3.9
***n*-bit block cipher**
block cipher with the property that plaintext blocks and ciphertext blocks are n bits in length

[SOURCE: ISO/IEC 18033-1:2015, 2.29]

3.10
padding
appending extra bits to a data string

3.11
plaintext
unencrypted information

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

3.12
starting variable
SV
variable possibly derived from some initialization value and used in defining the starting point of the modes of operation

Note 1 to entry: The “starting variable” (*SV*) used in this document is similar to (possibly identical to) the “initialization value” or “initialization vector” (IV) used in some other International Standards. If the starting variable referred to in this document is derived from some initialization value, then it needs to be described in any application of the modes of operation. A method for deriving a starting variable from an initializing value is not defined in this document.

4 Symbols, abbreviated terms and notation

4.1 Symbols and abbreviated terms

C	Ciphertext (string of bits)
CTR	Counter value
dK	Decryption function of the block cipher keyed by key K. The decryption relation defined by the block cipher is written $P = dK(C)$ where — P is the plaintext block; — C is the ciphertext block; — K is the key.
E	Intermediate variable
eK	Encryption function of the block cipher keyed by key K. The encryption relation defined by the block cipher is written $C = eK(P)$ where — P is the plaintext block; — C is the ciphertext block; — K is the key.
F	Intermediate variable
FB	Feedback buffer
i	Iteration
j	Size of plaintext/ciphertext variable
K	Key
m	Number of stored ciphertext blocks
n	Plaintext/ciphertext block length for a block cipher
P	Plaintext (string of bits)
q	Number of plaintext/ciphertext variables
r	Size of feedback buffer
SV	Starting variable (see Clause 5)
X	Block cipher input block
Y	Block cipher output block

$\|$	Concatenation of bit strings
$(a_1, a_2, . . . , a_m)$	A one-dimensional array of bits. For example, $A = (a_1, a_2, . . . , a_m)$ and $B = (b_1, b_2, . . . , b_m)$ are two arrays of m bits, numbered from 1 to m. All arrays of bits are written with the bit with the index 1 in the leftmost position. When interpreting a bit array as an integer, the leftmost bit shall be the most significant bit.
$\oplus$	The operation of bitwise addition, modulo 2, also known as the “exclusive or” function, is shown by the symbol $\oplus$. The operation when applied to arrays A and B of the same length is defined as $A \oplus B = (a_1 \oplus b_1, a_2 \oplus b_2, . . . , a_m \oplus b_m)$.

4.2 Notation

a mod n	For integers a and n, a mod n denotes the (non-negative) remainder obtained when a is divided by n. Equivalently, if $b = a$ mod n, then b is the unique integer satisfying: — $0 \leq b < n$, and — $(b - a)$ is an integer multiple of n.
~	The operation of selecting the j leftmost bits of an array $A = (a_1, a_2, . . . , a_m)$ to generate a j-bit array is written $(j \sim A) = (a_1, a_2, . . . , a_j)$ The operation of selecting the j rightmost bits of an array $A = (a_1, a_2, . . . , a_m)$ to generate a j-bit array is written $(A \sim j) = (a_{m-j+1}, a_{m-j+2}, . . . , a_m)$ The operation is defined only when $1 \leq j \leq m$.
$1(t)$	t-bit string where the value 1 is assigned to every bit.
S_t	Given an m-bit variable X and a t-bit variable F where $1 \leq t \leq m$, the effect of the shift function $S_t(X \| F)$ is to produce the m-bit variable $S_t(X \| F) = (x_{t+1}, x_{t+2}, . . . , x_m, f_1, f_2, . . . , f_t)$ $(t < m)$ $S_t(X \| F) = (f_1, f_2, . . . , f_t)$ $(t = m)$ The effect is to shift the bits of array X left by t places, discarding $x_1, x_2, . . . , x_t$, and to place the array F in the rightmost t places of X. When $t = m$ the effect is to totally replace X by F.

5 Requirements

For all of the described modes, a block cipher shall be selected from ISO/IEC 18033-3 and/or ISO/IEC 29192-2.

For the Electronic Codebook (ECB) mode, there are no additional parameter values that need to be selected. For the Cipher Block Chaining (CBC) mode of operation (see Clause 7), one parameter m shall be selected. For the Cipher Feedback (CFB) mode of operation (see Clause 8), three parameters r, j and k need to be selected. For the Output Feedback (OFB) mode of operation (see Clause 9) and the Counter (CTR) mode of operation (see Clause 10), one parameter j needs to be selected. When one of these modes of operation is used, the same parameter value(s) need to be chosen and used by all communicating parties. These parameters need not be kept secret.

All modes of operation specified in this document require the parties encrypting and decrypting a data string to share a secret key K for the block cipher in use. All modes of operation apart from the electronic Codebook (ECB) mode also require the parties to share a starting variable SV, where the length of SV will depend on the mode in use. The value of the starting variable should normally be different for every data string encrypted using a particular key. How keys and starting variables are managed and distributed is outside the scope of this document; however, important security guidance related to starting variables is provided in Annex B.

The encrypter and all potential decrypters shall agree on a padding method, unless messages to be encrypted are always a multiple of m bits ($m = n$ for ECB and CBC modes, $m = j$ for CFB, OFB and CTR modes) in length or unless the mode does not require padding. Such a padding method shall take a (non-empty) bit string P of arbitrary length as input and give a sequence of m-bit plaintext blocks as output. Unless the length of the plaintext message is fixed or otherwise defined by the application, then the padding method shall have the property that no two distinct input bit strings can give the same sequence of m-bit blocks as output, i.e. padded data strings can always be uniquely unpadded.

This document recommends the following padding method: the (non-empty) bit string P is right-padded with a single ‘1’ bit. The resulting string is then right-padded with as few (possibly none) ‘0’ bits as necessary to obtain a string whose length (in bits) is a positive integer multiple of m.

NOTE This padding method is identical to padding method 2 in both ISO/IEC 9797-1 and ISO/IEC 10118-1 except that it only considers non-empty bit strings.

6 Electronic Codebook (ECB) mode

6.1 Preliminaries

ECB mode shall be used with an agreed padding method if the bit length of the plaintext might not be a multiple of the block cipher block length n.

The variables employed by the ECB mode of encryption are

a) the input variables

   1) A plaintext string of bits P.

   2) A key K.

b) the output variables, i.e. a ciphertext string of bits C.

6.2 Encryption

If a padding method is to be used, then first pad the data string P to obtain a sequence of q plaintext blocks P_1, P_2, ..., P_q, each of n bits. Otherwise, simply divide the data string P into a sequence of q plaintext blocks P_1, P_2, ..., P_q, each of n bits.

The ECB mode of encryption continues as follows:

$$C_i = eK(P_i) \text{ for } i = 1, 2, \ldots, q.$$

6.3 Decryption

The ECB mode of decryption operates as follows:

$$P_i = dK(C_i) \text{ for } i = 1, 2, \ldots, q.$$

If a padding method is not in use, then P is equal to the concatenation of P_1, P_2, ..., P_q.

If a padding method is in use, apply the inverse of the agreed padding procedure to the sequence of q plaintext blocks P_1, P_2, ..., P_q to obtain the decrypted plaintext string P.

NOTE Unique removal of padding is guaranteed because of the constraint on the choice of padding method given in Clause 5.

7 Cipher Block Chaining (CBC) mode

7.1 Preliminaries

CBC mode encryption and decryption as defined in 7.2 and 7.3 shall use an agreed padding method if the bit length of the plaintext might not be a multiple of the block cipher block length n. If no padding method is agreed, then the plaintext shall be a whole number of complete blocks unless it is agreed to use the approach described in 7.4.

The CBC mode of operation is defined by an interleave parameter $m > 0$, the number of independent encryption processes that could operate in parallel.

NOTE 1 The value of m can be small (typically $m = 1$) and can rarely exceed 1 024.

The variables employed by the CBC mode when being used for encryption are

a) the input variables

   1) A plaintext string of bits P.

   2) A key K.

   3) A sequence of m starting variables $SV_1, SV_2, \dots, SV_m$ each of n bits. Refer to Annex B for security guidance related to the value of the starting variables.

   NOTE 2 If $m = 1$ and no padding is applied, then this mode is compatible with the CBC mode described in the second edition of this document.

b) the output variables, i.e. a ciphertext string of bits C.

7.2 Encryption

If a padding method is to be used, then first pad the data string P to obtain a sequence of q plaintext blocks P_1, P_2, ..., P_q, each of n bits. Otherwise, simply divide the data string P into a sequence of q plaintext blocks P_1, P_2, ..., P_q, each of n bits.

The CBC mode of encryption operates as follows:

$$C_i = eK(P_i \oplus SV_i),\ 1 \leq i \leq \min(m, q)$$

If q > m, all subsequent plaintext blocks are encrypted as:

$$C_i = eK(P_i \oplus C_{i-m}),\ m + 1 \leq \mathrm{i} \leq q$$

At any time during the computation, the values of the m most recent ciphertext blocks need to be stored, e.g. in a cyclically used “feedback buffer” FB (see Figure C.2).

This procedure is shown on the left side of Figure C.2.

7.3 Decryption

The CBC mode of decryption operates as follows:

$$P_i = dK(C_i) \oplus SV_i,\ 1 \leq i \leq \min(m, q)$$

If $q > m$, all subsequent ciphertext blocks are decrypted as:

$$P_i = dK(C_i\ ) \oplus C_{i-m},\ m + 1 \leq i \leq q$$

At any time during the computation, the values of the m most recent ciphertext blocks need to be stored, e.g. in a cyclically used “feedback buffer” FB (see Figure C.2).

This procedure is shown on the right side of Figure C.2.

If a padding method is not in use, then P is equal to the concatenation of P_1, P_2, ..., P_q.

If a padding method is in use, apply the inverse of the agreed padding procedure to the sequence of q plaintext blocks P_1, P_2, ..., P_q to obtain the decrypted plaintext string P.

NOTE Unique removal of padding is guaranteed because of the constraint on the choice of padding method given in Clause 5.

7.4 Avoiding ciphertext expansion

7.4.1 General

In order to avoid ciphertext expansion caused by padding, it is possible to implement ‘ciphertext stealing’. With this method, if the final plaintext block is partial, then the fewest ‘0’ bits are appended so as to complete it (i.e., padding method 1 in ISO/IEC 9797-1:2011) and the resulting plaintext is encrypted as above. Any expansion that would be caused by padding bits is avoided, because bits of the penultimate ciphertext block are discarded as they can be recovered from the decryption of the final ciphertext block.

Specifically, if m=1 (no interleaving) and the plaintext has been padded so that the rightmost p bits of the final plaintext block P_q are the ‘0’ padding bits, then the rightmost p bits of the penultimate ciphertext block $C_{q\text{-}1}$ are not transmitted but can be recovered as the rightmost p bits of $dK(C_q)$.

NOTE Ciphertext stealing cannot be applied if the number of bits in the plaintext is less than n.

7.4.2 Three ciphertext stealing variants of CBC

7.4.2.1 Preliminaries

This document defines three ciphertext stealing variants (CBC_CS) of CBC mode. All three are variants of the basic CBC encryption/decryption defined above using padding method 1 in ISO/IEC 9797-1:2011 [1]. These CBC_CS variants differ from the basic CBC mode only in how the two ciphertext blocks $C_{q\text{-m}}$ and C_q are processed after encryption and before decryption.

Although the CBC_CS variants can be used when the interleave parameter m>1, as noted in 7.1, typically, m=1 in which case the two ciphertext blocks $C_{q\text{-}m}$ and C_q blocks are the final two ciphertext blocks $C_{q\text{-}1}$ and C_q. The description below applies only to the case m=1 but can be extrapolated to the case m>1.

7.4.2.2 Processing after CBC encryption (ciphertext contraction)

After CBC encryption the ciphertext $C_1 \mid C_2 \mid ... \mid C_{q\text{-}1} \mid C_q$ is contracted (reduced in length) by the number p ($0 \le p < n$) of padding bits. If no padding occurred (p=0), then no contraction occurs but, as specified below, the third CBC_CS variant still modifies the ciphertext by swapping the final two ciphertext blocks $C_{q\text{-}1}$ and C_q :

For all CBC_CS variants define, $C_{q\text{-}1}{}^* = (n\text{-}p) \sim C_{q\text{-}1}$. The three CBC_CS variants are defined as follows:

— For the first CBC_CS variant, the final two ciphertext blocks $C_{q\text{-}1} \mid C_q$ are replaced by $C_{q\text{-}1}{}^* \mid C_q$.

— For the second CBC_CS variant, the final two ciphertext blocks $C_{q\text{-}1} \mid C_q$ are replaced by $C_q \mid C_{q\text{-}1}{}^*$ only if the plaintext was padded (otherwise no change is made).

— For the third CBC_CS variant, the final two ciphertext blocks $C_{q\text{-}1} \mid C_q$ are always replaced by $C_q \mid C_{q\text{-}1}{}^*$.

7.4.2.3 CBC decryption pre-processing (ciphertext extension)

Before CBC decryption, the received contracted ciphertext is extended to a whole number of n-bit blocks.