INTERNATIONAL STANDARD

ISO/IEC 29167-19

Second edition
2019-06

Information technology — Automatic identification and data capture techniques —

Part 19: Crypto suite RAMON security services for air interface communications

Technologies de l'information — Techniques automatiques d'identification et de capture de données —

Partie 19: Services de sécurité par suite cryptographique RAMON pour communications par interface radio

Reference number
ISO/IEC 29167-19:2019(E)

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

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 Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 31, *Automatic identification and data capture*.

This second edition cancels and replaces the first edition (ISO/IEC 29167-19:2016), which has been technically revised.

The main changes compared to the previous edition are as follows:

— It was thought that the fixed RAMON key length (KE) of 1 024 bits for tag authentication, defined in the first edition of this document, would maybe not be sufficient for all future uses. The method proposed in this edition allows extending the length of the cryptographic RAMON key by discrete steps of 128 bits. Beyond the previously defined key length of 1 024 bits, key lengths of 1 152, 1 280, 1 408, 1 536, 1 664 bits and beyond become feasible. The current method does not limit the possible key length in any way. The key length only is limited by the ability to send the cryptographic authentication response, which is of equal length to the cryptographic key, back to the interrogator. Allowing extended key length makes sure that the RAMON encryption is future-proofed and security can be improved as needed.

— To support different key lengths in a generic approach, the mix function has been revised.

— To improve the readability and consistency of this document, the specification of the cipher and the description of the TLV record have been separated into independent subclauses.

— A new TLV-Structure, supporting data identifiers according to ASC MH 10, was added.

A list of all parts in the ISO 29167 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

This document specifies the security services of a Rabin-Montgomery (RAMON) crypto suite. It is important to know that all security services are optional. The crypto suite provides Tag authentication security service.

The International Organization for Standardization (ISO) and International Electrotechnical Commission (IEC) draw attention to the fact that it is claimed that compliance with this International Standard may involve the use of patents concerning radio-frequency identification technology given in the clauses identified below.

ISO and IEC take no position concerning the evidence, validity and scope of these patent rights. The holders of these patent rights have ensured the ISO and IEC that they are willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statements of the holders of these patent rights are registered with ISO and IEC.

Information on the declared patents may be obtained from:

NXP B.V.

411 East Plumeria
San Jose
CA-95134 1924
USA

Impinj, Inc.

400 Fairview Ave N, # 1200
Seattle, WA 98109
USA

Giesecke & Devrient GmbH

Prinzregentenstrasse 159
D-81607 Munich
Germany

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 — Automatic identification and data capture techniques —

Part 19: Crypto suite RAMON security services for air interface communications

1 Scope

This document defines the Rabin-Montgomery (RAMON) crypto suite for the ISO/IEC 18000 series of air interfaces standards for radio frequency identification (RFID) devices. Its purpose is to provide a common crypto suite for security for RFID devices that can be referred to by ISO/IEC for air interface standards and application standards.

This document specifies a crypto suite for Rabin-Montgomery (RAMON) for air interface for RFID systems. The crypto suite is defined in alignment with existing air interfaces.

This document defines various authentication methods and methods of use for the cipher. A Tag and an Interrogator can support one, a subset, or all of the specified options, clearly stating what is supported.

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 8825-1, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) — Part 1*

ISO/IEC 15962:2013, *Information technology — Radio frequency identification (RFID) for item management — Data protocol: data encoding rules and logical memory functions*

ISO/IEC 18000-3, *Information technology — Radio frequency identification for item management — Part 3: Parameters for air interface communications at 13,56 MHz*

ISO/IEC 18000-4, *Information technology — Radio frequency identification for item management — Part 4: Parameters for air interface communications at 2,45 GHz*

ISO/IEC 18000-63:2015, *Information technology — Radio frequency identification for item management — Part 63: Parameters for air interface communications at 860 MHz to 960 MHz Type C*

ISO/IEC 19762, *Information technology — Automatic identification and data capture (AIDC) techniques — Harmonized vocabulary*

3 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 19762 and the following 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
authentication
service that is used to establish the origin of information

3.2
confidentiality
property whereby information is not disclosed to unauthorized parties

3.3
integrity
property whereby data have not been altered in an unauthorized manner since they were created, transmitted or stored

3.4
non-traceability
protection ensuring that an unauthorized interrogator is not able to track the tag location by using the information sent in the tag response

3.5
secure communication
communication between the tag and the interrogator by use of the *Authenticate* command, assuring authenticity, *integrity* (3.3) and *confidentiality* (3.2) of exchanged messages

4 Conformance

4.1 Claiming conformance

An Interrogator or Tag shall comply with all relevant clauses of this document, except those marked as "optional".

4.2 Interrogator conformance and obligations

An Interrogator shall implement the mandatory commands defined in this document and conform to ISO/IEC 18000-3, ISO/IEC 18000-4 or ISO/IEC 18000-63, as relevant.

An Interrogator may implement any subset of the optional commands defined in this document.

The Interrogator shall not

— implement any command that conflicts with this document, or

— require the use of an optional, proprietary or custom command to meet the requirements of this document.

4.3 Tag conformance and obligations

A Tag shall implement the mandatory commands defined in this document for the supported types and conform to ISO/IEC 18000-3, ISO/IEC 18000-4 or ISO/IEC 18000-63, as relevant

A Tag may implement any subset of the optional commands defined in this document.

A Tag shall not

— implement any command that conflicts with this document, or

— require the use of an optional, proprietary or custom command to meet the requirements of this document.

5 Symbols and abbreviated terms

5.1 Symbols

xx_2	binary notation
xxh	hexadecimal notation
||	concatenation of syntax elements in the order written

5.2 Abbreviated terms

AES	Advanced encryption standard
CBC	Cipher block chaining
CH	Challenge
CH_{I1}, CH_{I2}	Interrogator random challenge, 16 bytes
CH_T	Tag random challenge, 16 bytes
CG	Cryptogram
CMAC	Ciphered message authentication code
CRC	Cyclic redundancy check
CRC–16	16-bit CRC
CS	Crypto suite
CSI	Crypto suite identifier
DEC(key, data)	AES decryption of enciphered “data” with secret “key”
ENC(key, data)	AES encryption of plain “data” with secret “key”
EPC™	Electronic product code
IID	Interrogator identifier, 8 bytes
IV	Initialization vector for CBC-encryption, 16 bytes
KDF	Key derivation function
k	Bit length of public RAMON key K_E and private key K_D k shall be divisible by 128 and ≥1 024.
K_E	Public key for encryption stored on Tag
K_D	Private decryption key stored on Interrogator
K_V	Public signature verification key stored on Interrogator
K_S	Private signature generation key stored in the tag issuer facility
K_{ENC}	Shared secret message encryption key
K_{MAC}	Shared secret message authentication key

KESel	Key select (determines which K_E will be used)
KSel	Key select (determines which pair of K_{ENC}, K_{MAC} will be used)
MAC(key, data)	Calculation of a MAC of (enciphered) "data"' with secret "key"; internal state of the tag's state machine
MAMx,y	Mutual authentication method x.y
MCV	MAC chaining value
MIX(CH, RN, SID)	RAMON mix function
PRF	Pseudorandom function
R	Tag response
RAMON	Rabin-Montgomery
RFU	Reserved for future use
RM_ENC(key, data)	RAMON encryption of plain "data" with public "key"
RM_DEC(key, data)	RAMON decryption of enciphered "data" with private "key"
RN	Random number
RNT	Random number generated by the tag, 16 bytes
S_{ENC}	Message encryption session key
S_{MAC}	Message authentication session key
SID	Secret IDentifier, 8 bytes, identifying the tag
SSC	Send sequence counter for replay protection, 16 bytes
TAMx,y	Tag authentication method x.y; internal state of the tag's state machine
TLV	Tag length value
UHF	Ultra high frequency
UII	Unique item identifier
WORM	Write once, read many

5.3 Notation

This document uses the notation of ISO/IEC 18000-63.

The following notation for key derivation corresponds to Reference [7].

$PRF(s,x)$	A pseudo-random function with seed s and input data x.
K_I	Key derivation key used as input to the KDF to derive keying material. K_I is used as the block cipher key, and the other input data are used as the message defined in Reference [9].
K_O	Keying material output from a key derivation function, a binary string of the required length, which is derived using a key derivation key.

Label	A string that identifies the purpose for the derived keying material, which is encoded as a binary string.
Context	A binary string containing the information related to the derived keying material. It may include identities of parties who are deriving and/or using the derived keying material and, optionally, a nonce known by the parties who derive the keys.
L	An integer specifying the length (in bits) of the derived keying material K_0. *L* is represented as a binary string when it is an input to a key derivation function. The length of the binary string is specified by the encoding method for the input data.
h	An integer that indicates the length (in bits) of the output of the PRF.
i	A counter that is input to each iteration of the PRF.
r	An integer, smaller or equal to 32, that indicates the length of the binary representation of the counter *i*. in bits.
00h	An all zero byte. An optional data field used to indicate a separation of different variable length data fields.
$\lceil X \rceil$	The smallest integer that is larger than or equal to *X*. The ceiling of *X*.
$\{X\}$	Indicates that data *X* is an optional input to the key derivation function.
$[T]_2$	An integer *T* represented as a binary string (denoted by "2") with a length specified by the function, an algorithm, or a protocol which uses *T* as an input.
$\emptyset$	The empty binary string.

6 Crypto suite introduction

6.1 Overview

The RAMON crypto suite permits two levels of implementation. The first level provides secure identification and tag authentication, while the second level extends the functionality by mutual authentication to securely communicate between Interrogator and Tag, e.g. for secure reading and writing non-volatile memory.

Basic RAMON Tags can provide only the first level of implementation, while more sophisticated Tags also provide the second level. See Figure 1 for the different implementation levels for the RAMON crypto suite.

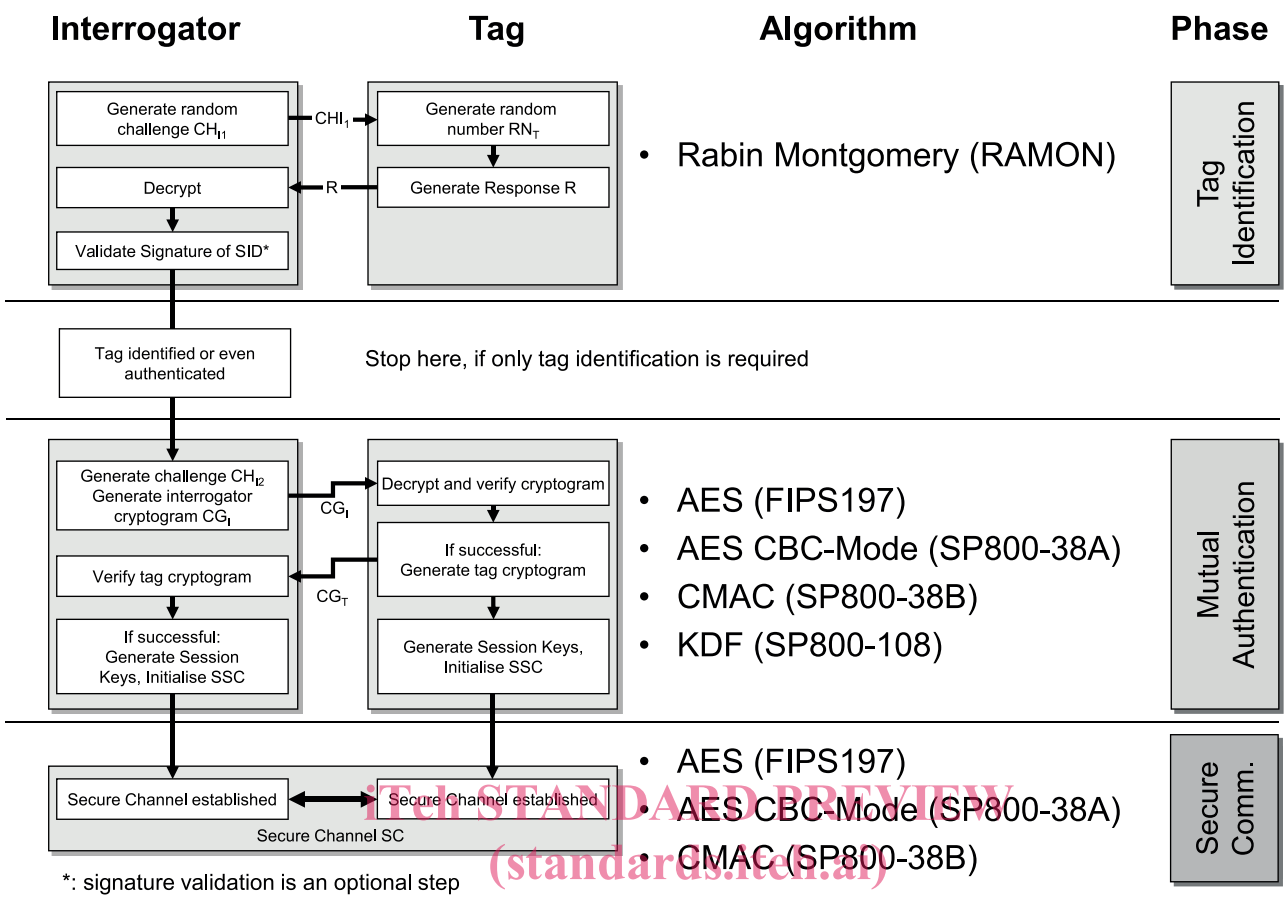

Figure 1 — Overview of the different implementation levels for the RAMON crypto suite

6.2 Authentication protocols

6.2.1 Tag identification

The Rabin-Montgomery crypto suite provides non-traceable and confidential Tag identification. Confidentiality and privacy for the Tag's identifier are provided without requiring the Tag to store a private key.

The crypto suite is based on the asymmetric cryptosystem developed by Michael O. Rabin[14]. The original algorithm is augmented by a method detected by Peter Montgomery[11], which avoids the division of long numbers in modular arithmetic. The advantage of combining Rabin encryption with the concept of Montgomery multiplication is related to the fact that no "costly" division is required.

The Tag performs only public key operations. The Interrogator performs the "expensive" private key operation. The steps necessary to carry out RAMON are outlined in Table 1. RAMON encryption performed by the Tag and decryption shall be performed by the Interrogator as specified in C.3 and C.4. The cryptographic keys are specified in 6.6. A test example for the RAMON encryption with a 1 024-bit public encryption-key K_E is defined in Annex D.

Annex G details the structure of the clear text record, the TLV record, used for authentication of the Tag, comprising the Tag identity record and random data originating in part from the Tag and from the Interrogator for the other part. The TLV record shall be structured in accordance with Annex G.

Table 1 — Protocol steps for Tag identification

Interrogator (K_D, K_V)		Tag (SID, K_E)
Generate random challenge CH_{I1} and send it to the Tag.	(CH_{I1}) →	Generate random number RN_T.
	(R) ←	Generate response cryptogram: R = RM_ENC(K_E, MIX(CH_{I1}, RN_T, TLV record)).
Decrypt Tag response and apply the inverse of the MIX function to get the plaintext P: P = MIX^{-1}(RM_DEC (K_D, R)).		
Obtain CH_{I1}, RN_T and SID from plaintext P.		
Compare previously generated Interrogator challenge with the value received from Tag. If successful, Tag is identified.		
If a signature is provided along with the SID, use K_V to validate the signature. If successful, Tag is authenticated.		

6.2.2 Symmetric mutual authentication

This crypto suite allows combining the Rabin-Montgomery scheme for Tag identification with symmetric mutual authentication. The mutual authentication specified by this crypto suite is based on AES, according to Reference [12]. The CBC mode for encryption is specified in Reference [8]. For MAC generation, CMAC according to Reference [9] is used. For derivation of secure messaging keys, the KDF in counter mode specified in 5.1 of Reference [7] is used.

The protocol steps for mutual authentication are outlined in Table 2.

Table 2 — Protocol steps for mutual authentication

Phase	Interrogator (*IID*, Database, K_D, K_V)		Tag (SID, K_E, K_{ENC}, K_{MAC})
(1) Tag Identification	Generate random challenge CH_{I1} and send it to the Tag.	(CH_{I1}) →	Generate random number RN_T. Generate response:
	Decrypt Tag response and apply the inverse of the MIX function to get the plaintext P: P = MIX^{-1}[RM_DEC (K_D, R)]. Obtain CH_{I1}, RN_T and SID from plaintext P. Compare previously generated Interrogator challenge with the value received from Tag. If successful, Tag is identified. If a signature is provided along with the SID, use K_V to validate the signature. If successful, Tag is authenticated. Set CH_T = RN_T.	(R) ←	R = RM_ENC(K_E, MIX(CH_{I1}, RN$_T$, TLV record, '00' byte)).
The Interrogator has successfully identified (and authenticated) the Tag. In the following phase, CH_T and SID are used in the mutual authentication.			

Table 2 *(continued)*

Phase	Interrogator (IID, Database, K_D, K_V)		Tag (SID, K_E, K_{ENC}, K_{MAC})
(2) Mutual Authentication	Generate CH_{I2}. Generate cryptogram: $S = CH_{I2} \| \| IID \| \| CH_T \| \| SID$; C = ENC (K_{ENC}, S); M = MAC (K_{MAC}, C); $CG_I = C \| \| M$.	(CG_I)	Decrypt and verify the cryptogram:
		→	MAC (K_{MAC}, C); DEC (K_{ENC}, C). Verify CH_T and SID. If equal, generate Session Keys S_{ENC}, S_{MAC}. Initialize SSC. Generate Tag cryptogram: $S = CH_T \| \| SID \| \| CH_{I2} \| \| IID$; C = ENC (K_{ENC}, S); M = MAC (K_{MAC}, C); $CG_T = C \| \| M$
	Verify the cryptogram: MAC (K_{MAC}, C); DEC (K_{ENC}, C). Verify CH_{I2}, CH_T, SID and IID. If equal, generate session keys: S_{ENC}, S_{MAC}. Initialize SSC.	(CG_T) ←	
Mutual authentication is now complete and a secure channel is established.			

The Interrogator has access to a list of SIDs (Secret identifiers) with the associated K_{ENC} and K_{MAC} for each Tag. This is represented by the "Database" on Interrogator's site.

After having successfully identified the Tag in Phase 1, the Interrogator is able to find secret keys K_{ENC} and K_{MAC} that it shares with the Tag. K_{ENC} is used in CBC mode. The IV for encryption is set to all zeroes 00h...00h. As the size of S is on both sides a multiple of the AES block size, no padding is applied. K_{MAC} is used to calculate a 16-byte MAC.

CH_T and CH_{I2} are used as challenges in the challenge-response protocol for mutual authentication and for generation of the starting value of the SSC. See 6.3 for details.

The session encryption key, S_{ENC}, is used for confidentiality of data in transit. AES encryption, including an SSC, is illustrated in Figure 18; decryption is illustrated in Figure 19. The session MAC key, S_{MAC}, is used for data and protocol integrity. This crypto suite derives session keys as specified in 6.4.

If the Tag cannot verify the interrogator's MAC, it reports a crypto suite error (see Annex B for further information) and assumes state **Init**. If the interrogator cannot verify the tag's MAC, the tag is not authenticated.

6.3 Send sequence counter

The send sequence counter (SSC) ensures that the initial values (IVs) are different for every encryption and the MAC chaining values (MCVs) are different for every MAC generation. To this end, the SSC is incremented (+1) each time before a Secure Communication command or response is processed.

After mutual authentication, the initial value of the send sequence counter SSC is generated as follows:

$SSC = CH_T$ (<algorithm block size/2> least significant bytes) ||

CH_{I2} (<algorithm block size/2> least significant bytes)

After receiving a secure command, the Tag increments SSC, then checks the MAC and then decrypts the command. In turn, before sending a secure response, the Tag increments SSC, encrypts the response and generates the MAC. Each particular step is under control of the security flags. Thus, if SSC has the value x at idle time, x+1 is used for processing the next secure command, and x+2 is used for processing the response. SSC may overflow to 0h during the increment without particular action.

6.4 Session key derivation

6.4.1 General

The derivation of the session keys, S_{ENC} and S_{MAC}, is based on the KDF in counter mode specified in 5.1 of Reference [7]. This method uses CMAC as the PRF with AES as underlying block cipher with full 16 bytes output length. The input to the PRF for this cipher suite is as specified in 6.4.3.

6.4.2 KDF in counter mode

The key derivation function iterates a pseudorandom function n times and concatenates the output until L bits of keying material are generated, where n: = .$\lceil L / h \rceil$.. In each iteration, the fixed input data is the string *Label* || *00h* || *Context* || $[L]_2$. The counter $[i]_2$ is the iteration variable and is represented as a binary string of r bits.

Figure 2 illustrates the process.

The input to the PRF [see step d) of **Process**] is explained in 6.4.2.

For the derivation of session encryption key S_{ENC}, K_I is set to K_{ENC}. For the derivation of session MAC key S_{MAC}, K_I is set to K_{MAC}.

Fixed values

— h – The length of the output of the PRF in bits;

— r – The length of the binary representation of the counter i. in bits.

Input: K_I, *Label, Context*, and L.

Process

a) n: = $\lceil L / h \rceil$.

b) If $n > 2^r - 1$, then indicate a crypto suite error and stop.

c) *result*(*0*) := ∅.

d) For i = 1 to n, do

— $K(i)$:= PRF (K_I, $[i]_2$ || *Label* || *00h* || *Context* || $[L]_2$);

— *result*(i) := *result*(i–1) || $K(i)$.

e) Return: K_O := the leftmost L bits of *result* (n).

Output: K_O.