Belgian Electronic Identity Card content

Document Change History

Version	Date	Description
1.0		Internal version
1.1	14-01-2003	Version for distribution
1.2	29-01-2003	Changes in PKCS#15 files
1.3	10-02-2003	Changes in PKCS#15 files
1.4	21-02-2003	Changes in PKCS#15 files Specified the address file format
1.5	27-02-2003	Adapted the address file max. length
1.6	18-03-2003	Added the ***Preference*** file format Added ***W*** in the "sex" field from the ID file Changes in PKCS#15 files
1.7	24-03-2003	Corrected national number length from the ID file
1.8	04-04-2003	Changed birth date type to UTF-8 the ID file and added the dot (.) as separator for German
1.9	09-04-2003	Added birth months table Corrected the ***PKCS#15 Authority*** flag for the certificates
2.0	25-04-2003	Changed ***EF(ID#Address)*** length
2.1	26-05-2003	Added "document type" field from the ID file
2.3	03-09-2003	Corrected card number length in ID file Corrected address file length
2.4	26-11-2003	Minor corrections
2.5	19-12-2003	Added ***TokenInfo*** version detail Added optional version in files Added envisioned max. length for fields
2.6	24-12-2003	Typo correction in AID Added default values for version
2.7	20-01-2004	Generalised to include Applet V2 specs Added picture resolution
2.8a	16-03-2004	Detailed some explanations Removed unused files/keys
3.00	07-05-2013	Various updates. Added information about Kids cards and resident cards for EU and non-EU citizens. Introduction of 0 and 1 user certificates.
3.01	03-07-2013	Correction: for the address signature calculation, the padding bytes in the address file must be removed first.
3.02	24/07/2014	Correction: some typo's
4.00	10/09/2014	Remove info about the content of the personal information on the chip (ID file, picture, address etc.)
4.01	18/09/2014	Add specification of ID file in the chip

Table of content

1. Scope

This document describes the specifications of the card files and some objects of the **Belgian eID card**.

The same Belpic applet that is present on the eID cards is also used for the **Kids cards**, and for the **resident cards for EU and non-EU citizens**. Moreover, the same files are present on those cards; mostly with the same or similar contents. Therefore most of the contents also apply to those cards too, unless stated otherwise.

As a result, several documents exist that describe the specific contents for the various versions of those cards. So this document should be viewed as a basis, describing the contents that are common to all versions and for all cards.

The information in this document comes from different sources. Therefore real cards need to be used for development and testing.

1.1. Terms and definitions

For the purposes of this document, the following definitions apply:

AODF = authentication object directory file	optional elementary file containing information about authentication objects known to the PKCS#15 application
binary coded decimal	Number representation where a number is expressed as a sequence of decimal digits and then each decimal digit is encoded as a four bit binary number. Example – Decimal 92 would be encoded as the eight bit sequence 1001 0010.
cardholder	person for whom the card was issued
card issuer	organization or entity that issues smart cards and card applications
CDF = certificate directory file	optional elementary file containing information about certificate known to the PKCS#15 application
dedicated file	file containing file control information, and, optionally, memory available for allocation, and which may be the parent of elementary files and/or other dedicated files
directory (DIR) file	optional elementary file containing a list of applications supported by the card and optional related data elements
elementary file	set of data units or records that share the same file identifier, and which cannot be a parent of another file
file identifier	2-byte binary value used to address a file on a smart card
master file	mandatory unique dedicated file representing the root of the structure
ODF = object directory file	elementary file containing information about other directory files in the PKCS #15 application

path	concatenation of file identifiers without delimitation NOTE – If the path starts with the MF identifier ($3F00_{16}$), it is an absolute path; otherwise it is a relative path. A relative path shall start with the identifier '$3FFF_{16}$' or with the identifier of the current DF.
personal identification number (PIN)	4 to 8 digit number entered by the cardholder to verify that the cardholder is authorized to use a functionality of the card
PrKDF = private key directory file	optional elementary file containing information about private keys known to the PKCS#15 application
provider	authority who has or who obtained the right to create the MF or a DF in the card
record	string of bytes which can be handled as a whole by the card and referenced by a record number or by a record identifier
token	portable device capable of storing persistent data

1.2. Symbols, abbreviated terms and document conventions

1.2.1. Symbols

DF(x) Dedicated file *x*

EF(x) Elementary file *x*

1.2.2. Abbreviated terms

For the purposes of this document, the following abbreviations apply:

AID	Application Identifier
BCD	Binary-Coded Decimal
DER	Distinguished Encoding Rules
DF	Dedicated File (directory)
EF	Elementary File
hexa	hexadecimal
MF	Master File
PIN	Personal Identification Number

2. Versions

2.1. Overview

There are several types of versions. These can be divided into 3 groups

1. Versions related to the chip, OS and Belpic applet.
2. Versions related to the electrical personalisation: which files are created on the card, their sizes, contents and access conditions.
3. Versions related to the graphical personalisation: to track changes in what is printed on the card, or changes in the physical security measures. These versions are no relevant for this document.

The various version numbers are not described here but can be found in documents such as

- *Description Belpic EID-version numbering vers x y z.doc*
- *Belpic Kids-ID-version numbering document vx.y.doc*
- *Description Belpic eVKeTS for EUcitizen-version numbering vx.y.doc*
- *Description Belpic eVKeTS for non-EU-citizen-version numbering vx y.docx*

2.2. Applet version

Some objects are hardcoded into the applet and are therefore linked to the version of the applet used. These objects are:

- The PINs and PUKs
- The public and private keys
- The MF, DF(BELPIC) and DF(ID) directories

When applicable, we will refer in this document to **"Applet version x"**. This version can be received with the command **"GetCardData"** that can be sent to the card and that returns a.o. the applet version.

2.3. Card contents versions

During personalisation, various files, as detailed below, are created and contents are written to these files. The PIN, public and private key objects, that already exist in the Belpic applet, are then also given their (initial) value or are generated in the case of a private key.

Two perso version numbers are located in the file ***TokenInfo*** (see below).

Electrical personalisation version: this number increases at every change – even minor – in the personalisation format

- ***Electrical personalisation interface version***: this number increases when a change in the personalisation format introduces an incompatibility of the file structure with the old format.

An application can thus use newer cards that have additional files, because the interface version will be the same.

Note that individual files may have an internal version number corresponding to the data in the file. The "***Electrical personalisation interface version***" should be used to check the file structure; the internal file version should be used to check the field format in the file.

3. Security Objects

The JavaCard framework provides a number of Java objects such as PINs and public and private keys. It also provides memory space in the form of a byte array.

Applets such as the Belpic applet can uses these Java objects to implement the desired behaviour of the card. The memory space is used to implement a file system with directories (DFs) and files (EFs).

3.1. PINs and PUKs

Two types of PIN usage exist:

- ***Permanent***: once the PIN has been validated, its granted access right is permanent until current access condition specifically changes (card reset, logoff, external auth…). This is the usual way of using a PIN.
- ***Transient***: the PIN access right granted is available for the next command only.

	PIN reference (Java Object)	**Type (transient/permanent)**	**PUK**	**PIN_{Reset}**	**Max. trials before blocked**
PUK	03	transient	**-**	**-**	12
PIN_{Reset}	02	transient	**-**	**-**	10
Activate	84	transient	**-**	**-**	15
Authentication	01	permanent	03	02	3
Non-repudiation	01	transient	03	02	3

3.2. Keys

3.2.1. Keys and certificates relationships

	Private Key (Java Object)	**Public Key (Java Object)**	**X.509 Certificates (Transparent file)**
Basic	PrK#1		
Authentication	PrK#2	In Cert#2	Cert#2
Non-repudiation	PrK#3	In Cert#3	Cert#3
Certification Authority (CA)		In Cert#4	Cert#4
Root		In Cert #6	Cert#6

CA Role		PuK#7	
RN		In Cert #8	Cert#8

Each key or certificate is indicated by means of a reference number (#). Some keys do not have a corresponding private/public key or certificate.

Remark: Puk#7 is not readable on the card.

3.2.2. Keys Access Control

	Reference (hex)	*Generate Key*	*Get Key*	*Put Key*	*Erase Key*	*Activate Key*	*Deactivate Key*	*PSO: Compute Digital Signature*	*Internal Authenticate*	*External Authenticate*
PrK#1	81	✗	✗	✗	✗	✗	✗	✗	ALW	✗
PrK#2	82	CTV(3)	✗	✗	✗	CTV(4)	CTV(3)	CHV(PIN_{Auth})	✗	✗
PrK#3	83	CTV(3)	✗	✗	✗	CTV(4)	CTV(3)	CHV($PIN_{Non-Rep}$)	✗	✗
PuK#7	87	✗	✗	CTV(8)	CTV(8)	✗	✗	✗	✗	CHV(PIN_{Auth})

✗	Not possible (forbidden by the card Operating System/applet)
NEV	Never
ALW	Always
CHV(x)	Card Holder Verification with PIN '**x**'
CTV(x)	Certificate Verification with Role '**x**'

3.2.3. Certificates and role identifiers

In compliance with ISO/IEC FDIS 7816–9 (sub–clause 7.4) card verifiable certificates will be applied in public key based authentication procedures. Such certificates contain certificate holder authorisations (e.g. role identifiers). This role identifier is used in the security conditions to be fulfilled for access to data or functions.

These certificates will be X.509 compliant, and thus will not use the ISO tags.

A number of role identifiers are hardcoded by the applet, i.e. the roles that the specify access conditions of certain operations on PINs, public and private keys. Other - and the same - roles identifiers can be specified for certain operations on a file, when that file is created during card personalization.

The roles are retrieved from the certificates after an **External Authentication** with certificate verification. A ***Mutual Authentication*** with ***Secure Messaging –*** with or without encryption - is advised to secure the connection. For example, after a successful Certificate Verification with a certificate contains Role 8, it is possible to Put and Erase *PuK#7.*

4. Files

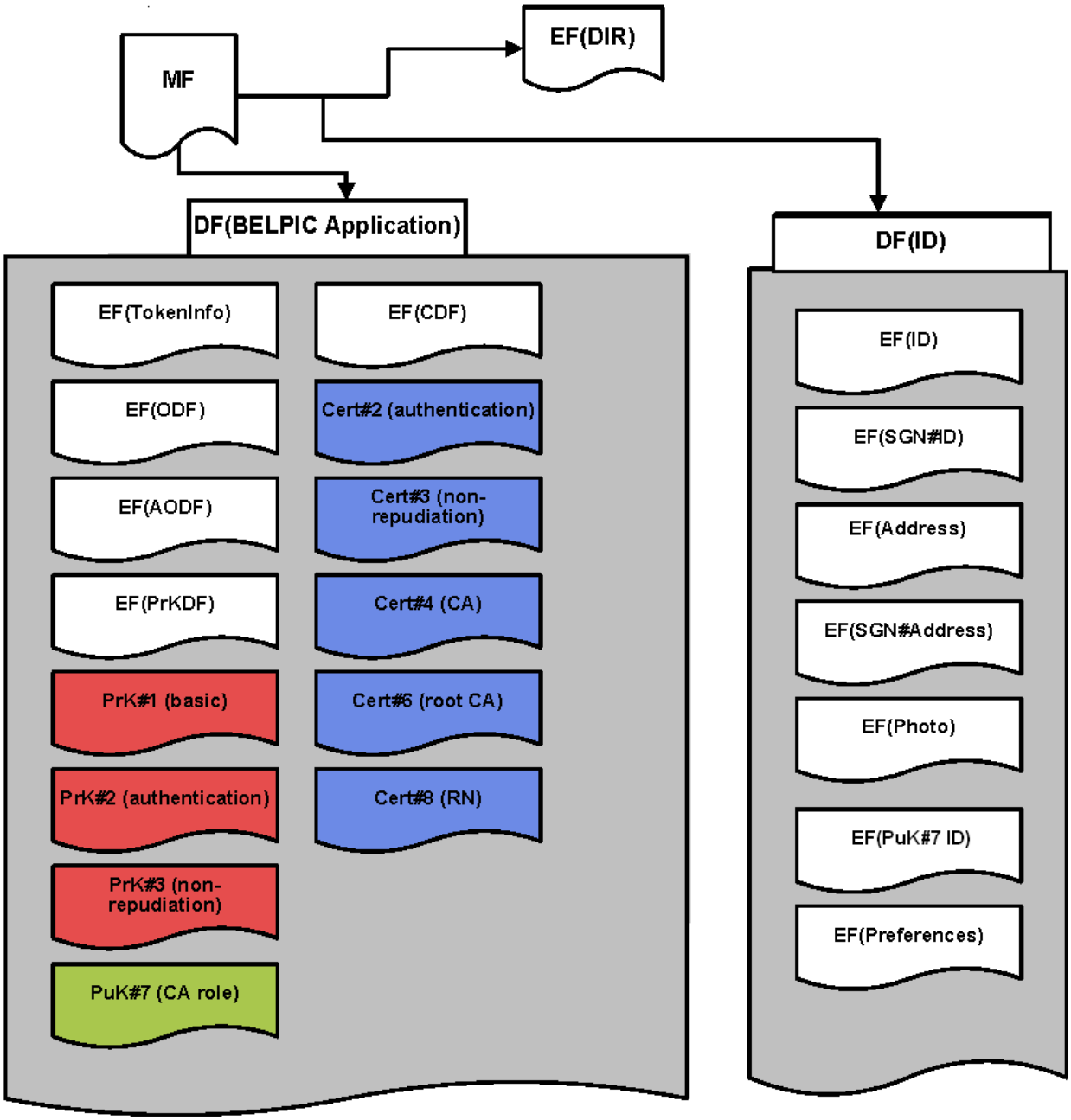
All EF file types are transparent, as defined in ISO/IEC 7816–4, sub–clause 5.1.3.

Files in the EID card is organised into a hierarchical structure according to ISO/IEC 7816–4.

The electronic signature and electronic identification applications are separated in the card by means of two application directories: ***DF(BELPIC)*** and ***DF(ID)***.

4.1. File structure

The file structure of the card is described in the figure below.

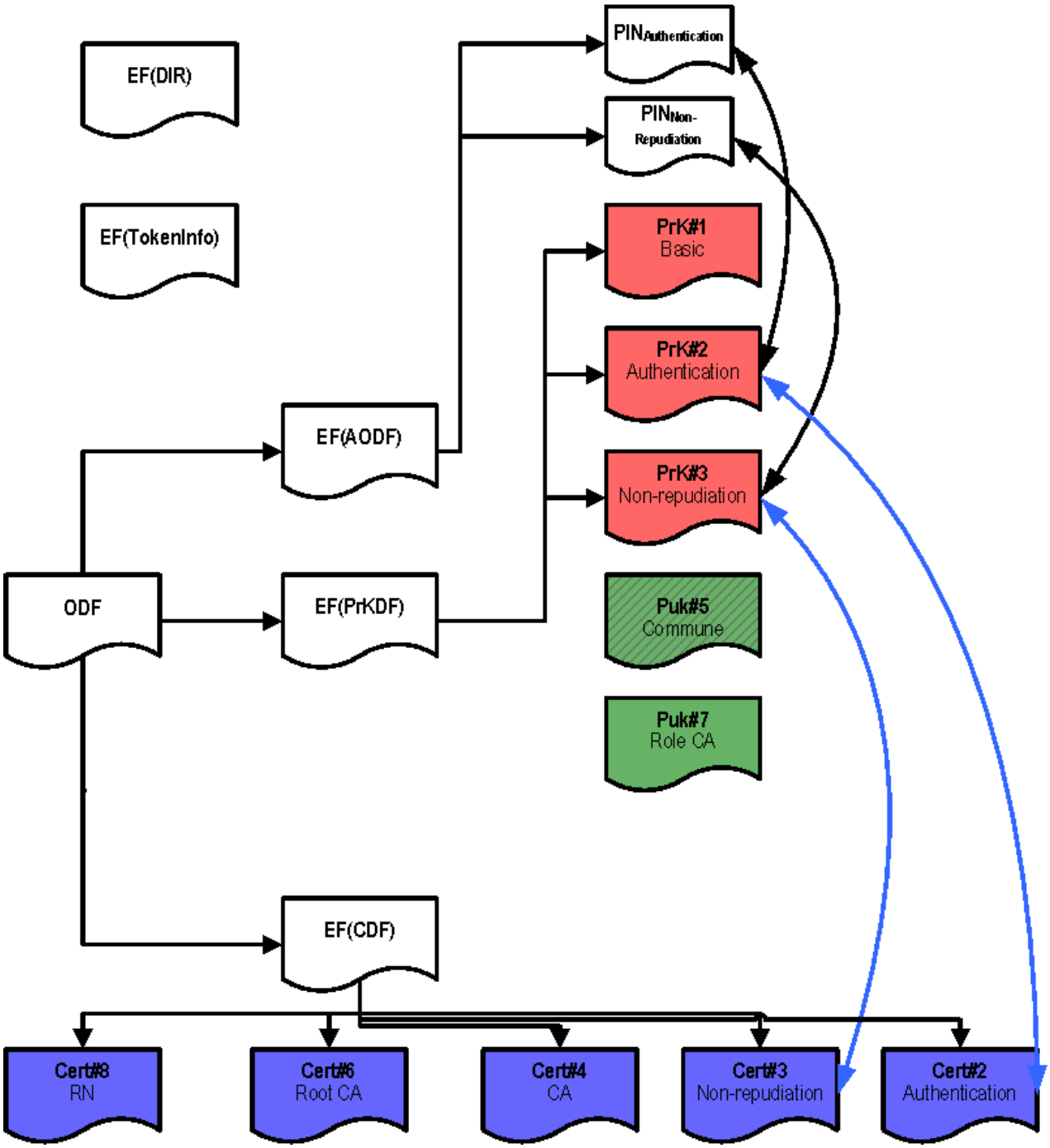


Each directory (MF, DF) and each file (EF) has a 2-byte **file identifier**. For example the file ID of the MF (= the root directory) is *3F00* (hexadecimal notation).

4.2. PKCS#15 files

The content of the ***DF(BELPIC)*** application directory files is compliant with PKCS#15 v1.1.

A directory file, ***EF(DIR)***, containing the AID (ISO/IEC 7816–5) for each application in the EID card is present in the ***Master File***. The PKCS#15 AID, and other AID, are also directly selectable.



The purpose of the figure above is to show the relationship between the PKCS#15 files ***EF(ODF)***, ***EF(AODF)***, ***EF(PrKDF)*** and ***EF(CDF)*** in the ***DF(BELPIC)*** directory. ***EF(ODF)*** points to other the other PKCS#15 files.

EF(PrKDF) contains cross-reference pointers to authentication objects (PIN) used to protect access to the keys. Arrows between PIN and PrK indicate this.

Some certificates (***Cert#2*** & ***Cert#3***) contain a public key whose private key also resides on the card, so these certificates contain the same identifier as the corresponding private key. Arrows between Certificates and Private Keys indicate this.

4.3. Files, identifiers and permissions

Command	**File type Access method Meaning**
Activate	The MF, DF or EF can be activated
Deactivate	The MF, DF or EF can be deactivated
Read Binary	The contents of the EF can be read
Update/Erase Binary	The contents of the EF can be updated and erased

	Reference (hexa)	**Activate**	**Deactivate**	**Read Binary**	**Update/Erase Binary**
MF	3F00	CHV($PIN_{activate}$)	NEV	✗	✗
EF(DIR)	2F00	NEV	NEV	ALW	CTV(1)
- DF(BELPIC)	DF00	NEV	NEV	✗	✗
EF(ODF)	5031	NEV	NEV	ALW	CTV(1)
EF(TokenInfo)	5032	NEV	NEV	ALW	CTV(1)
EF(AODF)	5034	NEV	NEV	ALW	CTV(1)
EF(PrKDF)	5035	NEV	NEV	ALW	CTV(1)
EF(CDF)	5037	NEV	NEV	ALW	CTV(1)
EF(Cert#2) (auth)	5038	CTV(4)	CTV(3)	ALW	CTV(4)
EF(Cert#3) (non-rep)	5039	CTV(4)	CTV(3)	ALW	CTV(4)
EF(Cert#4) (CA)	503A	CTV(4)	CTV(3)	ALW	CTV(4)
EF(Cert#6) (root)	503B	CTV(5)	CTV(5)	ALW	CTV(5)
EF(Cert#8) (RN)	503C	NEV	NEV	ALW	CTV(6)
- DF(ID)	DF01	NEV	NEV	✗	✗
EF(ID#RN)	4031	NEV	NEV	ALW	NEV
EF(SGN#RN)	4032	NEV	NEV	ALW	CTV(6)
EF(ID#Address)	4033	NEV	NEV	ALW	CTV(7)

	Reference (hexa)	Activate	Deactivate	Read Binary	Update/Erase Binary
EF(SGN#Adress)	4034	NEV	NEV	ALW	CTV(7)
EF(ID#Photo)	4035	NEV	NEV	ALW	NEV
EF(PuK#7 ID)	4038	NEV	NEV	ALW	CTV(8)
EF(Preferences)	4039	NEV	NEV	ALW	CHV(PIN_{Auth})

✗	Not possible (forbidden by the card Operating System/applet)
NEV	Never
ALW	Always
CTV(x)	Certificate Verification with Role '**x**'

4.4. Application selection

As an alternative to selecting a DF ('Application') by its 2 byte file identifier, the EID card support direct application selection as defined in ISO/IEC 7816–4, Section 9 and ISO/IEC 7816–5, Section 6 (the full AID is to be used as parameter for a 'SELECT FILE' command).

The operating system of the card keeps track of the currently selected application (=DF) and only allow the commands applicable to that particular application while it is selected.

The following AIDs are defined by the applet:

	AID (Application Identifier)
Belpic applet	A0 00 00 00 30 29 05 70 00 AD 13 10 01 01 FF
DF(BELPIC)	A0 00 00 01 77 50 4B 43 53 2D 31 35
DF(ID) *(not on applet V1.0)*	A0 00 00 01 77 49 64 46 69 6C 65 73

The Belpic applet is selected by default (= after a power on or reset of the card). After the Belpic applet is selected, the MF is the current directory.

4.5. Note for non-eID cards

The name Belpic does not only refer to the Belgian eID cards. All DFs, files, PINs and keys are also present on **Kids cards** and the **resident cards for EU and non-EU citizens**.

In some cases however, the contents of certain files may differ.
For example, the **EF(ID#RN)** file contains extra fields on resident cards for EU and non-EU citizens.
Or for Kids cards, the **EF(Cert#2) (auth)** file may be empty, depending on the age of the child.

5. MF directory contents

5.1. EF(DIR)

This file contains all application templates as defined in ISO/IEC 7816–5. Each application template (tag '61'H) for a PKCS#15 application must at least contain the following Data Objects:

- ***Application Identifier***: tag '4F', UTF-8 encoded
- ***Path***: tag '51', DER-encoded

Other tags from ISO/IEC 7816–5 may, at the application issuer's discretion, be present as well. In particular, it is recommended that application issuers include the following Data Objects:

- ***Application Label***: tag '50', UTF-8 encoded
- ***Discretionary Data Objects***: tag '51', DER-encoded

Hexadecimal dump	**Meaning (ASN.1)**
61 23	-- [APPLICATION 1] IMPLICIT SEQUENCE
4F 0C A0 00 00 01 77 50 4B 43 53 2D 31 35	Application ID: -- [APPLICATION 15] IMPLICIT OCTET STRING -- AID of the EF(BELPIC)
50 06 42 45 4C 50 49 43	Label: -- [APPLICATION 16] IMPLICIT UTF8 String -- 'BELPIC'
51 04 3F 00 DF 00	-- [APPLICATION 17] IMPLICIT OCTET STRING -- MF ID / DF(BELPIC) ID
73 05	Discretionary Data Object: -- [APPLICATION 19] IMPLICIT SEQUENCE
06 03 60 38 02	ObjectID: -- OBJECT IDENTIFIER -- Belgian citizen (2.16.56.2)

Remark: the Object Identifier ***2.16.56.2*** was originally intended to signify "Belgian citizen". However, since this Object Identifier is also used for **Kids cards** and resident **cards for EU and non-EU citizens**, it should now be interpreted more generally as "Belpic card".

6. DF(BELPIC) Application directory contents

This DF is the directory of the BelPIC application.

The meaning of the fields (such as TokenFlags, PinFlags, KeyUsageFlags, ..) can be found in the PKCS#15 standard.

6.1. EF(TokenInfo)

This file contains generic information about the token as such and its capabilities. This information includes the token serial number, file types for object directory files, algorithms implemented on the token, etc.

Hexadecimal dump	**Meaning (ASN.1)**
30 27	-- SEQUENCE
02 01 00	Version: -- INTEGER -- 0
04 10 {16 bytes}	Serial Number: -- OCTET STRING -- chip serial number
80 06 42 45 4C 50 49 43	Application Label: -- [0] Label IMPLICIT UTF8 String -- "BELPIC"
03 02 04 30	TokenFlags: -- BIT STRING -- prnGeneration(2), eidCompliant (3)
9E 04 {4 bytes}	-- [30] BELPIC Application IMPLICIT INTEGER -- Version bytes

Version bytes:

- Graphical personalisation version (default = 0)
- Electrical personalisation version (default = 0)
- Electrical personalisation interface version[1] (default = 0)
- Reserved for future use (0)

6.2. EF(ODF)

The Object Directory File (***ODF***) is a transparent elementary file, which contains pointers to other elementary files (***PrKDF***, ***CDF***, ***AODF***) of the EID card. The information is presented in ASN.1 syntax according to PKCS#15.

An application using the EID card must use this file to determine how to perform security services with the card.

Hexadecimal dump	**Meaning (ASN.1)**

[1] This is used to indicate to an application which file system organisation is used. This value only changes when a new version is no more compatible with the previous one.

A0 0A	-- [0] Private Keys
30 08	Path: -- SEQUENCE
04 06 3F 00 DF 00 50 35	Path: -- OCTET STRING -- MF/Belpic/PrKDF
A4 0A	-- [4] Certificates
30 08	Path: -- SEQUENCE
04 06 3F 00 DF 00 50 37	Path: -- OCTET STRING -- MF/Belpic/CDF
A8 0A	-- [8] Authentication Objects
30 08	Path: -- SEQUENCE
04 06 3F 00 DF 00 50 34	Path: -- OCTET STRING -- MF/Belpic/AODF

6.3. EF(AODF)

This elementary file (Authentication Object Directory File) contains generic authentication object attributes such as allowed characters, PIN length, PIN padding character, etc. It also contains the pointers to the authentication objects themselves (in the case of PINs, pointers to the DF in which the PIN file resides). The authentication objects are used to control access to other objects such as keys. The content of this file is according to PKCS#15.

Hexadecimal dump	**Meaning (ASN.1)**
30 33	-- SEQUENCE
30 0F	Common Object Attributes: -- SEQUENCE
0C 09 42 61 73 69 63 20 50 49 4E	Label: -- UTF8 String -- "Basic PIN"
03 02 06 C0	Common Object Flags: -- BIT STRING -- private(0), modifiable(1)
30 03	Common Authentication Object Attributes: -- SEQUENCE
04 01 01	Authority ID: -- OCTET STRING -- '01'
A1 1B	-- [1] Pin Attributes
30 19	-- SEQUENCE
03 02	Pin Flags: -- BIT STRING

02 0C	-- initialized(4), needs-padding(5)
0A 01 00	PinType: -- ENUMERATED -- bcd(0)
02 01 04	Min Length: -- INTEGER -- 4
02 01 08	Stored Length: -- INTEGER -- 8
80 01 01	Pin Reference: -- [0] Pin Reference IMPLICIT INTEGER -- 1
04 01 FF	Pad Char: -- OCTET STRING -- 'FF'
30 04	Path: -- SEQUENCE
04 02 3F 00	Path: -- OCTET STRING -- 'MF'

6.4. EF(PrKDF)

This transparent elementary file (Private Key Directory File) contains general key attributes such as labels, intended usage, identifiers etc. It also contains the pointers to the keys themselves. The keys reside in the BELPIC application directory on the card.

Hexadecimal dump	**Meaning (ASN.1)**
30 3A	**Private Authentication Key:** -- SEQUENCE
30 17	Common Object Attributes: -- SEQUENCE
0C 0E 41 75 74 68 65 6E 74 69 63 61 74 69 6F 6E	Label: -- UTF8 String -- "Authentication"
03 02 06 C0	Common Object Flags: -- BIT STRING -- private(0), modifiable(1)
04 01 01	Authority ID: -- OCTET STRING -- '01'
30 0F	Common Key Attributes: -- SEQUENCE
04 01 02	Identifier: -- OCTET STRING -- '02'
03 02 05 20	KeyUsageFlags: -- BIT STRING -- Sign(2)
03 02 03 B8	Key Access Flags: -- BIT STRING -- sensitive(0), alwaysSensitive(2) neverextractable(3), local(4)

```
    02 02
    00 82

  A1 0E

    30 0C

      30 06

        04 04
        3F 00 DF 00

      02 02
      04 00



30 39

  30 15

    0C 09
    53 69 67 6E 61 74 75 72 65

    03 02
    06 C0

    04 01
    01

    02 01
    01

  30 10

    04 01
    03

    03 03
    06 00 40

    03 02
    03 B8


    02 02
    00 83
  A1 0E

    30 0C

      30 06

        04 04
        3F 00 DF 00
```

KeyReference:
-- INTEGER
-- '82'

-- [1] Private RSA Key Attributes

-- SEQUENCE

Path:
-- SEQUENCE

Path:
-- OCTET STRING
-- MF/Belpic

Modulus Length:
-- INTEGER
-- 1024

Private Non-repudiation Key:
-- SEQUENCE

Common Object Attributes:
-- SEQUENCE

Label:
-- UTF8 String
-- "Signature"

Common Object Flags:
-- BIT STRING
-- private(0), modifiable(1)

Authority ID:
-- OCTET STRING
-- '01'

UserConsent:
-- INTEGER
-- 1

Common Key Attributes:
-- SEQUENCE

Identifier:
-- OCTET STRING
-- '03'

KeyUsageFlags:
-- BIT STRING
-- NonRepudiation(9)

Key Access Flags:
-- BIT STRING
-- sensitive(0) alwaysSensitive(2) neverextractable(3) local(4)

KeyReference:
-- INTEGER
-- '83'

-- [1] Private RSA Key Attributes

-- SEQUENCE

Path:
-- SEQUENCE

Path:
-- OCTET STRING
-- MF/Belpic

Modulus Length:

02 02 04 00	-- INTEGER -- 1024

6.5. EF(CDF)

This transparent elementary file contains attributes and pointers to the authentication certificate (Cert #2), non-repudiation signature certificate (Cert #3), CA certificate (Cert#4) and root certificate (Cert #6). Information in this file contains certificate attributes such as labels, key identifiers, pointers to certificate files etc. The format of the file is specified in PKCS#15.

Depending on the citizen's choice or the type of card, there can be 3 cases:

1. **All certificates are present**: In this case, the ***EF(CDF)*** is exactly as show below.
2. **No Non-repudiation certificate is present**. In this case, the information about the *Non-repudiation certificate* (bytes *30 27 30 12 ... DF 00 50 29*) is not present. The information about the *Intermediate CA certificate* immediately follows the information about the *Authentication certificate*, and the remainder of the file is filled with zero bytes. Additionally, the *Non-repudiation certificate* file is filled with 1300 zero bytes.
3. **No Authentication and Non-repudiation certificates are present**. In this case, the information about the *Authentication* and *Non-repudiation certificates* is not present. The file starts with the information about the *Intermediate CA certificate* (bytes *30 23 30 3B ...*), and the remainder of the file is filled with zero bytes. Additionally, the *Authentication* and *Non-repudiation certificate* files are filled with 1300 zero bytes.

Hexadecimal dump	**Meaning (ASN.1)**
30 2C	**Authentication Certificate:** -- SEQUENCE
30 17	Common Object Attributes: -- SEQUENCE
0C 0E 41 75 74 68 65 6E 74 69 63 61 74 69 6F 6E	Label: -- UTF8String -- "Authentication"
03 02 06 C0	Common Object Flags: -- BIT STRING -- private(0), modifiable(1)
04 01 01	AuthID: -- OCTET STRING -- '01'
30 03	Common Certificate Attributes: -- SEQUENCE
04 01 02	Identifier: -- OCTET STRING -- '02'
A1 0C	-- [1] 509CertificateAttributes:
30 0A	-- SEQUENCE
30 08	Path: -- SEQUENCE
	Path: -- OCTET STRING

```
        04 06
        3F 00 DF 00 50 38



30 27

  30 12

    0C 09
    53 69 67 6E 61 74 75 72 65

    03 02
    06 C0

    04 01
    01

  30 03

    04 01
    03

  A1 0C

    30 0A

      30 08

        04 06
        3F 00 DF 00 50 39



30 23

  30 0B

    0C 02
    43 41

    03 02
    06 C0

    04 01
    01

  30 06

    04 01
    04

    01 01
    ff
```

-- MF/Belpic/Cert#2(auth)

Non-repudiation Certificate:

-- SEQUENCE

Common Object Attributes:
-- SEQUENCE

Label:
-- UTF8String
-- “Signature”

Common Object Flags:

-- BIT STRING
-- private(0), modifiable(1)

AuthID:
-- OCTET STRING
-- '01'

Common Certificate Attributes:
-- SEQUENCE

Identifier:
-- OCTET STRING
-- '03'

-- [1] 509CertificateAttributes:

-- SEQUENCE

Path:
-- SEQUENCE

Path:
-- OCTET STRING
-- MF/Belpic/Cert#3(non-rep)

Intermediate CA Certificate:

-- SEQUENCE

Common Object Attributes:
-- SEQUENCE

Label:
-- UTF8String
-- “CA”

Common Object Flags:

-- BIT STRING
-- private(0), modifiable(1)

AuthID:
-- OCTET STRING
-- '01'

Common Certificate Attributes:
-- SEQUENCE

Identifier:
-- OCTET STRING
-- '04'

--[3] ImplicitTrust IMPLICIT BOOLEAN
-- True

-- [1] 509CertificateAttributes:

A1 0C 30 0A 30 08 04 06 3F 00 DF 00 50 3a 30 25 30 0D 0C 04 52 6f 6f 74 03 02 06 C0 04 01 01 30 06 04 01 06 01 01 ff A1 0C 30 0A 30 08 04 06 3F 00 DF 00 50 3b	-- SEQUENCE Path: -- SEQUENCE Path: -- OCTET STRING -- MF/Belpic/Cert#4(CA) **Intermediate CA Certificate:** -- SEQUENCE Common Object Attributes: -- SEQUENCE Label: -- UTF8String -- “Root” Common Object Flags: -- BIT STRING -- private(0), modifiable(1) AuthID: -- OCTET STRING -- '01' Common Certificate Attributes: -- SEQUENCE Identifier: -- OCTET STRING -- '06' --[3] ImplicitTrust IMPLICIT BOOLEAN -- True -- [1] 509CertificateAttributes: -- SEQUENCE Path: -- SEQUENCE Path: -- OCTET STRING -- MF/Belpic/Cert#4(Root)

7. Public and Private Keys detail

7.1. Private RSA Key #1

This file contains the private RSA ***Basic Key***. It is involved in the ***internal authentication*** process. The corresponding public key is not present on the card.

7.2. Private RSA Key #2

This file contains the private RSA ***authentication key***.

7.3. Private RSA Key #3

This file contains the private RSA ***non-repudiation key***.

The userConsent element in ***PrKDF*** contains value 1 for this key i.e. the cardholder must manually enter the corresponding PIN for each private key operation.

7.4. Public RSA Key #7

This file contains the public RSA ***CA role key*** used for external and mutual authentication.

8. Certificates detail

All certificates stored in the card are DER encoded (not Base 64).

8.1. Certificate #2

This file contains the citizen's X.509 ***authentication certificate*** containing the public key corresponding to the private RSA ***authentication key*** (Private RSA Key #2). When the file is created and written to during personalisation, 50 zero bytes are appended to it. If **no authentication certificate** is issued for this person, this file consists of 1300 zero bytes.

8.2. Certificate #3

This file contains the citizen's X.509 ***non-repudiation certificate*** containing the public key corresponding to the private RSA '***non-repudiation key***' (Private RSA Key #3). When the file is created and written to during personalisation, 50 zero bytes are appended to it. If **no non-repudiation certificate** is issued for this person, this file consists of 1300 zero bytes.

8.3. Certificate #4

This file contains the X.509 ***Citizen's CA certificate*** used to sign the ***authentication certificate*** (#2) and the ***non-repudiation certificate*** (#3). When the file is created and written to during personalisation, 50 zero bytes are appended to it.

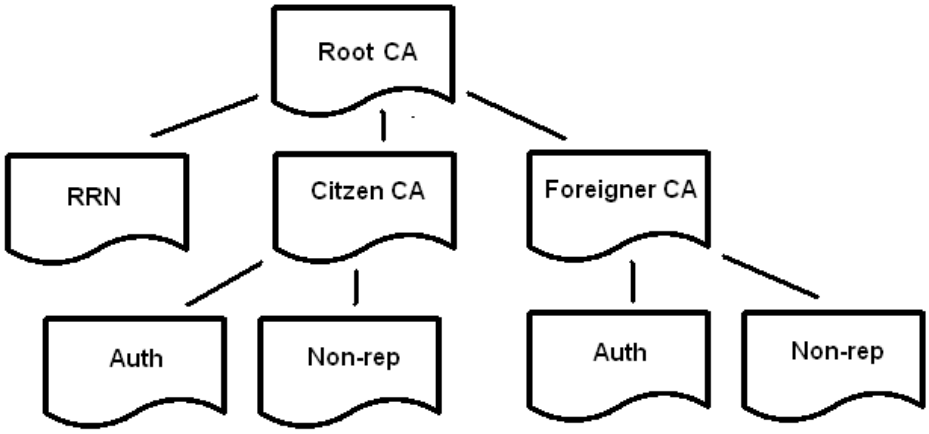
In the case of a resident **card for an EU or non-EU citizen**, this file contains the **Foreigner CA certificate** Instead.

8.4. Certificate #6

This file contains the X.509 ***Root certificate*** used to sign the ***Citizen's CA certificate*** (#4) or the **Foreigner CA certificate**(#4) and the **RRN certificate** (#8)

8.5. Certificate #8

This file contains the X.509 ***RRN certificate***. This certificate corresponds to the private key used to sign the files ***EF(ID#RN)*** and ***EF(ID#Address)***.



9. Specification of ID file in de chip

File contents and format								
Tag (decimal)	Tag (hexa)	Current max. # bytes (decimal)	Data	Encoding type	Default value	Envisioned max. # bytes (decimal)	eID	Foreigner
0	**00**	2	File structure version	Binary	N/A	2	✓	✓
1	**01**	12	Card Number	ASCII	M	12	✓	✓
2	**02**	16	Chip Number	Binary	M	16	✓	✓
3	**03**	10	Card validity date begin: ***DD.MM.YYYY***	ASCII	M	10	✓	✓
4	**04**	10	Card validity date end: ***DD.MM.YYYY***	ASCII	M	10	✓	✓
5	**05**	(42) *47	Card delivery municipality	UTF-8	M	80	✓	✓
6	**06**	11	National Number	ASCII	M	11	✓	✓
7	**07**	(62) *90	Name	UTF-8	M	110	✓	✓
8	**08**	(52) *75	2 first given names	UTF-8	""	95	✓	✓
9	**09**	3	First letter of 3rd given name	UTF-8	""	3	✓	✓
10	**0A**	(50) *65	Nationality	UTF-8	M	85	✓	✓
11	**0B**	(40) *60	Birth location	UTF-8	M	80	✓	✓
12	**0C**	12	Birth date: ***DD mmmm YYYY*** or ***DD.mmm. YYYY*** (German)	UTF-8	M	12	✓	✓
13	**0D**	1	Sex ***M***: man ***F/V/W***: woman	ASCII	M	1	✓	✓
14	**0E**	(21) *30	Noble condition	UTF-8	""	50	✓	✓

File contents and format								
Tag (decimal)	**Tag (hexa)**	**Current max. # bytes (decimal)**	**Data**	**Encoding type**	**Default value**	**Envisioned max. # bytes (decimal)**	**eID**	**Foreigner**
15	**0F**	2	Document type ***1***:Belgian citizen ***6***:kids card (< 12 year) ***7***:bootstrap card ***8***:"habilitation/machtigings" card ***11:*** card A ***12:*** card B ***13:*** card C ***14:*** card D ***15:*** card E ***16:*** card E+ ***17:*** card F ***18:*** card F+ ***19:*** card H	ASCII	M	2	✓	✓
16	**10**	1	Special status ***0***: No status ~~***1***: White cane (blind people)~~ ~~2: Extended minority~~ ~~***3***: White cane + extended minority~~ ~~***4***: Yellow cane (partially sighted people)~~ ~~***5***: Yellow cane + extended minority~~	ASCII	0	2	✓	✓
17	**11**	20	hash picture	Binary (SHA-1)	M	20	✓	✓
18	**12**	2	Duplicate	ASCII	0	2		✓

File contents and format								
Tag (decimal)	**Tag (hexa)**	**Current max. # bytes (decimal)**	**Data**	**Encoding type**	**Default value**	**Envisioned max. # bytes (decimal)**	**eID**	**Foreigner**
19	**13**	1	Special organisation ***1***: SHAPE ***2***: NATO ***4***: old-carte-bleue-euro ***5***: researcher	ASCII	""	1		✓
20	**14**	0	Member of family	Boolean	absent			✓
21	**15**	13	Date and country of protection dd.MM.yyyy-A2	ASCII	""	13		✓

Default Value:

N/A = Not applicable

M = Mandatory

"" = empty

0 = value equal to zero

Absent = tag not present