
**Information technology — Personal
identification — ISO-compliant driving
licence**

**Part 3:
Access control, authentication and
integrity validation**

*Technologies de l'information — Identification des personnes — Permis
de conduire conforme à l'ISO*

Partie 3: Contrôle d'accès, authentification et validation d'intégrité



Reference number
ISO/IEC 18013-3:2009(E)

AVA Documents
Document Sharing Hub
© ISO/IEC 2009

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.



COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2009

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

Contents

Page

Foreword.....	iv
Introduction	v
1 Scope	1
2 Conformance	1
3 Normative references	1
4 Terms and definitions.....	3
5 Abbreviated terms	6
6 Functional requirements.....	7
6.1 Access control	7
6.2 Document authentication.....	7
6.3 Data integrity validation	7
7 Mapping of mechanisms to requirements and technologies.....	10
8 Mechanisms	13
8.1 Passive authentication	13
8.2 Active authentication.....	18
8.3 Scanning area identifier	19
8.4 Non-match alert.....	25
8.5 Basic access protection.....	28
8.6 Extended access protection	30
9 Security mechanism indicator.....	31
10 SIC LDS	31
10.1 EF.SOD – Document security object (short EF identifier = '1D', Tag = '77')	33
10.2 EF.DG12 Non-match alert (short EF identifier= '0C', Tag = '71')	33
10.3 EF.DG13 Active authentication (short EF identifier = '0D', Tag = '6F').....	33
10.4 EF.DG14 Extended access protection (short EF identifier = '0E', Tag = '6E')	33
Annex A (informative) Public key infrastructure (PKI)	34
Annex B (normative) Basic access protection.....	43
Annex C (normative) Extended access protection	82
Annex D (normative) SIC command set.....	106
Annex E (normative) List of tags used.....	108
Annex F (normative) Brainpool curves	109
Bibliography	117

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.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. 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 document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 18013-3 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and personal identification*.

ISO/IEC 18013 consists of the following parts, under the general title *Information technology — Personal identification — ISO-compliant driving licence*:

- *Part 1: Physical characteristics and basic data set.* Part 1 defines the basic terms for ISO/IEC 18013, including physical characteristics, basic data element set, visual layout, and physical security features.
- *Part 2: Machine-readable technologies.* Part 2 defines the technologies that may be used for ISO/IEC 18013, including the logical data structure and data mapping for each technology.
- *Part 3: Access control, authentication and integrity validation.* Part 3 defines the electronic security features that may be incorporated under ISO/IEC 18013, including mechanisms for controlling access to data, verifying the origin of an ISO-compliant driving licence, and confirming data integrity.

Introduction

This part of ISO/IEC 18013 prescribes requirements for the implementation of mechanisms to control access to data recorded in the machine-readable technology on an ISO-compliant driving licence (IDL), verifying the origin of an IDL, and confirming data integrity.

One of the functions of an IDL is to facilitate international interchange. Whilst storing data in machine-readable form on the IDL supports this function by speeding up data input and eliminating transcription errors, certain machine-readable technologies are vulnerable to being read without the knowledge of the card holder and to other means of unauthorized access by unintended persons, that is other than driving licence or law enforcement authorities. Controlling access to IDL data stored in machine-readable form protects the data on the card from being read remotely by electronic means without the knowledge of the card holder.

Identifying falsified driving licences, or an alteration to the human-readable data on authentic driving licences present a major problem for driving licence and law enforcement authorities, both domestically and in the context of international interchange. Verifying the authenticity of an IDL and confirming the integrity of the data recorded on an IDL provide driving licence and law enforcement authorities with a means to identify an authentic IDL from a falsified or altered one in the interests of traffic law enforcement and other traffic safety processes.

Information technology — Personal identification — ISO-compliant driving licence —

Part 3: Access control, authentication and integrity validation

1 Scope

ISO/IEC 18013 establishes guidelines for the design format and data content of an ISO-compliant driving licence (IDL) with regard to human-readable features (ISO/IEC 18013-1), machine-readable technologies (ISO/IEC 18013-2), and access control, authentication and integrity validation (ISO/IEC 18013-3). It creates a common basis for international use and mutual recognition of the IDL without impeding individual countries/states to apply their privacy rules and national/community/regional motor vehicle authorities in taking care of their specific needs.

This part of ISO/IEC 18013

- a) is based on the machine-readable data content specified in ISO/IEC 18013-2;
- b) specifies mechanisms and rules available to issuing authorities (IAs) for
 - 1) access control (i.e. limiting access to the machine-readable data recorded on the IDL),
 - 2) document authentication (i.e. confirming that the document was issued by the claimed IA),
 - 3) data integrity validation (i.e. confirming that the data has not been changed since issuing).

This part of ISO/IEC 18013 does not address issues related to the subsequent use of data obtained from the IDL, e.g. privacy issues.

2 Conformance

A driving licence is in conformance with this part of ISO/IEC 18013 if it meets all mandatory requirements specified directly or by reference herein. Compliance with ISO/IEC 18013-2 is required for compliance with this part of ISO/IEC 18013.

Compliance with ISO/IEC 18013-1 is not required for compliance with this part of ISO/IEC 18013. Conversely, the incorporation of a machine-readable technology which is not compliant with this part of ISO/IEC 18013 does not render the IDL non-compliant with ISO/IEC 18013-1.

3 Normative references

The following referenced documents are indispensable for the application 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 1831:1980, *Printing specifications for optical character recognition*

ISO/IEC 7816-4:2005, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*

ISO/IEC 7816-8:2004, *Identification cards — Integrated circuit cards — Part 8: Commands for security operations*

ISO/IEC 8825-1:2002, *Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)*

ISO/IEC 8859-1:1998, *Information technology — 8-bit single-byte coded graphic character sets — Part 1: Latin alphabet No. 1*

ISO/IEC 9797-1:1999, *Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher*

ISO/IEC 10118-3:2004, *Information technology — Security techniques — Hash-functions — Part 3: Dedicated hash-functions*

ISO/IEC 11770-2:1996, *Information technology — Security techniques — Key management — Part 2: Mechanisms using symmetric techniques*

ISO/IEC 11770-2:1996/Cor.1:2005, *Information technology — Security techniques — Key management — Part 2: Mechanisms using symmetric techniques — Corrigendum 1*

ISO/IEC 11770-3, *Information technology — Security techniques — Key management — Part 3: Mechanisms using asymmetric techniques*

ISO/IEC 18013-1, *Information technology — Personal identification — ISO-compliant driving licence — Part 1: Physical characteristics and basic data set*

ISO/IEC 18013-2, *Information technology — Personal identification — ISO-compliant driving licence — Part 2: Machine-readable technologies*

ISO/IEC 18033-3:2005, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers*

ISO/IEC 18033-3:2005/Cor.1:2006, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers — Corrigendum 1*

ISO/IEC 18033-3:2005/Cor.2:2007, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers — Corrigendum 2*

ANSI X9.62:2005, *Public Key Cryptography For The Financial Services Industry: The Elliptic Curve Digital Signature Algorithm (ECDSA)*

FIPS 186-2 (including Change Notice), *Digital Signature Standard (DSS)*, Federal Information Processing Standards Publication, National Institute of Standards and Technology, 27 January 2000

NIST SP 800-38B, *Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication*, May 2005

RFC 2631, E. Rescorla, *Diffie-Hellman Key Agreement Method*, June 1999, <http://www.ietf.org/rfc.html>

RFC 3279, W. Polk et al., *Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, April 2002, <http://www.ietf.org/rfc.html>

RFC 3280, R. Housley et al., *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, April 2002, <http://www.ietf.org/rfc.html>

RFC 3369, R. Housley, *Cryptographic Message Syntax*, August 2002, <http://www.ietf.org/rfc.html>

RFC 4055, J. Schaad, B. Kaliski, R. Housley, *Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*, June 2005, <http://www.ietf.org/rfc.html>

4 Terms and definitions

For the purposes of this document, the terms and definitions given in ISO/IEC 18013-1, ISO/IEC 18013-2 and the following apply.

4.1

active authentication

mechanism that uses information stored in a secure area of a secure integrated circuit (SIC) to confirm that the SIC and the other machine-readable data were issued together

NOTE See 8.2.

4.2

basic access protection

BAP

mechanism to confirm that an inspection system (IS) has physical access to a proximity integrated circuit card (PICC) before the IS is allowed access to the data stored on the PICC and to ensure that communication between the IS and the PICC (once access is authorized) is protected

NOTE See 8.5 and Annex B.

4.3

chip authentication

ephemeral-static key agreement protocol that provides authentication of the secure integrated circuit and strong secure messaging

NOTE See 8.6 and C.3.

4.4

clone

unauthorized exact copy of a document that has the same security characteristics as the original document and that cannot be distinguished from the legitimate one

4.5

eavesdropping

unauthorized interception and interpretation of information-bearing emanations

NOTE Adapted from ISO/IEC 2382-8:1998, 08.05.25.

4.6

extended access protection

EAP

protocol for limiting access to select optional data groups to reading authorities

NOTE See 8.6.

4.7

input string

string of characters printed on an ISO-compliant driving licence [as human-readable text, optionally (or by specification) accompanied by or consisting of a machine-readable rendering thereof] used as input (either manually or automatically through the use of suitable equipment) for the non-match alert and basic access protection mechanisms

4.8

issuing authority

IA

licensing authority, or issuing country if separate licensing authorities have not been authorized, which applies a digital signature to an ISO-compliant driving licence and is responsible for the associated key management

NOTE Adapted from ISO/IEC 18013-1.

4.9

non-match alert

mechanism to detect any differences between the machine-readable information and (some of) the human-readable information on an ISO-compliant driving licence

NOTE See 8.4.

4.10

passive authentication

mechanism to confirm that machine-readable data on an ISO-compliant driving licence (IDL) has not been changed since the IDL was issued

NOTE See 8.1.

4.11

pseudo issuing authority

PIA

authority that does not issue ISO-compliant driving licences [but that is similar to an issuing authority (IA) in all other respects] and which does not issue document keys, but which does have a root key pair with which it can sign documents of other IAs or PIAs that it trusts

4.12

public key infrastructure

PKI

technologies and products using public key (asymmetric) cryptography

NOTE Both passive authentication and extended access protection use this technology.

4.13

reading authority

RA

authorized entity reading the machine-readable data on an ISO-compliant driving licence (IDL)

NOTE Driving licence authorities other than the authority that issued the IDL and law enforcement authorities are examples of reading authorities.

4.14

reference string

string of characters used as a reference against which to compare the input string when using the non-match alert mechanism, and used for session key calculation purposes by the secure integrated circuit during execution of the basic access protection mechanism

4.15

scanning area identifier

SAI

one or more graphical elements that demarcate an input string

4.16

secure integrated circuit

SIC

integrated circuit that includes both a security feature (or security features), and memory and/or a central processing unit

NOTE 1 An integrated circuit card with contacts and a proximity integrated circuit card (PICC) are examples of a SIC.

NOTE 2 A SIC can be embedded in different solutions, for example in ID-1 sized cards (as used for the ISO-compliant driving licence) and in a booklet (as found in passports).

4.17

secure memory

integrated circuit (IC) memory of which the content (once populated by an issuing authority during the personalization process) is accessible only by the IC operating system for internal use, and cannot be made available by the operating system to any reading device

4.18

skimming

reading data from a proximity integrated circuit card (PICC) without the card holder's awareness

4.19

trust chain

sequential set of trust points that a verifying authority references to verify a specific issuing authority's public root key

4.20

trust model

description of the functional and logical aspects of a traditional public key infrastructure, specifically excluding technical implementation details

4.21

trust network

component of a trust model that describes the trust relationships and chains between issuing authorities

4.22

trust point

issuing authority or pseudo issuing authority that publishes a trust list (and the related public root keys) that verifying entities can reference

4.23

twinning

copying the data and/or integrated circuit of a physically and/or biometrically similar driver to the attacker's integrated circuit or ISO-compliant driving licence

4.24

unpacked BCD

binary coding of a sequence of integers using 4 bits for each integer (where the bit weights are 8421) and encoding one integer in the least significant bits of each byte

NOTE Only unsigned BCD is used in this part of ISO/IEC 18013.

4.25

verifying authority

VA

verifying entity that is part of a trust network, i.e. that also is an issuing authority or a pseudo issuing authority

NOTE 1 Not all verifying entities are VAs: A car rental company can be a verifying entity, but is not a VA as it is not part of the trust network.

NOTE 2 VAs can be divided into immediate VAs and non-immediate VAs.

4.25.1

immediate VA

VA that acquired the public root key of the issuing authority via out-of-band means

4.25.2

non-immediate VA

VA that acquired the public root key of the issuing authority from another VA

4.26

verifying entity

entity that tries to determine if a digital signature is valid (i.e. if the data to which a certificate has been applied has not been changed, and if the signature was generated by the issuing authority the verifying entity expects)

5 Abbreviated terms

APDU	application protocol data unit
BAC	basic access control
BAP	basic access protection
BCD	binary coded decimal
BER-TLV	basic encoding rules – tag-length-value (see ISO/IEC 8825-1:2002)
CA	certification authority
CBC	cipher block chaining
DER	distinguished encoding rules (see ISO/IEC 8825-1:2002)
DF	dedicated file
DG	data group
DO	data object
DST	control reference template for digital signature (see ISO/IEC 7816-4:2005)
EAP	extended access protection
EF	elementary file
IA	issuing authority
IC	integrated circuit
ICC	integrated circuit card
IDL	ISO-compliant driving licence
IS	inspection system
IV	initialization vector
KAT	control reference template for key agreement (see ISO/IEC 7816-4:2005)
LDS	logical data structure
MAC	message authentication code
MSE	manage security environment (see ISO/IEC 7816-4:2005)
OCR	optical character recognition
OID	object identifier
PIA	pseudo issuing authority
PIC	proximity integrated circuit

PICC	proximity integrated circuit card
PKI	public key infrastructure
PSO	perform security operation (see ISO/IEC 7816-4:2005)
RA	reading authority
SAI	scanning area identifier
SIC	secure integrated circuit
SM	secure messaging
SOD	document security object
SSC	send sequence counter
TRCA	trust root certificate authority
UTC	coordinated universal time
VA	verifying authority
2D	two-dimensional

6 Functional requirements

6.1 Access control

Access control can be broken down into the following functional requirements:

- a) Prevent skimming of machine-readable data on a PICC by ensuring that physical access to the IDL is acquired prior to reading.
- b) Prevent unnoticed alteration of communication between a reader and a SIC.
- c) Prevent eavesdropping between a reader and a SIC.
- d) Selectively restrict access to specific optional machine-readable data groups for specific reading authorities.

6.2 Document authentication

Document authentication can functionally be established by allowing for verification of the origin of an IDL.

6.3 Data integrity validation

Data integrity validation can be broken down into the following functional requirements:

- a) Verify that the IDL (including the machine-readable data) is not a clone of another IDL. A cloning attempt can schematically be illustrated as shown in Figure 1.

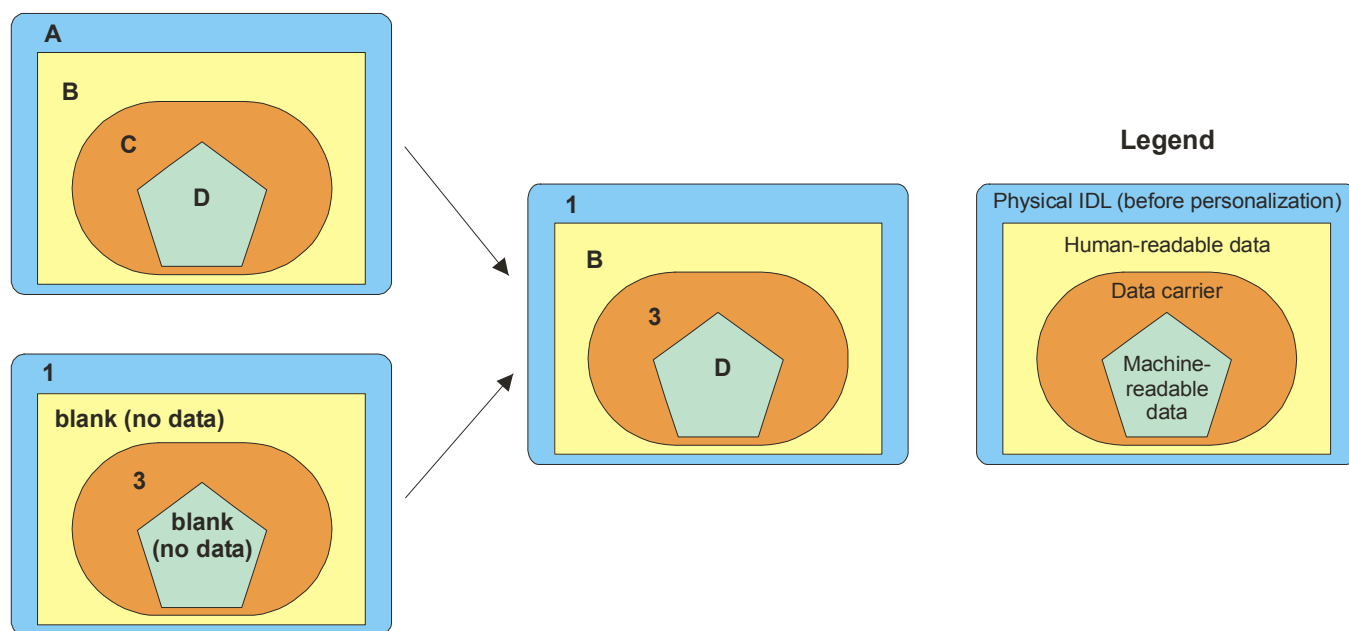


Figure 1 — Data integrity validation: IDL cloning

- b) Protect against the exchange of machine-readable data carriers between otherwise authentic IDLs¹. This type of attack can schematically be illustrated as shown in Figure 2.

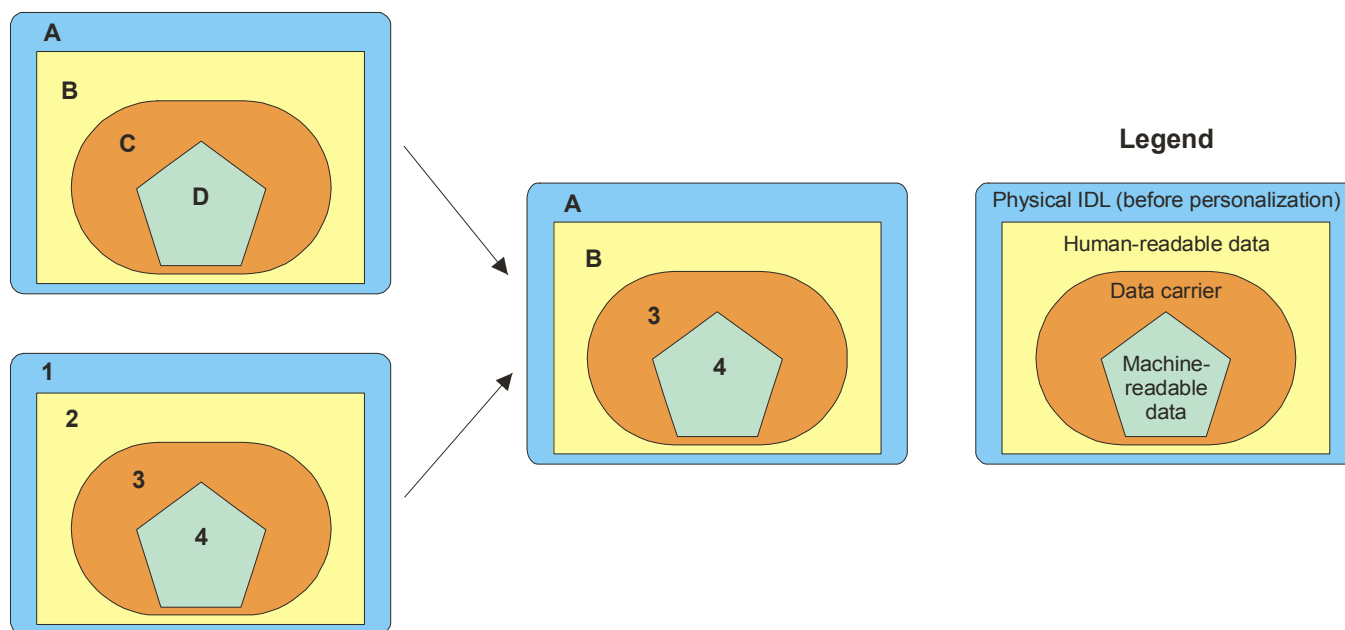


Figure 2 — Data integrity validation: Data carrier exchange or twinning

¹ This guards (amongst others) against an IC "twinning" attack. This type of attack is of particular concern in inspection environments where machine-readable data and human-readable data is not compared (or only cursorily compared by an operator using e.g. a portrait image). Finding a biometrically similar driver is possible by skimming the data of a few thousand IDL PICs.

- c) Verify that the physical IDL and the machine-readable data thereon were issued (belong) together. This type of attack can schematically be illustrated as shown in Figure 3.

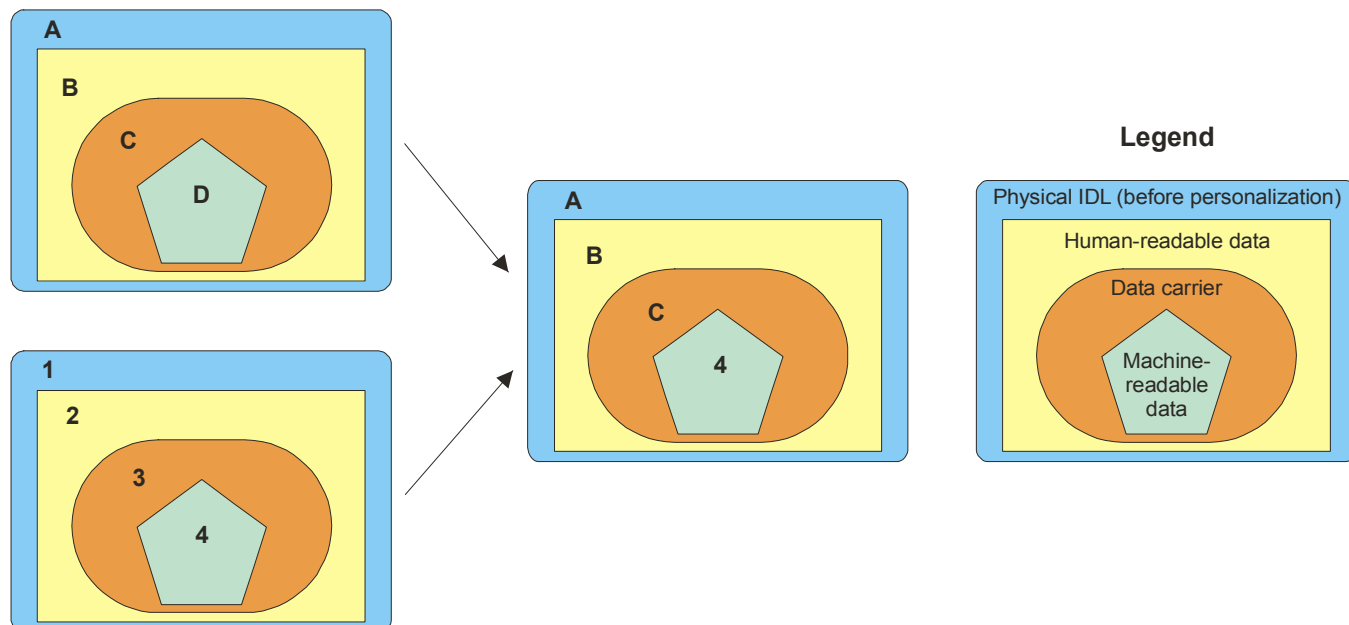


Figure 3 — Data integrity validation: Machine-readable data exchange

- d) Validate the integrity of the human readable data (i.e. confirm that the human-readable data has not changed since issuing). This type of attack can schematically be illustrated as shown in Figure 4.

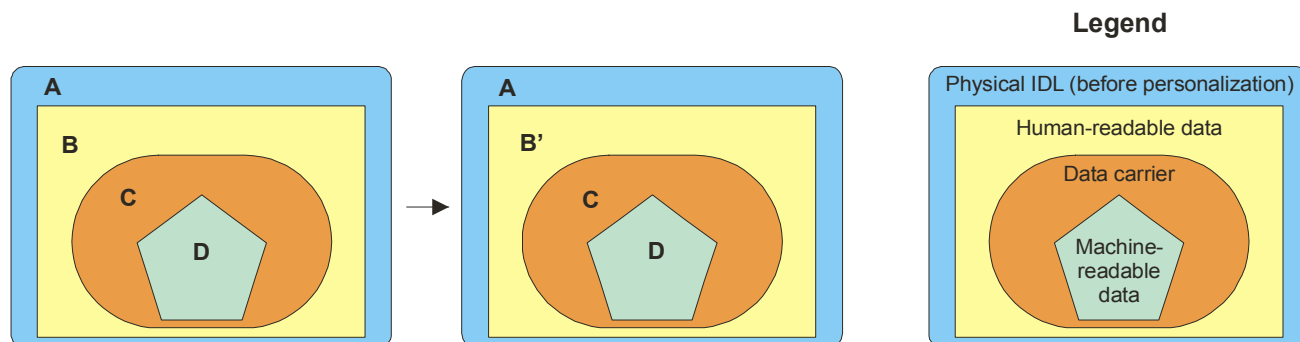


Figure 4 — Data integrity validation: Human-readable data alteration

- e) Validate the integrity of the machine-readable data (i.e. confirm that the machine-readable data has not changed since issuing). This can schematically be illustrated as shown in Figure 5.

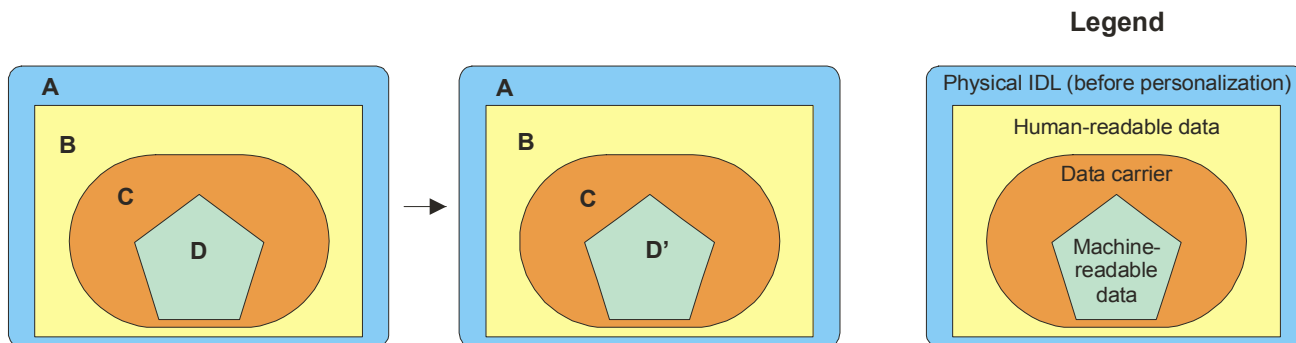


Figure 5 — Data integrity validation: Machine-readable data alteration

7 Mapping of mechanisms to requirements and technologies

Mechanisms that, when used individually, can address specific functional requirements are specified in Clause 8. Mechanisms can also be used together to address a broader range of functional requirements. Not all machine-readable technologies have the same functional requirements, and some mechanisms are applicable only to specific machine-readable technologies. Table 1 specifies the relationships between the mechanisms, machine-readable technologies, and functional requirements.

Table 1 — Applicable mechanisms

Functional requirement	Mechanisms		
	2D bar code, optical memory, magnetic stripe	ICC with contacts	PICC
Prevent skimming	Physical access to the IDL is required to read the machine-readable data	Physical access to the IDL is required to read the machine-readable data	BAP
Prevent unnoticed alteration of communication between a reader and an IDL	N/A	Chip authentication (EAP) in combination with BAP secure messaging	BAP Chip authentication (EAP) in combination with BAP secure messaging
Prevent eavesdropping	N/A	Chip authentication (EAP) in combination with BAP secure messaging	BAP ^a Chip authentication (EAP) in combination with BAP secure messaging
Selectively restrict access to specific optional machine-readable data groups for specific reading authorities	N/A	Terminal authentication (EAP)	Terminal authentication (EAP)
Allow for verification of the origin of an IDL	Step 1: Verify trustworthiness of / obtain a trustworthy asymmetric/public key (via a PKI and a protected distribution communication channel, or via alternative trust building methods) Step 2: Use trusted key to perform passive authentication, followed by use of the non-match alert mechanism, or followed by visual comparison of the human-readable data printed on the document with the authenticated machine-readable data	Step 1: Verify trustworthiness of / obtain a trustworthy asymmetric/public key (via a PKI and a protected distribution communication channel, or via alternative trust building methods) Step 2: Use trusted key to perform passive authentication; if the passive authentication was preceded by BAP the IDL is authentic, if BAP is not implemented passive authentication can be followed by use of the non-match alert mechanism, or followed by visual comparison of the human-readable data printed on the document with the authenticated machine-readable data	Step 1: Verify trustworthiness of / obtain a trustworthy asymmetric/public key (via a PKI and a protected distribution communication channel, or via alternative trust building methods) Step 2: Use trusted key to perform passive authentication; if the passive authentication was preceded by BAP the IDL is authentic, if BAP is not implemented passive authentication can be followed by use of the non-match alert mechanism, or followed by visual comparison of the human-readable data printed on the document with the authenticated machine-readable data
Verify that the IDL (including the machine-readable data) is not a clone of another IDL	2D bar code and magnetic stripe: Cannot be verified through available international standards Optical memory: Include unique card serial number in data protected by passive authentication	Active authentication via challenge response Chip authentication (EAP)	Active authentication via challenge response Chip authentication (EAP)

Functional requirement	Mechanisms		
	2D bar code, optical memory, magnetic stripe	ICC with contacts	PICC
Protect against the exchange of machine readable data carriers between otherwise authentic IDLs	Non-match alert Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data
Verify that the physical IDL and the machine-readable data thereon were issued (belong) together (i.e. that the machine-readable data has not been copied from another IDL)	Non-match alert Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data Active authentication via challenge response ^a	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data Active authentication via challenge response ^a
Validate the integrity of the human readable data (i.e. confirm that the human-readable data has not changed since issuing)	Non-match alert Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data Visual inspection of the security features incorporated into the IDL (see ISO/IEC 18013-1)	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data Visual inspection of the security features incorporated into the IDL (see ISO/IEC 18013-1)	Non-match alert BAP Passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data Visual inspection of the security features incorporated into the IDL (see ISO/IEC 18013-1)
Validate the integrity of the machine-readable data (i.e. confirm that the machine-readable data has not changed since issuing)	Passive authentication	Passive authentication	Passive authentication
NOTE Visual comparison of human-readable information printed on an IDL with machine-readable information obtained from the same IDL is not formally specified as a mechanism in this part of ISO/IEC 18013; it is assumed that such comparison can be implemented by any RA in a manner that meets local needs.			
^a The entropy of the reference string used has to be commensurate with the IA's assessment of the threat.			

IA's may selectively implement the mechanisms identified above, subject to any dependencies noted in Clause 8.

8 Mechanisms

8.1 Passive authentication

8.1.1 Purpose

The purpose of passive authentication is to confirm that machine-readable data has not been changed since the IDL was issued.

8.1.2 Applicability

Passive authentication is applicable to all machine-readable technologies.

8.1.3 Description

Passive authentication is implemented by way of a digital signature over specified machine-readable data on the IDL, using a public-private (asymmetric) key pair.

In the case of standard encoding, a separate message digest is calculated for each data group, and included in the machine-readable data. The collection of message digests is then digitally signed (using a private key that is kept secret by the IA), and the digital signature is added to the machine-readable data.

In the case of compact encoding, no message digests are calculated separately. The contents of the data groups present is directly signed (using a private key that is kept secret by the IA), and the digital signature is added to the machine-readable data.

NOTE A message digest has the following properties:

- a) It is very small in size compared to the IDL data.
- b) The probability of finding any two (different) IDL data sets that lead to the same message digest is negligible. This has the following implications:
 - i. The probability of finding an IDL data set A that produces the same message digest as a given IDL data set B is negligible.
 - ii. The probability that a message digest (for the data on an IDL) remains the same upon a change in the data is negligible.

When the IDL is presented to a RA, the RA uses the IA's public key to verify the digital signature. The RA also computes the message digests of each of the data groups that it is interested in, and compares them to the corresponding message digests stored in the machine-readable data. If:

- a) the digital signature verifies,
- b) the calculated message digests are the same as the message digests stored in the machine-readable data, and
- c) the RA is confident that the public key used to verify the digital signature belongs to the claimed IA,

the RA can consider the data groups that it is interested in to be authentic. If the digital signature does not verify, either an incorrect public key was used, or the data on the IDL has been changed. Depending on the digital signature method used, it may be possible to further narrow down the cause of the non-verification.

This part of ISO/IEC 18013 does not prescribe methods to obtain and/or to establish trust in public keys. It is the responsibility of each RA to obtain and/or to establish trust in the public keys used to verify a digital signature on an IDL. Informative methods and approaches to establish such trust are however provided in

Annex A, which describes the principles for a PKI that may be used for public key distribution in the absence of one global certification authority.

This part of ISO/IEC 18013 does not prescribe methods for the generation, administration and safekeeping of key pairs. It is the responsibility of each issuing jurisdiction to ensure that keys are generated, administered and protected as necessary.

8.1.4 Hash function

8.1.4.1 Standard encoding

For standard encoding, IA's shall choose the SHA-1, SHA-224, SHA-256, SHA-384 or the SHA-512 hash function.

NOTE SHA-256 is recommended. SHA-1 remains for compatibility with ICAO Doc 9303-1.

A message digest is calculated separately for each data group present, and stored in the machine-readable data (see 8.1.5.1). The same hash function is used for all data groups. A message digest for a data group is calculated on the concatenation of those data elements present in the data group in the order specified in ISO/IEC 18013-2.

NOTE This approach allows reading authorities to read only those data groups it is interested in.

8.1.4.2 Compact encoding

For compact encoding, IA's shall not calculate separate message digests for each data group. Therefore, no hash function is specified for compact encoding (except as required as part of any digital signature mechanism – see 8.1.5.3).

8.1.5 Signing method

8.1.5.1 Standard encoding

An IDL digital signature is generated over the concatenation of the message digests of the data groups present.

IA's may use either ECDSA or RSA as digital signature methods for standard encoding.

IA's using RSA shall use RFC 4055. RFC 4055 specifies two signature mechanisms, RSASSA-PSS and RSASSA-PKCS1-v1_5. It is recommended to generate signatures according to RSASSA-PSS, but reading authorities shall also be prepared to verify signatures according to RSASSA-PKCS1-v1_5. The minimum size of the modulus, n , shall be 1024 bits.

IA's implementing ECDSA shall use ANSI X9.62. The elliptic curve domain parameters used to generate the ECDSA key pair shall be described explicitly in the parameters of the public key, i.e. parameters shall be of type ECParameters (no named curves, no implicit parameters) and shall include the optional cofactor. ECPoints shall be in uncompressed format. The minimum size for the base point order shall be 160 bits.

In addition to EF.COM and the data groups specified in ISO/IEC 18013-2, IA's shall add the SOD to accommodate the hashes of the individual data groups (see 8.1.4.1) and the digital signature of the data on the IDL. The SOD is implemented as a `SignedData` Type, as specified in RFC 3369 (including processing rules). The SOD shall be produced in DER format.

Table 2 — SignedData Type

Value	Type ^a	Comments
SignedData	m	
Version	m	
digestAlgorithms	m	
encapContentInfo	m	
eContentType	m	id-icao-ldsSecurityObject
eContent	m	The encoded contents of an ldsSecurityObject
certificates	o	
Crls	x	
signerInfos	m	
SignerInfo	m	
version	m	
Sid	m	
issuerandSerialNumber	c	It is recommended that IA's support this field over subjectKeyIdentifier.
subjectKeyIdentifier	c	
digestAlgorithm	m	The algorithm identifier of the algorithm used to produce the hash value over encapsulatedContent and SignedAttrs.
signedAttrs	m	IA's may wish to include additional attributes for inclusion in the signature, however these do not have to be processed by a RA except to verify the signature value.
signatureAlgorithm	m	The algorithm identifier of the algorithm used to produce the signature value and any associated parameters.
signature	m	The result of the signature generation process.
unsignedAttrs	o	IA's may wish to use this field, but it is not recommended and reading authorities may choose to ignore them.
^a m = mandatory (the field shall be present); x = do not use (the field shall not be populated); o = optional (the field may be present); c = choice (the field contents is a choice from alternatives)		

ASN.1 sequence

```
LDSSecurityObject { joint-iso-itu-t(2) international-organizations(23) icao(136) mrt(1)
security(1) ldsSecurityObject(1) }
```

```
DEFINITIONS IMPLICIT TAGS ::=
```

```
BEGIN
```

```
-- Imports from RFC 3280 [PROFILE], Appendix A.1
```

```
AlgorithmIdentifier FROM
```

```
PKIX1Explicit88 { iso(1) identified-organization(3) dod(6)
internet(1) security(5) mechanisms(5) pkix(7)
id-mod(0) id-pkix1-explicit(18) }
```

```
-- Constants
```

```
ub-DataGroups INTEGER ::= 16
```

```
-- Object Identifiers
id-icao OBJECT IDENTIFIER ::= {2.23.136}
id-icao-mrtd OBJECT IDENTIFIER ::= {id-icao 1}
id-icao-mrtd-security OBJECT IDENTIFIER ::= {id-icao-mrtd 1}
id-icao-ldsSecurityObject OBJECT IDENTIFIER ::= {id-icao-mrtd-security 1}

-- LDS Security Object

LDSSecurityObjectVersion ::= INTEGER {V0(0)}

DigestAlgorithmIdentifier ::= AlgorithmIdentifier

LDSSecurityObject ::= SEQUENCE {
    version LDSSecurityObjectVersion,
    hashAlgorithm DigestAlgorithmIdentifier,
    dataGroupHashValues SEQUENCE SIZE (2..ub-DataGroups) OF DataGroupHash }

DataGroupHash ::= SEQUENCE {
    dataGroupNumber DataGroupNumber,
    dataGroupHashValue OCTET STRING }

DataGroupNumber ::= INTEGER {
    dataGroup1 (1),
    dataGroup2 (2),
    dataGroup3 (3),
    dataGroup4 (4),
    dataGroup5 (5),
    dataGroup6 (6),
    dataGroup7 (7),
    dataGroup8 (8),
    dataGroup9 (9),
    dataGroup10 (10),
    dataGroup11 (11),
    dataGroup12 (12),
    dataGroup13 (13),
    dataGroup14 (14),
    dataGroup15 (15),
    dataGroup16 (16) }
END
```

NOTE 1 The field `dataGroupHashValue` contains the calculated hash over the complete contents of the Data Group EF, specified by `dataGroupNumber`.

NOTE 2 Data Groups 15 and 16 may be defined in future.

8.1.5.2 Standard encoding for optical memory

The DER encoded SOD defined in 8.1.5.1 shall be stored as one data object in tag 15140.

In order to allow identification of clone attempts, IA's may optionally include the card serial number (found at Track 4, Sector 14, in 43-byte sector format; see ISO/IEC 11694-4) in DG3 using tag 1500. With reference to Table D.11 in ISO/IEC 18013-2, tag 1500 shall follow tag 1438 (Optional Data Discriminator) in sequence.

8.1.5.3 Compact encoding

An IDL digital signature is generated over the full data stream from the start of DG1 (including the data group delimiter between DG1 and the header) to the end of DG12 (including the data group delimiter that terminates DG12, but excluding DG.SOD, if present). The digital signature is stored in DG.SOD.

NOTE The header is not included in the information to be signed as the length information in the header pertains to DG11 as well.

IA's shall use ECDSA (as defined in ANSI X9.62) as a signing method for compact encoding, and in order to reduce the storage requirements for the domain parameters, shall use one of the curves specified in FIPS 186-2, Appendix 6.

DG.SOD shall consist of a concatenation of two Type 2 data groups and one Type 1 data group, storing the following information:

- a) DG.SOD.1: Digital signature, shall be the DER encoded ASN.1 sequence of two integers, r and s :

$$\text{SEQUENCE} ::= \{ r \text{ INTEGER}, s \text{ INTEGER} \}$$
- b) DG.SOD.2: Public key; octet string representation of the public point in uncompressed form according to X9.62
- c) DG.SOD.3: Curve identifier

DG.SOD shall be added after DG12, as follows:

[header] × [Data Group 1] × [Data Group 2] × [Data Group 3] × [Data Group 4] × [Data Group 7] × [Data Group 11] × [Data Group 12] × [DG.SOD.1 length] [digital signature] × [DG.SOD.2 length] [public key] × [DG.SOD.3: named curve] ¶

The inclusion of DG.SOD.2 and DG.SOD.3 (as a pair) is optional. Length information is encoded using ASN.1 rules.

NOTE Reading authorities should be able to verify (or obtain) a public key (and domain parameters) from the IA using the data in other data groups (specifically the ISO issuer ID number and document discriminator in DG3 and the licence number and date of issue in DG1).

The curve identifier shall consist of one byte, as shown in Table 3.

Table 3 — Curve identifiers

Curve identifier	Curve name in FIPS 186-2	Curve identifier	Curve name in Annex F
'01'	P-192	'11'	brainpoolP160r1
'02'	P-224	'12'	brainpoolP160t1
'03'	P-256	'13'	brainpoolP192r1
'04'	P-384	'14'	brainpoolP192t1
'05'	P-521	'15'	brainpoolP224r1
'06'	Reserve for future use	'16'	brainpoolP224t1
'07'	Reserve for future use	'17'	brainpoolP256r1
'08'	Reserve for future use	'18'	brainpoolP256t1
		'19'	brainpoolP320r1
		'1A'	brainpoolP320t1
		'1B'	brainpoolP384r1
		'1C'	brainpoolP384t1
		'1D'	brainpoolP512r1
		'1E'	brainpoolP512t1

EXAMPLE Suppose that a compact encoded data string contains the following data groups: DG1, DG2, DG7 and DG.SOD.1. A digital signature is included, but the public key and curve identifier are not included. The sequence of data groups and data group delimiters will be as follows:

[header] × [DG1] × [DG2] × × × [DG7] × × × [DG.SOD.1] ¶

8.2 Active authentication

8.2.1 Purpose

Active authentication uses information stored in a secure area of an SIC to confirm that the SIC and the other machine-readable data were issued together.

8.2.2 Applicability

This mechanism is limited to SICs.

8.2.3 Description

A challenge-response protocol matches the private and public keys of an SIC-individual key pair. The private key is stored in the SIC's secure memory and cannot be copied. The public key is stored as part of the signed data on the SIC. (The challenge-response protocol thus has to be preceded by passive authentication.) The SIC (or more specifically the SIC operating system) can prove knowledge of the SIC-individual private key in a challenge-response protocol. In this protocol the SIC digitally signs a challenge randomly chosen by the IS. The IS is convinced that the SIC is genuine if and only if the returned signature is correct.

NOTE Active authentication using a challenge-response protocol is open to a "traceability" attack. The challenge-response protocol does not place any limitation on the format or content of the challenge. If the challenge includes time, date, location, and a secret key, a log of challenge responses can be used to track IDLs and furthermore to prove presence (due to the inclusion of the secret key). The counterargument is that basic access protection would safeguard against unknown read attempts, and that a driver's location is known in any case in those instances where an IDL is handed over for inspection.

8.2.4 Mechanism

8.2.4.1 General

The SIC-individual public key shall be stored in DG13, an additional data group specifically intended for use with the active authentication mechanism. The format of the structure (`SubjectPublicKeyInfo`) is specified in RFC 3280. `SubjectPublicKeyInfo` shall be produced in DER format as follows:

```
ActiveAuthenticationPublicKeyInfo ::= SubjectPublicKeyInfo
```

```
SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm AlgorithmIdentifier,
    subjectPublicKey BIT STRING }
```

```
AlgorithmIdentifier ::= SEQUENCE {
    algorithm OBJECT IDENTIFIER,
    parameters ANY DEFINED BY algorithm OPTIONAL }
```

Allowed algorithm object identifiers and parameters are specified in RFC 3279.

Active authentication shall be performed using the ISO7816 INTERNAL AUTHENTICATE command as defined by ISO/IEC 7816-4:2005. The input shall be a terminal-generated nonce (RND.IDF) that shall be 8 bytes in length. The SIC computes a signature which shall be sent back to the IS. The IS shall check the response and verify the signature.

8.2.4.2 Signature generation using RSA

The signature shall be computed according to ISO 9796-2 Digital Signature Scheme 1. M shall consist of M1 and M2, where M1 is an SIC-generated nonce of c-4 bits and M2 is RND.IFD. The SHA-1 hashing algorithm and trailer option 1 shall be used. The result of the signature generation shall be the signature Σ without the recoverable message part M2.

8.2.4.3 Signature generation using ECC

The signature shall be computed according to ANSI X9.62 ECDSA. SHA-1 shall be used as hashing algorithm. The result of the signature generation shall be the DER encoded ASN.1 sequence of two integers, r and s :

```
SEQUENCE ::= { r INTEGER, s INTEGER }.
```

8.2.4.4 Security mechanism indicator

SICs which implement active authentication may optionally indicate this in the security mechanism indicator (see 9). The object identifier shall be `id-sm-AA`.

```
id-sm-AA OBJECT IDENTIFIER ::= {
    iso(1) standard(0) driving-licence(18013) part-3(3)
    security-mechanisms(2) 3}
```

The parameters are mandatory and shall be of type `param-AA`.

```
param-AA ::= SEQUENCE {
    version INTEGER,
    publicKeyDG INTEGER}
```

The version shall be set to `v1(0)` for this version of active authentication. The `publicKeyDG` field shall be set to the number of the data group of the active authentication public key, i.e. 13.

8.3 Scanning area identifier

8.3.1 Applicability

This mechanism is used with the non-match alert or BAP mechanisms, which collectively are applicable to SICs, 2D barcode, magnetic stripe and optical memory (see 8.4 and 8.5).

8.3.2 Description

8.3.2.1 General

The primary purpose of this mechanism is to facilitate machine assisted automatic or interactive verification procedures that match machine-readable data to human-readable data. This mechanism is not a solution on its own, but is intended for use as a component of either of the following mechanisms:

- a) Non-match alert (see 8.4), which checks if the machine-readable and human-readable data belong together.
- b) BAP (see 8.5), which ensures that a PICC cannot be read without physical access to the IDL.

NOTE: The BAP mechanism also performs the function of the non-match alert mechanism, but potentially at a different level of security.

The SAI demarcates the input string. The input string can be entered manually by an operator, or can be read automatically using machine-readable technologies.

The SAI and any machine-readable information demarcated by it shall be adapted for reading in the B900 infrared band defined in ISO 1831:1980. Under such illumination the printing background and any overlays

within the SAI will backscatter light in a homogeneous way (i.e. the SAI shall contain no optically variable device, local deviations in properties, security features, or surface gloss exceeding natural transparent overlay gloss, that will interfere with readability). In addition, machine-readable printed information (e.g. bar codes and OCR characters) shall comply with the associated standards. The brightness of the printing background (including the effects of overlays) will not be less than 40% compared to an ideal white surface under the same illumination. Human-readable information inside the SAI shall be rendered such that it can be read without difficulty.

The contrast ratio between character lines and the background shall be a minimum of 4:1. Any reflective layer covering the SAI area shall not further deteriorate this contrast value when illuminated under angles of incidence up to 45 degrees and looked at under angles more than 10 degrees outside of the nominal glance angle.

The presence of the SAI is identified as follows:

- a) If PICC access is required and the access conditions are unsatisfied (i.e. BAP is in place), the reader searches for a SAI on the IDL.
- b) If access to DG12 is available, BAP is not applicable and the non-match alert mechanism is present. DG12 may indicate the location of the SAI, alternatively, the reader searches for a SAI on the IDL.

Not more than one SAI shall be present on a single IDL.

The reference string (and the input string, when stored in a barcode) shall be encoded in accordance with ISO/IEC 8859-1:1998.

The content of the SAI can follow one of three standards, i.e. based on an existing text field, containing a dedicated text field, or containing a barcode.

8.3.2.2 SAI content based on existing text field

The SAI may be constructed around an existing field on an IDL. In this case the SAI shall consist of a double lined rectangle as illustrated by the example in Figure 6.

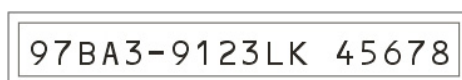


Figure 6 — SAI around an existing field (not to scale)

Each line shall have a thickness of $0.2\text{mm} \pm 0.1\text{mm}$, and the distance between the centre of each line shall be $0.6\text{mm} \pm 0.1\text{mm}$. The lines shall be black in colour. The clear distance between the inside line and the extremities of the input string shall be at least 1mm.

The input string shall be printed using OCR-B size 1 or a character set with the same symbols and character shapes but using a reduced size of either 25% or 50% reduction of all linear dimensions compared to OCR-B size 1.

To prevent confusion about the input sequence of the characters, text within the rectangle shall be limited to one line. The data element name (if reflected on the IDL) and/or the data field reference code shall not be included within the rectangle.

In order to facilitate manual entry of the input string when needed, the input string may contain only Latin capital letters (hexadecimal range 41 – 5A of ISO/IEC 8859-1:1998²), Latin small letters (hexadecimal range 61 – 7A of ISO/IEC 8859-1:1998²), N characters, and the S characters shown in Table 4.

NOTE Latin capital letters (hexadecimal range 41 – 5A of ISO/IEC 8859-1:1998) are recommended.

² References to ISO/IEC 8859-1:1998 are only for purposes of identifying the characters. Encoding methods are specified elsewhere.

Table 4 — S characters allowed for SAI content based on existing field

S character	Hexadecimal number in ISO/IEC 8859-1:1998 ^a
<space>	20
–	2D
<	3C

^a Reference to ISO/IEC 8859-1:1998 is only for the purpose of identifying the characters. Encoding methods are specified separately.

The reference string shall not contain any space characters (i.e. hexadecimal character '20' of ISO/IEC 8859-1:1998). The IS thus shall delete all space characters (hexadecimal code 20 of ISO/IEC 8859-1:1998) from the input string before further processing.

NOTE 1 IA's should carefully consider the level of entropy of existing fields before use within the SAI. Any rules followed in the construction of the field will decrease the entropy.

NOTE 2 IA's considering the use of an SAI based on an existing field should keep in mind that a check digit is not provided for in this case.

EXAMPLE Figure 7

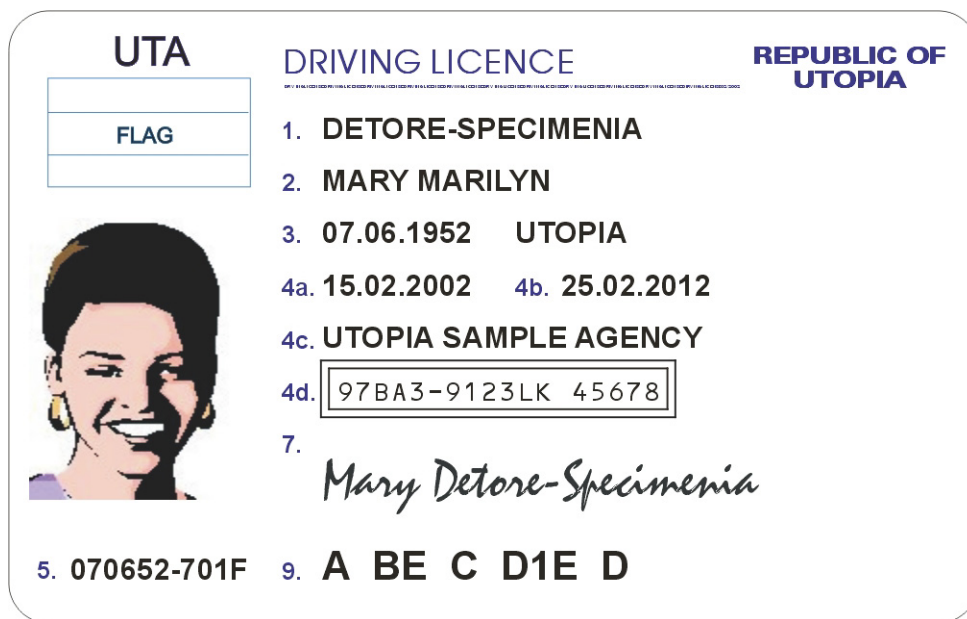


Figure 7 — Existing data field on portrait side of IDL designated as input string (not to scale)

8.3.2.3 SAI content consisting of a dedicated text field

A dedicated field (i.e. a field designed specifically for this purpose) may be used within a SAI. In this case, the SAI shall consist of a rectangle constructed by a black single line with a thickness of $0.65 \text{ mm} \pm 0.1 \text{ mm}$. The clear distance between the line and the extremities of the printed input string shall be at least 1mm.

EXAMPLE Figure 8

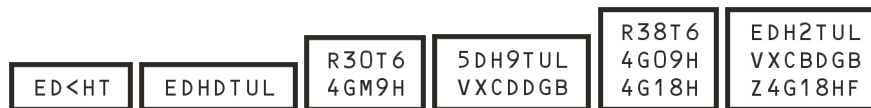


Figure 8 — SAI around a dedicated field (not to scale)

The content of the SAI shall comply with the following requirements:

- a) The text shall be limited to the set of ICAO MRZ characters, i.e.: 0 1 2 3 4 5 6 7 8 9 A B C D E F G H I J K L M N O P Q R S T U V W X Y Z <
- b) An odd number of characters per line shall be used.
- c) If characters are printed in more than one line, each line shall have the same number of characters.
- d) A line shall not have more than seven characters.
- e) No more than three lines shall be used.
- f) The text shall be printed using OCR-B font at 100% of OCR-B size 1.
- g) The middle character of each line shall be reserved for a check digit. Before adding the check digit(s), the input string may have the following lengths:
 - i. For one line only, string lengths of 4 or 6 characters are permitted.
 - ii. For two lines, string lengths of 8 or 12 characters are permitted.
 - iii. For three lines 12 or 18 characters are permitted.

To construct an input string (with check digits) consisting of i lines, the input string (before check digits are added) is parsed into i parts of equal length. The leftmost part is used to form line 1, the following parts each to form an additional line. Each part is split in the middle into a left and a right subsection. The left subsection forms the leftmost part of the corresponding line, the right subsection forms the rightmost part of the line. An additional character position is inserted in the middle of each line (for the check digit). The value of the check digit of each line will be calculated as explained further below.

For the purpose of check digit calculation each character in the input string (before the check digits are added) represents a numerical value V_C defined by the formula $V_C = E_C$, where E_C is a unique number allotted to each character value, as defined in Table 5.

Table 5 — Character numerical equivalents

Char	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F	G	H	I
E_C	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18
Char	J	K	L	M	N	O	P	Q	R	S	T	U	V	W	X	Y	Z	<	
E_C	19	20	21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	

If line 1 is the only line then the unique check digit will be calculated as follows:

- a) A check sum S is calculated as $S = \sum V_{Ci}$, where i runs from 0 to $n-1$ and V_{Ci} indicates the respective value V_C for the character in the i -th position in the input string of length n , counted from right to left, starting with position 0 for the rightmost character.

- b) From S the corresponding check digit c is calculated as $c = S \text{ modulo } 37$; (integer remainder when dividing S by 37).

If there are two or three lines, then the check digits c_1 for line one and c_2 for line two will be calculated as follows:

- a) A check sum S_V is calculated over the entire input string (before it is split into lines, and before adding the check digits) as $S_V = \sum V_{Ci}$, where i runs from 0 to $n-1$ and indicates the character in the i -th position in the input string of length n , counted from right to left, starting with position 0 for the rightmost character.
- b) A check sum S_P is calculated over the entire input string (before it is split into lines, and before adding the check digits) as $S_P = \sum (V_{Ci} * i)$, where i runs from 0 to $n-1$ and i indicates the i -th position number in the input string of length n , counted from right to left, starting with position 0 for the rightmost character, and V_{Ci} indicates the respective value for the character in the i -th position.
- c) Check digits c_1 and c_2 are calculated as $c_1 = (S_V \text{ modulo } 37)$ and $c_2 = (S_P \text{ modulo } 37)$ respectively.

In the case of exactly three lines the inserted character c_3 in the centre of line three will represent the version number and will be set to "1" for this version of the standard. Version numbers are not supported for less than three lines.

When printed on an IDL, lines are numbered from top to bottom, as shown in Figure 9.



Figure 9 — Line configuration (not to scale)

EXAMPLE Input string for Figure 9 is EDHTULVXCDBZ4G8HF

Input string	E	D	H	T	U	L	V	X	C	D	G	B	Z	4	G	8	H	F
E_C	14	13	17	29	30	21	31	33	12	13	16	11	35	4	16	8	17	15
i	17	16	15	14	13	12	11	10	9	8	7	6	5	4	3	2	1	0
$V_{Ci} * i$	238	208	255	406	390	252	341	330	108	104	112	66	175	16	48	16	17	0

$S_V = 335$

$S_P = 3082$

Check digit	Numeric value	Code value
c_1	2	2
c_2	11	B

$c_3 = 1$

The input string submitted as input to the non-match alert or BAP mechanisms is the input string inclusive of the check digit, i.e. the concatenation $s_1 + s_2 + s_3$ where s_i is the string in line i of the corresponding input string, including the respective check digit and read from the left to the right, with i ranging from 1 to a maximum of 3.

NOTE Mathematically, the check digit scheme allows identifying any single digit reading error in the case of a one-line input string and to correct any single digit reading error as well as identifying any double error in the case of a two- or three-line input string. Less than 3% of any other statistical reading error may pass inadvertently. This is only true if adequate measures are taken on the side of the reading equipment to make use of the information offered by the check digit scheme.

It is recommended that the minimum entropy of the input string be 40 bits or more (before addition of the check digits). The use of random data is recommended.

NOTE IA's may consider imposing restrictions on allowable content of input strings to meet local requirements, e.g. to address the "nasty word" problem, or to limit the value range of practically existing check digits (e.g. numerical characters only). Methods exist that allow enforcement of such restrictions without violation of the rules set out above. The use of such methods is left to the discretion of the IA's. Attention is drawn to the fact that such restrictions will reduce the entropy of the input string.

EXAMPLE Figure 10

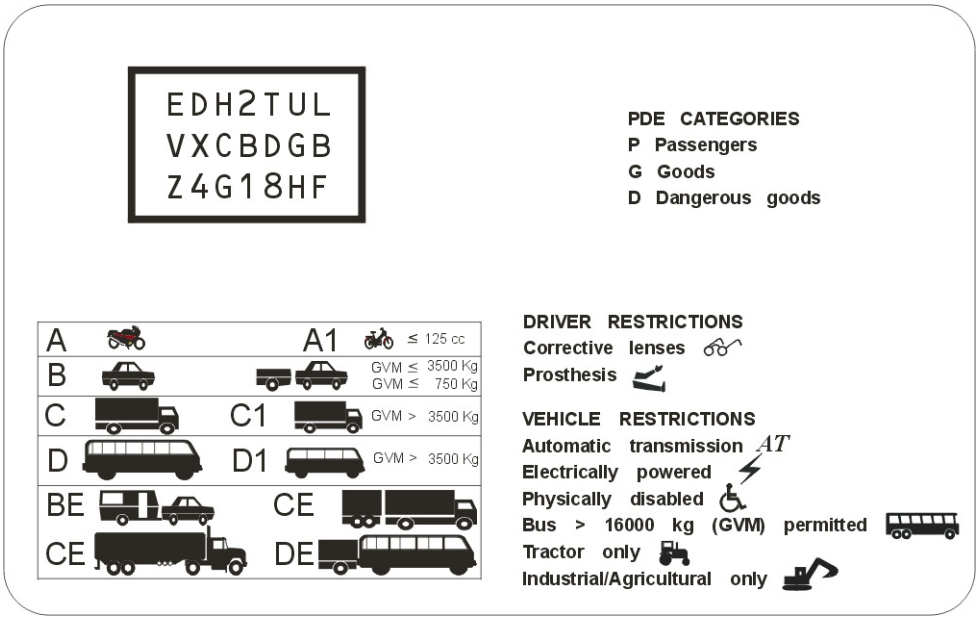


Figure 10 – Dedicated data field on non-portrait side of IDL designated as input string (not to scale)

8.3.2.4 SAI content consisting of a barcode

A barcode may be used within a SAI as the primary means of input. The SAI's corners shall be clearly shown. Each corner indicator shall be constructed of two perpendicular lines joined at the end points, with each line having a thickness of 0.5mm ±0.3mm, and a length of 4mm ±2mm.

EXAMPLE Figure 11



Figure 11 — SAI around a dedicated field (not to scale)

The input string may contain any A, N or S characters, and is not limited in length. However, it is recommended that the input string be limited to the characters specified in 8.3.2.3 to facilitate manual entry when required. The barcode may be accompanied by a human-readable version of the input string (in accordance with ISO/IEC 8859-1:1998), printed in a font not smaller than 6pt.

EXAMPLE Figure 12

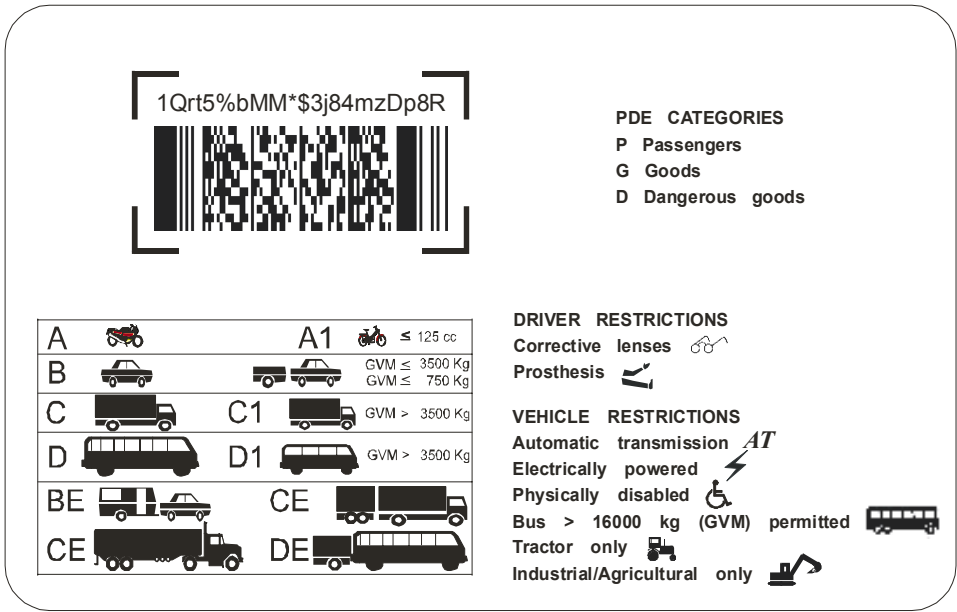


Figure 12 – Barcode on non-portrait side of IDL designated as input string (not to scale)

8.4 Non-match alert

8.4.1 Purpose

This mechanism allows detection of any differences between the machine-readable information and (some of) the human-readable information (printed on an IDL). This mechanism may be of value in the presence of machine-readable technologies with data content not accessible to visual inspection and on data carriers or physical support that may be transferred completely from one IDL to another without strong visual evidence of alteration.

8.4.2 Applicability

This mechanism is applicable to SICs, 2D barcode, magnetic stripe and optical memory.

8.4.3 Description

Conceptually, the reference string is verified via a digital signature, after which the reference string is compared to the input string. If the comparison is not successful, it means that the machine-readable data and the (human-readable) printed data are not consistent.

NOTE 1 It is assumed that the RA trusts the public key used to verify the digital signature.

NOTE 2 If the comparison is successful, the RA may conclude that the machine-readable data and the physical document are consistent if it can be assumed that alteration to any part of the printed data (i.e. not just the area that contains the input string) would be visually apparent.

The reference string is not separately verified with a dedicated digital signature, but is verified during passive authentication (the digital signature used for passive authentication already covers the reference string).

NOTE The function provided by the non-match alert mechanism can also be achieved by performing passive authentication followed by a visual comparison of the human-readable data printed on the document with the authenticated machine-readable data.

8.4.4 Mechanism

8.4.4.1 General

The manner in which the non-match alert mechanism is implemented is specified in DG12, using the parameters and values as shown in Table 6.

Table 6 — Non-match alert parameters

Name, Fixed (F) or Variable (V), Mandatory (M) or Optional (O)	Field format/ length/type	Example
SAI_referencestring, V, M	Byte 1: '00' if the input string follows '01' if a reference to where the input string can be obtained follows Subsequent bytes: If byte 1 = '00', the input string follows from byte 2 (inclusive). The input string is encoded in accordance with ISO/IEC 8859-1:1998. If byte 1 = '01', the reference to the field that contains the input string is constructed as aabb where aa is the data group and bb is the sequence number of the referenced data element, with aabb encoded as unsigned BCD	An input string of ABC4DEF contained in the SAI_referencestring field will be coded as '00 41 42 43 34 44 45 46', where '41 42 43 34 44 45 46' is the encoded form of ABC4DEF. If the licence number is used as the input string, this will be coded as '01 01 08'. If the 16th data element of Data Group 12 is used as the input parameter, the input string will be coded as '01 12 16'
SAI_inputmethod, V, O	Byte 1: SAI standard and input method. The four most significant bits (upper nibble) of byte 1 can take on any of the following values: '0x' if the input string is based on an existing field '1x' if the input string is based on a dedicated field '2x' if the input string is stored in a barcode The four least significant bits (lower nibble) of byte 1 denotes the input method, and can take on any of the following values: 'x0' if the input string is intended for manual input 'x1' if the input string is intended for OCR interpretation 'x2' if the input string is stored as a barcode. Byte 2: Barcode standard. If the first byte is of the form 'x2', byte 2 is mandatory, taking on any of the following values: '00' for PDF417 '01' for Code 39 (ISO/IEC 16388) '02' for Code 128 (ISO/IEC 15417) '03' for data matrix (ISO/IEC 16022)	If the licence number is used as the input string (i.e. a SAI is constructed around the existing licence number field on the IDL), the value of SAI_inputmethod will be '00'. If, in addition, the input string is printed in OCR-B font, the input string will be '00 01', or alternatively '00 01 00'. If, in addition, the SAI is located on the portrait side of the IDL, with the top left corner of the SAI at 29mm from the left edge of the card and 24mm from the bottom edge of the card, and the right bottom corner of the SAI at 59mm from the left edge of the card and 14mm from the bottom edge of the card, the input string will be '00 01 00 00 29 24 59 14'

Name, Fixed (F) or Variable (V), Mandatory (M) or Optional (O)	Field format/ length/type	Example
	'FE' for other barcode standards not provided for above 'FF' no barcode Byte 2 is also mandatory if Bytes 3 to 7 are present. Bytes 3 to 7: Position of the SAI, expressed as 'aa bb cc dd ee', where 'aa' is the side of the card on which the SAI appears ('00' for portrait side, and '01' for non-portrait side), 'bb cc' is the top left corner of the SAI (where 'bb' is the distance from the left edge of the IDL and 'cc' is the distance from the bottom edge of the card), and 'dd ee' is the bottom right corner of the SAI (where 'dd' is the distance from the left edge of the IDL and 'ee' is the distance from the bottom edge of the card), with all distances measured in millimetres, and encoded as BCD. The bytes are progressively mandatory. I.e., SAI_inputmethod can consist only of byte 1, or only of bytes 1 and 2, or of bytes 1 to 7.	

8.4.4.2 Compact encoding

When used with compact encoding, DG12 shall be constructed as a Type 1 data group as follows:

... × [SAI_referencestring] ÷ [SAI_inputmethod] × ...

8.4.4.3 Standard encoding

When used with standard encoding, the parameters shall be stored in DG12 as shown in Table 7.

Table 7 — Non-match alert parameters for standard encoding

Tag	Length	Value
'82'	X	SAI_referencestring as defined in Table 6.
'81'	X	SAI_inputmethod as defined in Table 6.

IA's may optionally indicate the presence of the non-match alert mechanism in the EF.COM file using the security mechanism indicator as specified in 9.

The object identifier for non-match alert shall be `id-sm-NMA`.

```
id-sm-NMA OBJECT IDENTIFIER ::= {
    iso(1) standard(0) driving-licence(18013) part-3(3)
    security-mechanisms(2) 4
}
```

The parameters are optional and shall be of type `param-NMA`.

```
param-NMA ::= SEQUENCE {
    version INTEGER, SAI_inputmethod OCTET STRING
}
```

The `version` field shall be set to `v1(0)` for this version of the non-match alert mechanism.

The `SAI_inputmethod` field shall be set to the corresponding value in DG12. If the `SAI_inputmethod` field is not present in DG12, the field shall also not be included in EF.COM.

8.4.4.4 Standard encoding for optical memory

When used with standard encoding for optical memory, the parameters shall be stored in DG12 as shown in Table 8.

Table 8 — Non-match alert parameters for standard encoding

Tag	Value
15200	SAI_referencestring as defined in Table 6.
15201	SAI_inputmethod as defined in Table 6.

8.5 Basic access protection

8.5.1 Purpose

BAP confirms that an IS has physical access to a PICC before the IS is allowed access to the data stored on the PICC. In addition, BAP ensures that communication between the IS and the PICC (once access is authorized) is protected.

NOTE Although BAP was designed for PICCs, it can also be used to satisfy a variety of functional requirements for ICs with contacts as listed in Table 1.

8.5.2 Applicability

This mechanism is limited to PICCs and ICs with contacts.

8.5.3 Description

For a SIC that is accessed via a contactless interface and protected by the BAP mechanism shall deny access to its content by the contactless interface unless the IS can prove that it is authorized to access the SIC. This proof shall be given in a challenge-response protocol, where the IS proves knowledge of the PICC-individual document basic access keys (K_{ENC} and K_{MAC}) that are derived from the input string.

After the IS has been authenticated successfully, the SIC shall enforce encryption and message authentication of the communication channel between the IS and the SIC by Secure Messaging techniques.

8.5.4 Mechanism

The document basic access keys are stored in the internal elementary file (see ISO/IEC 7816-4:2005).

A SIC that supports BAP shall respond to unauthenticated read attempts (including selection of files in the logical data structure) with 'Security status not satisfied' (0x6982). The presence of BAP thus is determined by an IS from the response to read attempts (see below) and the subsequent success in locating the SAI.

BAP is specified in Annex B. The following shall be used in the application of Annex B:

- a) The reference string shall be used as the document keying material (K_{doc}).
- b) The IS can automatically read the input string, or can allow an operator to manually enter the input string (if present) into the IS.
- c) The first byte of the input string shall identify the BAP configuration used.

NOTE BAP is intended mainly as an anti-skimming mechanism. However, provided that the entropy of the reference string is commensurate with the IA's assessment of the threat, this mechanism can also serve as protection against eavesdropping.

IA's may optionally add the BAP logo to an IDL to indicate that the SAI contains a BAP input string. Figure 13 shows the BAP logo. The minimum dimensions of the A-dimension of the logo shall be 5mm, and the B-dimension shall be scaled proportionally in a ratio of 5:3. It is recommended that the BAP logo be placed in close proximity to the associated SAI.

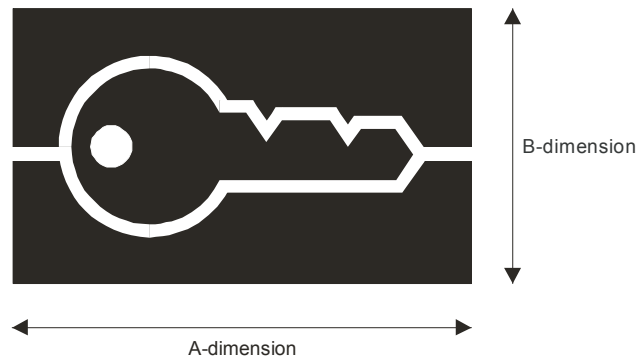


Figure 13 — Basic access protection logo (not to scale)

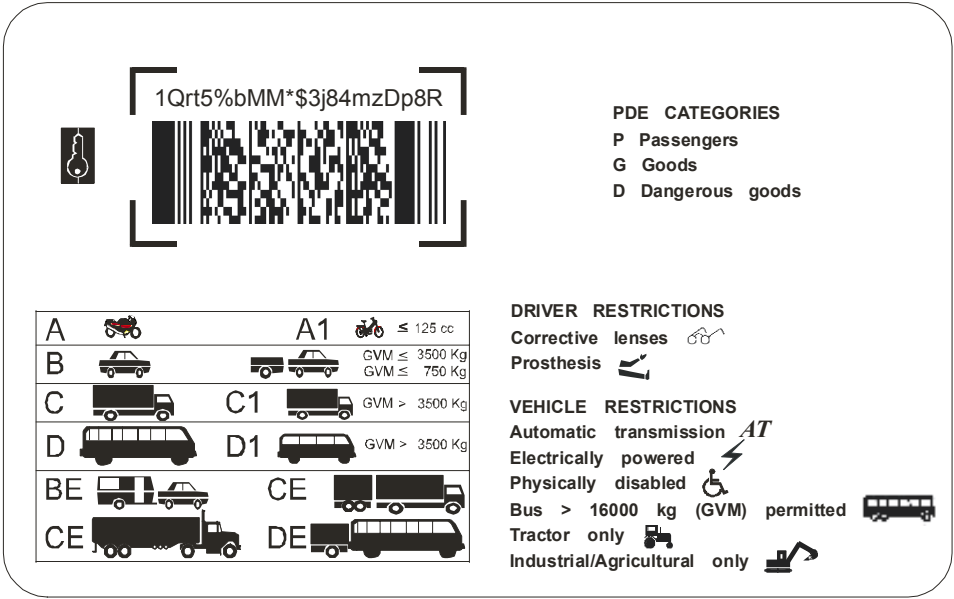


Figure 14 — Basic access protection logo on IDL (not to scale)

EXAMPLE Suppose the following barcode is printed on the IDL:



This barcode encodes the string “1462483345434115654434034118361284817041”, whose hexadecimal rendering is ‘31 34 36 32 34 38 33 33 34 35 34 33 34 31 31 35 36 35 34 34 33 34 30 33 34 31 31 38 33 36 31 32 38 34 38 31 37 30 34 31’

The first byte (‘31’) of the input string indicates BAP configuration 1. The entire input string is used as K_{doc} . The resulting derived document access keys are:

K_{enc} : ‘E4 80 0E 99 54 FF 39 EB F6 09 15 FD 46 C4 3F DD’

K_{mac} : ‘E7 EA 15 4F ED 39 38 77 A1 12 9D CB 7A 54 93 50’

The barcode in this example contains 40 numeric digits. As the first byte conveys predictable information, only the following 39 bytes may contribute entropy. Assuming they were generated randomly, the resulting K_{doc} contains $\log_2(10^{39}) \approx 129$ bits of entropy.

8.6 Extended access protection

8.6.1 Purpose

EAP consists of:

- a) Chip authentication, which provides for authentication of the SIC and strong secure messaging.
- b) Terminal authentication, which provides for conditional authenticated access to data groups.

8.6.2 Applicability

This mechanism is applicable only to SICs.

8.6.3 Description and mechanism

Extended access protection is specified in Annex C. The following shall be used in the application of Annex C:

- a) The SIC's key agreement public key shall be stored in DG14, formatted in accordance with C.3.4.
- b) The driving licence number, as it appears in DG1, shall be used as the SIC identifier.
- c) Strong secure messaging (established using chip authentication as described in C.3) shall be active before terminal authentication can take place.
- d) DG14 shall be accessible without terminal authentication.

NOTE Both BAP and EAP use the card commands GET CHALLENGE (INS = '84') and MUTUAL/EXTERNAL AUTHENTICATE (INS = '82'). Since a driving licence may implement both protocols, the two cases need to be distinguished in such a case. As per c) above, strong secure messaging that has been established through chip authentication shall be active before EAP Terminal Authentication can take place. Thus these two commands are used for EAP only if secure messaging is active (CLA = 0x0C), otherwise they are used for BAP.

9 Security mechanism indicator

The security mechanism(s) deployed on an IDL with an SIC can be identified³ to reading authorities by the addition of tag '86' to EF.COM. Tag '86' identifies the data groups subject to each mechanism used. Tag '86' shall have the following DER-encoded ASN.1 TLV structure:

```
SecurityMechanisms ::= SET OF SecurityMechanism
```

```
SecurityMechanism ::= SEQUENCE {
    mechanism MechanismIdentifier,
    datagroups SET OF INTEGER}
```

```
MechanismIdentifier ::= SEQUENCE {
    mechanism OBJECT IDENTIFIER,
    parameters ANY DEFINED BY mechanism OPTIONAL}
```

EXAMPLE Assume an implementation using LDS version 1.0 having the following data content – mandatory data (DG 1) and optional licence holder data (DG 2). The optional licence holder data is protected using EAP (see Annex C) with the current trust root set to the example in C.A.1, no alternate trust root, and BAP configuration 1 (see Annex B) as secure messaging configuration. EF.COM would be encoded as follows (in order to improve clarity, tags are printed in *RED ITALICS*; lengths are printed in UNDERLINED BLUE and values are printed in BLACK):

```
'60' '57'
'5F 01' '02'
'01 00'
'5C' '02'
'61 6B'
'86' '4C'
'31 4A 30 48 30 41 06 07 28 81 8C 5D 03 02 02 30
36 02 01 00 02 01 0E 06 08 28 81 8C 5D 03 02 01
01 04 11 10 C0 0D 1D 69 61 15 35 70 67 0B 2C 3A
EE 64 FF 36 04 11 00 00 00 00 00 00 00 00 00 00
00 00 00 00 00 00 00 00 31 03 02 01 02'
```

where '31 4A 30 46 ... 01 02' is the following DER-encoded ASN.1 structure:

```
SET
  SEQUENCE
    SEQUENCE
      OBJECT IDENTIFIER: 1.0.18013.3.2.2
      SEQUENCE
        INTEGER: 00
        INTEGER: 0E
        OBJECT IDENTIFIER: 1.0.18013.3.2.1.1
        OCTET STRING: 10C00D1D6961153570670B2C3AEE64FF36
        OCTET STRING: 00000000000000000000000000000000
    SET
      INTEGER: 02
```

10 SIC LDS

In addition to the data groups identified in Figure C.1 in ISO/IEC 18013-2, this part of ISO/IEC 18013 defines the data groups shown in Table 9.

³ Use of the security mechanism indicator is optional unless a mechanism mandates the security mechanism to be used.

Table 9 — Assignment of file identifiers and Data Group Tags

Elementary file	Name	Short EF Identifier	EFID	Tag
EF.SOD	Document security object	'1D'	'001D'	'77'
EF.DG12	Non-match alert	'0C'	'000C'	'71'
EF.DG13	Active authentication	'0D'	'000D'	'6F'
EF.DG14	Extended access protection	'0E'	000E	'6E'

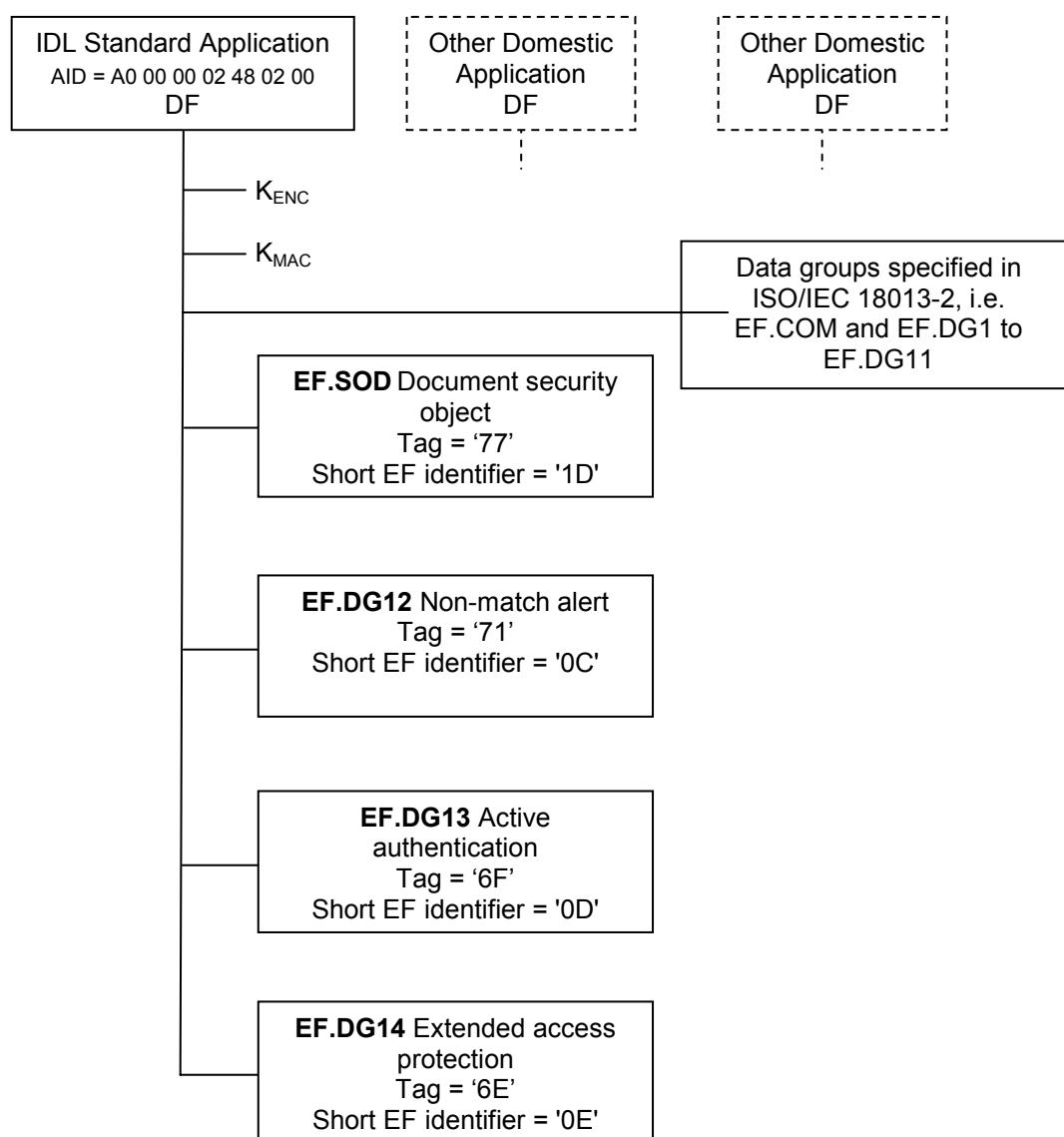


Figure 15 — Data groups