Federal Office
for Information Security

Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE
Premier ministre
ANSSI

Technical Guideline TR-03110

Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token –

Part 4: Applications and Document Profiles

Version 2.21

21. December 2016

History

Version	Date	Comment
1.00	2006-02-08	Initial public version.
1.01	2006-11-02	Minor corrections and clarifications.
1.10	2007-08-20	Revised version.
1.11	2008-02-21	Minor corrections and clarifications.
2.00	2008-10-27	Enhanced version.
2.01	2009-05-05	Minor corrections and clarifications. Additional Mapping for PACE.
2.02	2009-11-09	Adjustments to PACE required due to international standardization.
2.03	2010-03-24	Clarification on the definition of a session. Standardization of domain parameters. Introduction of a secondary security object.
2.04	2010-09-15	Clarifications on certificate extensions. Improved handling of chip-specific keys for privileged terminals.
2.05	2010-10-14	Clarifications on RFU-bits, “Read access to eID” deprecated
2.10	2012-03-20	Split into three parts
2.20	2015-02-03	Enhanced version with additional mechanisms. Split into four parts.
2.21	2016-12-21	Clarifications, minor corrections and optimizations. Simplification of authorization handling.

Federal Office for Information Security
Post Box 20 03 63
D-53133 Bonn

Phone: +49 22899 9582-0
E-Mail: ExtendedAccessControl@bsi.bund.de
Internet: https://www.bsi.bund.de

Contents

List of Figures

List of Tables

1 Introduction

This Part of the Technical Guideline specifies profiles of applications and profiles of documents and eIDAS tokens for electronic identification, authentication and trust services.

1.1 Requirements for eIDAS tokens and Terminals

This Technical Guideline specifies requirements for implementations of eIDAS tokens and terminals. While eIDAS tokens must comply with those requirements according to the terminology described in Section 1.2, requirements for terminals are to be interpreted as guidance, i.e. interoperability of eIDAS token and terminal are only guaranteed if the terminal complies with those requirements, otherwise the interaction with the eIDAS token will either fail or the behavior of the eIDAS token is undefined. In general, the eIDAS token need not enforce requirements related to terminals unless the security of the eIDAS token is directly affected.

1.2 Terminology

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119 [1]. The key word "CONDITIONAL" is to be interpreted as follows:

CONDITIONAL: The usage of an item is dependent on the usage of other items. It is therefore further qualified under which conditions the item is REQUIRED or RECOMMENDED.

When used in tables (profiles), the key words are abbreviated as shown in Table 1.

Key word		**Abbrev.**
MUST / SHALL	REQUIRED	m
MUST NOT / SHALL NOT	–	x
SHOULD	RECOMMENDED	r
MAY	OPTIONAL	o
–	CONDITIONAL	c

Table 1: Key words

2 Applications and Terminals

This specification supports three applications: *ePassport*, *eID* and *eSign* Application.

The following object identifier SHALL be used to identify the roles and authorization levels of different terminal types:

```
id-roles OBJECT IDENTIFIER ::= {
  bsi-de applications(3) mrtd(1) 2
}
```

Note: Access rights may be declared as reserved for future use (RFU). Those access rights may be assigned in later versions of this Technical Guideline. As a consequence eIDAS tokens already issued may have to import certificates with an unexpected certificate holder authorization or unexpected authorization extensions. Due to the calculation of access rights, described in Part 3, the effective authorization will always be restricted to the access rights known by the eIDAS token at the time of personalization.

2.1 ePassport Application

The ePassport Application is described in Part 1 of this Technical Guideline. The issuer of a MRTD (or eIDAS token with ePassport Application) implemented according to this Part of this Technical Guideline MAY define access conditions different from those mandated by Part 1.

2.1.1 Application Identifier

See ICAO Doc 9303 [2].

2.1.2 Authentication

Access to the ePassport application MUST be restricted to inspection systems, it is RECOMMENDED to require the terminal to be authenticated as extended inspection system by the General Authentication Procedure.

2.1.3 Data Groups and Access Conditions

2.1.3.1 Data Structures

See ICAO Doc 9303 [2] and Part 1 of this Technical Guideline.

2.1.3.2 Authorization

The following Object Identifier SHALL be used for inspection systems:

```
id-IS OBJECT IDENTIFIER ::= {id-roles 1}
```

The *relative authorization* of the certificate holder is encoded in one byte which is to be interpreted as binary bit map as shown in Table 2. In more detail, this bit map contains a role and access rights. Both are relative to the authorization of all previous certificates in the chain.

7	6	5	4	3	2	1	0	**Description**
x	x	-	-	-	-	-	-	**Role**
1	1	-	-	-	-	-	-	CVCA
1	0	-	-	-	-	-	-	DV (official domestic)
0	1	-	-	-	-	-	-	DV (official foreign)
0	0	-	-	-	-	-	-	Inspection System
-	-	x	x	x	x	x	x	**Access Rights**
-	-	x	x	x	x	-	-	RFU
-	-	-	-	-	-	1	-	Read access to ePassport application: DG 4 (Iris)
-	-	-	-	-	-	-	1	Read access to ePassport application: DG 3 (Fingerprint)

Table 2: Authorization of Inspection Systems

An authenticated inspection system

- SHALL always have read access to less-sensitive data groups (e.g. DG1, DG2, DG14) of the ePassport application
- SHOULD also have read access to all data groups of the eID application.

If ICAO compliance is not required, the eIDAS token SHOULD restrict access even to less-sensitive data to terminals authenticated as extended inspection systems via Extended Access Control.

2.2 eID Application

This section specifies the eID application, which is an application for electronic identification. The eID application contains data groups with personal data of the eIDAS token holder. Furthermore, the eID application MAY contain an Attribute capability comprising Specific or Generic Attributes.

2.2.1 Application Identifier

The eID application SHALL be identified by the standard application identifier 0xE80704007F00070302 that is based on the following object identifier:

```
id-eID OBJECT IDENTIFIER ::= {  bsi-de applications(3) 2  }
```

2.2.2 Authentication

The eID application requires the terminal to be authenticated as follows:

- To write to the eID application, the eIDAS token SHALL require the terminal to be authenticated as Authentication Terminal with authorization to write to the respective data groups of the eID application.
- To read from data groups of the eID application, the eIDAS token SHALL require the terminal to be authenticated as

- Authentication Terminal with authorization to read from the respective data groups of the eID application or as
- Inspection System, which implies authorization to read all data groups of the eID application.

- To read or write Specific Attributes or Attribute Requests to the eID application, the eIDAS token SHALL require the terminal to be authenticated as
  - Attribute Terminal with the corresponding authorization.

To authenticate a terminal as Authentication Terminal, Attribute Terminal or Inspection System, the General Authentication Procedure (cf. Part 2) MUST be used.

2.2.3 Data Groups and Generic Attributes

The eID application consists of 22 data groups (DG1 - DG22) containing personal data. The presence of each data group is OPTIONAL at the discretion of the issuer of the eIDAS token. An overview on the data groups is given in Table 3.

Additionally, the eID application MAY contain the capability to store additional Generic Attributes. Within the eID application, Global Generic Attributes and Local Generic Attributes are distinguished.

Global Generic Attributes are registered by a central registration authority that is responsible for the coordination and decision of the allocation and specification of Global Generic Attributes and for the publication of this information.

If Global Generic Attributes are supported, they MUST be stored in data groups with file IDs (short file IDs) in the range of 0x0117 (0x17) to 0x011E(0x1E) and 0x011E to 0x01FF, starting from 0x0117 (0x17) with the first Generic Attribute allocated by the registration authority.

Local Generic Attributes do not require interoperability among different eIDAS Token issuers. These Attributes SHALL be stored in data groups with file IDs in the range of 0x0301 to 0x03FF. Administration of local Generic Attributes is out of scope of this Technical Guideline.

DG	Content	FID	SFID	ASN.1 type	R/W	Access
DG1	Document Type	0x0101	0x01	`DocumentType`	R	PACE + TA + CA
DG2	Issuing State, Region and Municipality	0x0102	0x02	`IssuingEntity`	R	PACE + TA + CA
DG3	Date of Expiry	0x0103	0x03	`Date`	R	PACE + TA + CA
DG4	Given Names	0x0104	0x04	`GivenNames`	R	PACE + TA + CA
DG5	Family Names	0x0105	0x05	`FamilyNames`	R	PACE + TA + CA
DG6	Nom de Plume	0x0106	0x06	`NomDePlume`	R	PACE + TA + CA
DG7	Academic Title	0x0107	0x07	`AcademicTitle`	R	PACE + TA + CA
DG8	Date of Birth	0x0108	0x08	`DateOfBirth`	R	PACE + TA + CA
DG9	Place of Birth	0x0109	0x09	`PlaceOfBirth`	R	PACE + TA + CA
DG10	Nationality	0x010A	0x0A	`Nationality`	R	PACE + TA + CA
DG11	Sex	0x010B	0x0B	`Sex`	R	PACE + TA + CA
DG12	Optional Data	0x010C	0x0C	`OptionalDataR`	R	PACE + TA + CA
DG13	Birth Name	0x010D	0x0D	`BirthName`	R	PACE + TA + CA
DG14	Written Signature	0x010E	0x0E	`WrittenSignature`	R	PACE + TA + CA
DG15	Date of Issuance	0x010F	0x0F	`Date`	R	PACE + TA + CA
DG16	--	0x0110	0x10	`RFU`	R	PACE + TA + CA
DG17	Normal Place of Residence (multiple)	0x0111	0x11	`PlaceOfResidence`	R/W	PACE + TA + CA
DG18	Municipality ID	0x0112	0x12	`MunicipalityID`	R/W	PACE + TA + CA
DG19	Residence Permit I	0x0113	0x13	`ResidencePermitI`	R/W	PACE + TA + CA
DG20	Residence Permit II	0x0114	0x14	`ResidencePermitII`	R/W	PACE + TA + CA
DG21	Phone Number	0x0115	0x15	`PhoneNumber`	R/W	PACE + TA + CA
DG22	Email Address	0x0116	0x16	`EMailAddresse`	R/W	PACE + TA + CA

Table 3: Data Groups of the eID Application

2.2.3.1 Data Structures

Each elementary file contains an ASN.1-structure as defined below. The data is encoded according to the Distinguished Encoding Rules (DER) as specified in [3].

```
DocumentType     ::= [APPLICATION 1]  ICAOString (SIZE (2))
IssuingEntity    ::= [APPLICATION 2]  CHOICE {
                                        issuingState      ICAOCountry
                                        issuingPlace      Place
                                    }
DateOfExpiry     ::= [APPLICATION 3]  Date
GivenNames       ::= [APPLICATION 4]  UTF8String
FamilyNames      ::= [APPLICATION 5]  UTF8String
NomDePlume       ::= [APPLICATION 6]  UTF8String
```

```
AcademicTitle    ::= [APPLICATION 7]  UTF8String
DateOfBirth      ::= [APPLICATION 8]  Date
PlaceOfBirth     ::= [APPLICATION 9]  GeneralPlace
Nationality      ::= [APPLICATION 10] ICAOCountry
Sex              ::= [APPLICATION 11] ICAOSex
OptionalDataR    ::= [APPLICATION 12] SET OF OptionalData
BirthName        ::= [APPLICATION 13] UTF8String
WrittenSignature ::= [APPLICATION 14] OCTET STRING   -- JPEG-2000 [4]
DateOfIssuance   ::= {APPLICATION 15] Date
PlaceOfResidence ::= [APPLICATION 17] CHOICE {
                                       residence         GeneralPlace
                                       multResidence     SET OF GeneralPlace
                                    }
MunicipalityID   ::= [APPLICATION 18] OCTET STRING
ResidencePermitI ::= [APPLICATION 19] Text
ResidencePermitII::= [APPLICATION 20] Text
PhoneNumber      ::= [APPLICATION 21] PrintableString
                  -- telephone URI according to [5] containing a global-number
EmailAddress     ::= [APPLICATION 22] IA5String
ICAOString ::= PrintableString (FROM ("A".. "Z" | " "))

ICAOCountry ::= ICAOString (SIZE (1|3))  -- ICAO country code

ICAOSex ::= PrintableString (FROM ("M"|"F"|" "))

Date ::= NumericString (SIZE (8)) -- YYYYMMDD

Place ::= SEQUENCE {
  street     [10] UTF8String OPTIONAL,
  city       [11] UTF8String,
  state      [12] UTF8String OPTIONAL, -- can also be used to denote region
  country    [13] ICAOCountry,
  zipcode    [14] PrintableString OPTIONAL
}

GeneralPlace ::= CHOICE {
  structuredPlace   Place
  freetextPlace [1] UTF8String
  noPlaceInfo   [2] UTF8String
}

Text ::= CHOICE {
  uncompressed [1] UTF8String
  compressed   [2] OCTET STRING
      -- contains a DEFLATE-compressed UTF8String (cf. [6] for details on
      -- the compression algorithm)
}

OptionalData ::= SEQUENCE {
  type OBJECT IDENTIFIER,
  data ANY DEFINED BY type OPTIONAL
}
```

2.2.3.2 Authorization

The following Object Identifier SHALL be used for Authentication Terminals:

```
id-AT OBJECT IDENTIFIER ::= {id-roles 2}
```

The relative authorization of the certificate holder is encoded in five bytes which are to be interpreted as binary bit map as shown in Table 4. In more detail, this bit map contains a role and access rights. Both are relative to the authorization of all previous certificates in the chain.

39	38	37	…	32	31	30	29	…	8	7	6	5	4	3	2	1	0	Description
x	x	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Role
1	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	CVCA
1	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DV (official domestic)
0	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DV (non-official / foreign)
0	0	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	Authentication Terminal
-	-	x	x	x	-	-	-	-	-	-	-	-	-	-	-	-	-	Write Access (eID)
-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	DG 17
-	-	-	…	-	-	-	-	-	-	-	-	-	-	-	-	-	-	...
-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	-	-	DG 22
-	-	-	-	-	x	-	-	-	-	-	-	-	-	-	-	-	-	RFU
-	-	-	-	-	-	-	x	x	x	-	-	-	-	-	-	-	-	Read Access (eID)
-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	DG 22
-	-	-	-	-	-	-	-	…	-	-	-	-	-	-	-	-	-	...
-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	DG 1
-	-	-	-	-	-	-	-	-	-	x	x	x	x	x	x	x	x	Special Functions
-	-	-	-	-	-	1	-	-	-	-	-	-	-	-	-	-	-	PSA
-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	-	Install Qualified Certificate
-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	-	Install Certificate
-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	-	PIN Management
-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	-	CAN allowed
-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	-	Privileged Terminal
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	-	Restricted Identification
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	-	Municipality ID Verification
-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	-	1	Age Verification

Table 4: Authorization of Authentication Terminals

The following Authorization Extensions MAY additionally be used to encode authorizations for features of the eID application.

To denote the Authorization Extension for eID access, the following Object Identifier SHALL be used:

```
id-eIDAccess OBJECT IDENTIFIER ::= {id-AT 1}
```

The relative authorization of the certificate holder is encoded in an evolving binary bit map as defined in Table 5[1].

Evolution of this extension, i.e. adding authorization bits for new data groups, will follow the following encoding rules (Include to PSC DGx; Write/Erase DGx; Compare DGx; Read DGx).

47	46	45	44	43	...	22	21	...	0	Description
1	-	-	-	-	-	-	-	-	-	RFU (Include DG23 to PSC)
-	1	-		-	-	-	-	-	-	RFU (Write/Erase DG23)
-	-	1	-	-	-	-	-	-	-	RFU (Compare DG23)
-	-	-	1	-	-	-	-	-	-	RFU (Read DG23)
-	-	-	-	1	-	-	-	-	-	Include DG22 to PSC
-	-	-	-	-	...	-	-	-	-	...
-	-	-	-	-	-	1	-	-	-	Include DG1 to PSC
-	-	-	-	-	-	-	1	-	-	Compare DG22
-	-	-	-	-	-	-	-	...	-	...
-	-	-	-	-	-	-	-	-	1	Compare DG1

Table 5: Authorization for eIDAccess

Authorization Extensions to be used for local Generic Attributes are out of scope of this Technical Guideline.

To denote the Authorization Extension for Access to special functions the following Object Identifier SHALL be used:

```
id-specialFunctions OBJECT IDENTIFIER ::= {id-AT 2}
```

The relative authorization of the certificate holder is encoded in an evolving binary bit map as defined in Table 6[2].

7	6	5	4	3	2	1	0	Description
x	x	x	x	x	x	x	x	**Access Rights (Special Functions)**
1	-	-	-	-	-	-	-	PSC
-	1	-	-	-	-	-	-	PSM
-	-	x	x	x	x	x	x	RFU

Table 6: Authorization for Special Functions

An authenticated Authentication Terminal SHALL always have access to the following functions:

- Restricted Identification unless the eIDAS token requires the terminal to be authorized to use the function (`authorizedOnly` is set, cf. Part 3, Appendix A)

1 The bit *Include DGx to PSC* defines whether the Authentication Terminal is authorized to include the corresponding data group into the computation of the Pseudonymous Signature of Credentials or not.

2 The bits *Restricted Identification/PSA/PSM/PSC* define whether the Authentication Terminal is authorized to get access to a pseudonym for the corresponding protocol that requires explicit authorization.

- Document Validity Verification.

If Chip Authentication Version 3 is supported by the eIDAS Token, the following setting for `ps1-authInfo` and `ps2-authInfo` is RECOMMENDED in PSAInfo :

- `ps1-authInfo=0`
- `ps2-authInfo≠0`

If PSM is supported by the eIDAS Token, the following setting for `ps1-authInfo` and `ps2-authInfo` is RECOMMENDED in PSMInfo

- `ps1-authInfo=2`
- `ps2-authInfo≠2`

If PSC is supported by the eIDAS Token, the following setting for `ps1-authInfo` and `ps2-authInfo` is RECOMMENDED in PSCInfo

- `ps1-authInfo=2`
- `ps2-authInfo≠2`

2.2.4 Attribute Requests and Specific Attributes

The eID application MAY support Attribute Requests and Specific Attributes stored in data containers as specified in Part 3 of this Technical Guideline.

2.2.4.1 Authorization

The Object Identifier `id-ERAspecific` SHALL be used for Attribute Terminals:

```
id-CVCExtension OBJECT IDENTIFIER ::= {
iso(1) member-body(2) fr(250) type-org(1) anssi(223) eIDAStoken(1001) 1
}

id-ERAspecific OBJECT IDENTIFIER ::= { id-CVCExtension 3}
```

The relative authorization of the certificate holder is encoded in an evolving binary bit map as shown in Table 7[3].

If the access right *Access to all Terminal Sectors* is not set, the eIDAS token SHALL restrict read access and deletion of Attributes tied to the Terminals Sector included in the certificate of the Attribute Terminal. If the access right is set, the eIDAS token SHALL grant read access and deletion to all Attributes.

3 The bit *Include Specific Attributes to PSC* defines whether the Authentication Terminal is authorized to include Specific Attributes into the computation of the Pseudonymous Signature of Credentials or not.

7	6	5	4	3	2	1	0	Description
x	x	x	x	x	x	x	x	Access Rights (Attributes)
x	-	-	-	-	-	-	-	RFU
-	1	-	-	-	-	-	-	Include Specific Attributes to PSC
-	-	1	-	-	-	-	-	Access to all Terminal Sectors
-	-	-	1	-	-	-	-	Delete Specific Attributes
-	-	-	-	1	-	-	-	Write Specific Attribute
-	-	-	-	-	1	-	-	Read Specific Attribute
-	-	-	-	-	-	1	-	Write Attribute Request
-	-	-	-	-	-	-	1	Read Attribute Request

Table 7: Authorization for Specific Attributes

Attribute Providers SHALL have access right Read Attribute Requests. Attribute Providers with authorization to provide Specific Attributes MUST have access right Write Specific Attribute.

2.2.5 EID Application Info

The `EIDapplicationInfo` is a SecurityInfo (cf. Part 3 of this Technical Guideline) and MAY be used within CardSecurity and ChipSecurity to provide information on the functionality supported by the eID application. It SHOULD be present if the eIDAS token supports Generic or Specific Attributes or the compare command for Auxiliary Data Verification.

EIDApplicationInfo: This data structure provides information on the functionality supported by the eID application

- The sequence `eIDApplRequiredData` SHALL contain information about the eID application.
  - `supportedGlobalDGs` SHALL indicate the global generic data groups (including DG1-DG22 and Generic Attributes) that are supported by the eID application. If no global data groups are supported, this SHALL be indicated by an empty structure.
  - `supportedLocalDGs` SHALL contain local data groups (Local Generic Attributes) that are supported by the eID application. If no local data groups are supported, this SHALL be indicated by an empty structure.
  - The field `attrRequests` SHALL indicate whether or not storage of Attribute Requests is supported by the eID application.
  - The field `specAttributeCap` SHALL indicate whether or not Specific Attributes are supported by the eID application.
  - The boolean `compareInfo` MUST be used to indicate whether or not the ICC supports the compare command for all Auxiliary Data Verification information.
  - The set of object identifiers `supportedAuxDataVerification` MUST be used to indicate the supported Auxiliary Data Verification functionality. An empty structure indicates that Auxiliary Data Verification is not supported by the eIDAS token.

```
id-eIDApplicationInfo ::= {
```

```
      bsi-de protocols(2) smartcards(2) 13
}

EIDApplicationInfo ::= SEQUENCE {
  protocol              OBJECT IDENTIFIER(id-eIDApplicationInfo),
  eIDApplRequiredData   EIDApplRequiredData
}

EIDApplRequiredData ::= SEQUENCE {
  supportedGlobalDGs                SET OF INTEGER,
  supportedLocalDGs                 SET OF INTEGER,
  attrRequestsCap                   BOOLEAN,
  specAttributeCap                  BOOLEAN,
  compareInfo                       BOOLEAN,
  supportedAuxDataVerification      SET Of OBJECT IDENTIFIER,
}
```

2.2.6 Certificate Description Extension

The certificate description is used by the user device as part of the user interaction for online authentication of an eID server (cf. Part 2) and may be ignored by the eIDAS token.

The following object identifier SHALL be used for this extension:

```
id-description OBJECT IDENTIFIER ::= {id-extensions 1}
```

The following context specific data object is used to encode the certificate description:

- 0x80: Hash of `CertificateDescription`

```
CertificateDescription ::= SEQUENCE {
  descriptionType  OBJECT IDENTIFIER,
  issuerName       [1] UTF8String,
  issuerURL        [2] PrintableString OPTIONAL,
  subjectName      [3] UTF8String,
  subjectURL       [4] PrintableString OPTIONAL,
  termsOfUsage     [5] ANY DEFINED BY descriptionType,
  redirectURL      [6] PrintableString OPTIONAL,
  commCertificates [7] SET OF OCTET STRING OPTIONAL
}
```

The set `commCertificates` MAY contain hash values of admissible X.509 certificates of the remote terminal. The hash function to be used SHALL be defined by the hash function used to sign the CV-certificate. The input for the hash function is the respective DER-encoded X.509 certificate including tag and length. The hash function to be used to generate the content of the extension SHALL be defined by the hash function used to sign the certificate.

2.2.6.1 Plain Text Format

The following object identifier SHALL be used to identify terms of usage in plain text format:

```
id-plainFormat OBJECT IDENTIFIER ::= {id-description 1}
PlainTermsOfUsage ::= UTF8String
```

2.2.6.2 HTML Format

The following object identifier SHALL be used to identify terms of usage in HTML format:

```
id-htmlFormat OBJECT IDENTIFIER ::= {id-description 2}
```

```
HtmlTermsOfUsage ::= IA5String
```

2.2.6.3 PDF Format

The following object identifier SHALL be used to identify terms of usage in PDF format [7]:

```
id-pdfFormat OBJECT IDENTIFIER ::= {id-description 3}
PdfTermsOfUsage ::= OCTET STRING
```

2.2.7 Protocol support

If the eID application is supported by the eIDAS token, the following rules apply related to the availability of protocols:

- If supported by the eIDAS token, Restricted Identification, PSM and PSC MUST be available for Authentication Terminals within the eID application. Additionally, the protocols MAY be available on Master File level.
- If Chip Authentication Version 3 is supported by the eIDAS token, PSA MAY be available in the eID application.

2.3 eSign Application

The eSign application supported by this Technical Guideline is specified in the Technical Report [8]. The Technical Report contains two flavors of the eSign Application that differ in the complexity and functionality and may be accessed by different terminal types, i.e. Signature Terminal (Simple eSign Application) or. Signature Management Terminal (eSign Application with supplementary features), and the corresponding authentication procedures and access conditions.

3 Document Profiles

This section defines Document Profiles based on the Application Profiles from the preceding section.

3.1 European Passport

3.1.1 Passwords

This Document Profile requires support for the following Passwords:

- **MRZ (see Part 1)** **(REQUIRED)**
- **CAN (see Part 1)** **(OPTIONAL)**

3.1.2 Authentication Procedure

This Document Profile requires implementation of the following Authentication Procedure for access to DG 3 and DG 4 of the ePassport Application:

- **Advanced Inspection Procedure (see Part 1)** **(REQUIRED)**

3.1.3 Applications

This Document Profile requires implementation of the following Applications:

- **ePassport Application (see Section 2.1)** **(REQUIRED)**

3.1.4 Protocols

This Document Profile requires implementation of the following Protocols:

- **PACE (see Part 1)** **(REQUIRED)**
- **Terminal Authentication Version 1 (see Part 1)** **(REQUIRED)**
- **Chip Authentication Version 1 (see Part 1)** **(REQUIRED)**

3.2 Identity Card with Protected MRTD Application

3.2.1 Passwords

This Document Profile requires support for the following Passwords:

- **MRZ (see Part 1)** **(REQUIRED)**
- **CAN (see Part 1)** **(REQUIRED)**
- **PIN (see Part 2)** **(REQUIRED)**
- **PUK (see Part 2)** **(REQUIRED)**

The PIN-Management operation Change PIN MUST be supported for unauthenticated terminals for PACE with PIN (but not with PUK) and for Authentication Terminals with effective Authorization for PIN Management. The PUK MUST be non-blocking, i.e. Resume PUK MUST NOT be supported.

3.2.2 Authentication Procedure

This Document Profile requires implementation of the following Authentication Procedure

- **General Authentication Procedure (see Part 2)** **(REQUIRED)**

3.2.3 Applications

This Document Profile requires implementation of the following Applications:

- **ePassport Application (see Section 2.1)** **(REQUIRED)**

  Access to the ePassport application MUST be restricted to extended inspection systems authenticated by the General Authentication Procedure.
- **eID Application (see Section 2.2)** **(REQUIRED)**

  As part of the eID Application, the eIDAS token MUST support Auxiliary Data Verification for Data Groups 3, 8 and 18.
- **eSign Application (see Section 2.3)** **(REQUIRED)**

  The eSign Application MUST comply to the eSign Application according to [8].

3.2.4 Protocols

This Document Profile requires implementation of the following Protocols:

- **PACE (see Part 2)** **(REQUIRED)**
- **Terminal Authentication Version 2 (see Part 2)** **(REQUIRED)**
- **Chip Authentication Version 2 (see Part 2)** **(REQUIRED)**
- **Restricted Identification (see Part 2)** **(REQUIRED)**

3.3 Identity Card with optional EU-compliant MRTD Application

3.3.1 Passwords

This Document Profile requires support for the following Passwords:

- **MRZ (see Part 1)** **(CONDITIONAL)**

  MUST be supported if an EU-compliant MRTD Application is present.
- **CAN (see Part 1)** **(REQUIRED)**
- **PIN (see Part 2)** **(REQUIRED)**
- **PUK (see Part 2)** **(REQUIRED)**

Password Verification SHALL be based on PACE. The PIN-Management operation Change PIN MUST be supported.

3.3.2 Authentication Procedure

This Document Profile requires implementation of the following Authentication Procedure

- **General Authentication Procedure (see Part 2)** **(REQUIRED)**
- **Advanced Inspection Procedure (see Part 1)** **(CONDITIONAL)**

  MUST be supported if an EU-compliant MRTD Application is present.

3.3.3 Applications

This Document Profile requires implementation of the following Applications:

- **ePassport Application** **(OPTIONAL)**

  Access to DG 3 and DG 4 MUST be restricted to extended inspection systems with particular authorization.
- **eID Application (see Section 2.2)** **(REQUIRED)**

  As part of the eID Application, the eIDAS token MUST support Auxiliary Data Verification for Data Groups 3, 8 and 18.
- **eSign Application (see Section 2.3)** **(REQUIRED)**

  The eSign Application MUST comply to the eSign Application according to [8].

3.3.4 Protocols

This Document Profile requires implementation of the following Protocols:

- **PACE (see Part 2)** **(REQUIRED)**
- **Terminal Authentication Version 2 (see Part 2)** **(REQUIRED)**
- **Chip Authentication Version 2 (see Part 2)** **(REQUIRED)**
- **Restricted Identification (see Part 2)** **(REQUIRED)**
- **Terminal Authentication Version 1 (see Part 1)** **(CONDITIONAL)**

  MUST be supported if an EU-compliant MRTD Application is present.
- **Chip Authentication Version 1 (see Part 1)** **(CONDITIONAL)**

  MUST be supported if an EU-compliant MRTD Application is present.

Bibliography

[1] Bradner, Scott. Key words for use in RFCs to indicate requirement levels, RFC 2119, 1997

[2] ICAO. Machine Readable Travel Documents, ICAO Doc 9303, 2015

[3] ITU-T. Information Technology – ASN.1 encoding rules: Specification of Basic Encoding Rules(BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER), X.690, 2002

[4] ISO/IEC 15444-1. Information technology - JPEG 2000 image coding system: Core coding system, 2009

[5] H. Schulzrinne. The tel URI for Telephone Numbers, RFC 3966, 2004

[6] Deutsch, Peter. DEFLATE compressed data format specification version 1.3., RFC 1951, 1996

[7] ISO 32000-1:2008. Document management – Portable document format – Part 1: PDF 1.7, 2008

[8] Technical Report - Signature creation and administration for eIDAS Token, Version 1.0. TR-Sign,