INTERNATIONAL STANDARD

ISO/IEC 29167-22

First edition
2018-11

Information technology — Automatic identification and data capture techniques —

Part 22:
Crypto suite SPECK security services for air interface communications

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

Partie 22: Services de sécurité par suite cryptographique SPECK pour communications par interface radio

Reference number
ISO/IEC 29167-22:2018(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. 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*, Subcommittee, SC 31 *Automatic identification and data capture techniques.*

A list of all the parts in the ISO/IEC 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 a variety of security services provided by the lightweight block cipher SPECK. While SPECK supports various key and block sizes, the cipher versions that are supported in this cryptographic suite take the following block/key sizes in bits: 64/96, 96/96, 64/128, 128/128, and 128/256.

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 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 assured 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:

Contact details
Impinj, Inc.
400 Fairview Ave N, # 1200
Seattle, WA 98109 USA

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.

The latest information on IP that may be applicable to this document can be found at www.iso.org/patents.

Information technology — Automatic identification and data capture techniques —

Part 22: Crypto suite SPECK security services for air interface communications

1 Scope

This document defines the crypto suite for SPECK for the ISO/IEC 18000 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 committees for air interface standards and application standards. The crypto suite is defined in alignment with existing air interfaces.

SPECK is a symmetric block cipher that is parameterized in both its block length and key length. In this document, a variety of block/key length options are supported.

This document defines various 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 18000-63, *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, definitions, symbols and abbreviated terms

3.1 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/

3.1.1
bit string
ordered sequence of 0s and 1s

3.1.2
block cipher
family of permutations that is parameterized by a cryptographic key and, optionally, the block size

3.1.3
block size
number of bits in a *data block* (3.1.6) that is an input (or output) of the *block cipher* (3.1.2)

3.1.4
cryptographic key
string of bits of length given by *key size* (3.1.7) that is used by the *block cipher* (3.1.2) to transform some *data block* (3.1.6)

3.1.5
command
<message> data that the Interrogator sends to the Tag with Message as parameter

3.1.6
data block
string of bits whose length is given by the *block size* (3.1.3) of the *block cipher* (3.1.2)

3.1.7
key size
length in bits of the *cryptographic key* (3.1.4) that is used by the *block cipher* (3.1.2)

3.1.8
message
part of the *command* (3.1.5) that is defined by the crypto suite

3.1.9
nonce
data block (3.1.6) that, within the parameters of typical use, can be assumed to be non-repeating

3.1.10
SPECK-b/k-ENC(key, data)
SPECK encryption of a *b*-bit *data block* (3.1.6) using a *k*-bit *cryptographic key* (3.1.4)

3.1.11
SPECK-b/k-DEC(key, data)
SPECK decryption of a *b*-bit *data block* (3.1.6) using a *k*-bit *cryptographic key* (3.1.4)

3.1.12
Reply
<response> data that the Tag returns to the Interrogator with *Response* (3.1.13) as parameter

3.1.13
Response
part of the *Reply* (3.1.12) (stored or sent) that is defined within the crypto suite

3.2 Symbols

$XXXX_2$	Binary notation
$XXXX_h$	Hexadecimal notation
||	Concatenation of syntax elements, transmitted in the order written
∅	The empty string, typically used to indicate a deliberately empty input or omitted field
|A|	The bit-wise length of the string A expressed as an integer

Example 1: $|0000_2| = 4$.

Example 2: $|0000_h| = 16$.

Example 3: $|\emptyset| = 0$.

fix1(A) The string obtained by fixing the first (leftmost) bit to 1_2

Example 1: **fix**$1(0000_2) = 1000_2$.

Example 2: **fix**$1(0000_h) = 8000_h$.

Example 3: **fix**$1(\emptyset) = \emptyset$.

msb$_n$ (A) The *n*-bit binary string obtained by taking the first (leftmost) *n* bits of the binary representation of A

Example 1: **msb**$_3$ $(1010_2) = 101_2$.

Example 2: **msb**$_7$ $(ABCD_h) = 1010101_2$.

Example 3: **msb**$_7$ $(\emptyset) = \emptyset$.

Field [a:b] Selection of bits from a string of bits denoted Field

The selection ranges from bit "a" through to, and including, bit "b" where Field [0] represents the least significant or rightmost bit.

Example 1: Field [2:0] represents the selection of the three least significant bits of Field.

Example 2: Field, without a specified range, indicates the entirety of Field.

Example 3: Field [-1:0] is an alternative representation of the empty string Ø.

Key.KeyID The cryptographic key identified and indexed by the numerical value KeyID

3.3 Abbreviated terms

CS Crypto Suite

CSI Crypto Suite Indicator

RFU Reserved for future use

4 Conformance

4.1 Air interface protocol specific information

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 the relevant part of ISO/IEC 18000.

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 the relevant part of ISO/IEC 18000.

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 Introducing the SPECK cryptographic suite

SPECK is a lightweight Feistel block cipher that is suitable for extremely constrained environments such as RFID Tags. The details of the operation of the SPECK cipher are described in Annex C.

The background for the development of SPECK and its design principles are described in Reference [3].

SPECK is parameterized in terms of the block size, denoted b, and the key size denoted k. A particular variant of SPECK will be denoted SPECK-b/k throughout this document. While Reference [3] offers many different choices to the block and key size, this cryptographic suite only supports the five parameter combinations given in Table 1:

Table 1 — Variants of SPECK-*b*/*k* supported in this document

	SPECK-64/96	SPECK-64/128	SPECK-96/96	SPECK-128/128	SPECK-128/256
Block size (*b* bits)	64	64	96	128	128
Key size (*k* bits)	96	128	96	128	256

It is possible that not all variants of SPECK will be cryptographically suited to all applications. Guidance on the appropriate variant for a given application lies outside the scope of this document and a thorough security and risk assessment is advised before deployment.

Test vectors for the components of this document are provided in Annex D.

6 Parameter and variable definitions

Table 2 lists the variables and constants that are used in this document.

Table 2 — SPECK cryptographic suite variables and constants

Parameter	Description
IChallenge-*b*/*k*	A challenge generated at random by the Interrogator. The length of IChallenge-*b*/*k* depends on the values of *b* and *k*.
TChallenge-*b*/*k*	A challenge generated at random by the Tag. The length of TChallenge-*b*/*k* depends on the values of *b* and *k*.
TRnd-*b*/*k*	A salt value generated at random by the Tag. The length of TRnd-*b*/*k* depends on the values of *b* and *k*.
IRnd-*b*/*k*	A salt value generated at random by the Interrogator. The length of IRnd-*b*/*k* depends on the values of *b* and *k*.
C_TAM-*b*/*k*	A pre-defined constant. The length and value of C_TAM-*b*/*k* depends on the values of *b* and *k*.
C_IAM-*b*/*k*	A pre-defined constant. The length and value of C_IAM-*b*/*k* depends on the values of *b* and *k*.
C_MAM-*b*/*k*	A pre-defined constant. The length and value of C_MAM-*b*/*k* depends on the values of *b* and *k*.
Key.0 ... Key.255	A set of up to 256 keys Key.0 through to Key.255. Not all key values need to be specified. However Key.j shall not be specified when there remain unspecified Key.i with i<j.

Table 3 gives the values of C_TAM-*b*/*k*, C_IAM-*b*/*k* and C_MAM-*b*/*k* that are used in this document. For a given choice of operational parameters, the length of these constants depends on the block size *b*.

Table 3 — Values of C_TAM-*b*/*k*, C_IAM-*b*/*k*, and C_MAM-*b*/*k* for different values of *b* and *k* and different parameter sets PS

b*/*k	**64/96**	**64/128**	**96/96**	**128/128**	**128/256**
C_TAM-*b*/*k*	11_2	11_2	FF_h	$FFFF_h$	$FFFF_h$
C_IAM-*b*/*k*	10_2	10_2	FE_h	$FFFE_h$	$FFFE_h$
C_MAM-*b*/*k* for PS=00_2	01_2	01_2	FD_h	$FFFD_h$	$FFFD_h$
C_MAM-*b*/*k* for PS=01_2	1_h	1_h	D_h	FD_h	FD_h

7 Crypto suite state diagram

After power-up and after a reset, the Cryptographic Suite shall transition into the **Initial** state, state transitions shall be defined by Annex A, and error handling shall be defined by Annex B. See Figure 1.

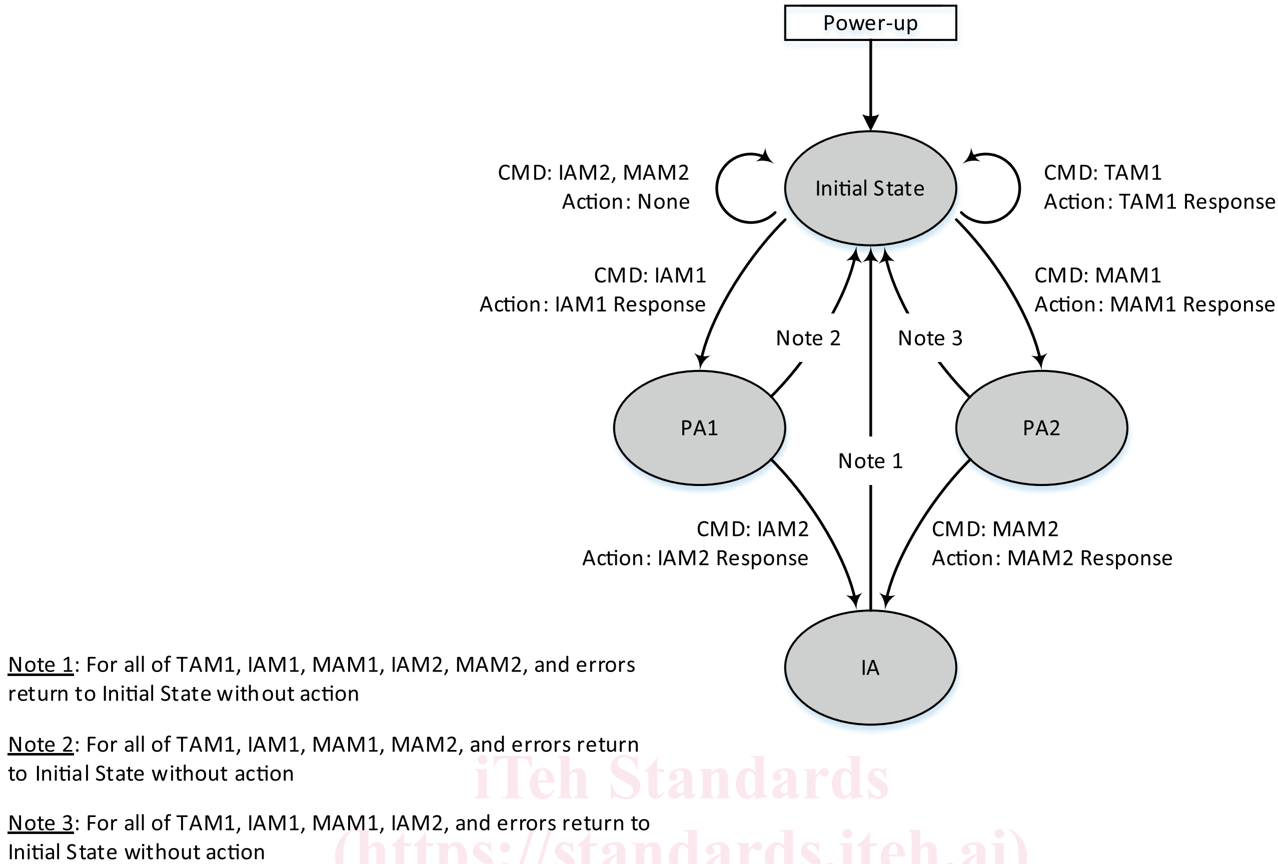

Figure 1 — Crypto suite state diagram

8 Initialization and resetting

After power-up and after a reset the cryptographic state machine transitions into the **Initial** state.

Implementations of this suite shall ensure that all memory used for any intermediate results is cleared:

— after the completion of each cryptographic protocol,

— if some cryptographic protocol is abandoned or incomplete, and

— after reset.

9 Authentication

9.1 General

This document supports Tag authentication, Interrogator authentication and Mutual authentication.

This clause describes the details of the messages and responses that are exchanged between the Interrogator and Tag for each of the authentication methods.

9.2 Message and response formatting

Messages and responses are part of the security commands described in the air interface specification. The following subclauses of this document describe the formatting of message and response for a

Tag authentication method, an Interrogator authentication method and a Tag-Interrogator mutual authentication method.

9.3 Tag authentication (AuthMethod "00")

9.3.1 General

Tag authentication uses a challenge-response protocol. See Figure 2.

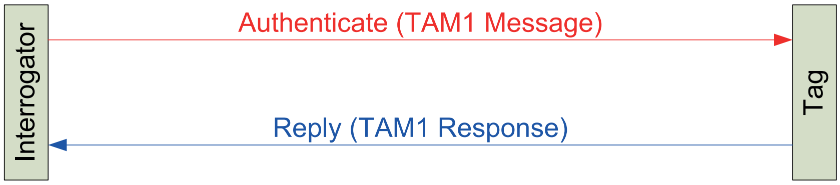

Figure 2 — Tag authentication via a challenge-response scheme

The parameter set PS defined in Table 4 gives the lengths of different fields for different block and key sizes.

NOTE The parameter set PS=00_2 closely matches other parts of the ISO/IEC 29167 series, most notably 29167-10. This provides some drop-in compatibility between SPECK-128/128 and AES-128.

Table 4 — Parameter set PS = 00_2 for Tag authentication

Parameter set PS= 00_2					
b/k	**64/96**	**64/128**	**96/96**	**128/128**	**128/256**
t **= │ IChallenge-*b/k* │**	42	42	56	80	80
r **= │ TRnd-*b/k* │**	20	20	32	32	32
c **= │ C_TAM-*b/k* │**	2	2	8	16	16

9.3.2 TAM1 message

The Interrogator shall generate a random Interrogator challenge (IChallenge-*b*/*k*) that is carried in the TAM1 message. The Interrogator shall also indicate the variant of SPECK to be used.

NOTE 1 The variant(s) of SPECK deployed on a device is (are) manufacturer dependent.

NOTE 2 Mechanisms to generate the random Interrogator challenge lie outside the scope of this document.

Table 5 — TAM1 message format

	Payload							
Field	**AuthMethod**	**Step**	**RFU**	**BlockSize**	**KeySize**	**KeyID**	**PS**	**Challenge**
Length (bits)	2	2	2	2	2	8	2	*t*
Value	00_2	00_2	00_2	00_2: *b*=64 01_2: *b*=96 10_2: *b*=128 11_2: RFU	00_2: *k*=96 01_2: *k*=128 10_2: *k*=256 11_2: RFU	variable	00_2	IChallenge-b/k

9.3.3 Intermediate Tag processing

The Tag shall accept the TAM1 message at any time (unless occupied by internal processing and not capable of receiving messages); *i.e.* upon receipt of the message with valid parameters, the Tag shall abort any cryptographic protocol that has not yet been completed and shall remain in the **Initial** state.

The Tag shall check if the Step is "00_2". If the value of Step is different, the Tag shall return a "Not Supported" error.

The Tag shall check if the RFU is "00_2". If the value of RFU is different, the Tag shall return a "Not Supported" error.

The Tag shall check whether the values of BlockSize and KeySize are supported by the Tag. If at least one of these checks is failed, the Tag shall return a "Not Supported" error.

The Tag shall check whether the values of BlockSize and KeySize are supported by Key.KeyID and that Key.KeyID is authorized for use in Tag authentication. If either (or both) of these checks is (are) failed, the Tag shall return a "Not Supported" error.

The Tag shall check whether the parameter set PS is supported. If the parameter set PS is not supported, the Tag shall return a "Not Supported" error.

Assuming that the TAM1 message is successfully parsed by the Tag, the Tag shall prepare the TAM1 response.

9.3.4 TAM1 response

The Tag shall generate a random salt TRnd-*b*/*k* of length *r* bits were *r* is given for the parameter set in Table 3.

The Tag shall use Key.KeyID and SPECK encryption to form a *b*-bit string TResponse such that:

TResponse = SPECK-b/k-ENC (Key.KeyID, C_TAM-*b*/*k* || TRnd-*b*/*k* || IChallenge-*b*/*k*).

The Tags shall return TResponse to the Interrogator.

NOTE 1 Only one input block of *b* bits is encrypted and so only one invocation of SPECK-*b*/*k* is required.

NOTE 2 Appropriate mechanisms to generate TRnd-*b*/*k* lie outside the scope of this document.

Table 6 — TAM1 response format

	Payload
Field	**Tag Response**
Length (bits)	*b*
Value	TResponse

9.3.5 Final Interrogator processing

After receiving TAM1 response the Interrogator shall use Key.KeyID to compute the *b*-bit string S where:

S = SPECK-*b*/*k*-DEC (Key.KeyID, TResponse).

1. The Interrogator shall check that S[*t*-1:0] = IChallenge-*b*/*k*.

2. The Interrogator may check that S[*b*-1:*b*-*c*] = C_TAM-*b*/*k*.

If these verification steps are successfully completed, the Interrogator may conclude that the Tag and Interrogator possess matching values of Key.KeyID. When combined with an appropriate key