TECHNICAL SPECIFICATION

**ISO/TS 21719-2**

Second edition
2022-10

# Electronic fee collection — Personalization of on-board equipment (OBE) —

## Part 2: **Using dedicated short-range communication**

*Perception de télépéage — Personnalisation des équipements embarqués —*

*Partie 2: Utilisation des communications dédiées à courte portée*

Reference number
ISO/TS 21719-2:2022(E)

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

# Contents

Page

# Foreword

ISO (the International Organization for Standardization) is a worldwide federation of national standards bodies (ISO member bodies). The work of preparing International Standards is normally carried out through ISO technical committees. Each member body interested in a subject for which a technical committee has been established has the right to be represented on that committee. International organizations, governmental and non-governmental, in liaison with ISO, also take part in the work. ISO collaborates closely with the International Electrotechnical Commission (IEC) on all matters of electrotechnical standardization.

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 ISO documents should be noted. This document was drafted in accordance with the editorial rules of the ISO/IEC Directives, Part 2 (see www.iso.org/directives).

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

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

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

This document was prepared by Technical Committee ISO/TC 204, *Intelligent transport systems,* in collaboration with the European Committee for Standardization (CEN) Technical Committee CEN/TC 278, *Intelligent transport systems,* in accordance with the Agreement on technical cooperation between ISO and CEN (Vienna Agreement).

This second edition cancels and replaces the first edition (ISO/TS 21719-2:2018), which has been technically revised.

The main changes are as follows:

— addition of subclause 5.4 on the Conformance statement;

— minor updating of terms, including a reference to ISO/TS 17573-2 as the primary source.

A list of all parts in the ISO 21719 series can be found on the ISO website.

Any feedback or questions on this document should be directed to the user’s national standards body. A complete listing of these bodies can be found at www.iso.org/members.html.

# Introduction

On-board equipment (OBE) is an in-vehicle device that contains one or more application instances to support different intelligent transport system (ITS) implementations such as electronic fee collection (EFC).

To assign the EFC application in the OBE to a certain user or/and vehicle, personalization is performed. This means that unique user- and vehicle-related data needs to be transferred and stored in the OBE.

CEN/TR 16152 assessed many aspects of the personalization process and defined the overall personalization assets: application data, application keys and vehicle data.

Different communication media may be used for transferring the personalization assets to the OBE. An overall message exchange framework and required security functionality may be applied for all media common procedures, to ensure data protection and integrity.

By standardizing the personalization procedure, compatibility of personalization equipment is supported, and the entity responsible for the personalization [e.g. a toll service provider (TSP)] will further be able to outsource partial or complete personalization to a third party or to another service provider or personalization agent.

The scope of the personalization functionality is illustrated in Figure 1 and is limited to the dedicated short-range communication (DSRC) interface between the personalization equipment (PE) and the OBE.

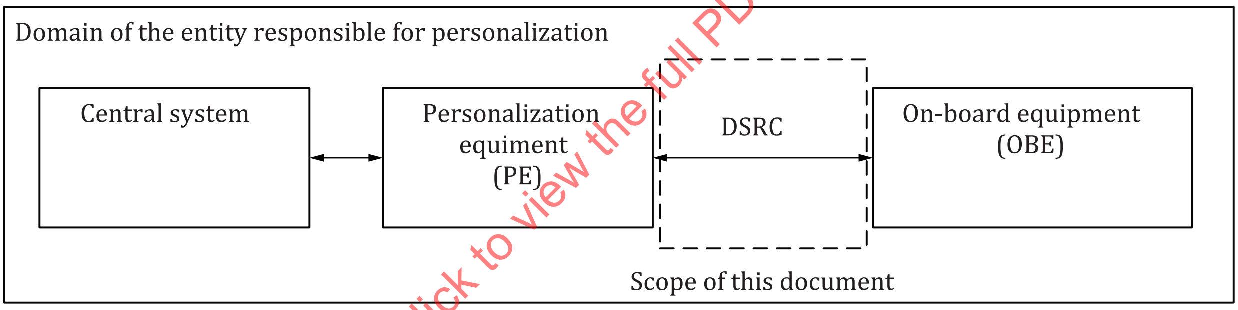
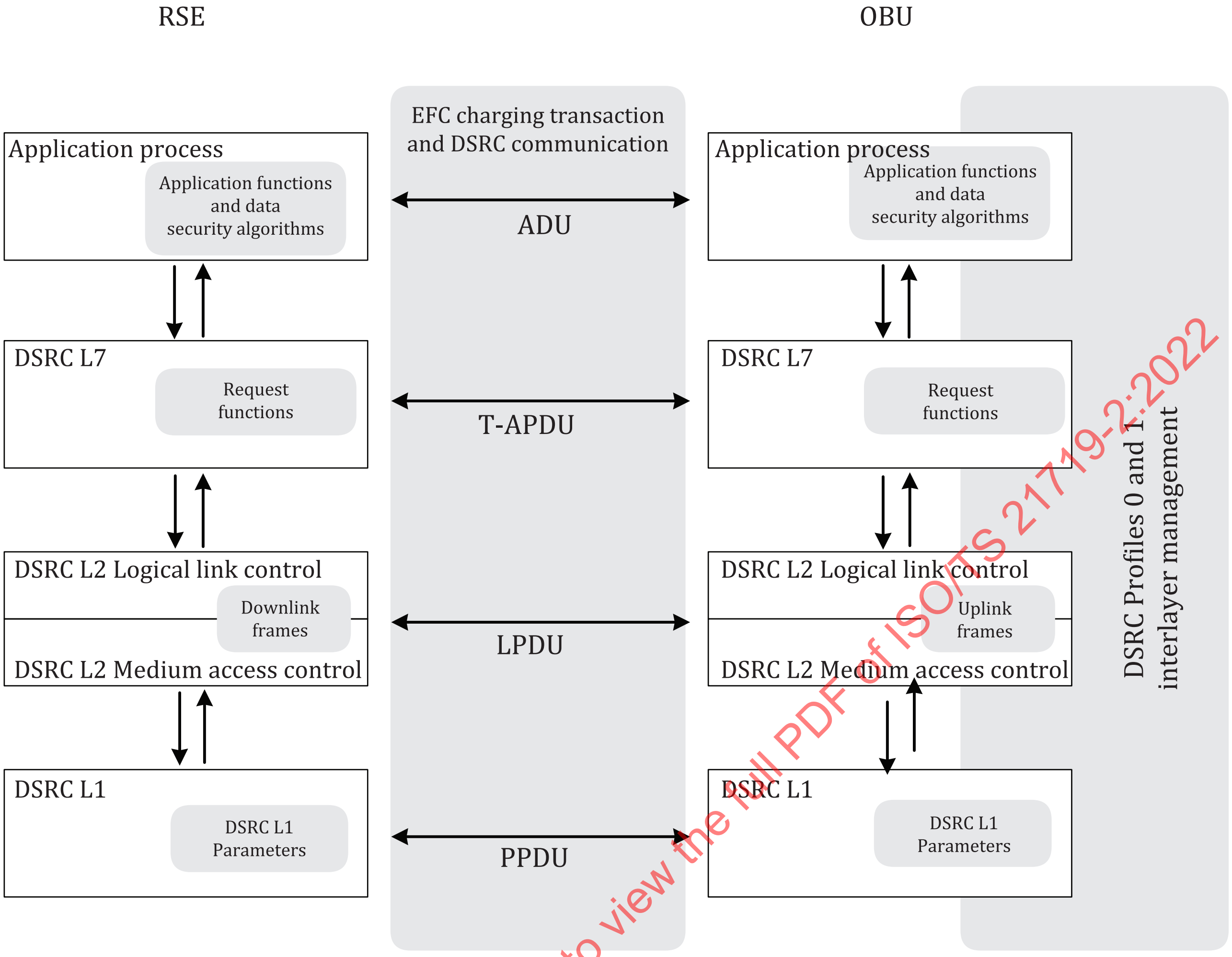

Domain of the entity responsible for personalization
Central system
Personalization equiment (PE)
DSRC
On-board equipment (OBE)
Scope of this document

**Figure 1 — Scope for this document (box delimited by a dotted line)**

This document defines a complete application profile using the personalization functionality described in ISO/TS 21719-1, on top of a CEN DSRC stack according to the DSRC communication profiles as specified in EN 13372 and using the EFC Application Interface according to ISO 14906.

This document further defines in the annexes the use of this application profile on top of other DSRC communication stacks that are compliant with the application layer interfaces as defined in ISO 14906 and EN 12834.

Figure 2 shows the scope of this document from a DSRC-stack perspective.


RSE
OBU
EFC charging transaction and DSRC communication
Application process
Application functions and data security algorithms
ADU
DSRC L7
Request functions
T-APDU
DSRC L2 Logical link control
Downlink frames
Uplink frames
DSRC L2 Medium access control
LPDU
DSRC L1
DSRC L1 Parameters
PPDU
DSRC Profiles 0 and 1 interlayer management

**Key**

| | |
|---|---|
| ADU | application data unit |
| T-APDU | transfer-application protocol data unit |
| LPDU | logical link control (LLC) protocol data unit |
| PPDU | physical layer protocol data unit |
| DSRC L1 | DSRC layer 1 (physical layer) |
| DSRC L2 | DSRC layer 2 (data link layer) |
| DSRC L7 | DSRC layer 7 (application layer) |

**Figure 2 — Relationship between this document and DSRC-stack elements**

# Electronic fee collection — Personalization of on-board equipment (OBE) —

# Part 2:
# Using dedicated short-range communication

## 1 Scope

This document defines:

— personalization interface: dedicated short-range communication (DSRC),

— physical systems: on-board equipment and the personalization equipment,

— DSRC-link requirements,

— EFC personalization functions according to ISO/TS 21719-1 when defined for the DSRC interface, and

— security data elements and mechanisms to be used over the DSRC interface.

A protocol information conformance statement (PICS) proforma is provided in Annex B, and security computation examples are provided in Annex E.

It is outside the scope of this document to define:

— conformance procedures and test specifications,

— setting-up of operating organizations (e.g. toll service provider, personalization agent, trusted third party), and

— legal issues.

NOTE Some of these issues are subject to separate standards prepared by ISO/TC 204, CEN/TC 278 or ETSI ERM.

## 2 Normative references

The following documents are referred to in the text in such a way that some or all of their content constitutes requirements of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 9797-1:2011, *Information technology — Security techniques — Message Authentication Codes (MACs) — Part 1: Mechanisms using a block cipher*

ISO/IEC 10116:2017, *Information technology — Security techniques — Modes of operations for an n-bit block cipher*

ISO 14906, *Electronic fee collection — Application interface definition for dedicated short-range communication*

ISO 15628, *Intelligent transport systems — Dedicated short range communication (DSRC) — DSRC application layer*

ISO/IEC 18033-3:2010, *Information technology — Security techniques — Encryption algorithms — Part 3: Block ciphers*

EN 12834, *Road transport and traffic telematics — Dedicated Short Range Communication (DSRC) — DSRC application layer*

EN 15509:2022, *Electronic fee collection — Interoperability application profile for DSRC*

ETSI/ES 200 674-1:2013, *Intelligent Transport Systems (ITS) — Road Transport and Traffic Telematics (RTTT) — Dedicated Short Range Communications (DSRC) — Part 1: Technical characteristics and test methods for High Data Rate (HDR) data transmission equipment operating in the 5,8 GHz Industrial, Scientific and Medical (ISM) band (V2.4.1, 2013-05)*

## 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**
**access credentials**
trusted attestation or secure module that establishes the claimed identity of an object or application

[SOURCE: ISO/TS 17573-2:2020, 3.4, modified — admitted term removed (listed in Clause 4).]

**3.2**
**attribute**
addressable package of data consisting of a single data element or structured sequences of data elements

[SOURCE: ISO/TS 17573-2:2020, 3.13]

**3.3**
**authentication**
security mechanism allowing verification of the provided identity

[SOURCE: ISO/TS 17573-2:2020, 3.15]

**3.4**
**authenticator**
data, possibly encrypted, that is used for *authentication* (3.3)

[SOURCE: ISO/TS 17573-2:2020, 3.16]

**3.5**
**base standard**
approved International Standard, Technical Specification or ITU-T Recommendation

Note 1 to entry: This includes but is not limited to approved standard deliverables from ISO, ITU, CEN, CENELEC, ETSI and IEEE.

[SOURCE: ISO/TS 17573-2:2020, 3.23]

**3.6**
**data integrity**
property that data has not been altered or destroyed in an unauthorized manner

[SOURCE: ISO/TS 17573-2:2020, 3.56]

**3.7**
**electronic fee collection**

fee collection by electronic means

[SOURCE: ISO/TS 17573-2:2020, 3.70, modified — admitted term removed (listed in Clause 4).]

**3.8**
**EFC Element**
coherent set of data and functionality

Note 1 to entry: The functionality includes, where applicable, the security-related functions and the associated security keys.

Note 2 to entry: EFC Elements are created by the applications and addressed using Element identifiers.

Note 3 to entry: In a given *on-board equipment (OBE)* (3.9), the EID is used to address a toll context, identified by the EFC-ContextMark, in which *attributes* (3.2) can be addressed unambiguously by AttributeIDs inside an EFC Element of the OBE.

[SOURCE: ISO/TS 17573-2:2020, 3.71]

**3.9**
**on-board equipment**
all required equipment on-board a vehicle for performing required *electronic fee collection* (*EFC*) (3.7) functions and communication services

[SOURCE: ISO/TS 17573-2:2020, 3.126, modified — admitted term removed (listed in Clause 4).]

**3.10**
**OBE personalization**
transferring *personalization assets* (3.11) to the *on-board equipment* (*OBE*) (3.9)

[SOURCE: ISO/TS 17573-2:2020, 3.123]

**3.11**
**personalization assets**
specific data stored in the *on-board equipment* (*OBE*) (3.9) related to the user and the vehicle

[SOURCE: ISO/TS 17573-2:2020, 3.137]

**3.12**
**personalization equipment**
equipment for transferring *personalization assets* (3.11) to the *on-board equipment* (*OBE*) (3.9)

[SOURCE: ISO/TS 17573-2:2020, 3.138]

**3.13**
**profile**
set of requirements and selected options from *base standards* (3.5) or international standardized profiles used to provide a specific functionality

[SOURCE: ISO/TS 17573-2:2020, 3.146]

**3.14**
**toll service provider**
entity providing toll services in one or more toll domains

[SOURCE: ISO/TS 17573-2:2020, 3.206, modified — admitted term removed (listed in Clause 4).]

**3.15**
**transaction**
whole of the exchange of information between two physically separated communication facilities

[SOURCE: ISO/TS 17573-2:2020, 3.211]

## 4 Abbreviated terms

For the purposes of this document, the following abbreviated terms apply.

| | |
|---|---|
| Ack | acknowledgement |
| AcK | access key |
| AC_CR | access credentials |
| ADU | application data unit |
| APDU | application protocol data unit |
| AP | application profile |
| ASN.1 | abstract syntax notation one |
| AVEI | automatic vehicle and equipment identification |
| BST | beacon service table |
| CBC | cipher block chaining |
| DSRC | dedicated short-range communication |
| EFC | electronic fee collection |
| EID | element identifier |
| ICS | implementation conformance statement |
| IUT | implementation under test |
| MAC | message authentication code |
| OBE | on-board equipment |
| PE | personalization equipment |
| PICS | protocol implementation conformance statement |
| SAM | secure application module |
| TSP | toll service provider |
| T-APDU | transfer-application protocol data unit |
| VST | vehicle service table |

# 5 Conformance

## 5.1 General

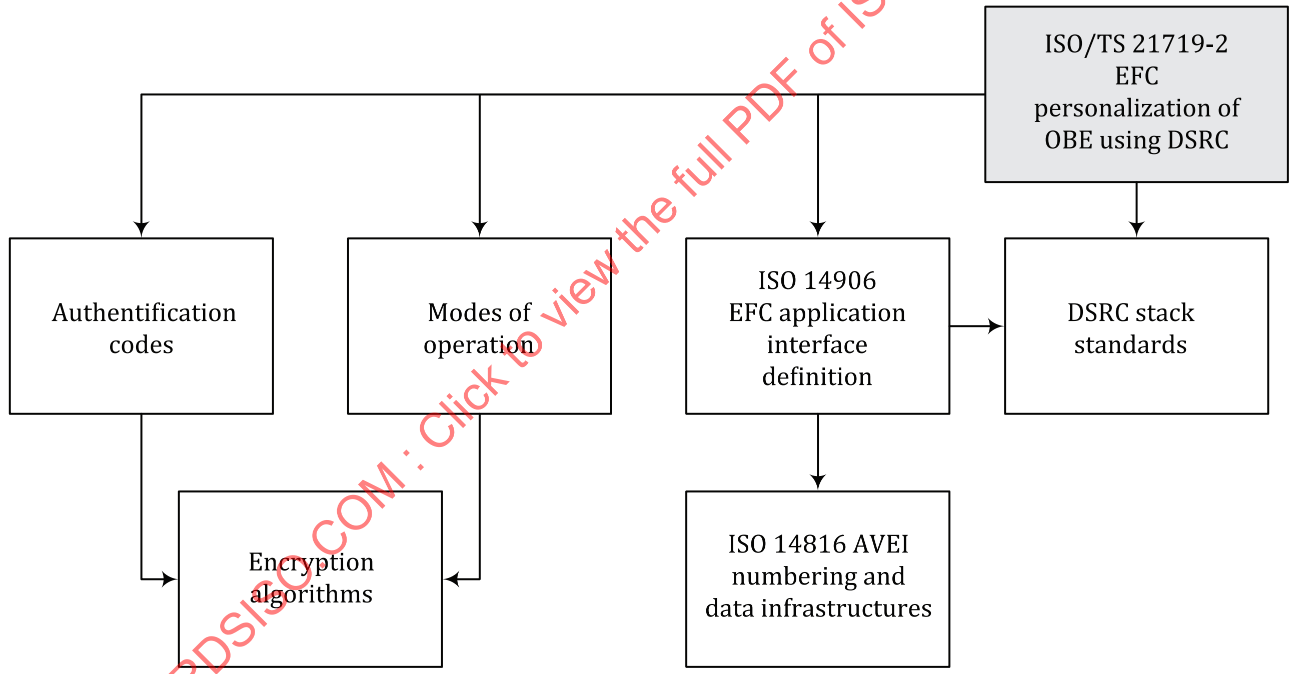
This clause describes in general terms what it means to be conformant with (the profile in) this document.

## 5.2 Base standards

This document defines one application profile (AP). The base standards that this AP is based upon are as follows:

- standards for security functionality;
- standards for EFC application definition as, e.g. ISO 14906;
- standards for the DSRC communication stack definition.

An overview of the relationship and references between base standards and this AP is illustrated in Figure 3.



**Key**

AVEI automatic vehicle and equipment identification

**Figure 3 — Relationship and references between base standards and this document**

All requirements defined in this document are either choices made from these base standards or more specific and limited requirement based on the general provisions of these standards.

## 5.3 Main contents of an EFC personalization AP

The conformance requirements of an AP are divided between requirements for the on-board equipment (OBE) and the personalization equipment (PE). The requirements are listed separately for OBE and PE. This applies for all parts, requirements, PICS and conformance testing.

The conformance requirements of an AP according to this document shall include the following parts (divided into separate requirements for OBE and PE):

- DSRC lower layer requirements;
- EFC personalization functions;
- security requirements;
- transaction requirements.

### 5.4 Conformance statement

A supplier of OBE that claims conformity of their OBE to this document shall provide a statement of conformance to this document by completing the protocol implementation conformance statement (PICS) as provided in B.5.

## 6 Personalization overview

### 6.1 Process

The overall personalization process is described in ISO/TS 21719-1:2018, 5.1.

Personalization means that an existing EFC application structure in the OBE is populated with personalization assets such as user or vehicle data.

Creation of the EFC application and entering initial data, such as initial security keys, is performed before the personalization and is out of scope of this document.

During personalization, the OBE shall be within the communication range of the PE in order for the data exchange according to this document to take place.

Application data and security keys are transferred to the OBE during the personalization process in an attribute list using standardized DSRC commands according to the requirements in this document.

### 6.2 System architecture

The overall system architecture is described in ISO/TS 21719-1:2018, 5.2.

For personalization over a DSRC interface, the OBE and PE shall contain a DSRC stack and the application services as described in this document.

Security functionality and secure key storage may either be implemented within the PE or the PE may be connected to a central system where this functionality may reside. This is outside the scope of this document.

## 7 OBE requirements

### 7.1 General

This clause contains the normative conformance requirements on the OBE for profile number 1: EFC-DSRC-Personalization Profile 1.

### 7.2 DSRC lower layer requirements

#### 7.2.1 Supported DSRC stacks

This document supports the DSRC stacks as defined in Table 1.

**Table 1 — Supported DSRC stacks**

| DSRC stack | Application layer | Lower layers | Detailed specifications |
|---|---|---|---|
| CEN-DSRC | ISO 15628<br>EN 12834 | EN 12795<br>EN 12253 | Specification in 7.2.2 |
| Italian DSRC | ETSI/ES 200 674–1:2013, Clause 11 and Annex C | ETSI/ES 200 674–1:2013, Clauses 7 to 10 and Annex C | Specification and implementation example in Annex C |
| Japanese DSRC | ARIB STD-T75 | ARIB STD-T75 | |
| Wave DSRC | IEEE1609.11 | IEEE 802.11p<br>IEEE 1609.3/4 | |

### 7.2.2 CEN DSRC stack

The following requirements apply for the personalization profile when using the CEN DSRC stack.

The OBE shall comply with EN 15509:2022, 6.1.2 which implicitly requires conformance with the underlying standards as shown in Figure 4.

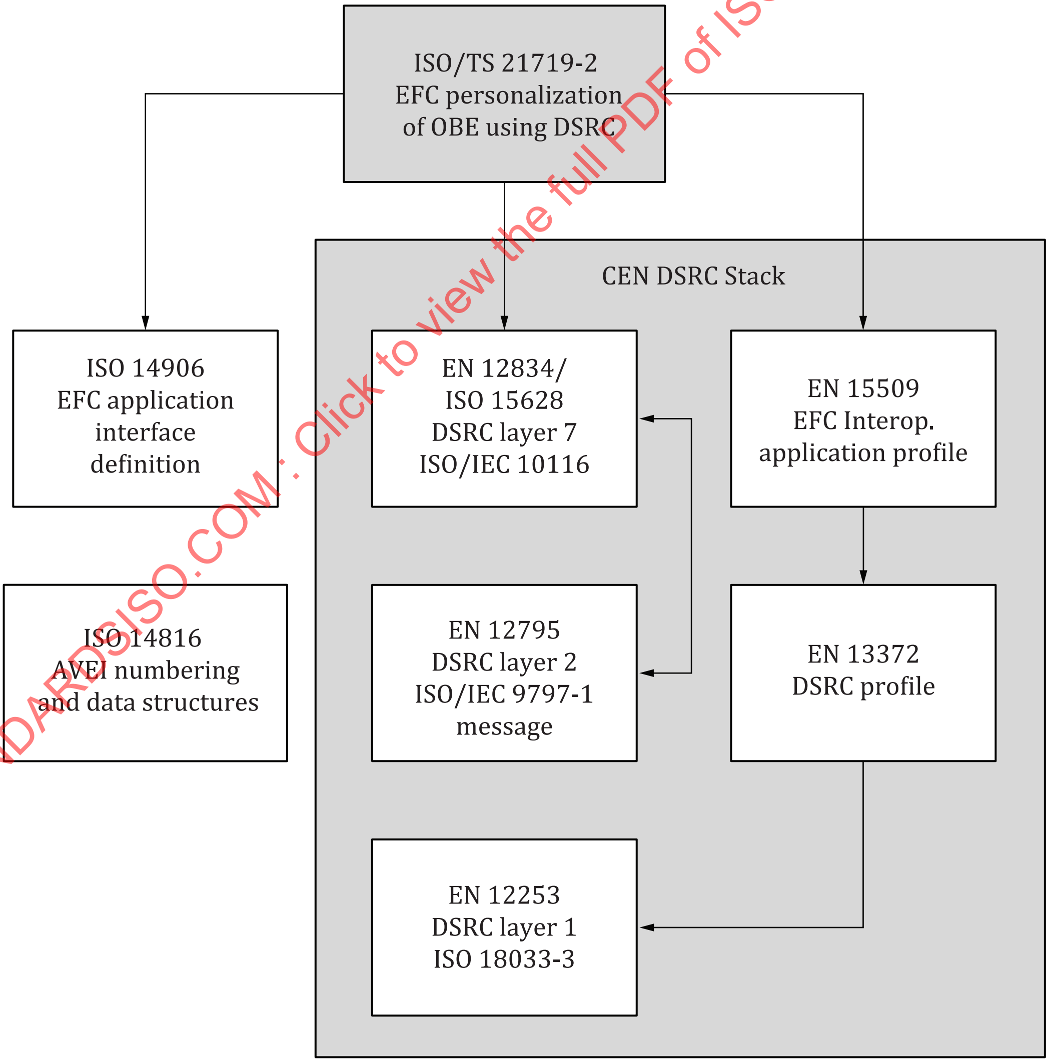


**Figure 4 — Relationship and references between standards for the CEN DSRC stack**

## 7.3 OBE personalization functions

### 7.3.1 General

The OBE shall offer the following functions in order to support personalization:

- initialization of communication: used to establish a communication session with the OBE;
- transferring OBE identifier(s) to the PE; (optional);
- writing of data: used to update data in the OBE;
- terminate session: used to terminate the personalization session with the OBE.

### 7.3.2 Initialization and termination

For CEN-DSRC, the OBE shall provide the following functions:

- INITIALIZATION, and RELEASE application layer services according to ISO 15628 and EN 12834.

DSRC stack implementations of initialization and termination according to ETSI ES 200 674-1 shall be in accordance with Annex C.

During initialization, the OBE shall transfer the following security parameters to the PE:

- random number from the OBE, RndOBE;
- key diversifier (optional);
- key reference (optional).

### 7.3.3 Retrieving the OBE identifier

In order for the PE to know the identity of the unit and, if necessary, provide a parameter for key derivation to the PE, the function GET according to ISO 15628 and EN 12834 may optionally be implemented.

It is out of the scope of this document to define the exact parameter to be used as the identifier.

### 7.3.4 Writing of data

The main functionality of personalization is to write or update data to already existing data fields (attributes) in an EFC application in the OBE.

The writing of data shall be performed by using the EFC function SET_SECURE as defined in ISO 14906.

Application attributes are defined with their container types in the application interface standard ISO 14906. Security keys are stored in attributes with container type 2 (octet string).

The **SET_SECURE.request** shall, for personalization, be used as shown in Table 2 where the settings of optional parameters are defined and shown in bold for the purpose of this document.

**Table 2 — SET_SECURE.request**

| parameter name | ASN.1 type | Value | Remark/constraints |
|---|---|---|---|
| Element identifier EID | Dsrc-EID | 1–127 | |
| ActionType | INTEGER(0..127,...) | 3 | |
| AccessCredentials | OCTET STRING | | PRESENT, Length = 8 octets |
| ActionParameter | OCTET STRING | | Content; see Table 3 |

**Table 2** *(continued)*

| parameter name | ASN.1 type | Value | Remark/constraints |
|---|---|---|---|
| Mode | BOOLEAN | TRUE | Confirmed mode |

The ActionParameter shall carry the attributes to be written into the OBE plus any information required by the algorithm providing the security measures. SET_SECURE.request shall be used in confirmed mode, and a reply shall always be expected.

The content of the action parameter (OCTET STRING) within the scope of this document is defined in Table 3.

**Table 3 — Action parameter content definition**

| Parameter | Length (octets) | Definition |
|---|---|---|
| Option_indicator request | 1 | Always present<br>Bit string that defines which optional parameters are present in Action Parameter; it is defined as follows:<br>$b_7$ – AttributeList present<br>$b_6$ – AttributeListEncrypted present<br>$b_5$ – KeyRefEnc present<br>$b_4$ – RndPE present<br>$b_3$ – Autenticator_Request present<br>$b_2$ – KeyRefAuthReq present<br>$b_1$ – KeyRefAuthRes present<br>$b_0$ – Not used<br>Table 4 shows allowed combinations of the Option Indicator. |
| AttributeList | n. | Optional<br>An attributeList according to ISO 14906-<br>Either the parameter AttributeList or AttributeListEncrypted shall be present. |
| AttributeListEncrypted | m. | Optional<br>An octet string that contains an AttributeList that has been padded to even 16 octet blocks and encrypted.<br>Either the parameter AttributeList or AttributeListEncrypted shall be present. |
| KeyRefEnc | 1 | Optional<br>Encryption Key reference.<br>Shall be present if AttributeListEncrypted is present. |
| RndPE | 16 | Optional<br>Random number from the PE.<br>Shall be present if AttributeListEncrypted is present or if KeyRefAuthRes is present. |
| Authenticator_Request | 8 | Optional<br>Authenticator to secure the integrity of the AttributeList. |
| KeyRefAuthReq | 1 | Optional<br>Shall be present if Authenticator_Request is present. |

**Table 3** *(continued)*

| Parameter | Length (octets) | Definition |
|---|---|---|
| KeyRefAuthRes | 1 | Optional<br>When present, an Authenticator_Response will be provided in the response. |

NOTE 1 The action parameter structure in Table 3 is equivalent to the following ASN.1 type definition encoded with unaligned packed encoding rules to be coded into the OCTET STRING of the Action Parameter.

```
ActionParameterContent:: = SEQUENCE {
   fill                    BOOLEAN,
   attributelist           OCTET STRING             OPTIONAL,
   attributelistEncrypted  OCTET STRING             OPTIONAL,
   keyRefEnc               INTEGER (0..255)         OPTIONAL,
   rndPE                   OCTET STRING (SIZE(8))   OPTIONAL,
   authenticatorReq        OCTET STRING (SIZE(8))   OPTIONAL,
   keyRefAuthReq           INTEGER (0..255)         OPTIONAL,
   keyRefAuthRes           INTEGER (0..255)         OPTIONAL
}
```

The content of the allowed combinations of option indicators in request within the scope of this document is defined in Table 4.

**Table 4 — Allowed combinations of option indicators in request**

| | Option set 1 | Option set 2 | Option set 3 | Option set 4 |
|---|---|---|---|---|
| $b_7$ – AttributeList present | X | X | | |
| $b_6$ – AttributeListEncrypted present | | | X | X |
| $b_5$ – KeyRefEnc present | | | X | X |
| $b_4$ – RndPE present | | X | X | X |
| $b_3$ – Autenticator_Request present | X | X | X | X |
| $b_2$ – KeyRefAuthReq present | X | X | X | X |
| $b_1$ – KeyRefAuthRes present | | X | | X |

The **SET_SECURE.response** is used according to ISO 14906 where the response parameter content is used according to Table 5 and Table 6.

**Table 5 — SET_SECURE.response**

| Parameter name | ASN.1 type | Value | Remark/constraints |
|---|---|---|---|
| ResponseParameter | OCTET STRING | See Table 6 | Optional use |
| Return Code (Ret) | ReturnStatus | | Optional use |

SET_SECURE.response shall carry the ResponseParameter and the confirmation of the corresponding request.

**Table 6 — Response parameter definition**

| Parameter | Length (octets) | Definition |
|---|---|---|
| Authenticator_Response | 8 | Optional<br>Authenticator to prove proper writing of AttributeList.<br>Present if an Authenticator_Response is requested. |

**Table 6** *(continued)*

| Parameter | Length (octets) | Definition |
|---|---|---|
| RndOBE2 | 8 | Optional<br>Random number used to calculate the Authenticator_Response.<br>Present if an Authenticator_Response is requested |

NOTE 2 The response parameter structure in Table 6 is equivalent to the following ASN.1 type definition encoded with unaligned packed encoding rules to be coded into the OCTET STRING of the ResponseParameter.

```
ResponseParameterContent:: = SEQUENCE {
   authenticatorRes         OCTET STRING (SIZE(8))
   rndOBE2                  OCTET STRING (SIZE(8))
}
```

## 7.4 Security requirements

This document defines security features and mechanisms based on the security framework defined in ISO/TS 19299.

All security functionality is implemented according to the following security primitives:

— AES128 encryption algorithm as defined in ISO/IEC 18033-3:2010, 5.2;

— message Authentication Code (MAC) calculation according to MAC algorithm 5 as defined in ISO/IEC 9797-1:2011, 7.6;

— cipher Block Chaining (CBC) algorithm according to ISO/IEC 10116:2017, Clause 7;

— padding method 4 according to ISO/IEC 9797-1:2011, 6.3.5.

The security related data elements listed in Table 7 shall be handled by the OBE.

**Table 7 — Security related data elements**

| Name | Length (octets) | Remarks |
|---|---|---|
| Personalization AccessKey | **16** | AccessKey used for computation of AccessCredentials that allows for writing attributes received in the AttributeList into the EFC application. |
| AccessCredentials | **8** | AccessCredentials that allows for writing attributes received in the AttributeList into the EFC application. |
| AC_CR-KeyReference | **2** | Reference to the key generation and the diversifier for the computation of PersonalizationAccessKey. |
| RndOBE | **8** | Random number received from the OBE in the initialization phase of the personalization transaction and used for computation of access credentials. If RndOBE received from the OBE is 4 octets, it shall be padded to eight octets by using the Padding method 2 according to ISO/IEC 9797-1. |
| RndOBE2 | **8** | Random number generated by the OBE and used in the calculation of the Authenticator_Response. |
| EncryptionKey | **16** | Encryption Key used to encrypt/decrypt the AttributeList. |
| KeyRefEnc | **1** | Reference to EncryptionKey used to encrypt the AttributeList<br>The reference corresponds to the Attribute ID where the key is stored. |
| RndPE | **16** | Random number, from PE used as Start Vector for encryption of the AttributeList and for the computation of the Authenticator_Response. |

**Table 7** *(continued)*

| Name | Length<br>(octets) | Remarks |
|---|---|---|
| AuthenticationReqKey | **16** | AuthenticationKey used for the computation of Authenticator_Request calculated over the ApplicationList (or AttributeListEncrypted) that shall be written. |
| AuthenticationResKey | **16** | AuthenticationKey used for the computation of Authenticator_Response calculated by the OBE over the ApplicationList when it has been written. |
| Authenticator_Request | **8** | Authenticator calculated over the ApplicationList (or AttributeListEncrypted) that shall be written. |
| Authenticator_Response | **8** | Authenticator calculated by the OBE over the ApplicationList when it has been written. |
| KeyRefAuthReq | **1** | Reference to AuthenticationKey used for the computation of Authenticator_Request calculated over the ApplicationList (or AttributeListEncrypted) that shall be written.<br>The reference corresponds to the Attribute ID where the key is stored. |
| KeyRefAuthRes | **1** | Reference to AuthenticationKey used for the computation of Authenticator_Response calculated by the OBE over the ApplicationList when it has been written.<br>The reference corresponds to the Attribute ID where the key is stored. |

The OBE shall be able to:

- generate a random number RndOBE that shall be transferred to the PE in the initialization phase;
- generate a random number RndOBE2 that shall be transferred to the PE in the response;
- check the received access credentials in order to authenticate the PE. This check uses the RndOBE and the PersonalizationAccessKey. If erroneous access credentials are received, no further processing of the received data shall be performed and the ReturnStatus in the SET_SECURE response shall be set to “Access Denied”;
- decrypt the received AttributeListEncrypted before performing the write operation to the EFC element. The decryption is performed by using the RndPE and the EncryptionKey defined by the KeyRefEnc;
- check the received attributes for correct size. If the check is negative, no writing of data shall be performed and the ReturnStatus shall be set to “Argument Error”;
- check the received Authenticator_Request calculated over the AttributeList, by using the received RndOBE and the AuthenticationReqKey defined by the KeyRefAuthReq, in order to check the data integrity, before writing the attribute list data into the EFC element. If the check is negative no writing of data shall be performed and the ReturnStatus shall be set to “Argument Error”;
- calculate an Authenticator_Response over the ApplicationList, the received RndPE and the RndOBE2, by using the AuthenticationResKey defined by the KeyRefAuthRes, in order to supply a proof that data was properly written to the EFC element.

If a Write command carries an AttributeList that includes an update of the security keys, authenticators shall be calculated or checked by using the already existing keys in the OBE.

Security calculations shall be as defined in Annex A.

## 7.5 Transaction requirements

An OBE that conforms to this document shall be able to perform a personalization transaction that includes the functions as defined in Clause 7.

Annex D provides an informative example of a personalization transaction when using CEN DSRC.

# 8 Personalization equipment requirements

## 8.1 General

This clause contains the normative conformance requirements on the personalization equipment (PE) for profile number 1; EFC-DSRC-Personalization Profile 1.

## 8.2 DSRC lower layer requirements

### 8.2.1 Supported DSRC stacks

The PE shall be able to support OBEs with corresponding DSRC stack as defined in 7.2.

### 8.2.2 CEN DSRC stack

The following requirements apply for the personalization profile when using the CEN DSRC stack:

— the PE shall conform to EN 15509:2022, 6.2.2.

## 8.3 PE personalization functions

The PE shall be able to support OBEs as defined in 7.3.

## 8.4 Security requirements

Since the security functionality may reside partly or fully within a central system of the entity responsible for the personalization, the requirements defined for the PE in this subclause shall be seen as requirements on the combination of the PE and a possible central system.

The PE shall support the security data elements as defined in the Table 7.

The PE shall be able to:

— generate access credentials using the received RndOBE and the PersonalizationAccessKey in order to authenticate itself towards the OBE;

— generate a random number RndPE;

— encrypt the AttributeList before sending it to the OBE. The encryption is performed using the RndPE and an EncryptionKey defined by the KeyRefEnc;

— calculate an Authenticator_Request calculated over the AttributeList and the RndOBE and send this authenticator to the OBE in order to secure the integrity of the data. The calculation is performed using an AuthenticationReqKey defined by the KeyRefAuthReq;

— check the Authenticator_Response that is received from the OBE and calculated on the AttributeList, the RndPE and the received RndOBE2, in order to assess that data was properly written. The check is performed by using the AuthenticationResKey defined by the KeyRefAuthRes.

## 8.5 Transaction requirements

A PE that conforms to this document shall be able to perform a personalization transaction that includes the functions as defined in Clause 8.

Annex D provides an informative example of a personalization transaction when using CEN DSRC.

# Annex A
(normative)

# Security calculations

## A.1 General

This annex contains detailed definitions of required security features and calculations.

## A.2 Access credentials

### A.2.1 General

Access credentials are used to protect against non-authorized access to data in the OBE, and they are a way for the OBE to verify that it is being accessed by an authentic PE.

### A.2.2 Principle of access credentials

This subclause describes the principle of access control to change the OBE information.

In the initialization phase between an OBE and the PE, the OBE sends an Application List that displays available EFC Applications in the OBE. For each application, the list contains an EFC Context mark, an Access Credential Key Reference (AC_CR-KeyReference) and a random number (RndOBE).

The AC_CR-Key reference is used as key diversifier for the PE to derive the Personalization Access Key from the Master Personalization Access Key. The calculations for this are described in A.3.

Knowing the Personalization Access Key, the PE is now able to calculate correct access credentials (AC_CR) over the RndOBE. The AC_CR is provided in messages to the OBE during the session.

The OBE compares the received access credentials with its own calculation. In case they are equal, the OBE accepts the PE as a genuine PE and is accepting the command from the PE.

### A.2.3 Calculation of access credentials

The calculation of access credentials in the OBE and PE uses the following standardized security primitives:

- Message Authentication Code calculation in Chained Block Cipher mode according to ISO/IEC 9797-1:2011, 7.6; MAC Algorithm 5 (CMAC) with Padding Method 4;
- Encryption algorithm AES-128 according to ISO/IEC 18033-3:2010, 5.2.

As input data string to the MAC algorithm, the 8 octet RndOBE shall be used. It shall be padded to 16 octets using Padding Method 4. The 16 octets output from the MAC algorithm shall be truncated and the 8 leftmost octets are used as AC-CR.

The encryption key for the encryption algorithm shall be the Personalization Access Key. This key is already present in the OBE but the PE derived it from the Master Personalization Access Key by using the AC-CR-Key reference parameter.

## A.3 Key derivation for the personalization access key

### A.3.1 General

In order to avoid that the Master Personalization Key is stored in all OBEs, different diversified Personalization Keys are used. The Master Personalization Key therefore only has to be available at time of personalization and can reside in a secure application module (SAM) to which the PE has access.

### A.3.2 Calculation of the Personalization Access Key

The Personalization Access Key of a Key Generation k shall have a length of 16 octets and shall be derived from the 32 octet Personalization Master Access Key using the AES-256 single block encryption as specified in ISO/IEC 18033-3:2010, 5.2 according to the description below and using the AC_CR_KeyRef as described below:

a) let the Personalization Master Access Key for a given generation k be: MPAcK(k);

b) let the derived Personalization Access Key be: PAcK(k);

c) make the concatenation of a multiple of AC_CR-KeyReference to obtain a 16 octets value, VAL:

VAL = ‘AC_CR_KeyReference || AC_CR-KeyReference || AC_CR-KeyReference || AC_CR-KeyReference || AC_CR-KeyReference || AC_CR-KeyReference || AC_CR-KeyReference || AC_CR-KeyReference’;

d) compute the PAcK(k) as follows:

PAcK(k) = e[MPAcK (k)] (VAL).

The generation of the Personalization Access Key is defined at the moment of personalization and should be associated to the OBE (e.g. through information in the ContextVersion). How this is done is outside the scope of this document.

## A.4 AttributeList Encryption

### A.4.1 General

The Attribute List that is being sent to the OBE can be encrypted in order to prevent sensitive data such as, for example, security keys from being exposed.

### A.4.2 Principle of Encryption of the Attribute List

In order to make the data in the Attribute List non-readable for unauthorized entities, the list may be encrypted before being transferred to the OBE.

The OBE will receive the encrypted Attribute List from the PE together with a RndPE and a KeyRefEnc.

RndPE is generated by the PE and used as the start vector for the block encryption. KeyRefEnc is the reference to the encryption key that was used by the PE.

The OBE will decrypt the Attribute List using the RndRSE and the referenced encryption key.

### A.4.3 Encryption of the Attribute List

Encryption of the Attribute List shall be performed by the PE that uses the following standardized security primitives:

- Padding Method 2 according to ISO/IEC 9797-1:2011, 6.3.5, in order to achieve even 16 octet blocks before encryption;
- encryption algorithm AES-128 according to ISO/IEC 18033-3:2010, 5.2;

- Cipher Block Chaining (CBC) according to ISO/IEC 10116:2017, 7.2, with m = 1 and with RndPE as Starting Variable.

The KeyRefEnc that the PE includes in the SET_SECURE.request defines the reference to the key that was used by the PE for encryption.

Decryption of the Attribute List shall be performed by the OBE that uses the following standardized security primitives:

- encryption algorithm AES-128 according to ISO/IEC 18033-3:2010, 5.2;
- Cipher Block Chaining (CBC) according to ISO/IEC 10116:2017, 7.3, with m = 1 and with RndPE as Starting Variable;
- Padding Method 2 according to ISO/IEC 9797-1:2011, 6.3.5, in order to delete padding after decryption.

The KeyRefEnc that the PE includes in the SET_SECURE.request defines the reference to the key that shall be used by the OBE for decryption.

## A.5 Authenticator in request

### A.5.1 General

In order to allow for the OBE to verify that the received Attribute List or Encrypted Attribute List originates from an authentic entity and not has been altered, the PE can apply an Authenticator_Request to the data.

### A.5.2 Principle of Authentication of request

The PE will in the SET_SECURE.request append an Authenticator_Request that is calculated over the (unencrypted) Attribute List and the RndOBE.

Before writing the data into the application memory, the OBE will check this authenticator by using the Authentication Key defined by the KeyRefAuthReq that also is supplied in the request.

### A.5.3 Calculation of Authenticator_Request

This Authenticator is calculated over the Attribute List and the RndOBE by the PE that uses the following standardized security primitives:

- Message Authentication Code calculation in Chained Block Cipher mode according to ISO/IEC 9797-1:2011, 7.6; MAC Algorithm 5 (CMAC) with Padding Method 4;
- encryption algorithm AES-128 according to ISO/IEC 18033-3:2010, 5.2.

As input to the MAC algorithm, the Attribute List concatenated with; the 8 octet RndOBE shall be used. Padding shall be applied according to Padding Method 4 in order to obtain full 16 octet blocks.

The 16 octets output from the MAC algorithm shall be truncated and the 8 leftmost octets are used as Authenticator_Request value.

The Authenticator_Request is sent to the OBE in the SET_SECURE.request together with the KeyRefAuthReq that defines the reference to the key that was used by the PE in the calculation.

## A.6 Authenticator in response

### A.6.1 General

In order to allow for the PE to verify that the submitted Attribute List or Encrypted Attribute List, was properly written into the OBE EFC application memory, the OBE can respond with an Authenticator_Response.

### A.6.2 Principle of Authentication of response

The OBE will after successfully having written the received Attribute List to the application memory, append an Authenticator_Response to the SET_SECURE.response. The Authenticator is calculated over the Attribute List and the RndPE.

The PE can, by checking this authenticator, get proof that the submitted Attribute List was properly stored in an authorized OBE.

### A.6.3 Calculation of Authenticator_Response

This Authenticator is calculated by the OBE over the Attribute List, the RndPE and the RndOBE2 when a KeyRefAuthRes is present in the SET_SECURE.request. The following standardized security primitives are used in the calculation:

— Message Authentication Code calculation in Chained Block Cipher mode according to ISO/IEC 9797-1:2011, 7.6; MAC Algorithm 5 (CMAC) with Padding Method 4;

— encryption algorithm AES-128 according to ISO/IEC 18033-3:2010, 5.2.

As input to the MAC algorithm, the Attribute List concatenated with the 8 octet RndPE and the 8 octet RndOBE2 shall be used. Padding shall be applied according to Padding Method 4 in order to obtain full 16 octet blocks.

The 16 octets output from the MAC algorithm shall be truncated and the 8 leftmost octets are used as Authenticator_Response value.

The authenticator is sent to the PE in the SET_SECURE.response.

## A.7 Derivation of Encryption and Authentication keys

### A.7.1 General

In order to avoid the Master Encryption or Authentication key from being stored in all OBEs, different diversified keys are used. The master keys, therefore, only have to be available at time of personalization and can reside in a secure application module (SAM) to which the PE has access.

### A.7.2 Calculation of an Encryption/Authentication Key

The Encryption/Authentication Keys stored in the OBE shall be derived from Master Authentication Keys, using a unique identifier related to the OBE such as EquipmentOBUID.

How this identifier is transferred to the PE from the OBE is out of scope of this document but optionally the identifier can be read from the OBE by a standard GET.request command. It can also be supplied together with the OBE as a printed number or code.

The Encryption or Authentication Key of a Key Generation k shall have a length of 16 octets and shall be derived from the 32 octets master key using the AES-256 single block encryption as specified in ISO/IEC 18033-3:2010, 5.2 according to the description below and using a key diversifier of in total 16 octets:

a) let the Master Authentication Key for a given generation k be: MAuthK(k);

b) let the derived Authentication Key be: AuthK(k);

c) let the 16 octets key diversifier be = VAL

d) Compute the AuthK(k) as follows:

AuthK(k) = e[MAuthK (k)] (VAL).

Example of VAL = ‘EquipmentOBUID || EquipmentOBUID || EquipmentOBUID || EquipmentOBUID´.

# Annex B
(normative)

# PICS proforma

## B.1 General

In order to evaluate the conformance of a particular implementation, it is necessary to have a statement of those capabilities and options that have been implemented. This is called an implementation conformance statement (ICS) or, more specifically when it covers transactions, a protocol implementation conformance statement (PICS).

This annex provides a PICS proforma (Table B.1 to Table B.12) to be filled in by equipment suppliers in order to declare conformance to this document.

## B.2 Purpose and structure

The purpose of this PICS proforma is to provide a mechanism whereby a supplier of an implementation of the requirements defined in this document can provide information about the implementation in a standardized manner.

The PICS proforma is subdivided into subclauses for the following categories of information:

- identification of the implementation;
- identification of the protocol;
- global statement of conformance;
- PICS proforma tables.

## B.3 Instructions for completing PICS proforma

### B.3.1 Definition of support

A capability is said to be supported if the implementation under test (IUT) can:

- generate the corresponding operation parameters (either automatically or because the end user requires that capability explicitly), and
- interpret, handle and, when required, make available to the end user the corresponding error or result.

A protocol element is said to be supported for a sending implementation if it is able to generate it under certain circumstances (either automatically or because the end user requires relevant services explicitly).

A protocol element is said to be supported for a receiving implementation if it is correctly interpreted and handled and also, when appropriate, made available to the end user.

### B.3.2 Status column

This column in the tables indicates the level of support required for conformance. The values are as follows:

m mandatory support is required;

o optional support is permitted for conformance to this document. If implemented, it should conform to the specifications and restrictions contained in this document. These restrictions may affect the optionality

c the item is conditional (support of the capability is subject to a predicate);

c: m the item is mandatory if the predicate is true, optional otherwise;

- the item is not applicable;

i the item is outside the scope of this PICS.

In the PICS proforma tables, every leading item marked “m” shall be supported by the IUT. Sub-items marked “m” shall be supported if the corresponding leading item is supported by the IUT.

### B.3.3 Support column

This column (see Table B.6 to Table B.12) shall be completed by the supplier or implementer to indicate the level of implementation of each item. The proforma has been designed such that values required are the following:

Y yes, the item has been implemented;

N no, the item has not been implemented;

- the item is not applicable.

All entries within the PICS proforma shall be made in ink. Alterations to such entries shall be made by crossing out, neither erasing nor making the original entry illegible, and by writing the new entry alongside. All such alterations to records shall be initialized by the person who made them.

### B.3.4 Item reference numbers

Each line within the PICS proforma which requires that implementation details be entered is numbered at the left-hand edge of the line. This numbering is included as a means of uniquely identifying all possible implementation details within the PICS proforma. This referencing is used both inside the PICS proforma, and for references from other test specification documents.

The means of referencing individual responses is done in the following sequence:

a) a reference to the smallest individual response enclosing the relevant item;

b) a solidus character (“/”);

c) the reference number of the row in which the response appears;

d) if — and only if — more than one response occurs in the row identified by the reference number, implicit labelling of each possible entry as “a”, “b”, “c”, etc., from left to right, with this letter appended to the sequence.

## B.4 PICS proforma for OBE

### B.4.1 Identification of the implementation

The proforma shown in Tables B.1 to B.3 are to be used to identify the implementation on the OBE side.

**Table B.1 — Identification of PICS**

| Item no. | Question | Response |
|---|---|---|
| 1 | Date of statement (DD/MM/YY) | |
| 2 | PICS serial number | |
| 3 | ISO/TS 21719-2 version | |
| 4 | Other information | |

**Table B.2 — Identification of the OBE supplier**

| Item no. | Question | Response |
|---|---|---|
| 1 | Organization name | |
| 2 | Contact name(s) | |
| 3 | Address | |
| 4 | Telephone number | |
| 5 | e-mail address | |
| 6 | Other information | |

**Table B.3 — Identification of the OBE**

| Item no. | Question | Response |
|---|---|---|
| 1 | Brand name | |
| 2 | Type, version | |
| 3 | Manufacturer ID | |
| 4 | Equipment Class | |
| 5 | Serial numbers of the supplied unit(s) | |
| 6 | Other information | |

### B.4.2 Global statement of conformance

The global statement of conformance is as follows:

— Are all mandatory capabilities implemented? (Yes/No) ...............

### B.4.3 PICS proforma tables

The proforma shown in Tables B.4 to B.6 are to be used to list the implementation supported by the OBE.

**Table B.4 — Implemented DSRC stacks**

| Item no. | Element | Reference | Status[a] | Support |
|---|---|---|---|---|
| 1 | CEN DSRC | 7.2.2 | o | |
| 2 | Italian DSRC | Annex D | o | |

[a] One of the DSRC stacks shall be implemented.

**Table B.5 — Implemented Layer 7/EFC functions**

| Item no. | Element | Reference | Status | Support |
|---|---|---|---|---|
| 1 | INITIALIZATION | 7.3.2 | m | |
| 2 | GET | 7.3.3 | o | |
| 3 | SET_SECURE | 7.3.4 | m | |
| 4 | EVENT_REPORT | 7.3.2 | m | |

**Table B.6 — Implemented security functions**

| Item no. | Element | Reference | Status | Support |
|---|---|---|---|---|
| 1 | Generation of RndOBE | Annex A | m | |
| 2 | Generation of RndOBE2 | Annex A | m | |
| 3 | Check access credentials | Annex A | m | |
| 4 | Decrypt AttributeListEncrypted | Annex A | m | |
| 5 | Check Authenticator_Request | Annex A | m | |
| 6 | Calculate Authenticator response | Annex A | m | |

## B.5 PICS proforma for PE

### B.5.1 Identification of the implementation

The proforma shown in Tables B.7 to B.9 are to be used to identify the implementation on the PE side.

**Table B.7 — Identification of PICS**

| Item no. | Question | Response |
|---|---|---|
| 1 | Date of statement (DD/MM/YY) | |
| 2 | PICS serial number | |
| 3 | ISO/TS 21719-2 version | |
| 4 | Other information | |

**Table B.8 — Identification of the PE supplier**

| Item no. | Question | Response |
|---|---|---|
| 1 | Organization name | |
| 2 | Contact name(s) | |
| 3 | Address | |
| 4 | Telephone number | |
| 5 | e-mail address | |
| 6 | Other information | |

**Table B.9 — Identification of the PE**

| Item no. | Question | Response |
|---|---|---|
| 1 | Brand name | |
| 2 | Type, version | |
| 3 | Serial numbers of the supplied unit(s) | |
| 4 | Other information | |

### B.5.2 Global statement of conformance

The global statement of conformance is as follows:

— Are all mandatory capabilities implemented? (Yes/No) ...............

### B.5.3 PICS proforma tables

The proforma shown in Tables B.10 to B.12. are to be used to list the implementation on the PE side.

**Table B.10 — Implemented DSRC stacks**

| Item no. | Element | Reference | Status[a] | Support |
|---|---|---|---|---|
| 1 | CEN DSRC | 7.2.2 | o | |
| 2 | Italian DSRC | Annex D | o | |

[a] One of the DSRC stacks shall be implemented.

**Table B.11 — Implemented Layer 7/EFC functions**

| Item no. | Element | Reference | Status | Support |
|---|---|---|---|---|
| 1 | INITIALIZATION | 7.3.2 | m | |
| 2 | GET | 7.3.3 | o | |
| 3 | SET_SECURE | 7.3.4 | m | |
| 4 | EVENT_REPORT | 7.3.2 | m | |

**Table B.12 — Implemented security functions**

| Item no. | Element | Reference | Status | Support |
|---|---|---|---|---|
| 1 | Calculation of access credentials | Annex A | m | |
| 2 | Generate RndPE | Annex A | m | |
| 3 | Encrypt AttributeList | Annex A | m | |
| 4 | Calculate Authenticator_Request | Annex A | m | |
| 4 | Check Authenticator response | Annex A | m | |

# Annex C
(normative)

# Personalization of OBE conforming to ETSI ES 200 674-1

## C.1 General

This annex describes the personalization procedures for OBE compliant to ETSI ES 200 674-1. In particular, the supported data and memory structure of the OBE are compliant to ETSI ES 200 674-1:2013, Annex D. In addition to that, the OBE shall support data structures, procedures, and protocol messages described in the following subclauses and summarized in C.2.5. Common ASN.1 data types definitions are from ETSI ES 200 674-1:2013, 11.3.

## C.2 Personalization functions

### C.2.1 General

The OBE offers the following functions in order to support personalization:

- initialization: used to establish a personalization session with the OBE;
- writing of data: used to update user data in the OBE;
- writing of keys: used to modify access, authentication and personalization keys;
- terminate session: used to terminate the personalization session with the OBE.

### C.2.2 Initialization

Personalization initialization is performed by means of the following protocol messages:

- OPEN-Rq: to start a configuration process, as defined in ETSI ES 200 674-1:2013, 11.5.2;
- Get-TBA-Random-Rq: to get a random number from the OBE, as defined in ETSI ES 200 674-1:2013, 11.5.22, without the limitations in length defined in ETSI ES 200 674-1:2013, Annex D;
- Read-Master_Core-Rq: to read the OBE manufacturer's identifier;
- Get-Master-Record-Rq: to read the AC_CR-KeyReference.

Personalization initialization protocol is performed through the following steps.

1) The PE opens a new session with the OBE by concatenating an OPEN-Rq protocol message with a Read-Master-Core-Rq protocol message that requests ManufacturerId value, a Get-Master-Record-Rq that requests the AC_CR-KeyReference, and a Get-TBA-Random-Rq protocol message that requires a random number of 8 octets in length. The addressed Application Process Invocation Identifier, as per ETSI ES 200 674-1:2013, 11.5.1, identifies the personalization application by bearing the value of '50FE'H.

2) If the OBE terminates the session with a negative Ack (see C.2.5), the requested personalization process is not supported.

3) If the OBE responds with a positive acknowledgement, the 4-octets random number is used to compute the PE's access credentials by using the security algorithm as described in C.2.6.4.

4) The PE then initiates the writing of data (see C.2.3) or writing of key (see C.2.4) phase.

## C.2.3 Writing of data

### C.2.3.1 General

Personalization writing of data is performed by means of the following protocol messages:

— Config-Set-Rq: to write configuration parameters onto the OBE memory, which is detailed in C.2.3.2;

— Config-Set-Rs: to answer to configuration requests, which is detailed in C.2.5.

### C.2.3.2 Config-Set-Rq protocol message

The Config-Set-Rq protocol message and its parameters are defined in Table C.1.

**Table C.1 — Config-Set-Rq**

| Parameter name | ASN.1 type | Limits and allowed values |
|---|---|---|
| Config-Set-Rq | Bit-String-1B | Directive code is ‘4C’Hexadecimal |
| AppToConfig | BIT STRING ((SIZE(16)) | Application to be configured. In this version, only the value ‘50F0’H is allowed, indicating the EETS application. |
| MemoryArea | INTEGER-0–1B | Indicates the memory area where the write operation is performed. Allowed values are<br>— 0, indicating Master Core,<br>— 1, indicating Master Record, and<br>— 2, indicating Application Core.<br>All other values are reserved. |
| Offset | INTEGER-0–1B | Pointer in the addressed data structure where the write operation is to start from. |
| ActualLength | INTEGER-1–1B | Length of the actual data to be stored. This length can be different from the transmitted data when encryption is used. |
| EncrLength | INTEGER-1–1B | Length of the encrypted data. This is the length of the transmitted data, which can be different from the actual length of data to be stored when encryption is used. |
| Data | OCTET STRING | Data to be stored. If encrypted, encryption is as described in C.2.6.2. |
| AccessCredentials | OCTET STRING | Access credentials of the PE in 8 octets, calculated according to C.2.6.1. |
| Mode | INTEGER-0–1B | Indicates how data are transferred. Allowed values are:<br>— 0, encrypted, and<br>— 1, not encrypted. |
| RndPE | INTEGER | Random number generated by the PE on 16 octets, to be used to decrypt encrypted data (if encryption is used) and to calculate OBE authenticator for the Config-Set-Rs protocol message. |
| PersonalizationKeyRef | INTEGER-1–1B | Integer indicating the personalization key to be used to encrypt/decrypt data. |

The Config-Set-Rq protocol message is used for memory areas to be personalized, keeping in mind that one memory can be personalized in one single request by wise use of the Offset and ActualLength parameters.

On receipt of a Config-Set-Rq protocol message, the OBE:

1) verifies the support of the personalization application and the possible encryption (indicated by the Mode parameter). If either of the two is not supported, the OBE terminates the session with a negative Ack (see C.2.5);

2) verifies the access credentials of the PE by implementing the algorithm in C.2.6.1. If access credentials are not valid, the OBE terminates the session with a negative Ack (see C.2.5);

3) if access credentials are valid, the OBE checks the value of the Mode parameter. If Mode indicates encryption and the OBE does not support encryption, the OBE terminates the session with a negative Ack (see C.2.5). Note that the use of Mode parameter allows a personalization process where encrypted data are intermixed with non-encrypted data;

4) if controls on access credentials and Mode are passed, and the Mode parameter indicates encryption, the OBE then uses the RndPE parameter value to decrypt data. The decryption keys for personalization are initially stored in the OBE at manufacturing time, and can be modified by using the services offered by the protocol message Config-Key-Set-Rq (see C.2.4);

5) the length of decrypted data is then checked with the length provided by the parameter ActualLength. If the two lengths do not match, the OBE terminates the session with a negative Ack (see C.2.5);

6) if length check is passed, data is written in the addressed memory area, and the OBE terminates the session with a positive Ack (see C.2.5).

## C.2.4 Writing of keys

### C.2.4.1 General

Personalization writing of keys is performed by means of the following messages:

- Config-Key-Set-Rq: to write configuration parameters onto the OBE memory, which is detailed in C.2.3.2;
- Config-Set-Rs: to answer to configuration requests, which is detailed in C.2.5.

### C.2.4.2 Config-Key-Set-Rq protocol message

The Config-Key-Set-Rq protocol message and its parameters are defined in Table C.2.

**Table C.2 — Config-Key-Set-Rq**

| Parameter name | ASN.1 type | Limits and allowed values |
|---|---|---|
| Config-Key-Set-Rq | Bit-String-1B | Directive code is '4D'Hexadecimal |
| AppToConfig | BIT STRING ((SIZE(16)) | Application to be configured. In this version, only the value '50F0'H is allowed, indicating the EETS application. |
| Key type | INTEGER-0–1B | Indicates the memory area where the write operation is to be performed. Allowed values are:<br>— 0, indicating Authentication Key,<br>— 1, indicating Access Key, and<br>— 2, indicating Personalization Key.<br>All other values are reserved. |
| KeyRef | INTEGER-1–1B | Integer indicating the key to be modified. |
| KeyLength | INTEGER-1–1B | Length of the key to be modified. |

**Table C.2** *(continued)*

| Parameter name | ASN.1 type | Limits and allowed values |
|---|---|---|
| Key | OCTET STRING | Encrypted key value to be stored. Encryption is as described in C.2.6.2. |
| AccessCredentials | OCTET STRING | Access credentials of the PE in 8 octets, calculated according to C.2.6.1. |
| RndPE | INTEGER | Random number generated by the PE on 16 octets, to be used to decrypt encrypted data and to calculate OBE authenticator for the Config-Set-Rs protocol message. |
| PersonalizationKeyRef | INTEGER-1–1B | Integer indicating the personalization key to be used to encrypt/decrypt data |

The Config-Key-Set-Rq protocol message is used for keys to be modified.

On receipt of a Config-Key-Set-Rq protocol message, the OBE:

1) verifies the support of the personalization application and of the encryption (in the case of keys, encryption is mandated). If either of the two is not supported, the OBE terminates the session with a negative Ack (see C.2.5);

2) verifies the access credentials of the PE by implementing the algorithm in C.2.6.1. If access credentials are not valid, the OBE terminates the session with a negative Ack (see C.2.5);

3) if controls on access credentials are passed, the OBE then can use the RndPE parameter value to decrypt the received key. The way keys are stored in the OBE (encrypted or not encrypted) is left to implementation choices, the only requirement is that they can be retrieved and used by means of the related KeyRef. If the received key is not acceptable for any reason, the OBE terminates the session with a negative Ack (see C.2.5);

4) if the received key is safely stored, the OBE terminates the session with a positive Ack (see C.2.5).

### C.2.5 Termination

The termination phase is performed by means of the Configuration-Set-Rs protocol message, which is used to answer to both write of data and write of keys functions.

The Configuration-Set-Rs protocol message can carry an authenticator if the following conditions apply:

1) personalization with encryption was used (see C.2.2);

2) the operation terminated correctly.

Authenticator in response is calculated as described in C.2.6.4. In case of no success, only result code and action code are transferred.

The used result codes and action codes are listed in the following Table C.3.

**Table C.3 — Acknowledgements of a personalization operation**

| Condition | Result code | Diagnostic code |
|---|---|---|
| Success | ‘06’H | ‘00’H |
| Personalization application not supported or encryption not supported | ‘15’H | ‘06’H |
| Invalid access credentials | ‘15’H | ‘11’H |
| Data length mismatch | ‘15’H | ‘04’H |

### C.2.6 Security calculations

#### C.2.6.1 Access credentials

Calculation of access credentials is performed according to the algorithm in A.2.3, where the random number to be used is the result of the Get-TBA-Random-Rq protocol message issued at initialization phase (see C.2.2).

#### C.2.6.2 Calculation of the personalization access key

Calculation of the Personalization Access Key is performed according to the algorithm in A.3.2.

#### C.2.6.3 Data Encryption and decryption

Data encryption is performed according to the algorithm described in A.4.3, where the KeyRefEnc to be used is provided by the PE in the PersonalizationKeyRef parameter of the Config-Set-Rq or Config-Key-Set-Rq protocol message.

#### C.2.6.4 Authenticator in response calculation

Authenticators in response are calculated according to the algorithms described in A.6.3, by using the received data (either Data or Key, according to the function performed) concatenated with the RndPE parameter value.

#### C.2.6.5 Calculation of encryption/authentication Keys

Encryption and authentication keys are derived from Master keys using the algorithm described in A.7.

# Annex D
(informative)

# Transaction example

## D.1 Overview

This annex presents an example of a personalization transaction where the personalization data is transferred to an OBE from a PE using the Option set 4 as defined in 7.3.4.

EquipmentOBUID is in this example used as an identifier for the OBE and is retrieved by a GET command before the SET_SECURE is submitted by the PE.

The overview of the transaction is given in Table D.1.

**Table D.1 — Transaction overview**

| Personalization equipment | | On-board equipment | Remarks |
|---|---|---|---|
| INITIALIZATION.request (BST)<br>AID = EFC | → | | The PE is asking for information of EFC applications in not yet connected OBEs. |
| | ← | INITIALIZATION.response (VST)<br>Application list of EFC<br>applications<br>EFC context mark<br>AC_CR key reference<br>RndOBE | The OBE sends an application list to the PE with information over available EFC applications and also security related information in order for the PE to calculate the access credentials. |
| GET.request<br>AttributeIdList<br>AC_CR | → | | The PE calculates access credentials and submits a GET.request in order to retrieve information about the OBE identity. |
| | ← | GET.response<br>Attribute list<br>Equipment OBE ID | The OBE answers with a GET.response that contains the requested Equipment OBE ID. |
| SET_SECURE.request<br>AC_CR<br>Attribute list encrypted<br>Key ref enc<br>RndPE<br>Authenticator request<br>Key ref auth req<br>Key ref auth res | → | | The PE can now use the Equipment OBE ID and knowledge about the master keys to calculate which keys are stored in this specific OBE.<br>With these derived keys the attribute list can be encrypted and authenticated. The RndPE that was used in these calculations is also sent to the OBE together with references to which exact keys were used.<br>The PE also sends a key reference to the authenticator to be submitted by the OBE in the response. |