INTERNATIONAL STANDARD

ISO/IEC 9797-1

First edition
1999-12-15

Information technology — Security techniques — Message Authentication Codes (MACs) —

Part 1: Mechanisms using a block cipher

Technologies de l'information — Techniques de sécurité — Codes d'authentification de message (MACs) —

Partie 1: Mécanismes utilisant un cryptogramme bloc

Reference number
ISO/IEC 9797-1:1999(E)

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 734 10 79
E-mail copyright@iso.ch
Web www.iso.ch

Printed in Switzerland

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 3.

In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1. 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.

Attention is drawn to the possibility that some of the elements of this part of ISO/IEC 9797 may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

International Standard ISO/IEC 9797-1 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 27, *IT Security techniques.*

This first edition of ISO/IEC 9797-1, together with the subsequent parts of ISO/IEC 9797, cancels and replaces ISO/IEC 9797:1994, which has been revised and extended to a multi-part standard. Note, however, that implementations which comply with ISO/IEC 9797:1994 will be compliant with this edition of ISO/IEC 9797-1.

ISO/IEC 9797 consists of the following parts, under the general title *Information technology — Security techniques — Message Authentication Codes (MACs)*:

— *Part 1: Mechanisms using a block cipher*

— *Part 2: Mechanisms using a hash-function*

Further parts may follow.

Annexes A and B of this part of ISO/IEC 9797 are for information only.

Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher

1 Scope

This part of ISO/IEC 9797 specifies six MAC algorithms that use a secret key and an n-bit block cipher to calculate an m-bit MAC. These mechanisms can be used as data integrity mechanisms to verify that data has not been altered in an unauthorised manner. They can also be used as message authentication mechanisms to provide assurance that a message has been originated by an entity in possession of the secret key. The strength of the data integrity mechanism and message authentication mechanism is dependent on the length (in bits) k^* and secrecy of the key, on the block length (in bits) n and strength of the block cipher, on the length (in bits) m of the MAC, and on the specific mechanism.

The first three mechanisms specified in this part of ISO/IEC 9797 are commonly known as CBC-MAC (CBC is the abbreviation of Cipher Block Chaining). The calculation of a MAC as described in ISO 8731-1 and ANSI X9.9 is a specific case of this part of ISO/IEC 9797 when $n = 64$, $m = 32$, MAC Algorithm 1 and Padding Method 1 are used, and the block cipher is DEA (ANSI X3.92: 1981). The calculation of a MAC as described in ANSI X9.19 and ISO 9807 is a specific case of this part of ISO/IEC 9797 when $n = 64$, $m = 32$, either MAC Algorithm 1 or MAC Algorithm 3 is used (both with Padding Method 1), and the block cipher is DEA (ANSI X3.92: 1981).

The fourth mechanism is a variant of CBC-MAC with a special initial transformation. It is recommended for applications which require that the key length of the MAC algorithm is twice that of the block cipher.

NOTES

1 For example, in the case of DEA (ANSI X3.92: 1981), the block cipher key length is 56 bits, while the MAC algorithm key length is 112 bits.

2 When used with DEA (which is also known as DES), this algorithm is called MacDES [12].

The fifth and sixth mechanism use two parallel instances of the first and fourth mechanism respectively, and combine the two results with a bitwise exclusive-or operation. They are recommended for applications which require an increased security level against forgery attacks (cf. Annex B). The fifth mechanism uses a single length MAC algorithm key, while the sixth mechanism doubles the MAC algorithm key length.

This part of ISO/IEC 9797 can be applied to the security services of any security architecture, process, or application.

2 Normative references

The following standards contain provisions which, through reference in this text, constitute provisions of this part of ISO/IEC 9797. At the time of publication, the editions indicated were valid. All standards are subject to revision, and parties to agreements based on this part of ISO/IEC 9797 are encouraged to investigate the possibility of applying the most recent editions of the standards indicated below. Members of IEC and ISO maintain registers of currently valid International Standards.

ISO 7498-2: 1989, *Information processing systems — Open Systems Interconnection — Basic Reference Model — Part 2: Security Architecture.*

ISO/IEC 9798-1: 1997, *Information technology — Security techniques — Entity authentication — Part 1: General.*

ISO/IEC 10116: 1997, *Information technology — Security techniques — Modes of operation for an n-bit block cipher.*

3 Definitions

3.1 This part of ISO/IEC 9797 makes use of the following general security-related term defined in ISO 7498-2.

3.1.1 data integrity: the property that data has not been altered or destroyed in an unauthorized manner.

3.2 For the purposes of this part of ISO/IEC 9797, the following definitions apply.

3.2.1 block: a bit-string of length n.

3.2.2 block cipher key: a key that controls the operation of a block cipher.

3.2.3 initial transformation: a function that is applied at the beginning of the MAC algorithm.

3.2.4 MAC algorithm key: a key that controls the operation of a MAC algorithm.

3.2.5 Message Authentication Code (MAC): the string of bits which is the output of a MAC algorithm.

NOTE — A MAC is sometimes called a cryptographic check value (see for example ISO 7498-2).

3.2.6 Message Authentication Code (MAC) algorithm: an algorithm for computing a function which maps strings of bits and a secret key to fixed-length strings of bits, satisfying the following two properties:

- for any key and any input string the function can be computed efficiently;
- for any fixed key, and given no prior knowledge of the key, it is computationally infeasible to compute the function value on any new input string, even given knowledge of the set of input strings and corresponding function values, where the value of the ith input string may have been chosen after observing the value of the first $i-1$ function values.

NOTES

1 A MAC algorithm is sometimes called a cryptographic check function (see for example ISO 7498-2).

2 Computational feasibility depends on the user's specific security requirements and environment.

3.2.7 output transformation: a function that is applied at the end of the MAC algorithm, before the truncation operation.

3.3 This part of ISO/IEC 9797 makes use of the following general security-related terms defined in ISO/IEC 9798-1.

3.3.1 ciphertext: data which has been transformed to hide its information content.

3.3.2 decipherment: the reversal of a corresponding encipherment.

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

3.3.4 key: a sequence of symbols that controls the operation of a cryptographic transformation (e.g., encipherment, decipherment, cryptographic check function computation, signature generation, or signature verification).

3.3.5 plaintext: unenciphered information.

3.4 This part of ISO/IEC 9797 makes use of the following general security-related term defined in ISO/IEC 10116.

3.4.1 n-bit block cipher: a block cipher with the property that plaintext blocks and ciphertext blocks are n bits in length.

4 Symbols and notation

Throughout this part of ISO/IEC 9797 the following symbols and notation are used:

D data string to be input to the MAC algorithm.

D_j a block derived from the data string D after the padding process.

$d_K(C)$ decipherment of the ciphertext C with the block cipher e using the key K.

$e_K(P)$ encipherment of the plaintext P with the block cipher e using the key K.

g output transformation, that maps the block H_q to the block G.

G the block that is the result of the output transformation.

H_j a block which is used in the MAC algorithm to store an intermediate result.

I initial transformation.

k the length (in bits) of the block cipher key.

k^* the length (in bits) of the MAC algorithm key.

K, K', K'', K''', K_1, K_2, K_1', K_2', K_1'', K_2'' secret block cipher keys.

L the length block, which is used in Padding Method 3.

L_D the length (in bits) of the data string D.

m the length (in bits) of the MAC.

n the block length (in bits) of the block cipher.

q the number of blocks in the data string D after the padding and splitting process.

$j \sim X$ the string obtained from the string X by taking the leftmost j bits of X.

$X \oplus Y$ exclusive-or of bit-strings X and Y.

$X||Y$ concatenation of bit-strings X and Y (in that order).

$:=$ a symbol denoting the 'set equal to' operation used in the procedural specifications of MAC algorithms, where it indicates that the value of the string on the left side of the symbol shall be made equal to the value of the expression on the right side of the symbol.

5 Requirements

Users who wish to employ a MAC algorithm from this part of ISO/IEC 9797 shall select:

- a block cipher e;
- a padding method from amongst those specified in Clause 6.1;
- a MAC algorithm from amongst those specified in Clause 7;
- the length (in bits) m of the MAC; and
- a common key derivation method if MAC algorithms 4, 5, and 6 are used; a common key derivation method may also be required for MAC algorithm 2.

Agreement on these choices amongst the users is essential for the purpose of the operation of the data integrity mechanism.

The length m of the MAC shall be a positive integer less than or equal to the block length n.

If Padding Method 3 is used, the length in bits of the data string D shall be less than 2^n.

The selection of a specific block cipher e, padding method, MAC algorithm, value for m, and key derivation method (if any) are beyond the scope of this part of ISO/IEC 9797.

NOTE — These choices affect the security level of the MAC algorithm. For a detailed discussion, see Annex B.

The same key shall be used for calculating and verifying the MAC. If the data string is also being enciphered, the key used for the calculation of the MAC shall be different from that used for encipherment.

NOTE — It is considered to be good cryptographic practice to have independent keys for confidentiality and for data integrity.

6 Model for MAC algorithms

The application of the MAC algorithm requires the following six steps: padding, splitting, initial transformation, iterative application of the block cipher, output transformation, and truncation. Steps 3 through 6 are illustrated in Figure 1.

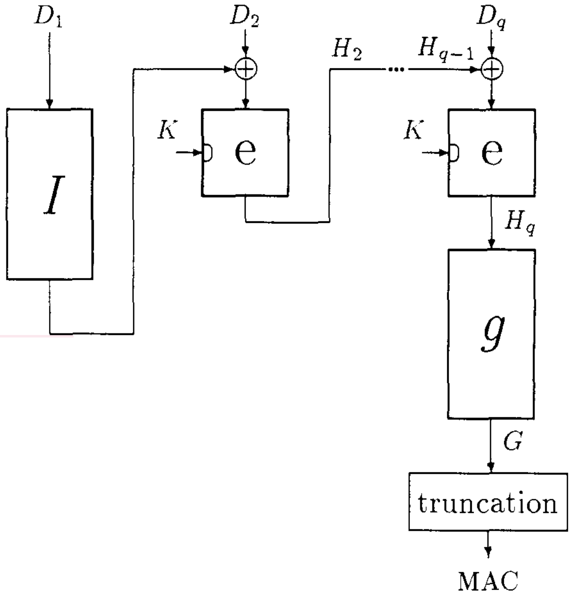

Figure 1: Application of Step 3, 4, 5 and 6 of the MAC algorithm.

6.1 Step 1 (padding)

This step involves prefixing and/or postfixing the data string D with additional 'padding' bits such that the padded version of the data string will always be a multiple of n bits in length. The padding bits that are added to the original data string, according to the chosen padding method, are only used for calculating the MAC. Consequently, these padding bits (if any) need not be stored or transmitted with the data. The verifier shall know whether or not the padding bits have been stored or transmitted, and which padding method is in use.

This part of ISO/IEC 9797 specifies three padding methods. Any of these three methods can be chosen for the six MAC algorithms specified in this part of ISO/IEC 9797.

6.1.1 Padding Method 1

The data string D to be input to the MAC algorithm shall be right-padded with as few (possibly none) '0' bits as necessary to obtain a data string whose length (in bits) is a positive integer multiple of n.

NOTES

1 MAC algorithms using Padding Method 1 may be subject to trivial forgery attacks. See informative Annex B for further details.

2 If the data string is empty, Padding Method 1 specifies that it is right-padded with n '0' bits.

6.1.2 Padding Method 2

The data string D to be input to the MAC algorithm shall be right-padded with a single '1' bit. The resulting string shall then be right-padded with as few (possibly none) '0' bits as necessary to obtain a data string whose length (in bits) is a positive integer multiple of n.

6.1.3 Padding Method 3

The data string D to be input to the MAC algorithm shall be right-padded with as few (possibly none) '0' bits as necessary to obtain a data string whose length (in bits) is a positive integer multiple of n. The resulting string shall then be left-padded with a block L. The block L consists of the binary representation of the length (in bits) L_D of the unpadded data string D, left-padded with as few (possibly none) '0' bits as necessary to obtain an n-bit block. The right-most bit of the block L corresponds to the least significant bit of the binary representation of L_D.

NOTE — Padding Method 3 is not suitable for use in situations where the length of the data string is not available prior to the start of the MAC calculation.

6.2 Step 2 (splitting)

The padded version of the data string D is split into q n-bit blocks D_1, D_2, ..., D_q. Here D_1 represents the first n bits of the padded version of D, D_2 represents the next n bits, and so on.

6.3 Step 3 (initial transformation)

The initial transformation I is applied to the first block D_1 of the padded data string to derive the block H_1.

Each of the six MAC algorithms specified in this part of ISO/IEC 9797 use one of two possible initial transformations.

6.3.1 Initial Transformation 1

This transformation requires only one block cipher key K. The block H_1 is computed by applying the block cipher with key K as follows:

$$H_1 := e_K(D_1)\,.$$

6.3.2 Initial Transformation 2

This transformation requires two block cipher keys K and K''. The block H_1 is computed by applying the block cipher with keys K and K'' as follows:

$$H_1 := e_{K''}(e_K(D_1))\,.$$

6.4 Step 4 (iteration)

The blocks H_2, H_3, ..., H_q are calculated by iteratively applying the block cipher to the bitwise exclusive-or of the data block D_i and the previous result H_{i-1}:

```
for $i$ from 2 to $q$:
    $H_i := e_K(D_i \oplus H_{i-1})$;
```

If q is equal to 1, Step 4 shall be omitted.

NOTE — This operation corresponds to the Cipher Block Chaining (CBC) mode as defined in ISO/IEC 10116.

6.5 Step 5 (output transformation)

The output transformation g is applied to the value H_q, obtained as a result of Step 4 (or Step 3 in the case $q = 1$).

This part of ISO/IEC 9797 specifies three output transformations.

6.5.1 Output Transformation 1

This output transformation is the identity function, i.e.,

$$G := H_q\,.$$

6.5.2 Output Transformation 2

This output transformation consists of applying the block cipher with block cipher key K' to H_q, i.e.,

$$G := e_{K'}(H_q)\,.$$

6.5.3 Output Transformation 3

This output transformation consists of applying the block cipher (in decryption mode) with the key K' to H_q followed by applying the block cipher with key K to the result of this operation, i.e.,

$$G := e_K(d_{K'}(H_q))\,.$$

6.6 Step 6 (truncation)

The MAC of m bits is derived by taking the leftmost m bits of the block G, i.e.,

$$\mathrm{MAC} := m \sim G\,.$$

7 MAC Algorithms

This part of ISO/IEC 9797 specifies six MAC algorithms. The initial transformation and output transformation are specified in each case. However, the padding method is not specified, i.e., each MAC algorithm may be used with any of the three padding methods specified in Clause 6.1.

NOTE — The choice of padding method affects the security of the MAC algorithm. See informative Annex B for further details.

7.1 MAC Algorithm 1

MAC Algorithm 1 uses Initial Transformation 1 and Output Transformation 1. The MAC algorithm key consists of the block cipher key K. MAC Algorithm 1 is illustrated in Figure 2.

7.2 MAC Algorithm 2

MAC Algorithm 2 uses Initial Transformation 1 and Output Transformation 2. The MAC algorithm key consists of two block cipher keys K and K'''. The value of K''' may be derived from the value of K in such a way that K and K''' are different.

NOTES

1 An example of how to derive K''' from K is to complement alternate substrings of four bits of K commencing with the first four bits. Another example is to derive both K and K''' from a common master key.

2 If K and K''' are equal, a simple xor forgery attack applies. See informative Annex B for further details.

3 If K and K''' are independent, the level of security against key recovery attacks is less than suggested by the MAC algorithm key size. See informative Annex B for further details.

MAC Algorithm 2 is illustrated in Figure 3.

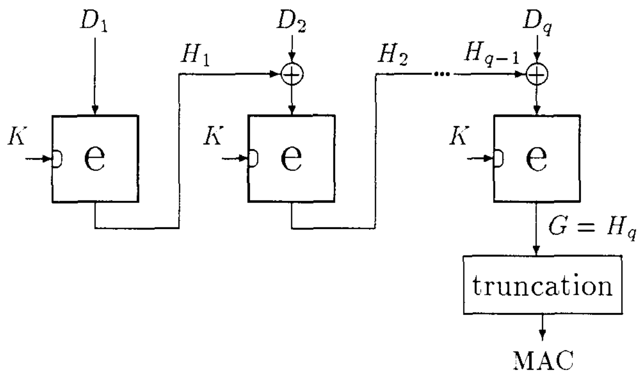

Figure 2 — MAC Algorithm 1.

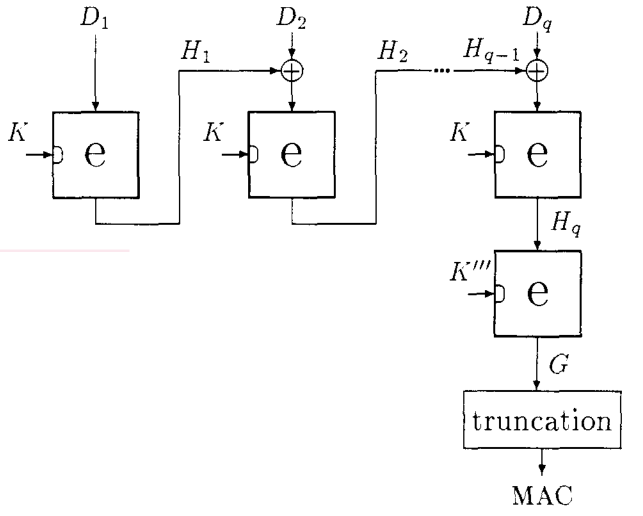

Figure 3 — MAC Algorithm 2.

7.3 MAC Algorithm 3

MAC Algorithm 3 uses Initial Transformation 1 and Output Transformation 3. The MAC algorithm key consists of two block cipher keys K and K'. The values of K and K' shall be chosen independently. If $K' = K$, MAC Algorithm 3 reduces to MAC Algorithm 1, which may not always be desirable. MAC Algorithm 3 is illustrated in Figure 4.

7.4 MAC Algorithm 4

MAC Algorithm 4 uses Initial Transformation 2 and Output Transformation 2. The MAC algorithm key con-

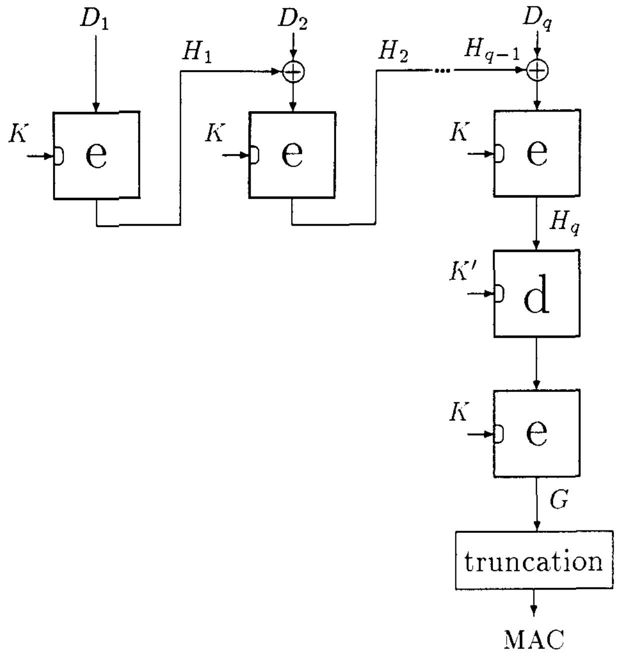

Figure 4 — MAC Algorithm 3.

sists of two block cipher keys K and K', that shall be chosen independently. A third block cipher key K'' shall be derived from K'. The values of K, K', and K'' shall be different. The block cipher keys K and K'' are used with Initial Transformation 2, and the block cipher keys K and K' are used with Output Transformation 2.

NOTE — An example of how to derive K'' from K' is to complement alternate substrings of four bits of K' commencing with the first four bits. Another example is to derive both K' and K'' from a common master key.

The number of blocks in the padded version of the data string shall be greater than or equal to two, i.e., $q \geq 2$.

MAC Algorithm 4 is illustrated in Figure 5.

7.5 MAC Algorithm 5

MAC Algorithm 5 uses two parallel instances of MAC Algorithm 1, resulting in two intermediate values, MAC$_1$ and MAC$_2$ respectively. The MAC algorithm key consists of the block cipher key K. The keys K_1 and K_2 used for the first and second instances are derived from the key K. The values of K_1 and K_2 shall be different.

NOTE — An example of how to derive K_1 and K_2 from K is to take K_1 equal to K and to construct K_2 by complementing alternate substrings of four bits of K commencing with the first four bits. Another example is to derive both K_1 and K_2 from a common master key in such a way that they are different.

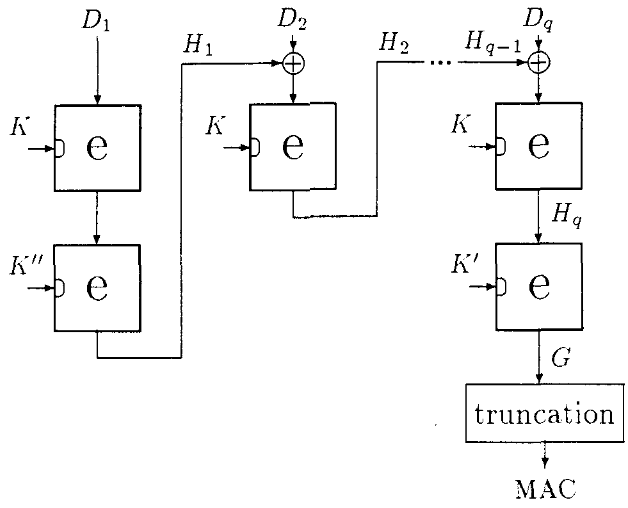

Figure 5 — MAC Algorithm 4.

The final MAC is obtained by a bitwise exclusive-or between intermediate values MAC$_1$ and MAC$_2$, i.e.,

$$\text{MAC} := \text{MAC}_1 \oplus \text{MAC}_2 \ .$$

7.6 MAC Algorithm 6

MAC Algorithm 6 uses two parallel instances of MAC Algorithm 4, resulting in two intermediate values, MAC$_1$ and MAC$_2$ respectively. The MAC algorithm key consists of two block cipher keys K and K', that shall be chosen independently.

The keys (K_1, K_1') and (K_2, K_2') used for the first and second instances respectively shall be derived from the keys (K, K') in such a way that K_1 and K_1' are different, K_2 and K_2' are different, and the pairs (K_1, K_1') and (K_2, K_2') are different.

NOTES

1 An example of how to derive (K_1, K_1') and (K_2, K_2') from (K, K') is to take K_1 equal to K, K_1' equal to K', and to construct K_2 (K_2') by complementing alternate substrings of eight bits of K_1 (K_1') commencing with the first eight bits.

2 MAC Algorithm 4 internally uses a derived key K'', which means that MAC Algorithm 6 uses in total six block cipher keys (K_1, K_1', K_1'') and (K_2, K_2', K_2''). It is recommended to check that all of these keys are different.

The number of blocks in the padded version of the data string shall be greater than or equal to two, i.e., $q \geq 2$.

The final MAC is obtained by a bitwise exclusive-or between intermediate values MAC$_1$ and MAC$_2$, i.e.,

$$\text{MAC} := \text{MAC}_1 \oplus \text{MAC}_2 \ .$$

Annex A

(informative)

Examples

This annex presents examples of the generation of a MAC using the DEA (see ANSI X3.92). The plaintexts are the 7-bit ASCII codes (no parity) for data string 1: “`Now␣is␣the␣time␣for␣all␣`” and data string 2: “`Now␣is␣the␣time␣for␣it`”, where “␣” denotes a blank. ASCII coding is equivalent to coding using ISO 646. The two key values used are K = 0123456789ABCDEF (hexadecimal), and K' = FEDCBA9876543210 (hexadecimal). The key parity bits are ignored. Derived keys are computed by complementing alternate substrings of four bits commencing with the first four bits. For MAC Algorithms 1, 2, 3, and 4, $m = 32$, while for MAC Algorithms 5 and 6, $m = 64$. All values are written in hexadecimal notation.

For data string 1, the results of the three padding methods are as follows:

- Padding Method 1: $q = 3$

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 61 6C 6C 20

- Padding Method 2: $q = 4$

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 61 6C 6C 20
D_4	80 00 00 00 00 00 00 00

- Padding Method 3: $q = 4$

D_1	00 00 00 00 00 00 00 C0
D_2	4E 6F 77 20 69 73 20 74
D_3	68 65 20 74 69 6D 65 20
D_4	66 6F 72 20 61 6C 6C 20

For data string 2, the results of the three padding methods are as follows:

- Padding Method 1: $q = 3$

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 69 74 00 00

- Padding Method 2: $q = 3$

D_1	4E 6F 77 20 69 73 20 74
D_2	68 65 20 74 69 6D 65 20
D_3	66 6F 72 20 69 74 80 00

- Padding Method 3: $q = 4$

D_1	00 00 00 00 00 00 00 B0
D_2	4E 6F 77 20 69 73 20 74
D_3	68 65 20 74 69 6D 65 20
D_4	66 6F 72 20 69 74 00 00

A.1 MAC Algorithm 1

Data string 1 with Padding Method 1

key (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 EC A9 E9 4A
$G = H_3$	70 A3 06 40 CC 76 DD 8B

MAC = 70 A3 06 40

Data string 1 with Padding Method 2

key (K)	01 23 45 67 89 AB CD EF
H_1	3F A4 0E 8A 98 4D 48 15
$D_2 \oplus H_1$	57 C1 2E FE F1 20 2D 35
H_2	0B 2E 73 F8 8D C5 85 6A
$D_3 \oplus H_2$	6D 41 01 D8 EC A9 E9 4A
H_3	70 A3 06 40 CC 76 DD 8B
$D_4 \oplus H_3$	F0 A3 06 40 CC 76 DD 8B
$G = H_4$	10 E1 F0 F1 08 34 1B 6D

MAC = 10 E1 F0 F1

Data string 1 with Padding Method 3