BS ISO/IEC 18013-5:2021

BSI Standards Publication

Personal identification — ISO-compliant driving licence

Part 5: Mobile driving licence (mDL) application

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 18013-5:2021

National foreword

This British Standard is the UK implementation of ISO/IEC 18013-5:2021.

The UK participation in its preparation was entrusted to Technical Committee IST/17, Cards and security devices for personal identification.

A list of organizations represented on this committee can be obtained on request to its committee manager.

Contractual and legal considerations

This publication has been prepared in good faith, however no representation, warranty, assurance or undertaking (express or implied) is or will be made, and no responsibility or liability is or will be accepted by BSI in relation to the adequacy, accuracy, completeness or reasonableness of this publication. All and any such responsibility and liability is expressly disclaimed to the full extent permitted by the law.

This publication is provided as is, and is to be used at the recipient's own risk.

The recipient is advised to consider seeking professional guidance with respect to its use of this publication.

This publication is not intended to constitute a contract. Users are responsible for its correct application.

© The British Standards Institution 2021
Published by BSI Standards Limited 2021

ISBN 978 0 580 90608 4

ICS 35.240.15

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 October 2021.

Amendments/corrigenda issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 18013-5:2021

INTERNATIONAL STANDARD

ISO/IEC 18013-5

First edition
2021-09

Personal identification — ISO-compliant driving licence —

Part 5: Mobile driving licence (mDL) application

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

Partie 5: Application permis de conduire sur téléphone mobile

Reference number
ISO/IEC 18013-5:2021(E)

Documents
Document Sharing Hub
© ISO/IEC 2021

Full document details: BS ISO/IEC 18013-5:2021

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2021

All rights reserved. Unless otherwise specified, or required in the context of its implementation, no part of this publication may be reproduced or utilized otherwise in any form or by any means, electronic or mechanical, including photocopying, or posting on the internet or an intranet, without prior written permission. Permission can be requested from either ISO at the address below or ISO's member body in the country of the requester.

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

Published in Switzerland

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 18013-5:2021

Contents

Page

Foreword ... **v**
Introduction ... **vi**
1 Scope ... **1**
2 Normative references ... **1**
3 Terms and definitions ... **3**
4 Abbreviated terms ... **5**
5 Conformance requirement ... **6**
6 mDL overview ... **6**
6.1 Interfaces ... 6
6.2 Functional requirements ... 7
6.3 Technical requirements ... 8
6.3.1 Data model ... 8
6.3.2 Data exchange ... 8
6.3.3 Security mechanisms ... 13
7 mDL data model ... **15**
7.1 mDL document type and namespace ... 15
7.2 mDL data ... 16
7.2.1 Overview ... 16
7.2.2 Portrait of mDL holder ... 21
7.2.3 Issuing authority ... 21
7.2.4 Categories of vehicles/restrictions/conditions ... 21
7.2.5 Age attestation: nearest “true” attestation above request ... 22
7.2.6 Biometric template ... 23
7.2.7 Signature or usual mark ... 23
7.2.8 Domestic data elements ... 23
7.3 Country codes ... 23
8 Transaction ... **23**
8.1 Encoding of data structures and data elements ... 23
8.2 Device engagement ... 24
8.2.1 Device engagement information ... 24
8.2.2 Device engagement transmission technology ... 26
8.2.3 Device engagement time-out ... 28
8.3 Data retrieval ... 29
8.3.1 Data model ... 29
8.3.2 Data retrieval methods ... 29
8.3.3 Data retrieval transmission technologies ... 36
9 Security mechanisms ... **47**
9.1 Device retrieval ... 47
9.1.1 Session encryption ... 47
9.1.2 Issuer data authentication ... 49
9.1.3 mdoc authentication ... 52
9.1.4 mdoc reader authentication ... 55
9.1.5 Session transcript and cipher suite ... 56
9.2 Server retrieval ... 58
9.2.1 TLS ... 58
9.2.2 JWS ... 58
9.3 Validation and inspection procedures ... 59
9.3.1 Inspection procedure for issuer data authentication ... 59
9.3.2 Inspection procedure for JWS ... 59
9.3.3 Certificate validation procedure ... 60

Documents
Document Sharing Hub

© ISO/IEC 2021 – All rights reserved

Full document details: BS ISO/IEC 18013-5:2021

Annex A (informative) **BLE L2CAP transmission profile** **61**
Annex B (normative) **Certificate and CRL profiles** **62**
Annex C (informative) **Verified issuer certificate authority list (VICAL) provider** **90**
Annex D (informative) **Data structure examples** **112**
Annex E (informative) **Privacy and security recommendations** **135**
Annex F (informative) **IANA Considerations** **149**
Bibliography **153**

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work.

The procedures used to develop this document and those intended for its further maintenance are described in the ISO/IEC Directives, Part 1. In particular, the different approval criteria needed for the different types of document should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives or www.iec.ch/members_experts/refdocs).

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights. Details of any patent rights identified during the development of the document will be in the Introduction and/or on the ISO list of patent declarations received (see www.iso.org/patents) or the IEC list of patent declarations received (see https://patents.iec.ch).

Any trade name used in this document is information given for the convenience of users and does not constitute an endorsement.

For an explanation of the voluntary nature of standards, the meaning of ISO specific terms and expressions related to conformity assessment, as well as information about ISO's adherence to the World Trade Organization (WTO) principles in the Technical Barriers to Trade (TBT) see www.iso.org/iso/foreword.html. In the IEC, see www.iec.ch/understanding-standards.

This document was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and security devices for personal identification*.

A list of all parts in the ISO/IEC 18013 series can be found on the ISO and IEC websites.

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 and www.iec.ch/national-committees.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Introduction

The ISO/IEC 18013 series 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), ISO machine-readable technologies (ISO/IEC 18013-2), access control, authentication and integrity validation (ISO/IEC 18013-3), and associated test methods (ISO/IEC 18013-4). It creates a common basis for international use and mutual recognition of the IDL without impeding individual countries/states in applying their privacy rules and national/community/regional motor vehicle authorities in taking care of their specific needs.

This document describes interface and related requirements to facilitate ISO-compliant driving licence (IDL) functionality on a mobile device. The requirements are specifically intended to enable verifiers not affiliated with or associated with the issuing authority to gain access to and authenticate the information. In addition, the requirements allow the holder of the driving licence to decide what information to release to a verifier. Other characteristics include the ability to update information frequently, and to authenticate information at a high level of confidence.

A mobile document conforming to this document primarily conveys the driving privileges associated with a person. It does so by providing means to associate the person presenting the mobile document with the mobile document itself. However, the transaction and security mechanisms in this document have been designed to support other types of mobile documents, specifically including identification documents. Consequently the mechanisms in this document can be used for any type of mobile identification document, regardless of the additional attributes the mobile document may convey. The details of the data elements to be used by other mobile documents are left to the respective issuing authority and are not within the scope of this document.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Personal identification — ISO-compliant driving licence —

Part 5:
Mobile driving licence (mDL) application

1 Scope

This document establishes interface specifications for the implementation of a driving licence in association with a mobile device. This document specifies the interface between the mDL and mDL reader and the interface between the mDL reader and the issuing authority infrastructure. This document also enables parties other than the issuing authority (e.g. other issuing authorities, or mDL verifiers in other countries) to:

— use a machine to obtain the mDL data;

— tie the mDL to the mDL holder;

— authenticate the origin of the mDL data;

— verify the integrity of the mDL data.

The following items are out of scope for this document:

— how mDL holder consent to share data is obtained;

— requirements on storage of mDL data and mDL private keys.

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 3166-1, *Codes for the representation of names of countries and their subdivisions — Part 1: Country code*

ISO 3166-2:2020, *Codes for the representation of names of countries and their subdivisions — Part 2: Country subdivision code*

ISO/IEC 5218, *Information technology — Codes for the representation of human sexes*

ISO/IEC 7816-4:2020, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*

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

ISO/IEC 18004, *Information technology — Automatic identification and data capture techniques — QR Code bar code symbology specification*

ISO/IEC 18013-1:2018, *Information technology — Personal identification — ISO-compliant driving licence — Part 1: Physical characteristics and basic data set*

ISO/IEC 18013-2:2020, *Personal identification — ISO-compliant driving licence — Part 2: Machine-readable technologies*

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

ISO/IEC 19785-3:2020, *Information technology — Common Biometric Exchange Formats Framework — Part 3: Patron format specifications*

BSI TR-03111, *Elliptic Curve Cryptography, Version 2.10, June 2018*

FIPS 186-4:2013, *D igital Signature Standard (DSS)*

NFC Forum, *Connection Handover (CH) Technical Specification, Version 1.5*

NIST SP 800-38D, *M. Dworkin, Recommendation for Block Cipher Modes of Operation: Galois/Counter Mode (GCM) and GMAC, November 2007*

OpenID Foundation *OpenID Connect Core 1.0 incorporating errata set 1*

OpenID Foundation *OpenID Connect Discovery 1.0 incorporating errata set 1*

RFC 4122, *P. Leach et al., A Universally Unique IDentifier (UUID) URN Namespace, July 2005*

RFC 4648, *S. Josefsson, The Base16, Base32, and Base64 Data Encodings, October 2006*

RFC 5246, *T. Dierks et al., The Transport Layer Security (TLS) Protocol Version 1.2, August 2008*

RFC 5280, *D. Cooper et al., Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, May 2008*

RFC 5639, *M. Lochter et al., Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, March 2010*

RFC 5869, *H. Krawczyk, HMAC-based Extract-and-Expand Key Derivation Function (HKDF), May 2010*

RFC 6066, *D. Eastlake 3rd, Transport Layer Security (TLS) Extensions: Extension Definitions, January 2011*

RFC 7049, *C. Bormann et al., Concise Binary Object Representation (CBOR), October 2013*

RFC 7515, *J. Bradley et al., JSON Web Signature (JWS), May 2015*

RFC 7518, *M. Jones et al., JSON Web Algorithms (JWA), May 2015*

RFC 7519, *J. Bradley et al., JSON Web Token (JWT), May 2015*

RFC 7748, *A. Langley et al., Elliptic Curves for Security, January 2016*

RFC 7905, *A. Langley et al., ChaCha20-Poly1305 Cipher Suites for Transport Layer Security (TLS), June 2016*

RFC 8032, *S. Josefsson et al., Edwards-Curve Digital Signature Algorithm (EdDSA), January 2017*

RFC 8152, *J. Schaad, CBOR Object Signing and Encryption (COSE), July 2017*

RFC 8252, *W. Denniss et al., Oauth 2.0 for Native Apps, October 2017*

RFC 8259, *T. Bray, The JavaScript Object Notation (JSON) Data Interchange Format, December 2017*

RFC 8410, *S. Josefsson et al., Algorithm Identifiers for Ed25519, Ed448, X25519, and X448 for Use in the Internet X.509 Public Key Infrastructure, August 2018*

RFC 8422, *Y. Nir et al., Elliptic Curve Cryptography (ECC) Cipher Suites for Transport Layer Security (TLS) Versions 1.2 and Earlier, August 2018*

RFC 8943, *M. Jones et al., Concise Binary Object Representation (CBOR) Tags for Date, November 2020*

RFC, *CBOR Object Signing and Encryption (COSE): Headers for carrying and referencing X.509 certificates*

Wi-Fi Alliance, *Neighbor Awareness Networking Specification, Version 3.1*

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 18013-5:2021

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

ISO and IEC maintain terminology databases for use in standardization at the following addresses:

— ISO Online browsing platform: available at https://www.iso.org/obp

— IEC Electropedia: available at https://www.electropedia.org/

3.1
mobile device
portable computing device that at least:

— has a small form factor such that it can easily be carried by a single individual;

— is designed to operate, transmit and receive information without a wired connection;

— possesses local, nonremovable or removable data storage;

— includes a self-contained power source;

— includes a display;

— includes a means for the holder of the portable computing device to interact with the device

Note 1 to entry: Adapted from NIST SP 800-157.

3.2
mdoc
document or application that resides on a *mobile device* (3.1) or requires a mobile device as part of the process to gain access to the document or application

3.3
mdoc reader
device that can retrieve *mdoc* (3.2) data for verification purposes

3.4
mdoc holder
individual to whom an *mdoc* (3.2) is issued

3.5
mdoc verifier
person or organization using and/or controlling an *mdoc reader* (3.3) to verify an *mdoc* (3.2)

3.6
mDL
driving licence that fulfils at least the same function as an IDL but, instead of being paper or plastic based, is an *mdoc* (3.2)

Note 1 to entry: ISO-compliant driving licence (IDL) is defined in ISO/IEC 18013-1.

3.7
mDL reader
mdoc reader (3.3) that can retrieve *mDL* (3.6) data

3.8
mDL holder
individual to whom an *mDL* (3.6) is issued, i.e. legitimate holder of the driving privileges reflected on an mDL

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

3.9
mDL verifier
person or organization using and/or controlling an *mDL reader* (3.7) to verify an *mDL* (3.6)

3.10
licensing authority
authorized agent organisation that issues a driving licence

EXAMPLE National, federal, state, provincial, regional, territorial, or local Ministry of Transport, Department of Motor Vehicles, or Police Agency.

[SOURCE: ISO/IEC 18013-1:2018, 3.15]

3.11
issuing country
country which issued the driving licence or within which the *licensing authority* (3.10) is located

[SOURCE: ISO/IEC 18013-1:2018, 3.12, modified — The words "according to Annex F" have been removed.]

3.12
issuing authority
licensing authority (3.10), or *issuing country* (3.11) if separate licensing authorities have not been authorized

[SOURCE: ISO/IEC 18013-1:2018, 3.11]

3.13
issuing authority infrastructure
infrastructure under control of the *issuing authority* (3.12)

3.14
issuing authority CA
certificate authority operated by or on behalf of an *issuing authority* (3.12)

3.15
device retrieval
method of data retrieval exclusively using the interface between the *mdoc* (3.2) and the *mdoc reader* (3.3)

3.16
server retrieval
method of data retrieval using the interface between the *mdoc reader* (3.3) and the *issuing authority infrastructure* (3.13)

3.17
server retrieval token
token identifying the *mdoc holder* (3.4) and the *mdoc* (3.2) to the *issuing authority* (3.12)

3.18
PCD mode
mode in which an NFC-enabled *mobile device* (3.1) operates as a PCD

[SOURCE: ISO/IEC 14443-3:2018, 3.7, modified — The words "a PXD" have been replaced with "an NFC-enabled mobile device".]

3.19
PICC mode
mode in which an NFC-enabled *mobile device* (3.1) operates as a PICC

[SOURCE: ISO/IEC 14443-3:2018, 3.8, modified — The words "a PXD" have been replaced with "an NFC-enabled mobile device".]

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

4 Abbreviated terms

AES	advanced encryption standard
APDU	application protocol data unit
BLE	Bluetooth® low energy
BT SIG	Bluetooth special interest group
CA	certificate authority
CBOR	concise binary object representation
CDDL	concise data definition language
COSE	CBOR object signing and encryption
CSPRNG	cryptographically secure pseudo-random number generator
CRL	certificate revocation list
DER	distinguished encoding rules
DS	document signer
ECDH	elliptic curve Diffie-Hellman key agreement
ECDSA	elliptic curve digital signature algorithm
EdDSA	Edwards-curve digital signature algorithm
GATT	generic attribute profile
HKDF	HMAC-based extract-and-expand key derivation function
HMAC	hash-based MAC
IA	issuing authority
IACA	issuing authority certificate authority
IANA	internet assigned number authority
IDL	ISO-compliant driving licence
IKM	input keying material
JSON	JavaScript object notation
JWK	JSON web key
JWS	JSON web signature
JWT	JSON web token
KDF	key derivation function
MAC	message authentication code
MITM	man-in-the-middle attack

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

MSO	mobile security object
MTU	maximum transmission unit
NDEF	NFC data exchange format
NFC	near field communication
OCSP	online certificate status protocol
OID	object identifier
OIDC	OpenID connect
PCD	proximity coupling device
PICC	proximity card or object
PKI	public key infrastructure
RF	radiofrequency
RFU	reserved for future use
SHA	secure hash algorithm
TLS	transport layer security
URI	uniform resource identifier
URL	uniform resource locator
UTC	coordinated universal time
UUID	universally unique identifier
VICAL	verified issuer certificate authority list

5 Conformance requirement

An mDL is in conformance with this document if it meets all the requirements specified directly or by reference herein. Conformance with ISO/IEC 18013-1, ISO/IEC 18013-2, ISO/IEC 18013-3, and ISO/IEC 18013-4 is not required for conformance with this document, except for those clauses normatively referenced in this document.

An mDL reader is in conformance with this document if it meets all the requirements specified directly or referenced herein.

An issuing authority infrastructure is in conformance with this document if it meets all the requirements specified directly or referenced herein.

6 mDL overview

6.1 Interfaces

Figure 1 shows the interfaces in scope for this document. The explanation of each interface is as follows.

— Interface 1 in Figure 1 is the interface between the issuing authority infrastructure and the mDL. This interface is out of scope for this document.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

— Interface 2 in Figure 1 is the interface between the mDL and the mDL reader. This interface is specified in this document. The interface can be used for connection setup and for the device retrieval method.

— Interface 3 in Figure 1 is the interface between the issuing authority infrastructure and the mDL reader. This interface is specified in this document. The interface can be used for the server retrieval method.

See Reference [9] for examples of use cases.

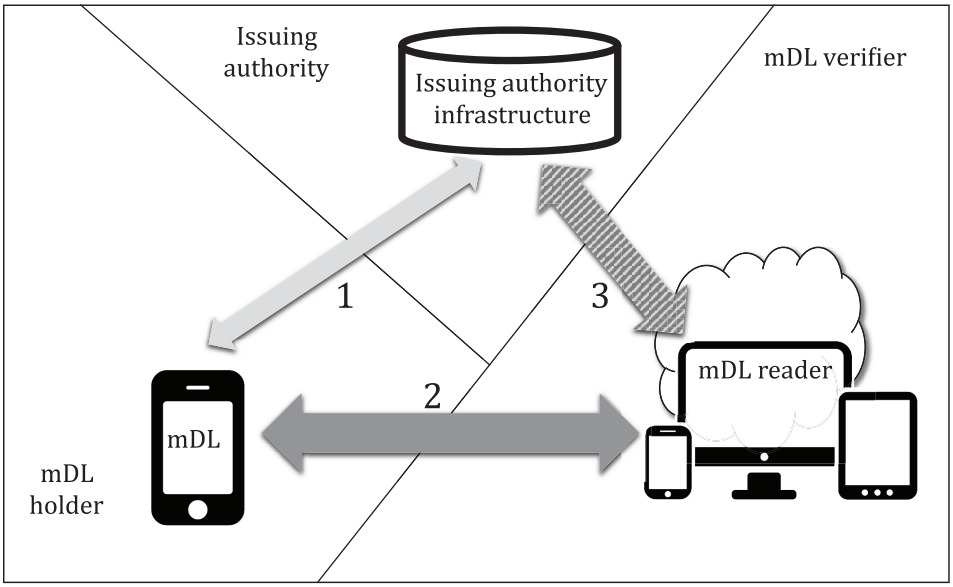

Figure 1 — mDL interfaces

6.2 Functional requirements

The functional requirements include at least the following.

a) An mDL verifier together with an mDL reader shall be able to request, receive and verify the integrity and authenticity of an mDL whether online connectivity is present or not for either the mDL or mDL reader.

b) An mDL verifier not associated with the issuing authority shall be able to verify the integrity and authenticity of an mDL.

c) An mDL verifier shall be enabled to confirm the binding between the person presenting the mDL and the mDL holder.

d) The interface between the mDL and the mDL reader shall support the selective release of mDL data to an mDL reader.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

6.3 Technical requirements

6.3.1 Data model

The mDL data is organized as individual data elements which can be requested and returned independently from each other. The mDL data model is described in Clause 7. It describes the identifier and format of the data elements.

NOTE The concepts used in this document have been designed so that other mobile credentials, e.g. mobile identity or other credentials that substitute Table 5 with a different set of data elements, can also make use of the engagement and retrieval protocols described in this document. Specifically, the mdoc data model, which is illustrated in Figure 2, is based on elements with unique identifiers within a namespace. The number of elements can vary, and the model is indifferent to the value and data format of each element. As such the data model is generic and can apply to any kind of document.

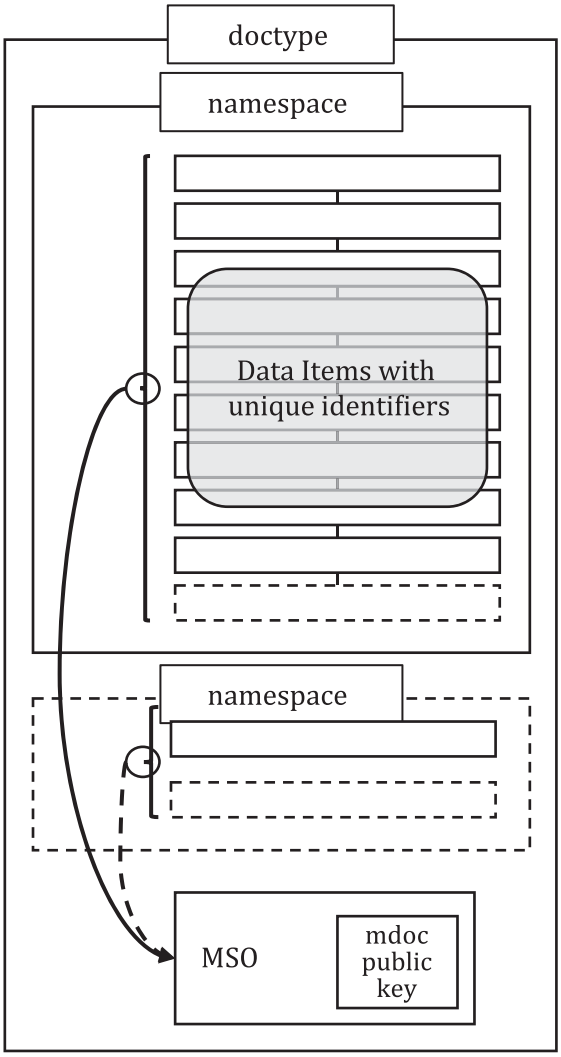

Figure 2 — mdoc data model

6.3.2 Data exchange

6.3.2.1 Overview

The mDL and mDL reader are implemented as an mdoc and mdoc reader, for which the requirements are described in Clause 8 and Clause 9.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Data exchange is divided into three phases: the initialization phase, the device engagement phase, and the data retrieval phase (as illustrated in Figure 3). After initialization between the mDL and the mDL reader three different transaction flows are distinguished:

— device engagement, followed by exchange of data using device retrieval between the mDL and the mDL reader [see (1) in Figure 3];

— device engagement, followed by exchange of server retrieval information using device retrieval between the mDL and the mDL reader, followed by exchange of data using server retrieval between the mDL reader and the issuing authority infrastructure [see (2) in Figure 3];

— device engagement, followed by exchange of data using server retrieval between the mDL reader and the issuing authority infrastructure [see (3) in Figure 3].

NOTE 1 For device retrieval, there is no requirement for any device involved in the transaction to be connected to the internet.

If the mDL reader receives the server retrieval token and URL from the mDL, either during device engagement or device retrieval, it may either use device retrieval or server retrieval. If it chooses to use device retrieval, either BLE, NFC or Wi-Fi Aware can be used to retrieve the information. If it chooses to use server retrieval, either OIDC or WebAPI can be used to retrieve the information.

NOTE 2 The transaction has been designed such that it is not necessary for the mDL holder to physically hand over the mobile device to the mDL verifier.

NOTE 3 The transaction protocols in this document provide generic means for a user to share connection information and optionally a server retrieval token.

The device data retrieval transport applies to any kind of data as it is designed to transport an encrypted blob.

The request and response commands (transported encrypted) are applicable to any kinds of document based on the mdoc data model and/or request for server retrieval token. Furthermore, the request and response commands are wallet compliant as elements from different documents can be requested and the response can include multiple documents from the same or different kinds.

The server retrieval method relies on OpenID Connect that is not specific to mDL, or on WebAPI that relies on the generic mdoc data model.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

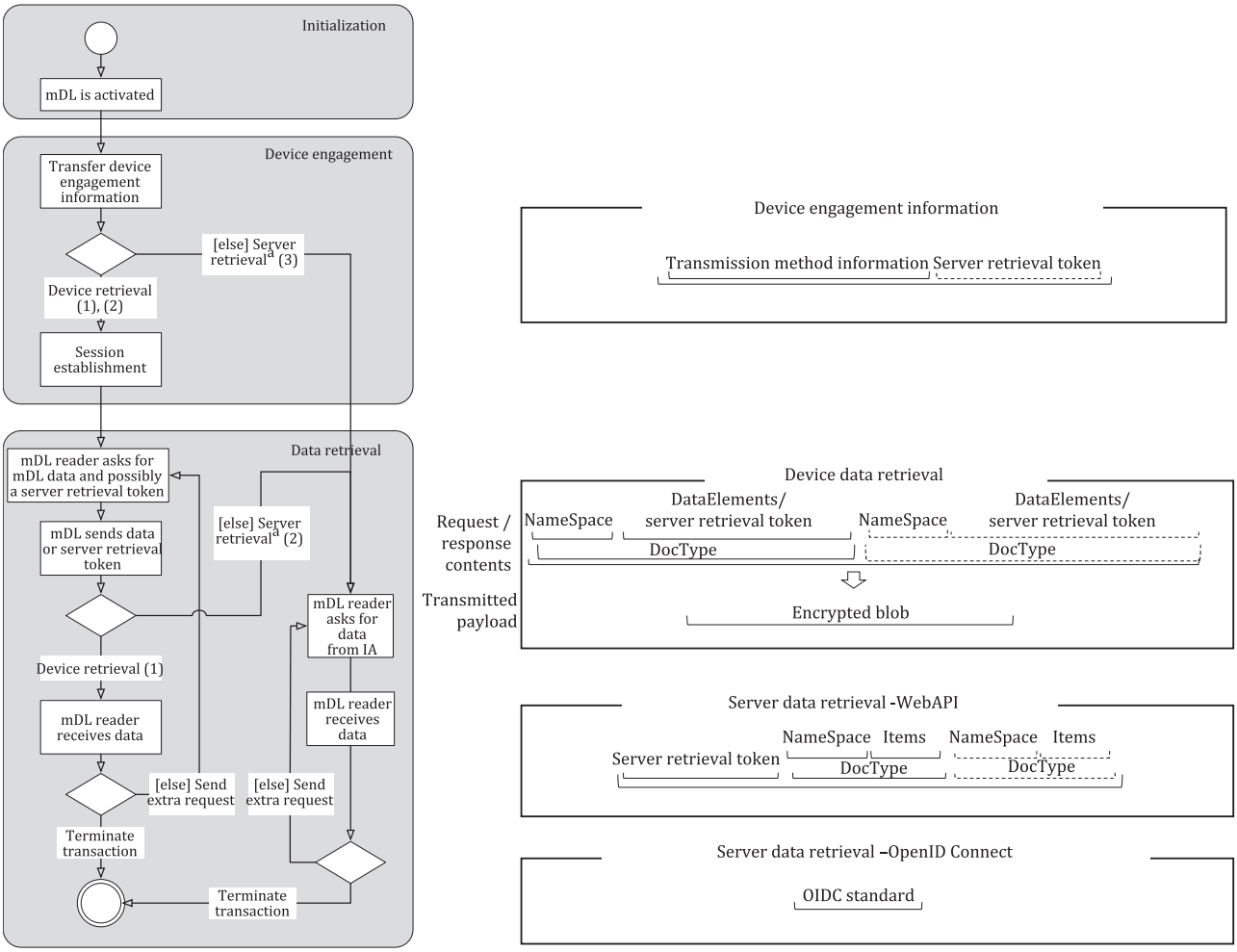

Figure 3 — mDL transaction flow

6.3.2.2 Initialization

During initialization, the mDL is activated. Activation is done by the mDL holder, or triggered by an mDL reader using NFC. Simultaneously, the mDL reader is activated. No requirements are specified for this phase.

NOTE It is important to avoid unauthorized access by an mDL reader if mDL activation is triggered by NFC.

6.3.2.3 Device engagement

During device engagement, information required to setup and secure data retrieval is exchanged between the mDL and the mDL reader. Transmission technologies available to transfer the device engagement data are as follows:

a) NFC,

b) QR code.

Table 1 shows the different transmission technologies for device engagement.

© ISO/IEC 2021 – All rights reserved
Documents Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 1 — Device engagement technologies

Transmission technologies	Support		Reference
	mDL	mDL reader	
NFC	C[a]	M	8.2.2.1
QR code	C[a]	M	8.2.2.3

Key

C conditional

M mandatory

[a] Support for at least one of these methods is mandatory.

The device engagement information, described in 8.2.1, is transferred using one of the transmission technologies, described in 8.2.2.

To ensure that the device engagement is always possible, an mDL shall support at least one of the transmission technologies in Table 1. An mDL reader shall support all transmission technologies.

If the mDL supports NFC for device engagement, it shall support Static Handover, Negotiated Handover, or both, as described in 8.2.2.1. The mDL reader shall support both handover methods.

6.3.2.4 Data retrieval architecture

Figure 4 shows the different data retrieval interfaces and the flow of the messages.

When using device retrieval, the mDL and mDL reader communicate using mdoc request and mdoc response messages encoded with CBOR. These messages are transported using a data retrieval method. The data retrieval methods are agnostic to the information that is transferred.

After device engagement, if the mDL reader sets up a device retrieval connection, the mDL reader asks for data as defined in 8.3.2.1.2.1. The mDL sends an mdoc response according to 8.3.2.1.2.2. The mdoc request may include a request for server retrieval information used to perform server retrieval. If server retrieval information is requested next to other mDL data, the mDL shall return either the server retrieval information or the other requested data, but not both.

The different data retrieval methods are described in 6.3.2.5.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

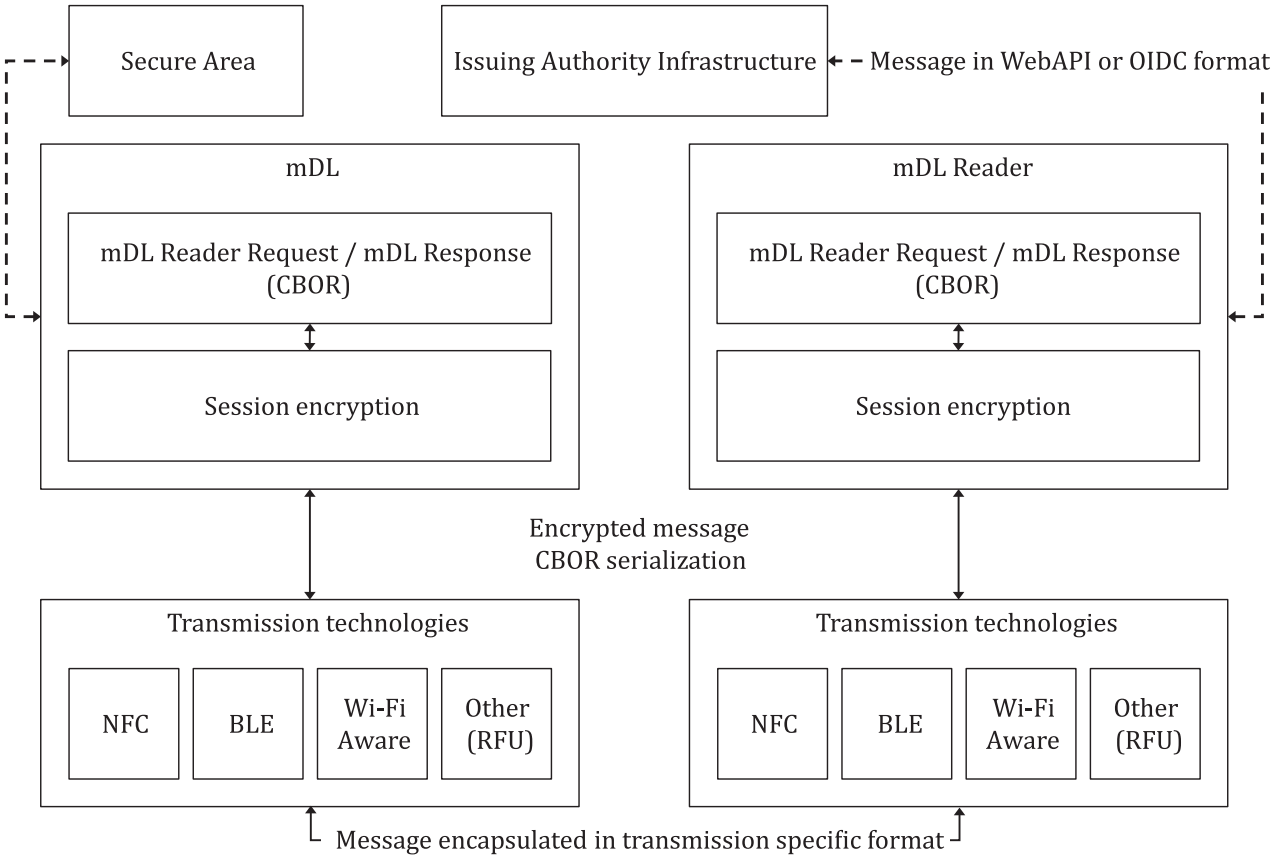

Figure 4 — Data retrieval architecture

NOTE 1 The secure area as present in Figure 4 indicates an area that provides additional protection of sensitive mDL related data. Security requirements regarding storage of credential information are outside the scope of this document. This includes the mDL private key and the mDL reader private key, if mdoc reader authentication or TLS client authentication is supported by the mDL reader. It is the responsibility of the issuing authority to ensure that all data stored on the mDL is stored securely.

NOTE 2 Implementation possibilities for a secure area are non-exhaustively listed in Clause E.5.

6.3.2.5 Data retrieval methods

The following methods are defined for retrieval of mDL data. Requirements for supporting these methods are defined in Table 2.

mDL data can be retrieved in two ways:

a) using device retrieval (interface 2 in Figure 1), see 8.3.2.1;

b) using server retrieval (interface 3 in Figure 1), see 8.3.2.2, where the server retrieval token may be retrieved by the mDL reader from the mDL during device engagement or during device retrieval.

Table 2 shows the transmission technologies and data retrieval methods.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 2 — Data retrieval methods

Data retrieval method	Transmission technology	Support			Reference
		mDL	mDL reader	issuing authority infrastructure	
Device retrieval	BLE	C [a]	M	N/A	8.3.3.1.1
	NFC	C [a]	M	N/A	8.3.3.1.2
	Wi-Fi Aware	O	R	N/A	8.3.3.1.3
Server retrieval	WebAPI	O	R	O	8.3.3.2.1
	OIDC	O	R	O	8.3.3.2.2

Key

M mandatory

C conditional

R recommended

O optional

N/A not applicable

[a] Support for at least one of these methods is mandatory.

To ensure that data retrieval is always possible, an mDL shall support device retrieval using BLE, NFC, or both transmission technologies. An mDL reader shall support the BLE and NFC transmission technology for device retrieval.

An mDL may support Wi-Fi Aware and an mDL reader should support Wi-Fi Aware.

An mDL and an issuing authority infrastructure may support WebAPI, OIDC or both and an mDL reader should support WebAPI and OIDC.

For device retrieval using BLE, the mDL reader shall support the mdoc central client mode and mdoc peripheral server mode, as defined in 8.3.3.1.1. The mDL and mDL reader may support the BLE L2CAP transmission profile as defined in Annex A.

All data retrieval methods shall use the data model as defined in Clause 7.

See Annex D for examples of data structures.

NOTE 1 If QR is used for device engagement and the mDL reader chooses to use NFC for data transfer, then there is no mechanism available for the mDL reader to indicate the choice for NFC data transfer to the mDL. It is possible that the mDL holder is not aware that the mDL needs to interface with the mDL reader using NFC. On the contrary, if NFC is used for device engagement, this problem does not exist.

NOTE 2 Due to the limited data transfer rate for NFC, if a large amount of data is required for a transaction, it is possible that it is neither practical nor reasonable to have an mDL holder hold the device within the RF range of the mDL reader for the duration of the transaction. Furthermore, due to the loss of signal when a device leaves the RF field, any mDL holder interactions with the mDL causing the mDL to leave the RF field require a new transaction to be initiated. This can be avoided by having all mDL holder interactions with the mDL done while the mDL stays in the field or if mDL does not require any mDL holder interactions while it is in the RF field.

6.3.3 Security mechanisms

The security of mDL data exchanged with an mDL reader is designed to preserve the triad of confidentiality, integrity, and authenticity by design and by default.

The security architecture aims to achieve four distinct goals.

a) Protection against forgery: data elements are signed by the issuing authority (IA). The degree of protection against forgery depends on the degree to which the IA's keys are protected. Minimizing the validity period of the data limits the value of the data.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

b) Protection against cloning: the mDL produces a signature or message authentication code over session data. The private key used to authenticate the session data is stored only in the mDL. The corresponding public key in turn is signed by the IA. The degree of protection against cloning depends on the degree to which the mDL authentication key or the TLS server key is protected.

c) Protection against eavesdropping: communications between mDL and mDL reader are encrypted and authenticated. Device engagement uses a separate communication channel to mitigate the risk of Man in the Middle (MITM) attacks. In addition, the mDL reader can detect MITM attacks by validating the anti-cloning signature or message authentication code, which is created using a key for which the public part is signed by the IA in the mobile security object (MSO, see 9.1.2.4). If mdoc reader authentication is used, the mDL can detect MITM attacks before returning any data. Server retrieval uses TLS for encryption to further protect against eavesdropping and MITM attacks.

d) Protection against unauthorized access: an mDL is protected from unauthorized access by an mDL reader by multiple mechanisms. If device retrieval is used, the encryption key used for communications between the mDL and mDL reader is derived from an ephemeral key pair from both the mDL and mDL reader. The public key of the mDL is shared only through short-range device engagement. This ensures data is only transferred between the mDL and a particular mDL reader. The mDL can optionally authenticate the mDL reader by means of an mDL reader certificate and a signature created by the mDL reader using the corresponding private key. The mDL reader certificate is signed by a certificate authority trusted by the mDL for this purpose.

   Server retrieval uses a server retrieval token which can be used to ensure that data is only transferred between the IA infrastructure and a particular mDL reader. Moreover, the IA can optionally authenticate the mDL reader by means of TLS client authentication.

Revocation of an mDL is out of scope for this document. However, the MSO includes update information and validity time frames which enable the mDL reader to check the freshness of the data. The issuing authority shall define appropriate periods of validity that balance freshness with offline capability, taking into account that a shorter validity period mitigates certain security risks.

Table 3 describes the security mechanisms that can be implemented for each data retrieval interface. For device retrieval, issuer data authentication, session encryption and mdoc authentication shall be implemented. mdoc reader authentication is optional for device retrieval and TLS client authentication is optional for server retrieval. For server retrieval, Transport Layer Security (TLS), and JSON Web Signature (JWS) shall be implemented.

Table 4 describes security goals and mechanisms for different data retrieval methods.

When server retrieval is used, the mDL should use one-time server retrieval tokens.

The certificate and CRL profile requirements in Annex B shall be applied.

All certificates issued by an IACA or another CA shall be validated according to 9.3.3.

An mDL reader needs access to the issuing authority's certificate authority (IACA) root certificate to verify issuer data authentication, verify the JWS and perform TLS. One optional method to get access to these certificates is described in Annex C.

See Annex E for additional information on privacy and security.

Table 3 — Security mechanisms

Data retrieval method	Security mechanisms	Reference
Device retrieval	Session encryption	9.1.1
	Issuer data authentication	9.1.2
	mdoc authentication	9.1.3
	mdoc reader authentication	9.1.4

[a] Only applicable if the server retrieval token is transferred using device retrieval.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 3 *(continued)*

Data retrieval method	Security mechanisms	Reference
Server retrieval	TLS	9.2.1
	JWS	9.2.2
	mdoc authentication [a]	9.1.3
[a] Only applicable if the server retrieval token is transferred using device retrieval.		

Table 4 — Security goals and mechanisms

Security goal	Functionality	Device retrieval	Server retrieval
Protection against forgery	Authenticate the origin of mDL data	Issuer data authentication	JWS
	Verify mDL data has not changed from issuing authority	Issuer data authentication	JWS
	Verify how up to date the mDL data is	Issuer data authentication	JWS
Protection against cloning	Protect against cloning of mDL/ binding mDL data to a specific device	mdoc authentication	mdoc authentication[a]
Protection against eavesdropping	Preserve confidentiality of mDL data	Session encryption	TLS
	Prevent unnoticed alteration of communication	Session encryption mdoc authentication	TLS
Protection against unauthorized access	Prevent unauthorized access of mDL data	Close-range device engagement with session encryption	Close-range device engagement (with session encryption)[a]
	Prevent unauthorized access of mDL data	mdoc reader authentication[b]	TLS client authentication[b]
[a] Only applicable if the server retrieval token is transferred using device retrieval. [b] This is an optional method.			

7 mDL data model

7.1 mDL document type and namespace

`DocType` and `NameSpace` are used to encapsulate the document type and the space in which the data elements are defined. The two concepts are further defined in 8.3.1.

The document type for an mDL document shall be “org.iso.18013.5.1.mDL”. The number “1” in the document type might be increased in future versions of the standard.

NOTE 1 The document type field follows the following general format: [Reverse Domain].[Domain Specific Extension]. The reverse domain (org.iso) was selected to avoid collisions. This approach can be used to define other doctypes.

If the mDL reader wants to retrieve an mDL, the `DocType` field in the device retrieval mdoc request or server retrieval mdoc request shall contain the mDL document type. The use of any other value for the `DocType` field in the device retrieval mdoc request or server retrieval mdoc request is beyond the scope of this document.

The namespace for mDL data defined in 7.2 shall be “org.iso.18013.5.1”. The number “1” in the namespace might be increased in future versions of the standard. Within this namespace, only data

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

elements defined in 7.2 and 8.2.1.2 may be used. To accommodate domestic data, an issuing authority may define its own namespace, as described in 7.2.8.

NOTE 2 The namespace field follows the following general format: [Reverse Domain].[Domain Specific Extension].

7.2 mDL data

7.2.1 Overview

The mDL data elements shall be as defined in Table 5 and belong to namespace “org.iso.18013.5.1”, see 7.1.

— The "Identifier" column is used for `DataElementIdentifier` in the device retrieval mdoc request or server retrieval mdoc request (see 8.3.1).

— The "Presence" column indicates whether the presence of the element on an mDL is mandatory (M), or optional (O).

NOTE 1 This does not mean that granting access to these elements to an mDL reader is mandatory.

— The “Encoding format” column indicates how the data elements shall be encoded. “tstr”, “uint”, “bstr”, “bool” and “tdate” are CDDL representation types as defined in RFC 8610. This document specifies “full-date” as `full-date = #6.1004(tstr)`, where tag 1004 is specified in RFC 8943.

— In accordance with RFC 7049, section 2.4.1, a tdate data item shall contain a date-time string as specified in RFC 3339. In accordance with RFC 8943, a `full-date` data item shall contain a full-date string as specified in RFC 3339.

— If data elements are encoded with JSON for the server retrieval methods, the data elements shall be encoded as specified in RFC 7049, section 4.1.

— The following requirements shall apply to the representation of dates in mDL data elements, unless otherwise indicated:

  — fraction of seconds shall not be used;

  — no local offset from UTC shall be used, as indicated by setting the `time-offset` defined in RFC 3339 to “Z”.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 5 — Data elements

Identifier	Meaning	Definition	Presence	Encoding format
family_name	Family name	Last name, surname, or primary identifier, of the mDL holder. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	M	tstr
given_name	Given names	First name(s), other name(s), or secondary identifier, of the mDL holder. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	M	tstr
birth_date	Date of birth	Day, month and year on which the mDL holder was born. If unknown, approximate date of birth	M	full-date
issue_date	Date of issue	Date when mDL was issued	M	tdate or full-date
expiry_date	Date of expiry	Date when mDL expires	M	tdate or full-date
issuing_country	Issuing country	Alpha-2 country code, as defined in ISO 3166-1, of the issuing authority's country or territory	M	tstr
issuing_authority	Issuing authority	Issuing authority name. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	M	tstr
document_number	Licence number	The number assigned or calculated by the issuing authority. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	M	tstr
portrait	Portrait of mDL holder	A reproduction of the mDL holder's portrait. See 7.2.2	M	bstr
driving_privileges	Categories of vehicles/ restrictions/ conditions	Driving privileges of the mDL holder. See 7.2.4	M	See 7.2.4

Key

Presence:

M mandatory

O optional

NOTE 1 The 'UN Distinguishing sign' element is added for purposes of the UN conventions[26][27] on driving licences.

NOTE 2 The 'sex' element is equal to the 'gender' fields in ISO/IEC 18013-1 and ISO/IEC 18013-2.

NOTE 3 The 'issuing jurisdiction' element can be used in cases where the issuing jurisdiction is different from the issuing authority or the issuing country.

[a] The mDL reader can convert to the local unit of measurement.

[b] Latin1 shall be as defined in ISO/IEC 8859-1 as Latin alphabet No. 1.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 5 *(continued)*

Identifier	Meaning	Definition	Presence	Encoding format
un_distinguishing_sign	UN distinguishing sign	Distinguishing sign of the issuing country according to ISO/IEC 18013-1:2018, Annex F. If no applicable distinguishing sign is available in ISO/IEC 18013-1, an IA may use an empty identifier or another identifier by which it is internationally recognized. In this case the IA should ensure there is no collision with other IA's.	M	tstr
administrative_number	Administrative number	An audit control number assigned by the issuing authority. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
sex	Sex	mDL holder's sex using values as defined in ISO/IEC 5218.	O	uint
height	Height (cm) [a]	mDL holder's height in centimetres	O	uint
weight	Weight (kg) [a]	mDL holder's weight in kilograms	O	uint
eye_colour	Eye colour	mDL holder's eye colour. The value shall be one of the following: "black", "blue", "brown", "dichromatic", "grey", "green", "hazel", "maroon", "pink", "unknown".	O	tstr
hair_colour	Hair colour	mDL holder's hair colour. The value shall be one of the following: "bald", "black", "blond", "brown", "grey", "red", "auburn", "sandy", "white", "unknown".	O	tstr
birth_place	Place of birth	Country and municipality or state/province where the mDL holder was born. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
resident_address	Permanent place of residence	The place where the mDL holder resides and/or may be contacted (street/house number, municipality etc.). The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
portrait_capture_date	Portrait image timestamp	Date when portrait was taken	O	tdate

Key

Presence:

M mandatory

O optional

NOTE 1 The 'UN Distinguishing sign' element is added for purposes of the UN conventions[26][27] on driving licences.

NOTE 2 The 'sex' element is equal to the 'gender' fields in ISO/IEC 18013-1 and ISO/IEC 18013-2.

NOTE 3 The 'issuing jurisdiction' element can be used in cases where the issuing jurisdiction is different from the issuing authority or the issuing country.

[a] The mDL reader can convert to the local unit of measurement.

[b] Latin1 shall be as defined in ISO/IEC 8859-1 as Latin alphabet No. 1.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 18013-5:2021

Table 5 *(continued)*

Identifier	Meaning	Definition	Presence	Encoding format
age_in_years	Age attestation: How old are you (in years)?	The age of the mDL holder	O	uint
age_birth_year	Age attestation: In what year were you born?	The year when the mDL holder was born	O	uint
age_over_NN	Age attestation: Nearest "true" attestation above request	See 7.2.5	O	bool
issuing_jurisdiction	Issuing jurisdiction	Country subdivision code of the jurisdiction that issued the mDL as defined in ISO 3166-2:2020, Clause 8. The first part of the code shall be the same as the value for issuing_country.	O	tstr
nationality	Nationality	Nationality of the mDL holder as a two letter country code (alpha-2 code) defined in ISO 3166-1	O	tstr
resident_city	Resident city	The city where the mDL holder lives. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
resident_state	Resident state/province/district	The state/province/district where the mDL holder lives. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
resident_postal_code	Resident postal code	The postal code of the mDL holder. The value shall only use latin1[b] characters and shall have a maximum length of 150 characters.	O	tstr
resident_country	Resident country	The country where the mDL holder lives as a two letter country code (alpha-2 code) defined in ISO 3166-1.	O	tstr
biometric_template_xx	Biometric template XX	See 7.2.6	O	bstr
family_name_national_character	Family name in national characters	The family name of the mDL holder using full UTF-8 character set.	O	tstr

Key

Presence:

M mandatory

O optional

NOTE 1 The 'UN Distinguishing sign' element is added for purposes of the UN conventions[26][27] on driving licences.

NOTE 2 The 'sex' element is equal to the 'gender' fields in ISO/IEC 18013-1 and ISO/IEC 18013-2.

NOTE 3 The 'issuing jurisdiction' element can be used in cases where the issuing jurisdiction is different from the issuing authority or the issuing country.

[a] The mDL reader can convert to the local unit of measurement.

[b] Latin1 shall be as defined in ISO/IEC 8859-1 as Latin alphabet No. 1.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 5 *(continued)*

Identifier	Meaning	Definition	Presence	Encoding format
given_name_national_character	Given name in national characters	The given name of the mDL holder using full UTF-8 character set.	O	tstr
signature_usual_mark	Signature / usual mark	Image of the signature or usual mark of the mDL holder, see 7.2.7	O	bstr

Key

Presence:

M mandatory

O optional

NOTE 1 The 'UN Distinguishing sign' element is added for purposes of the UN conventions[26][27] on driving licences.

NOTE 2 The 'sex' element is equal to the 'gender' fields in ISO/IEC 18013-1 and ISO/IEC 18013-2.

NOTE 3 The 'issuing jurisdiction' element can be used in cases where the issuing jurisdiction is different from the issuing authority or the issuing country.

[a] The mDL reader can convert to the local unit of measurement.

[b] Latin1 shall be as defined in ISO/IEC 8859-1 as Latin alphabet No. 1.

If any data element is returned to the mDL reader, then the information necessary to verify that the person presenting the mDL is the mDL holder shall be returned to the mDL reader if that information is requested.

NOTE 2 From the set of data elements with mandatory presence in column "Presence" from Table 5, the portrait of the mDL holder is the only data item for verifying that the person presenting the mDL is the mDL holder.

An mDL may require mdoc reader authentication (see 9.1.4) before releasing data elements not marked as mandatory in Table 5. An mDL shall not require mdoc reader authentication as a precondition for the release of any of the mandatory data elements. An mDL may offer functionality to the mDL holder to pre-authorize the release of mandatory data elements selected by the mDL holder to mDL readers using mdoc reader authentication.

NOTE 3 The intention of these requirements is that the mDL holder is always able to use the mDL as a driving licence if the mDL holder chooses to do that, including if an mDL reader does not use mdoc reader authentication.

If device retrieval is used, the "issue_date" data element shall not be after the `validFrom` element as defined in 9.1.2.4.

If server retrieval is used, the "issue_date" data element shall not be after the `iat` claim as described in 8.3.2.2.2.2 and 8.3.3.2.2.

If the mDL reader retrieved the "issuing_country" data element, it shall verify that the value of that element matches the `countryName` element in the subject field within the DS certificate or the JWS signer certificate (see Annex B).

If device retrieval is used, if the mDL reader retrieved the "issuing_jurisdiction" data element, it shall verify that the value of that element matches the `stateOrProvinceName` element in the subject field within the DS certificate (see Annex B). This is only required if the `stateOrProvinceName` element is present in the DS certificate.

If server retrieval is used, if the mDL reader retrieved the "issuing_jurisdiction" data element, it shall verify that the value of that element matches the `stateOrProvinceName` element in the subject field within the JWS signer certificate (see Annex B). This is only required if the `stateOrProvinceName` element is present in the JWS signer certificate."

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

If device retrieval is used, any element in Table 5 that is returned by the mDL shall be returned as part of the `IssuerSignedItems`. Domestic data elements (see 7.2.8) may be returned in either `IssuerSignedItems` or `DeviceSignedItems`.

NOTE 4 Issuing authorities have the responsibility to ensure that controls to prevent the unintended release of mDL data are implemented, including providing mDL holders with the means to control what data is released.

7.2.2 Portrait of mDL holder

The portrait of mDL holder consists of one portrait image and shall follow the requirements on the face image as specified in ISO/IEC 18013-2:2020, Annex D. One of the following image formats shall be used: JPEG or JPEG2000. The image data shall be encoded as binary data.

7.2.3 Issuing authority

The issuing authority element identifies the administrative authority entitled to issue the driving licence, or the issuing country if separate licensing authorities have not been authorized. The issuing authority element is represented by a string.

NOTE The contents of this field correspond to the contents of the issuing authority element on the IDL, and can indicate a local, regional, or national organisation. Note that, like in ISO/IEC 18013-1, the term issuing authority can also refer to a central government agency, acting on behalf of multiple local or regional issuing authorities.

7.2.4 Categories of vehicles/restrictions/conditions

The categories of vehicles/restrictions/conditions contain information describing the driving privileges of the mDL holder. The definition of the elements in the `DrivingPrivilege` structure can be found in ISO/IEC 18013-1:2018, Clause 5. The possible values for the elements are defined in ISO/IEC 18013-1:2018, Annex B and ISO/IEC 18013-2:2020, Annex A.

NOTE 1 Regardless of driving privileges' status, other mDL data elements can be used for identification purposes as described by this document.

The driving privileges structure shall be encoded as CBOR for device retrieval and JSON for server retrieval and shall be formatted as follows:

```
DrivingPrivileges = [
    * DrivingPrivilege
]

DrivingPrivilege = {
    "vehicle_category_code" : tstr      ; Vehicle category code as per ISO/IEC 18013-1
Annex B
    ? "issue_date" : full-date          ; Date of issue encoded as full-date
    ? "expiry_date" : full-date         ; Date of expiry encoded as full-date
    ? "codes" : [+Code]                 ; Array of code info
}

Code = {
    "code": tstr                        ; Code as per ISO/IEC 18013-2 Annex A
    ? "sign": tstr                      ; Sign as per ISO/IEC 18013-2 Annex A
    ? "value": tstr                     ; Value as per ISO/IEC 18013-2 Annex A
}
```

NOTE 2 The `DrivingPrivileges` structure can be an empty array.

Full-date is defined in 7.2.1.

An example can be found in D.2.1.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

7.2.5 Age attestation: nearest "true" attestation above request

This set of elements is used to convey to an mDL verifier, in a data-minimized fashion, if the mDL holder is as old or older than a specified age, or if the mDL holder is younger than a specified age. To achieve this, the mDL contains age attestation identifiers. An age attestation identifier has the format age_over_NN where NN is a value from 00 to 99. The value of an age attestation identifier can be TRUE or FALSE.

If an mDL verifier includes age_over_NN in a request, it has the meaning of "provide the nearest age attestation equal to or larger than NN with value TRUE, or smaller than NN with value FALSE". More specifically, after receiving an age_over_NN request, the logic to determine the appropriate response shall be equivalent to the following.

1. For all age attestations of the form age_over_nn stored on the mDL, consider all the attestations with value TRUE. From among these attestations, check if an attestation exists where nn is equal to or larger than NN. If one and only one such attestation exists, this is the response. If more than one such attestation exists, the response shall be the attestation with the smallest difference between nn and NN.

2. If step 1 does not produce a response, for all age attestations of the form age_over_nn stored on the mDL, consider all the attestations with value FALSE. From among these attestations, check if an attestation exists where nn is equal to or smaller than NN. If one and only one such attestation exists, this is the response. If more than one such attestation exists, the response shall be the attestation with the smallest difference between NN and nn.

3. If step 2 does not produce a response, no age_over_nn data element shall be returned.

In case of device retrieval, the value of an age_over_NN data element shall be calculated by the issuing authority infrastructure to be valid at the value of the timestamp in the `validFrom` element in the MSO from 9.1.2.4.

In case of server retrieval, the value of an age_over_NN data element shall be valid at the value of the `iat` timestamp as defined in 8.3.2.2.2.2 and 8.3.3.2.2.

During a single data retrieval phase (see Figure 3), an mDL reader shall not request more than two age_over_NN data elements, and an mDL or issuing authority infrastructure should not return more than two age_over_NN data elements.

Examples of requests and responses and their meaning can be found in D.2.2.

NOTE 1 Including more rather than less age_over_NN statements in an mDL has the following consequences, both of which are beneficial from a privacy perspective.

1. If an mDL does not respond to an age_over_NN request, the most likely next step of a verifier will be to ask for the mDL holder's date of birth. This would be the only remaining way in which the original business question could be answered. Including additional age_over_NN statements decreases the probability of a no response.

2. Given the question "are you older than 18?", the response "I am older than 21" and the response "I am older than 60" would both answer the question. However, the response "I am older than 21" is considered better from a privacy point of view, since the statement is closer to the request. Including additional age_over_NN statements will on average decrease the difference between the requested age and the response age.

Including more rather than less age_over_NN statements in a mDL on the other hand could allow a verifier to gain a more accurate estimate of a person's true age, if for whatever reason such a verifier chooses not to ask directly for the mDL holder's date of birth. This would require repeated requests.

NOTE 2 A request can include two age_over_NN statements to support effectively asking whether the age of the mDL holder falls within a certain range.

NOTE 3 It is possible that an mDL does not have an age_over_NN element available to respond to the request from the mDL reader. If an mDL reader does not receive an age_over_NN response, the mDL reader has the option of making another request with different data elements that fulfil the business requirements that it has. Examples of these are 'age_in_years' and 'birth_date'.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

7.2.6 Biometric template

This element contains optional facial, fingerprint, iris, or other biometric information of the mDL holder. Biometric information is encoded according to the biometric template defined in ISO/IEC 18013-2:2020, Table 7 and C.4.7. However, the first tag in ISO/IEC 18013-2:2020, Table C.11 ('75' for Facial, '63' for Finger, '76' for Iris, etc) shall be omitted. The value of the biometric_template_xx data element shall therefore start with tag '7F 61'. A biometric template identifier has the format biometric_template_xx where xx shall be replaced with the corresponding "Abstract value name" found in ISO/IEC 19785-3:2020, Table 7, according to the following convention: capitalized characters are replaced with their lowercase equivalent and spaces or non-alphanumeric characters are replaced by underscores (_).

EXAMPLE The "FACE" template corresponds to "biometric_template_face" and the "SIGNATURE/SIGN" template corresponds to "biometric_template_signature_sign".

If the biometric_template_face is used, the biometric data block (tag '5F 2E') shall contain a JPEG or JPEG2000 format, as specified in 7.2.2.

7.2.7 Signature or usual mark

The signature or usual mark of the mDL holder consists of one image. One of the following image formats shall be used: JPEG or JPEG2000. The image data shall be encoded as binary data.

7.2.8 Domestic data elements

Domestic data are data elements which are not specified in this document. An issuing authority may specify its own data elements within its namespace. Since the namespace for mDL data in this document is "org.iso.18013.5.1", the issuing authority infrastructure should use country-specific or issuer-specific namespaces by appending the ISO 3166-1 alpha-2 country code or the ISO 3166-2 region code after a period.

EXAMPLE The United States namespace is "org.iso.18013.5.1.US" and the Iowa namespace is "org.iso.18013.5.1.US-IA".

7.3 Country codes

ISO 3166-1 shall be used as a source for country code identifiers. If no applicable country code is available in ISO 3166-1, an IA may use one of the user-assigned country code elements, as indicated in ISO 3166-1 or another identifier by which it is internationally recognized. In these cases, the IA should ensure there is no collision with other IAs. These provisions apply to all occurrences of country code identifiers in this document unless stated otherwise.

8 Transaction

8.1 Encoding of data structures and data elements

In this document CDDL (Concise Data Definition Language) as specified in RFC 8610 is used to express CBOR and JSON-encoded data structures.

CBOR structures shall be encoded according to RFC 7049. JSON structures shall be encoded according to RFC 8259.

RFC 7049, section 3.9 describes four rules for canonical CBOR. Three of those rules shall be implemented for all CBOR structures as follows:

— integers (major types 0 and 1) shall be as small as possible;

— the expression of lengths in major types 2 through 5 shall be as short as possible;

— indefinite-length items shall be made into definite-length items.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

The fourth rule regarding sorting of map keys is not required. Furthermore, maps (major type 5) shall not have multiple entries with the same key.

Because canonical map ordering is not required, all CBOR maps that are used in a cryptographic operation are communicated in a tagged CBOR bytestring. For any cryptographic operation, an mdoc, mdoc reader or issuing authority infrastructure shall use these bytestrings as they were sent or received, without attempting to re-create them from the underlying maps.

EXAMPLE A data structure `DataItem` that is to be used in a cryptographic operation is communicated in a structure `DataItemBytes`, specified as follows:

```
DataItemBytes = #6.24(bstr .cbor DataItem)
```

The CDDL in this example is defined in RFC 8610, section 3.6 and expresses a tagged data item (major type 6). As specified in RFC 7049, section 2.4, tag value 24 indicates that the content of the CBOR bstr following the tag is itself a CBOR data item. The .cbor control operator indicates that this data item is in fact a `DataItem`.

When processing a data structure, an mdoc, mdoc reader or issuing authority infrastructure shall ignore any value that is specified as RFU in this document.

Whenever data structures in this document use a version element that is encoded as a string, their contents follow the format of 'major version number'.'minor version number'. A major version number shall be incremented by 1 when any backwards incompatible changes are introduced. In a future version of this document. A minor version number shall be incremented by 1 when new, but backwards compatible functionality is introduced. A minor version number shall be reset to 0 if the major version number is incremented. An mdoc, mdoc reader or issuing authority infrastructure shall not give an error and continue a transaction if it receives a data structure having a known major version number but with an unknown minor version number.

8.2 Device engagement

8.2.1 Device engagement information

8.2.1.1 Device engagement structure

The device engagement structure contains information to perform device engagement. The device engagement structure shall be CBOR encoded and formatted as follows:

```
DeviceEngagement =
{
    0: tstr,            ; Version
    1: Security,
    ? 2: DeviceRetrievalMethods,  ; Is absent if NFC is used for device engagement
    ? 3: ServerRetrievalMethods,
    ? 4: ProtocolInfo,
    * int => any
}

Security = [
    int,                ; Cipher suite identifier
    EDeviceKeyBytes
]

DeviceRetrievalMethods = [
    + DeviceRetrievalMethod
]

ServerRetrievalMethods = {
    ?"webApi" : WebApi,
    ?"oidc" : Oidc
}

ProtocolInfo = any      ; The use of ProtocolInfo is RFU
```


© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

```
DeviceRetrievalMethod = [
    uint,                ; Type
    uint,                ; Version
    RetrievalOptions     ; Specific option(s) to the type of retrieval method
]

RetrievalOptions = WifiOptions / BleOptions / NfcOptions / any  ; The any option is RFU
```

The device engagement structure contains the following key-value pairs.

0. `Version`: the version of the device engagement structure, in the current version of this document its value shall be "1.0".

1. `Security`: an array that contains two mandatory elements. The first element is the cipher suite identifier, defined in 9.1.5.2. The second element is `EdeviceKeyBytes`, defined in 9.1.1.4.

2. `DeviceRetrievalMethods`: an array that shall contain one or more `DeviceRetrievalMethod` arrays when performing device engagement using the QR code. When using NFC to perform device engagement, the `DeviceRetrievalMethods` array shall be absent, because the data retrieval methods supported by the mdoc are specified in the Alternative Carrier Records, as specified in 8.2.2.1. `DeviceRetrievalMethods` lists the device retrieval methods supported by the mdoc. A `DeviceRetrievalMethod` array holds two mandatory values (type and version). The first element defines the type and the second element the version for the transfer method. The `RetrievalOptions` element may contain extra info for each connection. The values for the different `RetrievalOptions` are defined in 8.2.2.3.

3. `ServerRetrievalMethods`: a map that contains optional information on the server retrieval methods supported by the mdoc. The values for the different server retrieval methods are defined in 8.2.1.2. If the `ServerRetrievalMethods` is present, the map shall contain `webApi`, `oidc`, or both.

4. `ProtocolInfo`: to ensure the DeviceEngagement remains applicable to future solutions and updates, the contents of the `ProtocolInfo` are RFU in consideration for defining protocol information.

The following requirements apply for additional key-value pairs within the device engagement structure: positive integers for keys are RFU. An application-specific extension shall use a negative integer for the key. An mdoc or mdoc reader shall ignore any key-value pairs with a negative key value that it is not able to interpret.

An example of a device engagement structure can be found in D.3.1.

8.2.1.2 Server retrieval information

Setting up server retrieval requires the mdoc to provide the server retrieval information to the mdoc reader to facilitate the mdoc reader's use of the server retrieval method.

The server retrieval information can be transferred as part of the device engagement structure or as a data element during device retrieval.

— If the server retrieval information is transferred as part of device engagement, the `Oidc`, `WebApi` or both structures shall be transferred as part of `ServerRetrievalMethods` in the device engagement structure (see 8.2.1.1).

— If the server retrieval information is transferred as part of device retrieval, the `Oidc`, `WebApi` or both structures shall be transferred as data elements in the device retrieval mdoc response as defined in 8.3.2.1.2.2. They shall be returned in either `IssuerSignedItems` or `DeviceSignedItems`. The `Oidc` structure shall have the data element identifier "oidc_info". The `WebApi` structure shall have the data element identifier "webapi_info". The namespace of these data elements shall be "org.iso.18013.5.1".

If the mdoc reader wants to use server retrieval, it can request the "webapi_info" or "oidc_info" data elements as a part of the device retrieval mdoc request. The mdoc reader may also do this if the mdoc already included the server retrieval information in the device engagement structure.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

If the server retrieval information is transferred during device retrieval, its authenticity is protected by either issuer data authentication (9.1.2) or mdoc authentication (9.1.3). If the server retrieval information is transferred in the device engagement structure, it is not protected. Therefore, the IA is responsible for the server retrieval token sent in the device engagement structure to authorize the request as part of the transaction by or on behalf of the mdoc holder.

The structures that contain the server retrieval information shall be encoded as CBOR and formatted as follows:

```
Oidc = [
    uint,    ; Version
    tstr,    ; Issuer URL
    tstr     ; Server retrieval token
]

WebApi = [
    uint,    ; Version
    tstr,    ; Issuer URL
    tstr     ; Server retrieval token
]
```

Both arrays consist of three fields: the version, the issuer URL, and the server retrieval token. The version indicates the version of the transfer methods, in the current version of this document, its value shall be 1 for both OIDC and WebAPI. The URL and the server retrieval token field are further defined in 8.3.2.2.

8.2.2 Device engagement transmission technology

8.2.2.1 Device engagement using NFC

Device engagement using NFC shall follow the Connection Handover protocol as defined by NFC Forum, *Connection Handover (CH) Technical Specification, Version 1.5*. Only Reader/Writer mode using the Type 4 Tag shall be used. The Connection Handover protocol shall be initiated by the mdoc reader. The mdoc reader shall take the role of Handover Requester. The mdoc shall be the NFC Tag Device and the mdoc reader shall be the NFC Reader Device. The mdoc shall use either Static Handover or Negotiated Handover.

When Static Handover is used, the Handover Select Message shall be retrieved by the mdoc reader from the mdoc in a Type 4 Tag and shall contain at least one Alternative Carrier Record. Each Alternative Carrier Record shall indicate a device retrieval method the mdoc supports. An mdoc reader shall select one of the transmission technologies from the ones provided in the Alternative Carrier Records.

When Negotiated Handover is used, the mdoc shall include the "urn:nfc:sn:handover" service in a Service Parameter record in the Initial NDEF message provided to the mdoc reader. The mdoc reader shall send a Handover Request Message to the mdoc after the it has selected this service. The Handover Request Message shall contain an Alternative Carrier Record for each alternative carrier that is supported by the mdoc reader. The mdoc confirms the handover by providing a Handover Select Message containing exactly one selected alternative carrier.

NOTE Use of Negotiated Handover for device engagement allows negotiation of transfer methods. For BLE and Wi-Fi Aware, it additionally allows negotiation of keys used by the transmission layer. This can provide better user experience and security of data transmission.

For the BLE device retrieval transmission technology, the contents of the Alternative Carrier Record and Carrier Configuration Record(s) shall comply with 8.3.3.1.1.2.

For the NFC device retrieval transmission technology, the contents of the Alternative Carrier Record and Carrier Configuration Record(s) shall comply with 8.2.2.2.

For the Wi-Fi Aware device retrieval transmission technology, the contents of the Alternative Carrier Record and Carrier Configuration Record(s) shall comply with the Wi-Fi Alliance Neighbor Awareness Networking Specification, version 3.1, section 12.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

The `DeviceEngagement` structure as defined in 8.2.1.1 shall be transferred from the mdoc to the mdoc reader as part of an auxiliary data record of the Handover Select Message with the type "iso.org:18013: deviceengagement" and the ID reference "mdoc", with type name format NFC Forum external type (0x04). For each ac record, "Auxiliary Data Reference" points to the NDEF record which contains the device engagement structure.

When Negotiated Handover is used, the mdoc reader may sent a `ReaderEngagement` structure to the mdoc as part of an auxiliary data record of the Handover Request Message with the type "iso.org:18013: readerengagement" and the ID reference "mdocreader", with type name format NFC Forum external type (0x04), to provide information from the mdoc reader to the mdoc.

The reader engagement structure shall be CBOR encoded and formatted as follows:

```
ReaderEngagement =
{
    0: tstr,            ; Version
    * int => any
}
```

The reader engagement structure contains the following key-value pairs:

0. `Version`: the version of the reader engagement structure. In the current version of this document its value shall be "1.0".

This document does not define any other key-value pairs for use in `ReaderEngagement`. The following requirements apply for additional key-value pairs within the reader engagement structure: positive integers for keys are RFU. An application-specific extension shall use a negative integer for the key. An mdoc or mdoc reader shall ignore any key-value pairs with a negative key value that it is not able to interpret.

8.2.2.2 Alternative Carrier Record for device retrieval using NFC

The Alternative Carrier Record for the NFC device retrieval transmission technology shall reference the Carrier Configuration Record with the ID reference "nfc".

The Carrier Configuration Record for NFC device retrieval transmission technology shall have the type "iso.org:18013:nfc" and the ID reference "nfc". The binary content of the Carrier Configuration Record shall be encoded according to Table 6.

Table 6 — Binary content of the Carrier Configuration Record for NFC device retrieval

Field	Size (Octet)	Sub-Field	Value	Presence
Version	1		0x01	Mandatory
Maximum command data length	1	Length	See below	Conditional
	1	Data type	0x01	
	Variable	Maximum command data length	See 8.3.3.1.2	
Maximum response data length	1	Length	See below	Conditional
	1	Data type	0x02	
	Variable	Maximum response data length	See 8.3.3.1.2	

The value of the version field shall be the mdoc NFC Connection Handover Version encoded as an unsigned integer. This is a mandatory field.

For each field, the value of the length sub-field shall be the sum of the length of the other two sub-fields within that field, encoded as an unsigned integer.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

The value of the maximum command data length field shall be the maximum length of command data field supported by the mdoc, encoded as an unsigned integer. This field is further defined in 8.3.3.1.2. This field is mandatory for the mdoc and shall not be used by the mdoc reader.

The value of the maximum response data length field shall be the maximum length of response data field supported by the mdoc, encoded as an unsigned integer. This field is further defined in 8.3.3.1.2. This field is mandatory for the mdoc and shall not be used by the mdoc reader.

8.2.2.3 Device engagement using QR code

If QR code is used for device engagement, the device engagement structure shall be transmitted as a barcode compliant with ISO/IEC 18004. The QR code shall contain a URI with "mdoc:" as scheme and the `DeviceEngagement` structure specified in 8.2.1.1 encoded using base64url-without-padding, according to RFC 4648, as path.

NOTE The requirements above result in the content of the QR code as "mdoc:" followed by the base64url-without-padding encoded device engagement structure.

An mdoc reader shall select one of the transmission technologies from the ones provided in the device engagement structure.

Table 7 defines the values that shall be used for the `DeviceRetrievalMethod` structure. Any other `DeviceRetrievalMethod` type is RFU.

Table 7 — DeviceRetrievalMethod parameters

	NFC	BLE	Wi-Fi Aware
type	1	2	3
version	1	1	1
options	`NfcOptions`	`BleOptions`	`WifiOptions`

The `WifiOptions`, `BleOptions` and `NfcOptions` structures shall be formatted as follows. Other `RetrievalOptions` are RFU.

```
WifiOptions = {
    ? 0: tstr,      ; Pass-phrase Info Pass-phrase
    ? 1: uint,      ; Channel Info Operating Class
    ? 2: uint,      ; Channel Info Channel Number
    ? 3: bstr       ; Band Info Supported Bands
}

BleOptions = {
    0 : bool,       ; Indicates support for mdoc peripheral server mode
    1 : bool,       ; Indicates support for mdoc central client mode
    ? 10 : bstr,    ; UUID for mdoc peripheral server mode
    ? 11 : bstr     ; UUID for mdoc client central mode
    ? 20 : bstr          ; mdoc BLE Device Address for mdoc peripheral server mode
}

NfcOptions = {
    0 : uint,       ; Maximum length of command data field
    1 : uint        ; Maximum length of response data field
}
```

The contents of these fields are further defined in 8.3.3.1.1, 8.3.3.1.2, and 8.3.3.1.3.

8.2.3 Device engagement time-out

If the mdoc reader implements a time-out for the time between the transaction initialization and receiving device engagement data, the time-out should be no less than 30 seconds. The mdoc verifier may terminate the session at any time.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

If the mdoc or mdoc reader implements a time-out for the time between receiving or sending device engagement data and sending or receiving the session establishment message (see 9.1.1.4), the time-out should be no less than 30 seconds. The mdoc holder and mdoc verifier may terminate the session at any time.

8.3 Data retrieval

8.3.1 Data model

The document type that is requested by an mdoc reader or returned by an mdoc or IA infrastructure is encapsulated in the `DocType` parameter.

NOTE 1 There is no requirement for the `DocType` format. An approach to avoid collisions is to use the following general format: [Reverse Domain].[Domain Specific Extension].

The `NameSpace` parameter provides the namespace within which the data elements requested by the mdoc reader or returned by the mdoc are defined. A document may contain data elements from multiple namespaces. The meaning of data elements is dependent on their namespace.

NOTE 2 There is no requirement for the `NameSpace` format. An approach to avoid collisions is to use the following general format: [Reverse Domain].[Domain Specific Extension].

The CDDL definitions for `DocType`, `NameSpace`, `DataElementIdentifier` and `DataElementValue` are common across different data retrieval methods and the applicable security mechanisms. The following CDDL definitions shall be applied to the CDDL structures defined in Clause 8 and Clause 9:

```
DocType = tstr
NameSpace = tstr
DataElementIdentifier = tstr     ; Data element identifier
DataElementValue = any           ; Data element value
```

8.3.2 Data retrieval methods

8.3.2.1 Device retrieval

8.3.2.1.1 General

The mdoc request and mdoc response messages are defined in 8.3.2.1.2. They are encrypted and subsequently included in a session establishment or session data message, see 9.1.1. These session establishment and session data messages are then transferred using one of the device retrieval transmission methods specified in 8.3.3.1.1, 8.3.3.1.2, or 8.3.3.1.3.

8.3.2.1.2 Message structures

8.3.2.1.2.1 Device retrieval mdoc request

The device retrieval mdoc request structure shall be CBOR encoded and formatted as follows:

```
DeviceRequest = {
    "version" : tstr,                   ; Version of DeviceRequest structure
    "docRequests" : [+ DocRequest]      ; Requested documents
}

DocRequest = {
    "itemsRequest" : ItemsRequestBytes,
    ? "readerAuth" : ReaderAuth         ; mdoc reader authentication
}

ItemsRequestBytes = #6.24(bstr .cbor ItemsRequest)

ItemsRequest = {
    "docType" : DocType,                ; Document type requested
```

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

```
    "nameSpaces" : NameSpaces,
    ? "requestInfo" : {* tstr => any}    ; Additional information
}

NameSpaces = {
    + NameSpace => DataElements          ; Requested data elements for each NameSpace
}

DataElements = {
    + DataElementIdentifier => IntentToRetain    ; Requested data element identifiers
                                                  ; with intent to retain values
}

IntentToRetain = bool
```

`version` is the version for the `DeviceRequest` structure: in the current version of this document its value shall be "1.0". If other versions are specified in the future, the major version (see 8.1) of a `DeviceRequest` structure shall not be higher than the major version of the device engagement structure (see 8.2.1.1) communicated by the mdoc in the same transaction.

`docRequests` contains an array of all requested documents.

`ItemRequestBytes` contains the `ItemsRequest` structure as a tagged CBOR bytestring.

`ReaderAuth` is used for mdoc reader authentication as defined in 9.1.4.

`docType` is the requested document type.

`requestInfo` may be used by the mdoc reader to provide additional information. This document does not define any key-value pairs for use in `requestInfo`. An mdoc shall ignore any key-value pairs that it is not able to interpret.

`NameSpaces` contains the requested data elements and the namespace they belong to.

`DataElements` contains the requested data elements identified by their data element identifier. For each requested data element, the `IntentToRetain` variable indicates whether the mdoc verifier intends to retain the received data element. The mdoc verifier shall not retain any data, including digests and signatures, or derived data received from the mdoc, except for data elements for which the `IntentToRetain` flag was set to true in the request. To retain is defined as "to store for a period longer than necessary to conduct the transaction in realtime".

The mdoc shall ignore all unknown data elements in a device retrieval mdoc request when processing the request.

An example of a device retrieval mdoc request can be found in D.4.1.1.

8.3.2.1.2.2 Device retrieval mdoc response

The device retrieval mdoc response shall be CBOR encoded and formatted as follows:

```
DeviceResponse = {
    "version" : tstr,                    ; Version of the DeviceResponse structure
    ? "documents" : [+Document],         ; Returned documents
    ? "documentErrors": [+DocumentError]; For unreturned documents, optional error codes
    "status" : uint                      ; Status code
}

Document = {
    "docType" : DocType,                 ; Document type returned
    "issuerSigned" : IssuerSigned,       ; Returned data elements signed by the issuer
    "deviceSigned" : DeviceSigned,       ; Returned data elements signed by the mdoc
    ? "errors" : Errors
}

DocumentError = {
```


© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

```
    DocType => ErrorCode                ; Error codes for unreturned documents
}

IssuerSigned = {
    ? "nameSpaces" : IssuerNameSpaces,  ; Returned data elements
    "issuerAuth" : IssuerAuth           ; Contains the mobile security object (MSO)
                                        ; for issuer data authentication
}


IssuerNameSpaces = {                    ; Returned data elements for each namespace
    + NameSpace => [ + IssuerSignedItemBytes ]
}

IssuerSignedItemBytes = #6.24(bstr .cbor IssuerSignedItem)

IssuerSignedItem = {
    "digestID" : uint,                  ; Digest ID for issuer data authentication
    "random" : bstr,                    ; Random value for issuer data authentication
    "elementIdentifier" : DataElementIdentifier,    ; Data element identifier
    "elementValue" : DataElementValue   ; Data element value
}

DeviceSigned = {
    "nameSpaces" : DeviceNameSpacesBytes,   ; Returned data elements
    "deviceAuth" : DeviceAuth               ; Contains the device authentication
                                            ; for mdoc authentication
}

DeviceNameSpacesBytes = #6.24(bstr .cbor DeviceNameSpaces)

DeviceNameSpaces = {
    * NameSpace => DeviceSignedItems    ; Returned data elements for each namespace
}

DeviceSignedItems = {
    + DataElementIdentifier => DataElementValue   ; Returned data element identifier and
value
}

DeviceAuth = {          ; Either signature or MAC for mdoc authentication
    "deviceSignature" : DeviceSignature //     ; "//" means or
    "deviceMac" : DeviceMac
}

Errors = {
    + NameSpace => ErrorItems               ; Error codes for each namespace
}

ErrorItems = {
    + DataElementIdentifier => ErrorCode    ; Error code per data element
}

ErrorCode = int                             ; Error code
```

`version` is the version for the `DeviceResponse` structure. In the current version of this document its value shall be "1.0". If other versions are specified in the future, the major version (see 8.1) of a `DeviceResponse` structure shall not be higher than the major version of the device engagement structure (see 8.2.1.1) communicated by the mdoc in the same transaction. The major version of a `DeviceResponse` structure shall also not be higher than the major version of the mdoc request (see 8.3.2.1.2.1) to which it is a response.

`documents` contains an array of all returned documents. `documentErrors` can contain error codes for documents that are not returned. `status` contains a status code according to 8.3.2.1.2.3.

In the `Document` structure, the document type of the returned document is indicated by the `docType` element. The document type shall match the document type as indicated in the issuer data

© ISO/IEC 2021 – All rights reserved

AVA Documents Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

authentication (see 9.1.2) and mdoc authentication structures (see 9.1.3). `errors` can contain error codes for data elements that are not returned.

`IssuerSigned` contains the mobile security object for issuer data authentication and the data elements protected by issuer data authentication. `nameSpaces` contains the returned data elements as part of their corresponding namespaces.

The `IssuerAuth` structure is defined in 9.1.2.4.

Individual data elements are returned as `IssuerSignedItem`. The `digestID` and `random` are defined in 9.1.2.5. `elementIdentifier` is the data element identifier and `elementValue` the data element value. Each `IssuerSignedItem` is returned as part of the corresponding namespace in the `IssuerNameSpaces` structure. The mdoc shall not include two or more `IssuerSignedItem` elements with the same `DataElementIdentifier` in a single `NameSpace` and `Document`.

`DeviceSigned` contains the mdoc authentication structure and the data elements protected by mdoc authentication. `nameSpaces` contains the returned data elements as part of their corresponding namespaces. `nameSpaces` is a mandatory element because the element is authenticated using mdoc authentication. The `DeviceNameSpaces` structure can be an empty structure. The `DeviceAuth` structure contains either the `DeviceSignature` or the `DeviceMac` element, both are defined in 9.1.3.

`DeviceSignedItems` contains the data element identifiers and values. `DeviceSignedItems` is returned as part of the corresponding namespace in `DeviceNameSpaces`.

If the device retrieval mdoc response structure does not include some data element or document requested in the device retrieval mdoc request, an error code may be returned as part of the `documentErrors` or `errors` structures.

If present, `ErrorCode` shall contain an error code according to 8.3.2.1.2.3.

An example of a device retrieval mdoc response can be found in D.4.1.2.

8.3.2.1.2.3 Device retrieval mdoc response status and error codes

The `status` element shall contain one of the status codes in Table 8. If the mdoc returns a status code different from 0, it shall not return any documents.

Table 8 — Response status

Status code	Status message	Description	Actions required
0	OK	Normal processing. This status message shall be returned if no other status is returned	No specific action required
10	General error	The mdoc returns an error without any given reason.	The mdoc reader may inspect the problem. The mdoc reader may continue the transaction.
11	CBOR decoding error	The mdoc indicates an error during CBOR decoding that the data received is not valid CBOR. Returning this status code is optional.	The mdoc reader may inspect the problem. The mdoc reader may continue the transaction.
12	CBOR validation error	The mdoc indicates an error during CBOR validation, e.g. wrong CBOR structures. Returning this status code is optional.	The mdoc reader may inspect the problem. The mdoc reader may continue the transaction.

If present, the `ErrorCode` element shall contain an error code from Table 9. An `ErrorCode` is specific to the document or data element requested within a namespace.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 9 — Data handling error

Error code	Error code message	Description
0	Data not returned	The mdoc does not provide the requested document or data element without any given reason. This element may be used in all cases.
Other positive integers	See description	RFU
Negative integers	See description	These error codes may be used for application-specific purposes.

8.3.2.2 Server retrieval

8.3.2.2.1 General

Data retrieval using server retrieval shall make use of WebAPI or OIDC. Subclause 8.3.2.2.2 specifies the structures of server retrieval mdoc requests and server retrieval mdoc responses for WebAPI. The messages structures for WebAPI are transferred using the server data retrieval transmission method specified in 8.3.3.2.1. Subclause 8.3.3.2.2 specifies the server data retrieval transmission method for OIDC.

The mdoc is identified by the issuing authority infrastructure using the server retrieval token as provided by the mdoc reader (see 8.2.1.2). The information encoded in the server retrieval token is outside the scope of this document.

NOTE 1 The server retrieval token can contain information about which data is to be provided to the mdoc reader. The issuing authority infrastructure can use a separate interface with the mdoc to retrieve information about which data is to be provided to the mdoc reader.

NOTE 2 The issuing authority infrastructure is involved in each server retrieval-based transaction; therefore, the issuing authority knows when an mdoc is used and what data is shared. If tracking is a concern, the issuing authority can implement mitigating strategies to ensure the mdoc and the mdoc holder are not tracked.

8.3.2.2.2 WebAPI structures

8.3.2.2.2.1 Server retrieval mdoc request

The server retrieval mdoc request shall be JSON encoded and formatted as follows:

```
ServerRequest = {
    "version" : tstr,                   ; Version of the structure
    "token" : tstr,                     ; Server retrieval token
    "docRequests" : [+ ItemsRequest]    ; Requested documents
}

ItemsRequest = {
    "docType" : DocType,                ; Document type requested
    "nameSpaces" : NameSpaces,
    ? "requestInfo" : {* tstr => any}   ; Additional information
}

NameSpaces = {
    + NameSpace => DataElements         ; Requested data elements for each NameSpace
}

DataElements = {
    + DataElementIdentifier => IntentToRetain   ; Requested data element identifiers
                                                ; with intent to retain values
}

IntentToRetain = bool
```

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

`version` is the version for the `ServerRequest` structure: in the current version of this document its value shall be "1.0". If other versions are specified in the future, the major version (see 8.1) of a `ServerRequest` structure shall not be higher than the major version of the device engagement structure (see 8.2.1.1) communicated by the mdoc in the same transaction.

`token` shall contain the server retrieval token (see 8.2.1.2) which identifies the mdoc.

`docRequests` contains an array of all requested documents.

`docType` is the requested document type.

`requestInfo` may be used by the mdoc reader to provide additional information. This document does not define any key-value pairs for use in `requestInfo`. An IA infrastructure shall ignore any key-value pairs that it is not able to interpret.

`NameSpaces` contains the requested data elements and the namespace they belong to.

`DataElements` contains the requested data elements identified by their data element identifier. For each requested data element, the `IntentToRetain` variable indicates whether the mdoc verifier intends to retain the received data element. The mdoc verifier shall not retain any data, including digests and signatures, or derived data received from the mdoc, except for data elements for which the `IntentToRetain` flag was set to true in the request. To retain is defined as "to store for a period longer than necessary to conduct the transaction in realtime".

The IA infrastructure shall ignore all unknown data elements in a server retrieval mdoc request when processing the request.

An informative example of a server retrieval mdoc request can be found in D.4.2.1.1.

8.3.2.2.2.2 Server retrieval mdoc response

The server retrieval mdoc response shall be JSON encoded and shall be formatted as follows:

```
ServerResponse = {
    "version" : tstr,                   ; Version of the structure
    ? "documents" : [+JWT],             ; Returned documents
    ? "documentErrors": [+DocumentError]; For unreturned documents, optional error codes
}

JWT = tstr                              ; JWT with JWTClaimsSet as JWT Claims Set

JWTClaimsSet = {
    "doctype" : DocType,                ; Document type returned
    "namespaces" : NameSpacesResponse,
    ? "errors" : Errors,
    * tstr => any                       ; Registered claims
}

NameSpacesResponse = {
    + NameSpace => DataElementsValues   ; Returned data elements for each namespace
}

DataElementsValues = {
    + DataElementIdentifier => DataElementValue
}

DocumentError = {
    DocType => ErrorCode                ; Error codes for unreturned documents
}

Errors = {
    + NameSpace => ErrorItems               ; Error codes for each namespace
}
```


© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

```
ErrorItems = {
    + DataElementIdentifier => ErrorCode     ; Error code per data element
}

ErrorCode = int                              ; Error code
```

`version` is the version for the `ServerResponse` structure: in the current version of this document its value shall be "1.0". If other versions are specified in the future, the major version (see 8.1) of a `ServerResponse` structure shall not be higher than the major version of the device engagement structure (see 8.2.1.1) communicated by the mdoc in the same transaction. The major version of a `ServerResponse` structure shall also not be higher than the major version of the server retrieval mdoc request (see 8.3.2.2.2.1) to which it is a response.

`documents` contains an array of all returned documents. Each document shall be returned as a JSON Web Token (JWT), as specified in RFC 7519. The claims conveyed by each JWT are in `JWTClaimsSet`. Each JWT is protected using a JSON Web Signature (JWS) as specified in 9.2.2.

`documentErrors` can contain error codes for documents that are not returned.

`doctype` is the returned document type.

`NameSpacesResponse` contains the returned mdoc data elements as part of their namespace. `DataElementsValues` contains the data element identifiers and the corresponding data element values.

`errors` can contain error codes for data elements that are not returned.

In addition to the `doctype`, `namespaces` and `errors` claims, also any registered claims (see RFC 7519) can be returned as part of `JWTClaimsSet`. Whenever any mdoc data is sent to an mdoc reader, the following registered claims shall be present in the JWT: exp (Expiration Time) and iat (Issued AT) as defined in RFC 7519.

If the server retrieval mdoc response structure does not include some data element or document requested in the server retrieval mdoc request, an error code may be returned as part of the `documentErrors` or `errors` structures. If present `ErrorCode` shall contain an error code according to Table 10. An `ErrorCode` is specific to the document or data element requested within a namespace.

Table 10 — Data handling errors

Error code	Error code message	Description
0	Data not returned	The issuing authority infrastructure does not provide the requested document or data element without any given reason. This element may be used in all cases.
Other positive integers	See description	RFU
Negative integers	See description	These error codes may be used for application-specific purposes.

An example of the server retrieval mdoc response can be found in D.4.2.1.2.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

8.3.3 Data retrieval transmission technologies

8.3.3.1 Device retrieval

8.3.3.1.1 Data retrieval using Bluetooth® low energy (BLE)

8.3.3.1.1.1 General

Bluetooth®[1)] low energy (BLE), implemented according to the Bluetooth Core Specification[10], may be used for device retrieval. An mdoc shall support version 4.0 of this specification. An mdoc should support version 4.2 and LE Data Packet Length Extension. An mdoc reader shall support version 4.2 and LE Data Packet Length Extension. An mdoc reader is recommended to support version 5.0 and LE 2M PHY.

Using BLE as a transmission technology consists of two phases, connection setup and data retrieval. During connection setup, the mdoc and mdoc reader connect to each other. After the connection is set up, data retrieval can be initiated.

BLE secure connections can be used if supported by both the mdoc and mdoc reader. However, an mdoc or an mdoc reader shall not require the use of BLE secure connections. Security of the transferred data is ensured by the security mechanisms specified in Clause 9.

All BLE session information shall be removed after each transaction.

As part of device engagement, an mdoc, and optionally the mdoc reader, indicates whether it supports the Central role, the Peripheral role or both, which are implemented according to the Bluetooth Core Specification[10].

NOTE It is possible for the mdoc or mdoc reader to support both roles, until data retrieval has started.

If the mdoc supports the Central role, it shall act as a GATT client. This mode is called mdoc central client mode. If the mdoc supports the Peripheral role, it shall act as a GATT server. This mode is called mdoc peripheral server mode.

If the mdoc indicates during device engagement that it supports both modes, the mdoc reader should select the mdoc central client mode.

8.3.3.1.1.2 Device engagement contents

This subclause describes the requirements for the contents of the device engagement information. Reference [13] gives further informative guidance.

Device engagement using NFC

For device engagement using NFC, the BLE alternative carrier configuration record Record Type Name shall be “application/vnd.bluetooth.le.oob”. The Supplement to the Bluetooth Core Specification[11] specifies data types used for OOB data blocks. The following requirements apply to the presence of the data types in the OOB data block.

— LE Role (0x1C). This type is mandatory. It shall be used by the mdoc and mdoc reader to indicate which LE roles they support or select.

— LE Device Address (0x1B). This field is recommended. If it is available, the connection process can take a shorter time compared to using the UUID to identify the correct device to connect to.

— Complete List of 128-bit Service UUIDs (0x07). The requirements for including this field are defined below. This field shall be used to transfer the UUIDs used for connection setup, if applicable.

1) Bluetooth is the trademark of a product supplied by the Bluetooth Special Interest Group. This information is given for the convenience of users of this document and does not constitute an endorsement by ISO/IEC of the product named. Equivalent products may be used if they can be shown to lead to the same results.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

NOTE When encoding the OOB data blocks, note that the Supplement to the Bluetooth Core Specification specifies that certain values should be little-endian, in Clause 1 it is stated that: "All numerical multi-byte entities and values associated with the following data types shall use little-endian byte order."

Other data types may be included in the OOB data block.

The following requirements apply for including the UUID field during NFC device engagement:

— for Negotiated Handover, if the mdoc reader supports mdoc central client mode, it shall include a UUID in the Handover Request message, to be used for mdoc central client mode;

— for Negotiated Handover, if the mdoc chooses to use mdoc peripheral server mode, it shall include a UUID in the Handover Select message, to be used for mdoc peripheral server mode;

— for Static Handover, the mdoc shall send one UUID in the handover select message, to be used for mdoc central client mode, mdoc peripheral server mode or both.

Device engagement using QR code

For device engagement using QR code, the mdoc shall indicate which modes it supports using the fields in the `BleOptions` structure as specified in 8.2.2.3. The UUIDs in the `BleOptions` structure shall be encoded using variant 1 ('10x'b) as a 16 octets element with the byte order as specified in RFC 4122, section 4.1.2. The value of the BLE Device Address field shall use the same encoding as the LE Device Address field used in device engagement using NFC as defined earlier in this subclause.

The UUID for peripheral server mode shall be present if mdoc peripheral server mode is supported and shall not be present if peripheral server mode is not supported.

The UUID for client central mode shall be present if mdoc central client mode is supported and shall not be present if central client mode is not supported.

The BLE Device Address field may be present if mdoc peripheral server mode is supported and it shall not be present if peripheral server mode is not supported. If it is available, the connection process can take a shorter time compared to using the UUID to identify the correct device to connect to.

8.3.3.1.1.3 Connection setup

The UUIDs used shall be 16-byte UUIDs that are unique for the transaction. The Peripheral device shall broadcast the service with the UUID as received or sent during device engagement in the advertising packet. The Central device is then able to scan for the UUID and connect to the advertised service. However, the Central device may use a different mechanism to identify the Peripheral device.

NOTE 1 BLE stacks in mobile devices can use scan filter and caching methods to manage congested environments and manage scan intervals for device energy consumption control. This can influence the connection time required when using UUIDs for the identification of the Peripheral device.

NOTE 2 Finding the correct device to connect to is purely a practical problem. Connecting to the wrong mdoc reader does not have security implications, since due to the security methods described in Clause 9, the mdoc and mdoc reader will not setup a session with the wrong mdoc reader. Note however, that these mechanisms do not provide complete protection against a bad actor aiming to cause a denial of service attack by advertising as a fake mdoc reader.

To ensure that the mdoc is connected to the correct mdoc reader, the mdoc may verify the Ident characteristic as described in 8.3.3.1.1.4. The Ident characteristic value shall be calculated using the following procedure:

Use HKDF as defined in RFC 5869 with the following parameters:

— Hash: SHA-256,

— IKM: `EdeviceKeyBytes` (see 9.1.1.4),

— salt: (no salt value is provided),

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

— info: “BLEIdent” (encoded as a UTF-8 string),

— L: 16 octets.

If the Ident characteristic received from the mdoc reader does not match the expected value, the mdoc shall terminate the connection.

NOTE 3 The purpose of the Ident characteristic is only to verify whether the mdoc is connected to the correct mdoc reader before starting data retrieval. If the mdoc is connected to the wrong mdoc reader, session establishment will fail. Connecting and disconnecting to an mdoc reader takes a relatively large amount of time and it is therefore fastest to implement methods to identify the correct mdoc reader to connect to and not to rely purely on the Ident characteristic to identify the correct mdoc reader.

After connection is setup, the GATT client may check to see if the GATT server supports the L2CAP transmission profile and, if so, use it to transfer data. See Annex A for more information. If the L2CAP transmission profile is used, 8.3.3.1.1.5, 8.3.3.1.1.6, 8.3.3.1.1.7, and 8.3.3.1.1.8 do not apply.

8.3.3.1.1.4 Service definition

Table 11 shows characteristics which the mdoc service shall contain if the mdoc is the GATT server. The services may contain other characteristics and properties besides the ones required in Table 11.

NOTE Supporting the Write property next to Write Without Response property can solve some interoperability issues. Using Write Without Response has a higher transmission rate.

Table 11 — mdoc service characteristics

Characteristic name	UUID	Mandatory properties
State	00000001-A123-48CE-896B-4C76973373E6	Notify, Write Without Response
Client2Server	00000002- A123-48CE-896B-4C76973373E6	Write Without Response
Server2Client	00000003- A123-48CE-896B-4C76973373E6	Notify

Table 12 shows characteristics which the mdoc reader service shall contain if the mdoc reader is the GATT server. The services may contain other characteristics and properties besides the ones required in Table 12.

Table 12 — mdoc reader service characteristics

Characteristic name	UUID	Mandatory properties
State	00000005- A123-48CE-896B-4C76973373E6	Notify, Write Without Response
Client2Server	00000006- A123-48CE-896B-4C76973373E6	Write Without Response
Server2Client	00000007- A123-48CE-896B-4C76973373E6	Notify
Ident	00000008- A123-48CE-896B-4C76973373E6	Read

Each service characteristic having the Notify property shall contain the Client Characteristic Configuration Descriptor, with UUID ‘0x29 0x02’ and default value of ‘0x00 0x00’. This value shall be set to ‘0x00 0x01’ by the GATT client to get notified for the characteristic associated to this descriptor.

8.3.3.1.1.5 Connection state

After the connection is setup, the GATT client shall subscribe to notifications of characteristic ‘State’ and ‘Server2Client’. For performance reasons, the GATT client should request for an MTU as high possible.

© ISO/IEC 2021 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

After these steps, the GATT client shall make a write without response request to 'State' where it sets the value to 0x01. This tells the GATT server that the GATT client is ready for the transmission to start.

The connection state is indicated by the 'State' characteristic. It is encoded as 1-byte binary data. Table 13 describes the different connection state values, which are communicated using Write Without Response and Notify.

Table 13 — Connection state values

Command	Data	Sender	Description
Start	0x01	GATT client	This indicates that the mdoc reader may/will begin transmission.
End	0x02	mdoc, mdoc reader	Signal to finish/terminate transaction. The mdoc reader shall use this value to signal the end of data retrieval. Both the mdoc and the mdoc reader can use this value at any time to terminate the connection. See also 9.1.1.4 for more information on session termination.

8.3.3.1.1.6 Data retrieval

Data retrieval shall start by signalling the 'Start' value to the 'State' characteristic.

The data sent shall be the `SessionEstablishment` or `SessionData` messages as defined in 9.1.1.4.

If the GATT client wants to send a message to the GATT server, it shall divide the message in parts with a length of 3 bytes less than the MTU size. It then sends these parts to the GATT server using the Write Without Response command via the 'Client2Server' characteristic. The first byte of each part is either 0x01, which indicates more messages are coming, or 0x00, to indicate it is the last part of the message.

If the GATT server wants to send a message to the GATT client, it shall divide the message in parts with a length of 3 bytes less than the MTU size. It then sends these parts to the GATT client using the Notify command via the 'Server2Client' characteristic. The first byte of each part is either 0x01, which indicates more parts are coming, or 0x00, to indicate it is the last part of the message.

The sequence of messages shall be repeated as long as necessary to finish data retrieval.

Figure 5 shows the informative sequence diagram for the data retrieval phase as described in this subclause.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

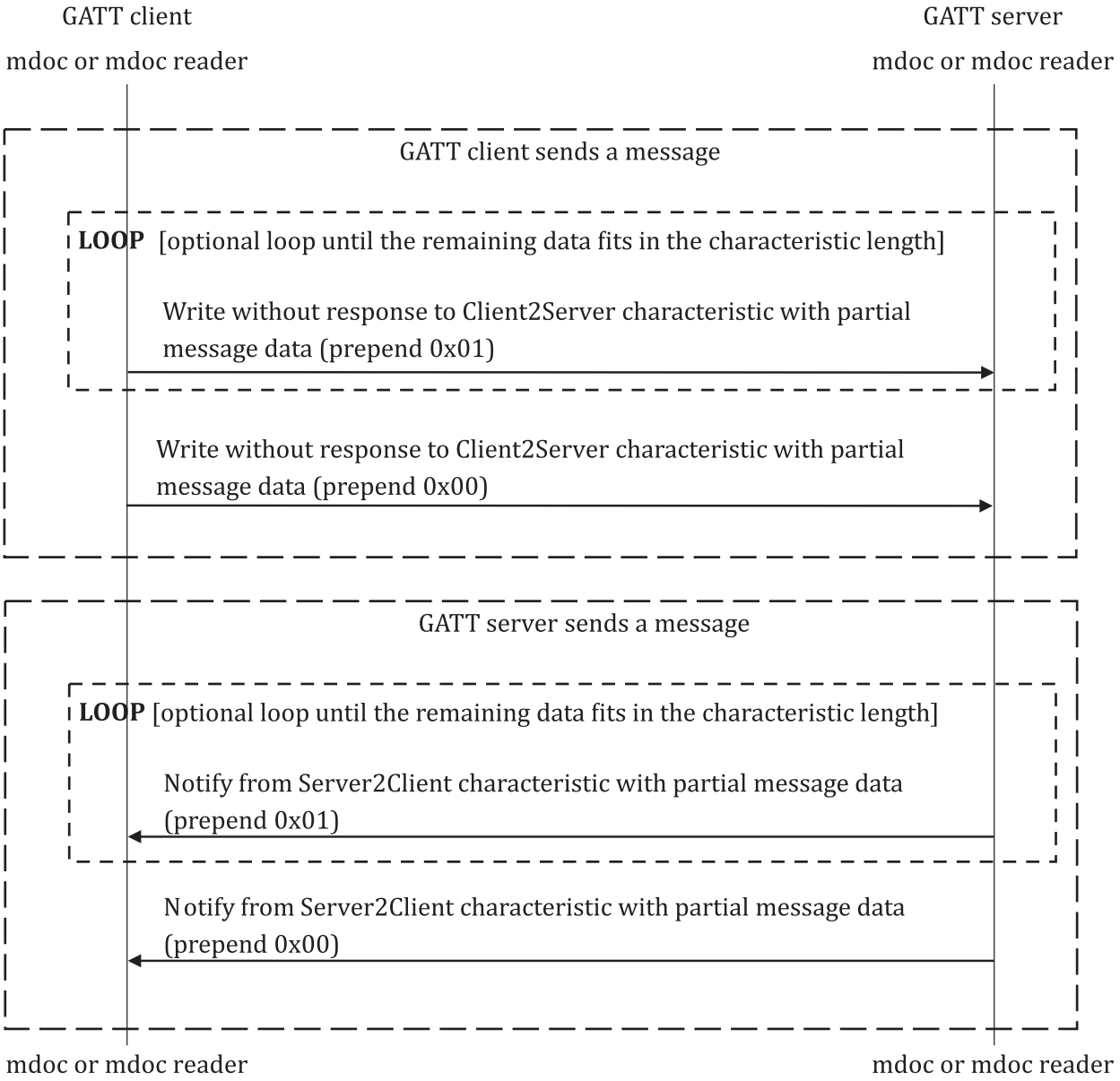

Figure 5 — Data transfer sequence diagram

8.3.3.1.1.7 Connection closure

After data retrieval, the GATT client shall unsubscribe from both the 'State' and 'Server2Client' characteristics and shall disconnect from the GATT server.

8.3.3.1.1.8 Connection re-establishment

In case of a lost connection before the 'State' characteristic has been set to a value of 0x01 (e.g. the transmission has not yet started), the mdoc and mdoc reader should terminate their current BLE session and try to reconnect according to 8.3.3.1.1.3.

In case of a lost connection after the 'State' characteristic has been set to value 0x01 (e.g. the transmission of data has started), a connection shall not be re-established, and a completely new mdoc transaction shall be initiated if required.

© ISO/IEC 2021 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

8.3.3.1.2 Data retrieval using near field communication (NFC)

NFC may be used for device retrieval. In case NFC is used, the mdoc shall support PICC mode and the mdoc reader shall support PCD mode.

An mdoc and mdoc reader shall support short-length fields as specified in ISO/IEC 7816-4:2020, 5.2 and should support extended-length fields as specified in ISO/IEC 7816-4:2020, 5.2. An mdoc shall indicate the maximum length of command data fields and of response data fields it supports during device engagement, as specified in 8.2.2.2 for device engagement using QR code and 8.2.2.3 for device engagement using NFC. The two fields shall indicate the maximum length of the command and response data fields as defined in ISO/IEC 7816-4:2020, 5.2 supported by the mdoc.

NOTE 1 For device engagement using QR code, this information is in the NfcOptions map. For device engagement using NFC, this information is in the NFC Handover Select message.

NOTE 2 The minimum and maximum possible values for the command data field limit are 'FF' and 'FF FF', i.e. the limit is between 255 and 65 535 bytes (inclusive). The minimum and maximum possible values for the response data limit are '01 00' and '01 00 00', i.e. the limit is between 256 and 65 536 bytes (inclusive).

The mdoc reader shall respect the data field size limitations of the mdoc.

NOTE 3 The mdoc reader determines the size of both the command and the response APDUs and can, therefore, make sure that its APDU size limitations (if any) are not being violated.

An mdoc and mdoc reader shall support command chaining and response chaining as specified in ISO/IEC 7816-4:2020, 5.3.

NOTE 4 Even if extended-length APDUs are supported by the mdoc and the mdoc reader, using command or response chaining can still be necessary, because it cannot be guaranteed that an mdoc reader request or an mdoc response fits in a single APDU.

The Application IDentifier (AID) of the mdoc shall be 'A0 00 00 02 48 04 00'.

NOTE 5 The AID of the mdoc application consists of the registered application provider identifier (RID) ('A0 00 00 02 48') followed by the proprietary application identifier extension (PIX) ('04 00').

An mdoc application shall be selected using the SELECT command defined in ISO/IEC 7816-4 with the AID listed above. Table 14 and Table 15 specify the SELECT command and response APDUs.

Table 14 — SELECT command

CLA	INS	P1-P2	Lc field	Data field	Le field
'00'	'A4'	'04 0C'	'07'	'A0 00 00 02 48 04 00'	Absent

Table 15 — SELECT response

Data field	SW1 – SW2
Absent	See ISO/IEC 7816-4:2020, Table 61

After the mdoc application is selected, the mdoc reader can start data retrieval. The mdoc reader shall use the ENVELOPE command with INS = 'C3', specified in ISO/IEC 7816-4, to communicate the `SessionEstablishment` en `SessionData` messages as defined in 9.1.1.4 to the mdoc. These messages shall be encapsulated in a data object '53' as specified in ISO/IEC 7816-4:2020, 11.7.2. The mdoc shall use the ENVELOPE response to communicate the `SessionData` messages as defined in 9.1.1.4 to the mdoc reader. These messages shall be encapsulated in a data object '53' as well.

Table 16 and Table 17 specify the ENVELOPE command and response APDUs.

© ISO/IEC 2021 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021

Table 16 — ENVELOPE command

CLA	INS	P1-P2	Lc field	Data field	Le field
'00' or '10', as defined in ISO/IEC 7816-4:2020, 5.4.1	'C3'	'0000'	Length of data field	Data object '53' or data object fragment	See below

Table 17 — ENVELOPE response

Data field	SW1 – SW2
Data object '53' or absent if an error occurred on ISO/IEC 7816-4:2020 protocol level	See ISO/IEC 7816-4:2020, Table 121, in particular '61 XY'.

For oversize incoming payload (from an mdoc reader to an mdoc), several ENVELOPE commands shall be chained as specified in ISO/IEC 7816-4:2020, 5.3. For oversize outgoing payload (from an mdoc to an mdoc reader), response chaining shall be used as specified in ISO 7816-4:2020, 5.3 and further detailed below, using one or more GET RESPONSE commands and responses.

Regarding the value of Le in the ENVELOPE and GET RESPONSE commands:

— For all ENVELOPE commands in a chain except the last one, Le shall be absent, since no data is expected in the response to these commands;

— For the last ENVELOPE command in a chain, Le shall be set to the maximum length of the response data field that is supported by both the mdoc and the mdoc reader. The mdoc reader shall encode Le as specified in ISO/IEC 7816-4:2020, 5.2;

— For the last ENVELOPE command or for a GET RESPONSE command,

  — if Le ≥ the number of available bytes, the mdoc shall include all available bytes in the response and set the status words to '90 00';

  — if Le < the number of available bytes ≤ Le + 255, the mdoc shall include as many bytes in the response as indicated by Le and shall set the status words to '61 XX', where XX is the number of available bytes remaining. The mdoc reader shall respond with a GET RESPONSE command where Le is set to XX;

  — if the number of available bytes > Le + 255, the mdoc shall include as many bytes in the response as indicated by Le and shall set the status words to '61 00'. The mdoc reader shall respond with a GET RESPONSE command where Le is set to the maximum length of the response data field that is supported by both the mdoc and the mdoc reader.

If the NFC connection is lost during data retrieval, a completely new mdoc transaction (including device engagement) shall be initiated.

8.3.3.1.3 Data retrieval using Wi-Fi Aware

8.3.3.1.3.1 General

Wi-Fi Aware may be used for device retrieval. Wi-Fi Aware shall be implemented according to the Wi-Fi Alliance Neighbor Awareness Networking Specification. The data retrieval using Wi-Fi Aware consists of three phases, connection setup, data retrieval and closure.

8.3.3.1.3.2 Connection setup

Wi-Fi Aware is setup using the information exchanged during device engagement. The Wi-Fi Alliance Neighbor Awareness Networking Specification, Version 3.1 describes the connection setup process. The Wi-Fi Alliance Neighbor Awareness Networking Specification, section 12 describes the connection setup process for NFC Negotiated Connection Handover and NFC Static Connection Handover. The structure used to transfer the required info if QR code is used for device engagement is described below.

© ISO/IEC 2021 – All rights reserved
AVA Documents Document Sharing Hub
Full document details: BS ISO/IEC 18013-5:2021