



BSI Standards Publication

Identification card systems — European Citizen Card

Part 2: Logical data structures and
security services

NO COPYING WITHOUT BSI PERMISSION EXCEPT AS PERMITTED BY COPYRIGHT LAW

raising standards worldwide™

National foreword

This Published Document is the UK implementation of CEN/TS 15480-2:2012. It supersedes DD CEN/TS 15480-2:2007 which is withdrawn.

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

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

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© The British Standards Institution 2012

Published by BSI Standards Limited 2012

ISBN 978 0 580 75764 8

ICS 35.240.15

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

This Published Document was published under the authority of the Policy Standards Policy and Strategy Committee on 31 July 2012.

Amendments issued since publication

Amd. No.	Date	Text affected
----------	------	---------------

TECHNICAL SPECIFICATION
SPÉCIFICATION TECHNIQUE
TECHNISCHE SPEZIFIKATION

CEN/TS 15480-2

June 2012

ICS 35.240.15

Supersedes CEN/TS 15480-2:2007

English Version

**Identification card systems - European Citizen Card - Part 2:
Logical data structures and security services**

Systèmes de cartes d'identification - Carte Européenne du
Citoyen - Partie 2: structures de données logiques et
services de sécurité

Identifikationskartensysteme - Europäische Bürgerkarte -
Teil 2: Logische Datenstrukturen und Sicherheitsfunktionen

This Technical Specification (CEN/TS) was approved by CEN on 9 January 2012 for provisional application.

The period of validity of this CEN/TS is limited initially to three years. After two years the members of CEN will be requested to submit their comments, particularly on the question whether the CEN/TS can be converted into a European Standard.

CEN members are required to announce the existence of this CEN/TS in the same way as for an EN and to make the CEN/TS available promptly at national level in an appropriate form. It is permissible to keep conflicting national standards in force (in parallel to the CEN/TS) until the final decision about the possible conversion of the CEN/TS into an EN is reached.

CEN members are the national standards bodies of Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, Former Yugoslav Republic of Macedonia, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and United Kingdom.



EUROPEAN COMMITTEE FOR STANDARDIZATION
COMITÉ EUROPÉEN DE NORMALISATION
EUROPÄISCHES KOMITEE FÜR NORMUNG

Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents

Page

Foreword.....	4
1 Scope	5
2 Normative references	5
3 Terms and definitions	6
4 Abbreviations	7
4.1 Abbreviations	7
4.2 Coding conventions and notation.....	9
5 Data elements and data structures	10
5.1 Supported data Structures	10
5.2 Access to data structures	10
5.3 Answer to reset (ATR) / answer to select (ATS)	11
5.4 General architecture and file supported	15
5.5 Selection of data structures	16
5.6 Access to files.....	17
6 Basic card services	18
6.1 General.....	18
6.2 Identification.....	18
6.3 User verification.....	20
6.4 Device authentication.....	20
6.5 Digital signature.....	23
6.6 Client/Server Authentication	24
6.7 Encryption key decipherment	24
7 Extended card services.....	25
7.1 General.....	25
7.2 Biometrics – on card matching	25
7.3 Passive Authentication	25
7.4 Basic Access Control.....	25
7.5 Active Authentication.....	25
7.6 Extended Access Control	26
7.7 Role authentication.....	26
7.8 Restricted Identification (RI).....	27
7.9 Age, Validity or Auxiliary Data Verification	28
7.10 Modular Enhanced Role Authentication (mERA)	28
Annex A (normative) Command set.....	29
A.1 CLASS byte coding.....	29
A.2 Command chaining mechanisms.....	29
A.3 Extended length mechanism	30
A.4 Logical channels.....	31
A.5 Short and extended length fields	31
A.6 Status words	31
A.7 Command set	32
Annex B (normative) Cryptographic Information Application	54
B.1 Description	54
B.2 CIA data organisation.....	63
Annex C (normative) Mandatory features	83
C.1 General.....	83
C.2 Data elements and data structures	83

C.3	Card services	84
C.4	Command set	84
C.5	Device Authentication and Key Derivation	85
C.6	Digital signature	85
C.7	Client/Server Authentication	86
C.8	Encryption Key Decipherment	86
Annex D	(informative) Optional features	87
D.1	General	87
D.2	Data elements and data structures	87
D.3	Card services	88
D.4	Command set	88
D.5	Device Authentication and Key Derivation	89
D.6	Digital signature	89
Annex E	(informative) Application Profiles	90
E.1	General	90
E.2	Application Profile 1: ICAO Application with EAC features	90
E.3	Application Profile 2: Travel Document Application	96
E.4	Application Profile 3: eID Application	101
E.5	Application Profile 4: Digital Signature Application	111
E.6	E.6 Application Profile 5: eServices Application using a trusted third party	121
E.7	Application Profile 6: Health Insurance Application	136
E.8	Application Profile 7: Combined eID and signature application	152
E.9	Application Profile 8: Multi-Service application	156
Annex F	(informative) Access rules in expanded format	161
F.1	Object protection by access rules in expanded format	161
F.2	Access rules in expanded format	161
F.3	Security attribute referencing expanded format	162
F.4	Security attribute template for physical interfaces	163
Annex G	(informative) Example of data structure: the Security Data Objects concept	164
G.1	SDO concept	164
Bibliography		176

Foreword

This document (CEN/TS 15480-2:2012) has been prepared by Technical Committee CEN/TC 224 “Personal identification, electronic signature and cards and their related systems and operations”, the secretariat of which is held by AFNOR.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. CEN [and/or CENELEC] shall not be held responsible for identifying any or all such patent rights.

This document supersedes CEN/TS 15480-2:2007.

CEN/TS 15480 *Identification card systems — European Citizen Card* consists of the following four parts:

Part 1, Physical, electrical and transport protocol characteristics

Part 2, Logical data structures and card services

Part 3, European Citizen Card Interoperability using an application interface

Part 4, Recommendations for European Citizen Card issuance, operation and use

According to the CEN/CENELEC Internal Regulations, the national standards organisations of the following countries are bound to announce this Technical Specification: Austria, Belgium, Bulgaria, Croatia, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, Turkey and the United Kingdom.

1 Scope

This Technical Specification specifies the logical characteristics and security features at the card/system interface for the European Citizen Card.

The European Citizen Card is a smart card with Identification, Authentication and electronic Signature (IAS) services. Therefore:

- the supported services are specified;
- the supported data structures as well as the access to these structures are specified;
- the command set is defined.

This Technical Specification aims to ensure the interoperability at card/system interface in the usage phase.

In order to reach the interoperability objective, IAS services are compliant with EN 14890 Part 1 and Part 2. As the EN documents offer options, this specification fully defines a complete profile.

This Technical Specification also considers ICAO Doc 9303.

This Technical Specification does not mandate the use of a particular technology, and is intended to allow both native and Java card technologies.

This specification encompasses mandatory and optional features. Optional features make up a toolbox of modular options from which issuers can pick up the necessary protocols to fulfil the requirements for use. Mandatory features shall be implemented for a smart card to be compliant with this Technical Specification. Mandatory features required for compliancy to ECC specification are given in Annex C, the optional features are given in Annex D. Two IAS-enabled smart cards issued by two different issuers, and compliant with this Technical Specification but implementing different application profiles out of this Technical Specification, can interoperate with a terminal provided that such a terminal supports both application profiles. Therefore, interoperability requires a specific agreement between issuers/governments in order to determine which cross-border services are to be shared, and consequently, which protocols are to be supported by the terminals in each country.

All the APDU commands described in this Technical Specification are in accordance with ISO/IEC 7816 Part 4 or Part 8. They are fully described here in order to provide the settings adopted by this specification and to prevent any ambiguity in case of several possible interpretations of the standards.

For physical, electrical and transport protocol characteristics, refer to CEN/TS 15480-1.

2 Normative references

The following documents, in whole or in part, are normatively referenced in this document and are indispensable for its application. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

EN 14890-1:2008:2008, *Application Interface for smart cards used as Secure Signature Creation Devices — Part 1: Basic requirements*

ISO/IEC 7816-3, *Information technology — Identification cards — Integrated circuit(s) cards with contacts — Part 3: Electronic signals and transmission protocols*

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

ISO/IEC 7816-4:2005/PDAM 2.1:2008, *Identification cards — Integrated circuit(s) cards — Part 4: Organisation, security and commands for interchange, AMENDMENT 2: Handling of Extended Length Information (Working Draft)*

ISO/IEC 7816-11, *Identification cards — Integrated circuit cards — Part 11: Personal verification through biometric methods*

ISO/IEC 7816-15, *Identification cards — Integrated circuit cards with contact — Part 15: Cryptographic information application*

ISO/IEC 7816-15:2004/Amd 1:2007, *Identification cards — Integrated circuit cards with contact — Part 15: Cryptographic information application, AMENDMENT 1: Examples of the use of the cryptographic information application*

ISO/IEC 7816-15:2004/Amd 2:2008, *Identification cards — Integrated circuit cards with contact — Part 15: Cryptographic information application, AMENDMENT 2: Error corrections and extensions for multi-application environments*

ISO/IEC 9796-2, *Information technology — Security techniques — Digital signature schemes giving message recovery — Part 2: Integer factorisation based mechanisms*

ISO/IEC 14443-4, *Identification cards — Contactless integrated circuit(s) cards — Proximity cards — Part 4: Transmission protocol*

ISO/IEC 19794-2, *Information technology — Biometric data interchange formats — Part 2: Finger minutiae data*

ANSI X9.63, *Public Key Cryptography For the Financial Services Industry: Key Agreement and Key Transport Using Elliptic Curve Cryptography*, January 8th 1999

BSI TR-03110 Version 1.11, *Advanced Security Mechanisms for Machine Readable Travel Documents – Extended Access Control (EAC)*

BSI TR-03111 Version 1.11, *Technical Guideline Elliptic Curve Cryptography*

ICAO Doc 9303, *Machine Readable Travel Documents, Part 3 – Machine Readable Official Travel Documents – Volume 2 Specifications for Electronically Enabled MRtds with Biometric Identification Capability*, Third Edition, 2008

3 Terms and definitions

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

3.1

Application Dedicated File (ADF)

structure hosting an application in a card

3.2

root

Master File MF in case of a native operating system, the applet instance having the default selection privilege in case of a Java card implementation

4 Abbreviations

4.1 Abbreviations

ADF	Application Dedicated File
AID	Application Identifier
AMB	Access Mode Byte
AT	Authentication Template
ATR	Answer to Reset
ATS	Answer to Select
BER	Basic Encoding Rules
BHT	Biometric Header Template
BIT	Biometric Information Template
CA	Certification Authority
CAR	Certification Authority Reference
CCT	Cryptographic Checksum Template
CED	Certificate Effective Date
CHA	Certificate Holder Authorisation
CHR	Certificate Holder Reference
CIA	Cryptographic Information Application
CPI	Certificate Profile Identifier
CRT	Control Reference Template
CT	Confidentiality Template
CV	Card Verifiable
CXD	Certificate Expiration Date
DES	Data Encryption Standard
DF	Dedicated File
DH	Diffie Hellman
DOCP	Data Object Control Parameters
DST	Digital Signature Template
ECDH	Elliptic Curve DH

ELC	Elliptic Curve Cryptosystem
ECDSA	Elliptic Curve Digital Signature Algorithm
EF	Elementary File
FCI	File Control Information
FCP	File Control Parameters
HT	Hash Template
IAS	Identification, Authentication and electronic Signature
ICC	Integrated Circuit Card
IFD	Interface Device
KAT	Control reference template for key agreement
LSB	Least Significant Byte
MAC	Message Authentication Code
MF	Master File
MSE	Manage Security Environment
OID	Object Identifier
PAN	Primary Account Number
PIN	Personal Identification Number
PK – DH	Public key – Diffie Hellman (asymmetric key base algorithm)
PSO	Perform Security Operation
RFU	Reserved for Future Use
RSA	Rivest Shamir Adleman
SDO	Security Data Object
SCB	Security Condition Byte
SE	Security Environment
SEID	Security Environment IDentifier byte
SM	Secure Messaging
TLV	Tag Length Value
UQB	Usage Qualifier Byte

4.2 Coding conventions and notation

4.2.1 Coding conventions

The following coding conventions apply throughout the Technical Specification:

'0' to '9' and 'A' to 'F' the sixteen hexadecimal digits;

" ... " ASCII-string;

B1 || B2 concatenation of bytes B1 (the most significant byte) and B2 (the least significant byte).

4.2.2 Notation

For private and public keys as well as for certificates the following simplified Backus-Naur notation is used:

<object descriptor>::=	<key descriptor> <certificate descriptor>
<key descriptor>::=	<asymmetric key>.<keyholder>.<usage>
<asymmetric key>::=	<private key> <public key>
<private key>::=	PrK
<public key>::=	PuK
<keyholder>::=	<cardholder> <device>
<cardholder>::=	CH
<device>::=	<integrated circuit(s) card> <interface device> <sender>
<integrated circuit(s) card>::=	ICC
<interface device>::=	IFD
<sender>::=	S
<usage>::=	<authentication> <role authentication> <key agreement> <key encipherment>
<authentication>::=	AUT
<key agreement>::=	KA
<role authentication>::=	RA
<key encipherment>::=	KE
<certificate descriptor>::=	<certificate>.<certholder>.<usage>
<certificate>::=	<card verifiable certificate> <X.509 certificate>
<card verifiable certificate>::=	C_CV
<X.509 certificate>::=	C_X509

<certholder>::= **<cardholder> | <device>**

EXAMPLES:

- 1) PrK.ICC.AUT = Private key of the ICC for device authentication;
- 2) PuK.S.KA = Public key of the sender used for key agreement;
- 3) C_X509.CH.AUT = Certificate of the card holder for client/server authentication.

In addition, the following notation is used:

K _{ICC/IFD}	Randomness provided by ICC and IFD used for session key derivation
RND.ICC	Random number of the ICC
RND.IFD	Random number of the IFD
SN.ICC	Serial number of the ICC
SN.IFD	Serial number of the IFD

5 Data elements and data structures

5.1 Supported data Structures

The European Citizen Card shall support the data structures described in 5.2. These data structures are used to store externally accessed data (certificates, card serial number, ...). Exceptions from this rule, i.e. cases where data objects are used that can be accessed by a GET DATA command are listed in the description of the services.

In addition, the card may support further data structures including proprietary structures to handle data as long as these structures have no effect on the services defined in this Technical Specification, i.e. on the interoperability. For example, the storage of private and secret keys, the storage of PIN reference data and of security environments is not defined in this Technical Report. The storage of these entities is out of the scope of this Technical Specification and implementation specific.

5.2 Access to data structures

5.2.1 File system considerations

The European Citizen Card might embed a virtual machine or be a native operating system.

- 1) The card may include an MF. The differentiation between cards with or without an MF is based on the card ATR/ATS or the content of EF.ATR/INFO. See ISO/IEC 7816-4:2005, 8.1 – card service data byte. Consequently, the card shall include the card service data byte when returning the ATR/ATS.
- 2) If an application is selected implicitly, i.e., always selected at the card reset, it has the default selection privilege. The corresponding AID shall be indicated in the historical bytes or the EF.ATR.
- 3) The root is the MF or the applet instance having the default selection privilege. This depends on the card manufacturer implementation choice (native or JavaCard implementation).
- 4) Three basic file types are supported (refer to ISO/IEC 7816-4 for definitions of EF, DF and ADF):

- i) transparent EF;
 - ii) dedicated files DF;
 - iii) application dedicated files ADF
- 5) For cards without MF, each applet instance matches with at least one application DF (ADF).
- 6) All cards shall contain an EF.DIR file.
- i) The EF.DIR is always under the root.
 - ii) The file identifier of EF.DIR is: '2F00', the short EF identifier is 30 = '1E' = 11110 bin.

5.3 Answer to reset (ATR) / answer to select (ATS)

5.3.1 General

The ATR of the card shall follow the rules indicated in ISO/IEC 7816-3 and ISO/IEC 7816-4. The ATS of the card shall follow the rules specified in ISO/IEC 14443-4.

Data objects for card identification shall be provided in the historical data bytes of the ATR/ATS or in an optional EF.ATR/INFO (see, 5.3.3). In case of ISO/IEC 14443-4 and protocol type B no ATS is available and for this reason the presence of the EF.ATR/INFO is mandatory in this case.

Table 1 provides the list of data objects which may be supported by the card in the ATR/ATS in the Compact-TLV format as defined in ISO/IEC 7816-4, Table 2 provides the data objects in EF.ATR/INFO for the BER-TLV structure. The data objects defined in Table 1 and Table 2 have to be used as given in the tables, these are not application specific.

5.3.2 Historical bytes

The ATR/ATS contains configuration data so that ICC and IFD can communicate together (protocol, speed, etc.).

The category indicator as the first historical byte shall be set to the value '00'. Therefore, the last three bytes shall be a status indicator, i.e. a card life cycle status indicator followed by two status bytes SW1-SW2.

Transmission in historical bytes for the data objects "card service data" and "card capabilities" is mandatory. For other data objects the decision is left to the card manufacturer.

If a data object from Table 1 is used in the historical bytes the length of the DO shall be used as defined in the table. Furthermore, the definitions for the coding of content of the data objects given in the table are mandatory. The coding of further parameters in the content of the data objects is left to the choice of the card manufacturer but shall follow the rules given in ISO/IEC 7816-4.

Table 1 — Card Identification Historical Bytes

Byte #	Name	Value	Description
1	Category indicator	'00'	COMPACT-TLV data objects followed by a status indicator shall be present as the last three historical bytes
2	Card service data tag	'31'	Tag for next byte
3	Card service data byte	'B9' or 'B8'	b8=1: Application selection by full DF name b6=1: BER-TLV DO are present in EF.DIR b5=1: BER-TLV DO present in EF.ATR/INFO ^a b4...b2=100: EF.DIR/EF.ATR is a transparent EF (use READ BINARY) b1 = 0: Card with MF b1 = 1: Card without MF
4	Pre-issuing DO tag	'64'	Tag for next 4 bytes
5	IC manufacturer	'XX'	IC Manufacturer according ISO/IEC 7816-6
6	Type of the IC	'XX'	defined by the IC or card manufacturer
7	OS Version	XX	Version of the operating system defined by card manufacturer
8	Discretionary data	'XX'	Discretionary data
9	Card capabilities data tag	'73'	Tag for next 3 bytes
10	Card capabilities data byte 1 → <i>Selection method</i>	'94'	DF selection b8=1: DF selection full name b5=1: DF selection using file identifier EF selection b3=1: file selection using short file identifier is supported
11	Card capabilities data byte 2 → <i>Data coding byte</i>	'01'	b4...b1 = 0001: data unit size is 1 byte
12	Card capabilities data byte 3 → <i>Miscellaneous</i>	'C0', '80', 'D0' or '90'	b8=1: command chaining is supported b7=0: Extended Lc and Le fields not supported b7=1: Extended Lc and Le fields supported b5, b4=00: no logical channel supported b5, b4=10: channel number assignment by the card Maximum number of channels supported: 4
13	Status indicator tag	'8x'	Tag for next byte, x is either '2' or '3'
14	Status indicator	[LCS] '9000'	Life Cycle State (optional) followed by status words SW1-SW2
^a DO may be present in EF.ATR/INFO for compatible tag			

The list and the order of the historical bytes are compulsory unless further items complete these historical bytes according to the smart card manufacturer's policy (e.g. reference and version of the application).

5.3.3 EF.ATR/INFO

At least one of the options ATR/ATS or EF.ATR/INFO shall be supported.

The support of an EF.ATR/INFO shall be indicated in the "card service data byte" of the historical bytes if an ATR/ATS is supported. With regard to ISO/IEC 14443-4, the ATS Application information bytes are not normalised for data objects interoperable description. Therefore, for both contact and contact-less cards, EF.ATR/INFO may host information.

For data objects stored in the EF.ATR/INFO the BER-TLV format is mandatory. All data objects given in Table 2 are mandatory to be contained in EF.ATR/INFO except if they are conditional or not applicable.

In order to identify the compatible tag allocation scheme and the authority responsible for the scheme, the interindustry template referenced by tag '78' is used by the present specification. The allocation authority shall be identified as CEN by an OID with the following root:

ISO (1) identified-organisation (3) CEN (162)

The specification number to be defined according to CEN conventions shall be added to this root to figure the full OID. The corresponding BER-TLV encoded OID shall be nested in the data object '06'.

The DO referenced by tag '78' may be hosted in EF.ATR while the historical byte denoting the Card Service Data shall has bit5 set to one to indicate that a DO is available in EF.ATR.

Alternatively, the D.O. '78' can be hosted by an ADF whereby denoting that the tag allocation scheme applies to data objects within this ADF and not to the whole card.

Table 2 — Card Identification Data Objects (1 of 2)

Tag	Length	What	Value	Meaning
'80'	N.A.	Category indicator	'00'	Indicate format of next historical bytes (compact TLV)
'43'	'01'	Card service data tag		Tag for next byte
		Card service data byte	'B9', 'B8'	b8=1 : Application selection by full DF name b6=1 : BER TLV DO are present in EF.DIR b5=1 : BER-TLV DO present in EF.ATR ^a b4...b2=100 : EF.DIR/EF.ATR is a transparent EF (use READ BINARY) b1 = 0 : Card with MF b1 = 1 : Card without MF
'46'	'04'	Pre-issuing DO		Tag for next 4 bytes
		IC manufacturer	'XX'	IC Manufacturer according ISO/IEC 7816-6
		Type of the IC	'XX'	defined by the IC or card manufacturer
		OS Version	'XX'	Version of the operating system defined by card manufacturer
		Discretionary data	'XX'	Discretionary data
'47'	'03'	Card capabilities tag		Tag for next 3 bytes
		Card capabilities data byte 1 → <i>Selection method</i>	'94'	DF selection b8=1 : DF selection full name b5=1 : DF selection using file identifier EF selection b3=1 : file selection using short file identifier is supported
		Card capabilities data byte 2 → <i>Data coding byte</i>	'01'	b4...b1 = 0001 : data unit size is 1 byte
		Card capabilities data byte 3 → <i>Miscellaneous</i>	'C0', '80', 'D0', '90'	b8=1 : command chaining is supported b7=0 : Extended Lc and Le fields NOT supported b7=1 : Extended Lc and Le fields supported b5, b4=00 : no logical channel supported b5, b4=10 : channel number assignment by the card Maximum number of channels supported :4
'4F'	'01'...'10'	Application Identifier	'XX'...'XX'	AID of implicitly selected application (1 to 16 bytes)

Table 2 — Card Identification Data Objects (2 of 2)

'E0' ^b	'10'	IO buffer size	'02' – L ₀₂ – 'WW'...'WW' '02' – L ₀₂ – 'XX'...'XX' '02' – L ₀₂ – 'YY'...'YY' '02' – L ₀₂ – 'ZZ'...'ZZ'	4 concatenated data objects with tag '02' (Universal class). The value field of each DO encodes the maximum number of bytes of the respective APDU. 'WW'...'WW' = DO maximum length of command APDU without secure messaging 'XX'...'XX' = DO maximum length of command APDU with secure messaging 'YY'...'YY' = DO maximum length of response APDU without secure messaging 'ZZ'...'ZZ' = DO maximum length of response APDU with secure messaging.
'7F66'	Variable	IO buffer size	'02' – L ₀₂ – 'XX'...'XX' '02' – L ₀₂ – 'YY'...'YY'	'XX'...'XX': Positive integer, the number of bytes in a command APDU shall not exceed this number 'YY'...'YY': Positive integer, if the card does not support response chaining for a particular command-response pair, then N _e shall be set such that the number of bytes in a response APDU does not exceed this number
'78'	'08'	Allocation scheme tag		Tag for next 8 bytes
		OID	'06 06 2B8022F87802'	OID of the allocation authority to be identified as CEN.
'82'	'02' or '03'	Status indicator		Tag for next 2 or 3 bytes
		Status Word	[LCS] '9000'	Life cycle state (optional) followed by SW1+ SW2
^a A DO(s) may be present in EF.ATR for the purposes of compatible tag allocation scheme or other purposes				
^b The use of the 'E0' data object is not recommended for future applications.				

5.4 General architecture and file supported

Figure 1 shows an example of directory and elementary file organisation.

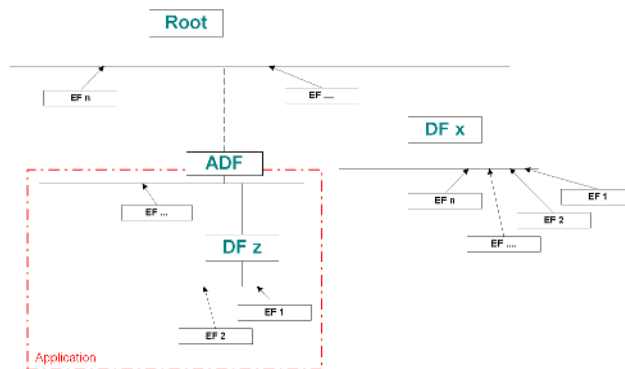


Figure 1 — Directory architecture example

The card supports the following type of files:

- the root is unique and identifies the default selected directory after reset;
- DF files that are attached to the root or to another DF;
- one or more ADFs each representing an application that may include dedicated files;
- transparent EF files.

An ADF may be stored under another ADF. Due to the fact that an ADF is always selected by its AID, this is not visible by the outside world.

For interoperability purposes, the European Citizen Card shall support at least one level of DF and ADF under the root and one level of DF under any ADF.

5.5 Selection of data structures

5.5.1 Selection of an application

After reset and return of the ATR/ATS, the root is automatically selected (refer to Clause 3 for definition of the root).

The terminal may select the EF.DIR, which is a mandatory EF, always present under the root. The EF.DIR is always selected by FID ('2F00') or addressed by its short EF identifier '1E' = 30.

The EF.DIR contains the list of the AID of the applications (ADF) present on the smart card.

ADF selection is always performed by name. An ADF may support the selection by more than one AID (e.g. a national and an international AID) in order to support interoperability.

In case of a card without MF (typically a Java card implementation), the SELECT by AID of an application shall be issued by the terminal always in clear, even if a secure messaging session is currently active. For such cards this SELECT by AID in clear shall not break the current secure messaging session (if any). For such cards this behaviour is necessary in order to allow the association of one applet instance to one application in a Java card.

5.5.2 Selection of files

Files selection is always relative to the ADF. The EF selection can be performed either:

- explicitly by the SELECT command using FID over two bytes, or
- implicitly by addressing the file content using short EF identifier.

All files, except ADF or root (when card without MF), may be selected by file ID (2 bytes-long). An elementary file may also be selected implicitly by an access with short EF identifier.

5.6 Access to files

5.6.1 File control parameter

The File Control Parameter (FCP) referenced by tag '62' provides the logical, structural and security attributes of a file (EF or DF or ADF). Table 3 — File Control Parameter defines the subset of ISO/IEC 7816-4 to be supported at least by the ICC. The FCP is returned by a SELECT command if the corresponding command parameter is set, see Annex A. The European Citizen Card may return additional data objects in the FCP (i.e. data objects with other tags). Those data objects are to be ignored by the IFD in the context of ECC usage.

The ICC may also support File Control Information (FCI) and File Management Data (FMD) that may contain proprietary data not relevant with respect to interoperability. For this reason the FCI as well as the FMD are out of the scope of the present specification.

Table 3 — File Control Parameter

Tag	Length	Description			
'62'	Var.	File Control Parameter (FCP)			
		Tag	Length	Description	Applies to
		'80'	'02'	Number of data bytes in the file excluding structural information	Transparent EFs
		'82'	'01'	File descriptor byte (data coding byte)	DFs, ADFs, transparent EFs
		'83'	'02'	File Identifier coded on two bytes	DFs, transparent EFs ^a
		'84'	'01' to '10'	DF name (AID)	ADF
		'88'	'01'	Short EF identifier (optional)	EFs
		'8A'	'01'	Life cycle status byte	Any file
		'A1'	variable	Security attribute template for physical interfaces ^b	Any file

^a An ADF shall be selected by AID but the tag '83' may be present inside the FCP of an ADF. For interoperability reasons, an ADF shall not be selected by File ID.

^b This data object will be substituted by the the data object defined in the upcoming revised CD version of ISO/IEC 7816-4 as soon as it is available.

Whether access rules in compact or in (referenced) expanded format compliant with ISO/IEC 7816-4 are supported is out of the scope of this Technical Specification.

Where an Application DF is selectable by more than one AID the FCP returned by a SELECT command shall contain at least the AID used to select the ADF.

The File Control Parameter data objects are encoded compliant with ISO/IEC 7816-4 and described in the following clauses.

There are no restrictions on file IDs, except for those defined by ISO/IEC 7816-4 and this standard (e.g. ID = '3F00' is reserved for the MF, '2F00' is reserved for EF.DIR, '2F01' is reserved for EF.ATR, '0000' is reserved for the current file, '3FFF' and 'FFFF' are forbidden etc.).

5.6.2 File descriptor byte

The file descriptor byte is encoded as defined in Table 2. Bit 4, 5 and 6 are not relevant with respect to interoperability and therefore not defined in the present specification.

Table 4 — File descriptor byte

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0							File Accessibility
0	1							Not shareable file
0	-	1	1	1	0	0	0	DF
0	-	Not all set to 1			-	-	-	EF category
0	-	-	-	-	0	0	1	EF structure
0	-	-	-	-	0	0	1	Transparent structure
- 'Shareable' means that the file supports at least concurrent access on different logical channels. - The support of shareable files is optional.								

5.6.3 Life cycle status byte

Referenced by tag '8A' a life cycle status byte is present in the FCP of any file. This byte shall be interpreted compliant with ISO/IEC 7816-4 according to Table 2.

Table 5 — Life cycle status byte

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	Applies to
0	0	0	0	0	0	0	1	Creation state ^a	Any file
0	0	0	0	0	0	1	1	Initialisation state ^a	Any file
0	0	0	0	0	1	-	1	Operational state (activated)	Any file
0	0	0	0	0	1	-	0	Operational state (deactivated)	Any file
0	0	0	0	1	1	-	-	Termination state	Any file
- Any other value is RFU									
^a Usage of creation and initialization states are implementation dependant.									

6 Basic card services

6.1 General

The card services described here from a functional standpoint relate to the IAS services as referred to in EN 14890-1:2008:2008 and EN 14890-2. Unless indicated as OPTIONAL, the following services shall be supported.

6.2 Identification

6.2.1 Identification of the cardholder

At personalisation time, any data dedicated to identify the cardholder are designated as identification data.

6.2.2 Identification of the European Citizen Card

If the European Citizen card makes use of the symmetric device authentication protocol according to 6.4.3 or an asymmetric device authentication protocol according to 6.4.4.2 or 6.4.4.3, the ECC shall contain a Primary Account Number (PAN). The structure of the PAN is in accordance with ISO/IEC 7812-1 and defined in Table 2. The PAN is BCD coded.

Table 6 — Primary Account Number

	ISO/IEC 7812 definition			
Length	6 digits (BCD)		12 digits (BCD)	1 digit (BCD)
Data	IIN Issuer Identification Number			Check digit
	MII	Country code	Issuer identifier	
			Individual card number	

For the Issuer Identification Number (IIN) the following definitions apply:

- the Major Industry Identifier (MI) typically is a single digit with the value 9, i.e., for assignment by national standard bodies according to ISO/IEC 7812-1; the MI may take other values as defined in ISO/IEC 7812.
- the country code is a three digit value in accordance with ISO 3166-1;
- the issuer identifier is coded on two digits and identifies for the major industry and the country the card issuer in a unique way. The value is assigned by the corresponding national standard body. Depending on National profile, the Issuer identifier value may be longer. In such cases, provision for extra digits shall be taken over the individual card number, the overall length of IIN and Individual card number being always 18 digits.

The individual account identification is typically coded on 12 digits (BCD) and allows a unique identification of the smart card over all cards having the same IIN.

The check digit is a single digit and allows verification of the PAN integrity. The check digit is calculated according to the Luhn formula, see ISO/IEC 7812-1.

The PAN is stored in the elementary file EF.SN under the root. The contents of EF.SN is BER-TLV coded using the tag '5A' that references the PAN, see Table 2. In this data object the PAN (19 digits) is concatenated with a filler digit (1 digit with value 'F', BCD encoded) in order to get a value field that is 10 bytes long. The (short) EF Identifier of EF.SN is provided by means of the Cryptographic Information Application.

Table 7 — Structure of EF.SN

Tag	Length	Value
'5A'	'0A'	PAN, (BCD encoded) Filler bit with value 'F'

Throughout the present specification the following elements of the concatenation of the PAN and the filler bit are used:

- the rightmost 8 bytes are called smart card serial number SN.ICC and used in the context of the device authentication protocols;
- the complete concatenation of the PAN and the filler bit is used in the Card Verifiable certificate of a European Citizen Card as Certificate Holder Reference (CHR), see EN 14890-1:2008:2008, 14.3.

6.3 User verification

6.3.1 General aspects

A retry counter and an initial value of the retry counter may be defined for each reference data. If the verification fails, the retry counter is decremented by one and an error status that contains the remaining attempts is returned by the application.

When the retry counter equals zero, the reference data is blocked and can no longer be used. Note that a successful verification of the reference data resets its retry counter to the initial value.

The ECC supports a resetting code mechanism to unblock reference data, i.e. to reset the corresponding retry counter to its initial value and optionally to set new reference data. Whether resetting codes are supported or not and which command variant is supported may be defined by the security attributes of the application as indicated in the corresponding DF.CIA.

6.3.2 Knowledge based

To ensure interoperability:

- for the VERIFY command, the referenced data shall be encoded in the P2 parameter;
- for the CHANGE REFERENCE DATA command, the reference data shall be selected by a password reference encoded in the P2 parameter. Upon successful command execution the reference data inside the card is replaced by the new reference data provided in the command data field;
- for the RESET RETRY COUNTER, the referenced data shall be encoded in the P2 parameter.
- for the GENERAL AUTHENTICATE command the data objects shall be encoded as defined in EN 14890 when used in a password based protocol

NOTE Command flows for the VERIFY and the CHANGE REFERENCE DATA command can be found in EN 14890-1:2008:2008. Refer to A.7.5.8 for more information on the usage of the RESET RETRY COUNTER command.

6.3.3 Biometrics – on card matching

Please refer to 7.2.

6.4 Device authentication

6.4.1 General

The European Citizen Card shall at least implement one device authentication protocol (see table Table C.2). For interoperability reasons a terminal shall implement all protocol(s) referenced in the specific application or card profile(s) the terminal claims to support.

6.4.2 Authentication environments

Two kinds of environments shall be distinguished, by considering their security level: trusted and untrusted environments. During the personalisation phase, the issuer will define the access rules to the application resources. Different access rules can be defined for trusted and un-trusted environments. The terminal shall adapt the authentication to the security level required by the resource it wishes to access.

For definition of trusted/un-trusted environments, refer to EN 14890-1:2008:2008, 5.2 and 8.2.

6.4.3 Symmetric authentication scheme

The smart card implements the mutual authentication scheme using symmetric keys as defined in EN 14890-1:2008:2008, 8.8.

6.4.4 Asymmetric device authentication schemes

6.4.4.1 Card verifiable certificate verification commands

Card verifiable (CV) certificates are used in the context of asymmetric device authentication schemes to import a public key and related attributes into the European Citizen Card. The structure of the supported CV certificates and the card internal verification process is specified in EN 14890-1:2008:2008, 14.3. Details about possible certification hierarchies can be found in EN 14890-1:2008:2008, Clause 8, which defines the command flow used to verify a command chain.

6.4.4.2 Key transport protocol

The ICC makes use of the key transport protocol defined in EN 14890-1:2008:2008, 8.4 for an asymmetric device authentication service. This protocol makes use of card verifiable (CV) certificates (see EN 14890-1:2008:2008, 14.3 of this Technical Specification) in order to import public keys into the ICC. Details about possible certification hierarchies and the calculation of the cryptographic authentication tokens can be found in the referenced document.

Five main phases may be distinguished:

- 1) transport of the IFD public key to the ICC;
- 2) transport of the ICC public key to the IFD;
- 3) internal authentication of the ICC;
- 4) external authentication of the IFD;
- 5) session key derivation and establishment of a secure channel.

6.4.4.3 Device authentication with privacy protection

The device authentication with privacy protection scheme is based on EN 14890-1:2008:2008, 8.5. To avoid the card disclosing private information, such as identity, a secure channel session is established before any other operation. To do so, the protocol starts with an unauthenticated Diffie-Hellman key exchange and then authenticates the IFD before the ICC.

Five main phases may be distinguished:

- 1) agreement of session keys and establishment of a secure channel for privacy protection;
- 2) transport of the IFD public key to the ICC;
- 3) external authentication of the IFD;
- 4) transport of the ICC public key to the IFD;
- 5) internal authentication of the ICC.

6.4.4.4 Modular EAC (mEAC) protocol family

The modular EAC (mEAC) protocol is based on EN 14890-1:2008, 8.6. The protocol is constructed to allow for interchangeability of protocol modules depending on the use case.

Five main phases may be distinguished:

- 1) User consent;
 - i) Accomplished by organisational means
 - ii) Presenting a password to the card unsecured
 - iii) Using a password based protocol to establish a secured channel (recommended to prevent skimming and/or eavesdropping attacks)
- 2) transport of the IFD public key to the ICC;
- 3) external authentication of the IFD;
- 4) transport of the ICC public key to the IFD;
- 5) chip authentication with agreement of session keys and establishment of a secured channel.

6.4.5 Session key computation

Session keys are required for the secure messaging protocol. These keys are derived from $K_{ICC/IFD}$ during the device authentication protocol according to EN 14890-1:2008, 8.9.

6.4.6 Send Sequence Counter (SSC) initialisation

The Send Sequence Counter is a number that is initialised immediately after successful device authentication. Its length depends on the block size of the algorithm used in secure messaging. The SSC is defined, initialised and handled as specified in EN 14890-1:2008, 8.10.

6.4.7 Secure messaging

6.4.7.1 General

Secure messaging session establishment begins with an IFD/ICC mutual authentication. SM session keys are computed during this mutual authentication step. The ECC platform shall support secure messaging; whether it is used depends on the use case.

The following is mandatory:

- an SM session always includes data confidentiality AND integrity. Data confidentiality only, or data integrity only are not supported;
- SM policy for incoming data and outgoing data shall be the same.

6.4.7.2 SM Data Objects

For the description of the SM data objects refer to EN 14890-1:2008, 9.2.

6.4.7.3 Treatment of secure messaging errors

For the treatment of secure messaging errors refer to EN 14890-1:2008, 9.3

For ending a secure messaging session refer to EN 14890-1:2008, 8.12.2:

- status bytes of the card application may be sent along with SM DOs. In this case the SM session is not aborted by the card;
- status bytes other than defined above for SM errors may be sent without SM DO. In this case the behaviour of the card concerning the deletion of SM session keys is not defined by this Technical Specification.

6.4.7.4 Commands and responses under SM

For the message structure of secure messaging APDUs refer to EN 14890-1:2008, 9.6.

6.4.7.5 SM session breaking events

The SM session is handled according to EN 14890-1:2008, 8.12.2. Further exceptions are outlined in 5.5.1.

6.5 Digital signature

6.5.1 General

The European Citizen Card platform shall support at least one of the following two cryptographic algorithms for the digital signature service:

- RSA;
- ECDSA in accordance with TR-03111 using elliptic curves over prime fields $GF(p)$.

For further details refer to EN 14890-1:2008, Clause 7.

6.5.2 Hashing

Three modes for data hashing may be used:

- partial hashing with last round on card;
- hashing off card;
- hashing on card .

When the European Citizen Card supports partial data hashing algorithm with last round processed by the card the IFD calculates the hash over the beginning of the data to sign and the ICC ends by finishing the hash calculation on the last block. In addition, the card may support complete hashing of a byte string that is shorter than the block length of the hash algorithm.

The on card data hashing (complete and partial) is performed with the command PSO:HASH. Details on the PSO:HASH command can be found in A.7.6.2.

The result is temporarily stored in the card until the next command execution that is PSO:COMPUTE DIGITAL SIGNATURE.

For further description refer to EN 14890-1:2008, 7.3 and 7.4.

For a list of supported algorithms refer to EN 14890-1:2008, 13.2 Table 145.

6.5.3 Digital signature

The European Citizen Card shall support at digital signature creation.

Table B.10 — defines the algorithms and the corresponding algorithm identifier used in this specification. Table C.6 defines the algorithms and input formats that are mandatory and optional to support. In the case of ECDSA the digital signature is calculated according to TR-03111.

Details on the PSO:COMPUTE DIGITAL SIGNATURE command can be found in Annex A.

For further information refer to EN 14890-1:2008, 7.4 and 7.5.

6.6 Client/Server Authentication

6.6.1 General

The European Citizen Card platform shall support at least one of the following two cryptographic algorithms for the client/server authentication service:

- RSA;
- ECDSA in accordance with TR-03111 using elliptic curves over prime fields $GF(p)$.

The use case for the client/server authentication service is described in detail in EN 14890-2 Clause 6. In case of RSA the ICC computes a signature using a message digest as input according to EN 14890-2. For the client/server authentication, the INTERNAL AUTHENTICATE command is used.

6.7 Encryption key decipherment

6.7.1 General

The European Citizen Card platform shall support at least one of the following two cryptographic algorithms for the Encryption Key Decipherment service:

- RSA
- ECDH

The Encryption Key Decipherment service is described in detail in EN 14890-2, 8.2 for RSA and 8.3 for DH and ECDH.

6.7.2 Preceding steps

Preceding steps that are application specific, e.g. user verification, are performed. The precondition for the usage of the service may be defined in the corresponding Cryptographic Information Application.

If the IFD does not already have the ICC's Encryption Key Decipherment certificate denoted as C_X509.CH.KE, the IFD reads the certificate from the transparent file EF_C.CH.KE. The (short) EF identifier is provided by means of the corresponding Cryptographic Information Application.

7 Extended card services

7.1 General

The extended card services described here from functional standpoint relate either to the ICAO requirements or to other services. Unless indicated OPTIONAL, the following services shall be supported.

7.2 Biometrics – on card matching

7.2.1 General

The support of biometric user verification (on-card matching) is optional. On-card matching (MOC) is provided only as a service to access on-card data or functions. On-card matching can be used in the same way as a PIN to set a card internal security status. It shall not be used to set a card external security status (for physical access etc.). In the latter case some information is required by the IFD about the quality of the matcher.

7.2.2 General aspects

If this service is supported, the Finger Minutiae Matcher compliant with ISO/IEC 19794-2 shall be used. The on-card biometric comparison format identified by the CBEFF format type '0005' shall be applied as encoding format for finger minutiae.

The ICC shall provide the IFD with information about the supported biometric verification service using a Biometric Information Template (BIT) compliant with ISO/IEC 7816-11 that shall be encoded as presented in EN 14890-1:2008, 6.3.3.3

The enrolment process is out of the scope of the present specification.

Verification flow: For verification with sensor off-card refer to EN 14890-1:2008, 6.3.3.2

7.3 Passive Authentication

Passive Authentication enables a user of the ECC to verify that data elements stored on the ICC have not been altered by verifying a signature over these data elements stored in the ICC. By using a trusted public key for signature verification the data is passively authenticated also.

For further description refer to ICAO Doc 9303, Part 3, Volume 2, Section IV, 5.6.1.

7.4 Basic Access Control

The Basic Access Control provides a mechanism to require the terminal to know some data of the ECC before being able to communicate with the ICC. This data has to be retrieved optically from the document. Basic Access Control is used as an anti-skimming protection for contactless ICCs, it is assumed that a terminal, that can read the optical data is also allowed to communicate with the contactless ICC.

For further description refer to ICAO Doc 9303, Part 3, Volume 2, Section IV, 5.7.

7.5 Active Authentication

Active Authentication enables a user of the ECC to verify the authenticity of the ICC by using a challenge-response mechanism. The public key of the used ICC key pair has to be stored in a data element that is covered by Passive Authentication.

For further description refer to ICAO Doc 9303, Part 3, Volume 2, Section IV, 5.6.2.

7.6 Extended Access Control

Extended Access Control comprises two mechanisms:

- Terminal Authentication provides a mechanism for restricting access to ICC data based on a secure authentication using certificates and digital signatures.
- Chip Authentication verifies the authenticity of the ICC by using a Diffie-Hellman based mechanism and provides session keys for establishment of a secure channel. The public key of the used ICC key pair has to be stored in a data element that is covered by Passive Authentication.

Extended Access Control is used in the European passports to protect sensible data like biometric features (e. g. fingerprints).

For further details refer to TR-03110 [1].

7.7 Role authentication

7.7.1 General

This authentication is used to present a specific authorization to the ICC.

When an administrator wants to get authenticated to the ICC (to gain access rights) he has to prove it with a challenge-response mechanism. This mechanism shall be based on a secret key or certificate

The ICC may allow to authenticate several different administrators by this means, to grant different access rights to different entities, e.g.

- A remote entity entitled to update a file (e.g. address);
- A citizen card is unblocked at card issuance by an administrator.

The ECC supports two mechanisms:

- a symmetric role authentication (external authentication);
- an asymmetric role authentication.

7.7.2 Symmetric role authentication

7.7.2.1 General

The following procedure describes:

- the cryptographic operation that allow the authentication;
- the specification of the associated role in the card.

7.7.2.2 Protocol

Two phases may be distinguished:

- 1) Establishment of the security context by;
 - i) setting a respective value in the Security Environment or

- ii) setting the value in following command (Step 2)

2) External Authentication.

For details refer to EN 14890-2, 7.3

7.7.2.3 Role description

The card implicitly knows the role associated with the KeySet. Prior to granting privileges associated with the role, the card validates the cryptogram delivered by the IFD as described above.

The identification of the role allows limited operation on files as described in 5.6.

Where the IFD addresses an MSE SET command conveying a CRT AT template within which the KeySet reference is nested, the card-application implicitly knows the role associated with this KeySet.

7.7.3 Asymmetric role authentication

7.7.3.1 General

Public keys are transported through Card Verifiable Certificates (CVC) which is fully described in EN 14890-1:2008, 14.3.

7.7.3.2 Protocol based on RSA

Three phases may be distinguished:

- 1) verifying the certificate chain and simultaneously checking the role ID authorisation;
- 2) establishment of the security context by setting a respective value in the Security Environment;
- 3) external Authentication.

For details, refer to EN 14890-2, 7.4.1

7.7.3.3 Role description

The roles associated with the key are mapped in the CHA field, more precisely in the RoleID field. The whole CHA, including its RoleID part, has to be validated. For Role description details refer to EN 14890-1:2008, 14.7.5.

7.8 Restricted Identification (RI)

Restricted Identification provides a privacy feature for ID cards to identify without transferring specific personal data. It is based on a sector-specific identifier for the ECC on base of a static Diffie-Hellman Protocol with the following properties:

- within each sector the sector-specific identifier of the ECC is unique;
- across any two sectors, it is computationally impossible to link the sector-specific identifiers of any ECC.

The sector-specific identifier is used to (re-)identify the ECC within each sector.

The application-specific security requirements for the use of this service have to be respected.

NOTE Depending on the hash function used to create the sector-specific identifier, hash collisions may occur.

For details refer to EN 14890-2.

7.9 Age, Validity or Auxiliary Data Verification

To perform verification of application-specific authenticated auxiliary data (e. g. age verification or validity verification), these data should be made available to the card in a way ensuring its authenticity, either by sending it beforehand to the ECC or conveying it during the comparison.

Repeating comparison attempts should be delayed by the card, e. g. requiring a reauthentication of the terminal before repeating a comparison attempt on the same authenticated auxiliary data.

- For Age Verification the terminal has sent the required date of birth.
- For Document Validity Verification the terminal has sent the current date.

The COMPARE command (see ISO/IEC CD 7816-4:2010) may be used to verify authenticated auxiliary data, i.e. to perform age verification, document validity verification, or other comparison verification. Thereby the OID of the verification data if required may be sent in the command data.

For details refer to EN 14890-2.

NOTE If auxiliary data verification is used, the underlying personal data may be compromised by direct read access to the data. Therefore this access should be denied if auxiliary data verification is supported for the terminal.

7.10 Modular Enhanced Role Authentication (mERA)

This protocol is fundamentally a Role Authentication enhanced with additional capability to allow for both recording application-specific data on the card those (e.g. criteria list for access to a e-service) and reading application-specific data out of the card (e.g. card public key for Key Agreement). mERA provides a means for a terminal to authenticate to the card whereas preserving card private identification data from disclosure. mERA protocol is modular and can be unrolled two ways :

- Option A :
 - 1) the card derives terminal symmetric key;
 - 2) both card and terminal compute session keys for authentication;
 - 3) the terminal authenticates to the card and optionally delivers securely application-specific information to the card.
- Option B :
 - 1) the card derives terminal symmetric key;
 - 2) both card and terminal compute session keys for authentication;
 - 3) the terminal challenges the card with a random and recovers securely application-specific data from the card;
 - 4) the terminal presents its certificate to the card for verification;
 - 5) both card and terminal use static public key from certificate to compute shared secret;
 - 6) both card and terminal derive session key from the secret to establish secure messaging.

For details refer to EN 14890-2.

Annex A (normative)

Command set

A.1 CLASS byte coding

Table A.1 — Interindustry values of CLA

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	Support
0	0	0	X	-	-	-	-	Command chaining control	
			0	-	-	-	-	— The command is the last or only command of a chain	Mandatory
			1	-	-	-	-	— The command is not the last command of a chain	Mandatory
			-	X	X	-	-	Secure messaging indication	
			-	0	0	-	-	— No SM or no indication	Mandatory
			-	0	1	-	-	— Proprietary SM format	Out of scope
			-	1	0	-	-	— SM, command header not processed	Out of scope
			-	1	1	-	-	— SM, command header authenticated	Mandatory
			-	-	-	X	X	Logical channel number from zero to three	Optional

A.2 Command chaining mechanisms

If indicated for a command in the present clause, command chaining shall be supported compliant with ISO/IEC 7816-4 as follows:

- the smart card shall indicate the support of this mechanism in the historical bytes by means of the third software function table, see 5.3;
- the CLA byte of the last command of a chain shall be coded as '0X', whereas all preceding commands shall be coded as '1X', see A.1;
- bit b1 to b4 of the CLA byte shall be identical for all commands of a chain;
- if a command is aborted, the execution of command chaining is aborted altogether;
- if a command chaining command with CLA '1X' is followed by a reset or a command that is not part of the chain, command chaining is aborted;
- if the smart card does not respond with status words SW1-SW2 set to '90 00' to a command of a chain with CLA '1X', command chaining is aborted;
- every command of a chain evaluates access rules separately and is secured by secure messaging separately, if secure messaging is to be applied. Where secure messaging is applied, the Send Sequence Counter (SSC) is incremented prior to every MAC verification of every command of the chain and prior to every MAC calculation of every response of the chain.

A.2.1 Command chaining for oversized command payloads

Refer to ISO/IEC CD 7816-4:2010 section 5.2.1.

If the command chaining mechanism is used to transmit and oversized command payload:

- the INS byte shall be identical in all commands of a chain. If P1 and / or P2 shall be identical as well, this is defined in the specification of the command;

A.2.2 Command chaining for application-defined processes

Refer to ISO/IEC CD 7816-4:2010 section 5.2.1.

The command chaining mechanism can be used to build application-defined processes involving several consecutive command-response pairs that shall not be interrupted by other APDUs, e. g. for authentication protocols. In this case, the APDUs are grouped using command chaining. There is no constraint on the values of the INS, P1 and P2 bytes of these commands.

A.3 Extended length mechanism

A.3.1 Use cases

In the following use cases the European Citizen Card shall be able to handle more than 255 bytes command and/or 256 bytes response payload (acc. to card capability descriptor and data object '7F66' in section 5.3.3):

- usage of the PSO: COMPUTE DIGITAL SIGNATURE command with a sufficient key length;
- usage of INTERNAL AUTHENTICATE for client/server authentication with a sufficient key length;
- usage of READ BINARY with sufficient response data and secure messaging. If the terminal requests e.g. 256 bytes plain data by means of the Le = '00' byte;
- usage of GENERAL AUTHENTICATE and PSO:VERIFY CERTIFICATE with a sufficient key length;
- usage of MSE:SET KAT with a sufficient key length.

The following clauses define how these cases shall be treated.

A.3.2 Byte transmission protocol T = 0

The card makes use of the GET RESPONSE command with an interindustry value of the CLA byte '0x', see A.7.7.1.

The number of bytes available is indicated by the status words of the command that generated the response data, see A.6.

A.3.3 Byte transmission protocol T = 1

If the European Citizen Card indicates in the ATR/ATS that both protocol T=1 and extended length are supported, the IFD shall make use of extended length for the use cases defined in A.3.1.

The European Citizen Card behaviour for protocol T=1 and short APDU length fields as well as for protocol T=0 and short APDU length fields (A.3.2) in these use cases is according to this Technical Specification.

A.4 Logical channels

The European Citizen Card supports up to 4 logical channels as an optional feature. If the support of logical channels is indicated in the historical bytes of the ATR/ATS, the implementation shall be compliant with the present specification. Cards not supporting this mechanism shall make use of the basic channel with channel number zero only.

If the European Citizen Card supports logical channels, the number shall be indicated by means of the third software function table in the historical bytes of the ATR/ATS, see 5.3.

For referring to logical channels in the interindustry class the following rules that are compliant with ISO/IEC 7816-4 apply:

- the CLA byte encodes the channel number from 0 to 3 of the command-response pair, see A.1;
- the basic channel (channel number zero) shall be permanently available, i.e., it cannot be closed;
- after a RESET of the European Citizen Card the logical channels one to three are closed;
- channels other than the basic channel shall be opened by completion of the MANAGE CHANNEL command, see A.7.2.2, with open function;
- channels other than the basic channel shall be closed by completion of the MANAGE CHANNEL command, see A.7.2.2, with close function. After closing, the channel shall be available for re-use;
- only one channel shall be active at a time;
- the file descriptor byte defines for every EF or (Application) DF whether more than one channel may be opened to the same structure.

Each logical channel has its own security status. A security status shall not be shared between different logical channels.

A.5 Short and extended length fields

The handling of short and extended length fields shall be compliant with ISO/IEC 7816-4:

If the European Citizen Card states its capability of handling extended L_C and L_E fields in the third software function table of the historical bytes, then the card handles both extended and short length fields. Otherwise the card handles short length field only. In addition, a card supporting extended length fields shall indicate the maximum number of bytes expected in the command data field by means of the EF.ATR/INFO, see 5.3.3.

In any command response pair comprising both L_C and L_E fields short and extended length fields shall not be combined: either both of them are short or both of them are extended.

A.6 Status words

In general status words shall be compliant with ISO/IEC 7816-4, but the actual values used are out of the scope of the document if not listed in the command description of this annex. In the command description only status words used in the user verification service with a meaning for the cardholder are mandated.

The following status words shall be used in the Secure Messaging protocol compliant with ISO/IEC 7816-4, EN 14890-1:2008 and the ICAO PKI report:

- '69 87' – Expected Secure Messaging data objects missing;

— '69 88' – Secure Messaging data objects incorrect;

If the European Citizen Card hosts an MRTD application, the status words listed in the ICAO PKI report shall be supported in this application.

If byte transmission protocol $T = 0$ is applied the number of bytes to be retrieved by one or more consecutive GET RESPONSE commands shall be indicated by the status words:

- '61 00' if more than 255 bytes are available;
- '61 xx' where 'xx' indicates the number of bytes available.

A.7 Command set

A.7.1 General

A.7.2 Commands for selection

A.7.2.1 SELECT

The SELECT command can be used to open logical channels and/or to set a current structure (e. g. an EF) within the logical channel. Subsequent commands may implicitly refer to the current structure through the logical channel.

The following tables define the variants of the SELECT command that are mandatory, optional and conditional to support.

Table A.2 — SELECT command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'A4'
P1	See Table A.3 — P1
P2	See Table A.4 — P2
L_C field	Absent or length of command data field
Data field	Absent or file identifier or DF name (according to P1)
L_e field	Present for encoding $N_e > 0$

Response Parameter	Meaning
Data field	Absent or file control information
SW1-SW2	See ISO/IEC 7816-4

The parameter P1 = '00' shall be supported if the card indicates in the ATR/ATS that an MF is available.

The parameter P1 = '09' shall be supported if the card indicates in the ATR/ATS that no MF is available.

Table A.3 — P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	Command data field	Support
0	0	0	0	0	0	X	X	Selection by file identifier		
				0	0	0	0	— Select MF	Absent	conditional
				0	0	0	1	— Select child DF	DF identifier	mandatory
				0	0	1	0	— Select EF under the current DF	EF identifier	mandatory
				0	0	1	1	— Select parent DF of the current DF	Absent	mandatory
				0	1	X	X	Selection by DF name		
				0	1	0	0	— Select by DF name	AID	mandatory
				1	0	X	X	Selection by path		
				1	0	0	1	— Select from the current DF	Path without the current DF identifier	conditional

Table A.4 — P2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	Support
0	0	0	0	-	-	X	X	File occurrence	
				-	-	0	0	— First or only occurrence	Mandatory
				X	X	-	-	File control information	
				0	0	-	-	— Return FCI template	Optional
				0	1	-	-	— Return FCP template	Mandatory
				1	1	-	-	— No response data, L _e byte absent	Mandatory

A.7.2.2 MANAGE CHANNEL

The support of the MANAGE CHANNEL command is conditional. If the card indicates in the historical bytes of the ATR/ATS or EF.ATR/INFO that logical channels are supported, the implementation shall be compliant with the present specification.

When completed, the command opens or closes a logical channel other than the basic one, i.e. a channel numbered from one to three.

The open function opens a logical channel other than the basic one. The channel number is assigned by the card and provided in the response data field, see Table A.5 — MANAGE CHANNEL command-response pair. After an open function is performed from the basic logical channel the root is selected, after an open function performed from a non-basic logical channel, the current DF on the channel numbered in CLA shall become current on the new channel. If the file descriptor byte of a DF indicates that the DF is not shareable, no further logical channel can be opened in this DF.

The close function closes a logical channel other than the basic one. After closing the channel shall be available for re-use. The channel to be closed is denoted in the P2 parameter byte, see Table A.5 — MANAGE CHANNEL command-response pair.

Table A.5 — MANAGE CHANNEL command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'70'
P1-P2	'00 00' for opening a logical channel to be numbered in the response data field '80 01' to '80 03' for closing the logical channel numbered in P2
L _C field	Absent
Data field	Absent
L _e field	Present for encoding N _e > 0

Response Parameter	Meaning
Data field	P1-P2 = '00 00': '01' to '03' else: absent
SW1-SW2	See ISO/IEC 7816-4

A.7.3 Commands for data unit handling

A.7.3.1 General

All commands of this group that make use of an even instruction code shall use P1-P2 as defined in Table A.6 — P1 P2 for data unit handling commands (even instruction code).

Table A.6 — P1 P2 for data unit handling commands (even instruction code)

P1								P2							
b8	b7	b6	b5	b4	b3	b2	b1	b8	b7	b6	b5	b4	b3	b2	b1
0	Offset in the currently selected file over 15 bits '00' ≤ Offset ≤ '7F FF'														
1	0	0	Short File Identifier 1 ≤ SFI ≤ 30					Offset in the file over 8 bits							

All commands of this group that make use of an odd instruction code shall use P1-P2 as follows:

- if bits 8 to 1 of P1 and bits 8 to 6 of P2 are set to zero and bits 5 to 1 of P2 are not all equal, then bits 5 to 1 of P2 encode a short EF identifier;
- if bits 8 to 1 of P1 and bits 8 to 6 of P2 are not all set to zero, then P1-P2 encodes a file identifier;
- if P1-P2 = '00 1F', then P1-P2 encodes a file identifier;
- P1-P2 set to '00 00' identifies the current EF.

NOTE If an odd instruction code is used, EFs with a file identifier '00 01' to '00 1E' and no short file identifier can only be addressed as current EF in P1-P2.

The offset Data Object shall be encoded as follows: '54' – '02' – {Offset encoded in two bytes}.

A.7.3.2 READ BINARY

The READ BINARY command can be used to request contents of an EF supporting data units. Depending on how the L_e field is set, also only a part of the content may be returned.

The support of the READ BINARY command with an even instruction code is mandatory.

Table A.7 — READ BINARY command-response pair (even instruction code)

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'B0'
P1-P2	See Table A.6 — P1 P2 for data unit handling commands (even instruction code)
L_e field	Present for encoding $N_e > 0$

Response Parameter	Meaning
Data field	Data read
SW1-SW2	See ISO/IEC 7816-4

The support of the READ BINARY command with an odd instruction code is mandatory.

Table A.8 — READ BINARY command-response pair (odd instruction code)

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'B1'
P1-P2	See A.7.3.1
L_c field	Length of command data field, '04'
Data field	Offset Data Object with tag '54'
L_e field	Present for encoding $N_e > 0$

Response Parameter	Meaning
Data field	Discretionary data object with tag '53'
SW1-SW2	See ISO/IEC 7816-4

As the ICC does not process the response data, the discretionary data object with tag '53' shall always be used in the response data field.

In this case the discretionary data object '53' – '7F' – {data read from the file} or '53' – '81 7F' – {data read from the card} shall be returned.

A.7.3.3 UPDATE BINARY

The UPDATE BINARY command initiates the update of data already present in an EF with the data given in the command data field.

The support of the UPDATE BINARY command with an even instruction code is mandatory.

Table A.9 — UPDATE BINARY command-response pair (even instruction code)

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'D6'
P1-P2	See Table A.6 — P1 P2 for data unit handling commands (even instruction code)
L _c field	Length of command data field
Data field	String of data units to be updated
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	See ISO/IEC 7816-4

The support of the UPDATE BINARY command with an odd instruction code is mandatory.

Table A.10 — UPDATE BINARY command-response pair (odd instruction code)

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'D7'
P1-P2	See A.7.3.1
L _c field	Length of command data field
Data field	Offset data object with tag '54' discretionary data object with tag '53' ^a
L _e field	Absent

^a The use of the data object '73' is deprecated in the command data field acc. to ISO/IEC 7816-4.

Response Parameter	Meaning
Data field	Absent
SW1-SW2	See ISO/IEC 7816-4

A.7.4 Commands for data object handling

A.7.4.1 GET DATA

The GET DATA command can be used to retrieve data objects or data elements from the ICC.

The support of the GET DATA command is conditional. The corresponding command variant shall be supported:

- if the device authentication protocol with privacy protection, see 6.4.4.3, is supported;
- if the privacy constrained device authentication with non traceability based on ELC, see 6.4.4.4, is supported;
- if a biometric user verification, see 7.2, is supported.

Table A.11 — GET DATA command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'CA' or 'CB', see Table A.12 — GET DATA command variants
P1-P2	see Table A.12 — GET DATA command variants
L _C field	length of command data field or absent
Data field	see Table A.12 — GET DATA command variants
L _e field	Present for encoding N _e > 0

Response Parameter	Meaning
Data field	see Table A.12 — GET DATA command variants
SW1-SW2	See ISO/IEC 7816-4

Table A.12 — GET DATA command variants

Service	INS	P1-P2	Command data field	Response data field
Device Authentication with privacy protection	'CB'	'3F FF' (current DF)	Extended header list containing a KAT referencing K _{ICC} '4D 04 A6 02 91 00'	KAT containing K _{ICC} 'A6' L _{A6} ('91' L ₉₁ K _{ICC})
Privacy constrained device authentication with non traceability based on ELC	'CA'	'7F 49' (public key)	Absent	'86' – L ₈₆ - PuK.ICC.KA
Biometric user verification	'CA'	'7F 60' (Biometric Information Template)	Absent	Biometric Information Template (BIT)

A.7.5 Commands for basic security handling

A.7.5.1 INTERNAL AUTHENTICATE

The INTERNAL AUTHENTICATE command initiates the computation of authentication data by the ICC using the challenge data sent by the interface device and a relevant secret (e.g. a key) stored in the card e. g. for the purpose of proving the authenticity of the ICC.

The support of the INTERNAL AUTHENTICATE command is mandatory for the IFD. The command is used in device authentication protocols as well as in the client/server authentication service. In general it is used for authentication of the ICC using a challenge-response mechanism e.g. for Active Authentication (see 7.5), in the Key transport protocol (see 6.4.4.2), in the Device authentication with privacy protection (see 6.4.4.3) or in the Modular Enhanced Role Authentication (mERA) (see 7.10). The command shall at least support one of the device authentication protocols.

With respect to the CSA service the European Citizen Card shall support at least one of the following two cryptographic algorithms:

- RSA;
- ECDSA in accordance with TR-03111

Table A.13 — INTERNAL AUTHENTICATE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'88'
P1	'00'
P2	'00'
L _c field	Length of command data field
Data field	Authentication related data
L _e field	Present for encoding N _e > 0

Response Parameter	Meaning
Data field	Authentication related data
SW1-SW2	See ISO/IEC 7816-4

A.7.5.2 GET CHALLENGE

The GET CHALLENGE command returns a random challenge e. g. for use in a security related procedure.

The support of the GET CHALLENGE command is mandatory.

Table A.14 — GET CHALLENGE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'84'
P1-P2	'00 00'
L _e field	See ISO/IEC 7816-4

Response Parameter	Meaning
Data field	Challenge
SW1-SW2	See ISO/IEC 7816-4

A.7.5.3 EXTERNAL AUTHENTICATE

The EXTERNAL AUTHENTICATE command updates the security status using the result (yes or no) of the computation by the card based on a challenge previously issued by the ICC (e.g. by a GET CHALLENGE command), e. g. during an authentication protocol.

The support of the EXTERNAL AUTHENTICATE command for at least one of the asymmetric device authentication protocols is mandatory. The support of the EXTERNAL AUTHENTICATE command for the role authentication is optional.

Table A.15 — EXTERNAL AUTHENTICATE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'82'
P1	'00'
P2	'00'
L _c field	Length of command data field
Data field	Authentication related data
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	See ISO/IEC 7816-4

A.7.5.4 MUTUAL AUTHENTICATE

The MUTUAL AUTHENTICATE command updates the security status using the result (yes or no) of the computation by the card based on a challenge previously issued by the ICC (e.g. by a GET CHALLENGE command), e. g. during an authentication protocol.

The support of the MUTUAL AUTHENTICATE command is mandatory for the IFD. The command is used in the symmetric device authenticate service.

Table A.16 — MUTUAL AUTHENTICATE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'82'
P1	'00'
P2	'00'
L _c field	Length of command data field
Data field	Authentication related data
L _e field	Present for encoding N _e > 0

Response Parameter	Meaning
Data field	Authentication related data
SW1-SW2	See ISO/IEC 7816-4

A.7.5.5 GENERAL AUTHENTICATE

The GENERAL AUTHENTICATE command can be used to perform authentication procedures, e. g. a multi-step authentication protocol.

The support of the GENERAL AUTHENTICATE command is mandatory if the asymmetric device authentication protocols "Device authentication with privacy protection" (see 6.4.4.3) or "Modular EAC (mEAC) protocol" (see 6.4.4.4) is supported.

Table A.17 — GENERAL AUTHENTICATE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'86' ^a or '87'
P1	'00'
P2	'00'
L _C field	Length of command data field
Data field	Authentication related data
L _e field	Present for encoding N _e > 0
^a For even INS code the data objects contained in the command and response data fields will be aligned the upcoming revised CD version of ISO/IEC 7816-4 as soon as it is available.	

Response Parameter	Meaning
Data field	Authentication related data
SW1-SW2	See ISO/IEC 7816-4

When present for GENERAL AUTHENTICATE command with odd INS code, each data field shall contain an interindustry template referenced by tag '7C'. In the dynamic authentication template, the context-specific class (first byte from '80' to 'BF') is reserved for dynamic authentication data objects as listed in Table A.18le 71.

Table A.18 — GENERAL AUTHENTICATE dynamic authentication data objects

Tag	Value
'7C'	Set of dynamic authentication data objects with the following tags
'80'	Witness (e.g., one or more positive numbers less than the public modulus in use)
'81'	Challenge (e.g., one or more numbers, possibly zero, less than the public exponent in use)
'82'	Response (e.g., one or more positive numbers less than the public modulus in use)
'83'	Committed challenge (e.g., the hash-code of a large random number including one or more challenges)
'84'	Authentication code (e.g., the hash-code of one or more data fields and a witness data object)
'85'	Exponential (e.g., a positive number for establishing a session key by a key agreement technique)
'86'	Encrypted data
'A0'	Identification data template
In this context, ISO/IEC JTC 1/SC 17 reserves any other data object of the context-specific class (first byte from '80' to 'BF').	

The following rules apply within the interindustry template for dynamic authentication:

- If a data object is empty in a template, then it shall be complete in the template in the next data field.

- In the first command data field, the template indicates the dynamic authentication function as follows.
 - A witness request, e.g., an empty witness, or an authentication request, e.g., an empty authentication code, denotes an INTERNAL AUTHENTICATE function.
 - A challenge request, e.g., an empty challenge, denotes an EXTERNAL AUTHENTICATE function.
 - The absence of empty data object denotes a MUTUAL AUTHENTICATE function. Then unless the card aborts the process, the template in the response data field shall contain the same data objects as the template in the command data field. The MUTUAL AUTHENTICATE function allows two entities to agree on a session key using a pair of “exponential” data elements referenced by tag '85' (see key agreement techniques in ISO/IEC 11770-3 [12]).

The dynamic authentication may protect data fields exchanged during a session. Both entities maintain a current hash-code, updated by including one command or response data field at a time. The data object with tag '84' conveys an authentication code resulting from updating the current code by including a witness data object with tag '80'. The verifier successively reconstructs a witness and an authentication code: if the reconstructed witness is not zero and if the two codes are identical, then the authentication is successful.

A.7.5.6 VERIFY

The VERIFY command initiates the comparison in the ICC of stored reference data with verification data sent from the interface device (e.g. password) or from a sensor on the card (e.g. fingerprint). The security status may be modified as a result of a comparison. The card may record unsuccessful comparisons (e.g. to limit the number of further uses of the reference data).

The support of the VERIFY command with an even instruction code for a knowledge based user verification is mandatory.

Table A.19 — VERIFY command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'20'
P1	'00'
P2	see Table A.22 — P2
L _c field	Length of command data field
Data field	Verification data
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	'63 CX' – Authentication failed, 'X' attempts remaining '69 83' – Authentication method blocked, i.e. the corresponding retry counter equals zero Further status words: see ISO/IEC 7816-4

The support of the VERIFY command with an odd instruction code is conditional. This command variant shall be applied if a biometric verification (on-card matching) is performed.

Table A.20 — VERIFY command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'21'
P1	'00'
P2	see Table A.22 — P2
L _C field	Length of command data field
Data field	Biometric data template, see Table A.21 — Biometric data template
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	'63 CX' – Authentication failed, 'X' attempts remaining '69 83' – Authentication method blocked, i.e. the corresponding retry counter equals zero Further status words: see ISO/IEC 7816-4

Table A.21 — Biometric data template

Tag	Length	Value
'7F 2E'	Var.	Biometric Data Template
		Tag Length Value
		'81' Var. Biometric data with standardised format (primitive encoding)

Table A.22 — P2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning	Support
0	0	0	0	0	0	0	0	No information given	Optional
0	-	-	-	-	-	-	-	Global reference data	Mandatory
1	-	-	-	-	-	-	-	Local reference data (DF-specific)	Mandatory
-	0	0	-	-	-	-	-	RFU	
-	-	-	X	X	X	X	X	Qualifier, i.e. number of reference data	Optional
-	-	-	-	-	X	X	X	Qualifier, i.e. number of reference data	Mandatory
-	-	-	-	-					
- P2 = '00' denotes an implementation specific default value									

A.7.5.7 CHANGE REFERENCE DATA

The CHANGE REFERENCE DATA command either replaces reference data stored in the card with new reference data sent from the interface device, or initiates their comparison with verification data sent from the interface device and then conditionally replaces them with new reference data sent from the interface device.

The support of the CHANGE REFERENCE DATA command variant with P1 = '00' by the card platform is mandatory, the support of the P1 = '01' variant is optional. The command variant actually supported for specific reference data of an application shall be indicated by means of the Cryptographic Information Application.

Table A.23 — CHANGE REFERENCE DATA command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'24'
P1	'00' (Mandatory) or '01' (optional)
P2	see Table A.22 — P2
L _c field	Length of command data field
Data field	P1='00' data field is composed by the verification followed by the new reference data P1='01' data field is composed only by the new reference data
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	'63 CX' – Authentication failed, 'X' attempts remaining '69 83' – Authentication method blocked, i.e. the corresponding retry counter equals zero Further status words: see ISO/IEC 7816-4

A.7.5.8 RESET RETRY COUNTER

The RESET RETRY COUNTER command either resets the reference data retry counter to its initial value, or changes reference data on completion of a reset of the reference data retry counter to its initial value.

The support of the RESET RETRY COUNTER command is mandatory.

Table A.24 — RESET RETRY COUNTER command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'2C'
P1	'02' or '03'
P2	see Table A.22 — P2
L _C field	P1 = '02': Length of command data field P1 = '03': Absent
Data field	P1 = '02': New reference data P1 = '03': Absent
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	See ISO/IEC 7816-4

In the case of a PIN for a qualified electronic signature shall be reset using RESET RETRY COUNTER the following flow shall be used in order to reset the retry counter: A VERIFY command is used to provide the corresponding resetting code followed by RESET RETRY COUNTER command. In the case of P1 = '02' this command sets new reference data for the signature PIN. For security reasons both commands shall be used with secure messaging.

A.7.5.9 MANAGE SECURITY ENVIRONMENT

The command prepares secure messaging and security commands (e.g. EXTERNAL, INTERNAL and GENERAL AUTHENTICATE or PERFORM SECURITY OPERATION).

The following variants of the MANAGE SECURITY ENVIRONMENT are mandatory to support:

- MSE: RESTORE;
- MSE: SET.

Table A.25 — MSE:RESTORE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'22'
P1	'F3' – RESTORE
P2	SEID byte (≠ '00', 'EF', 'FF')

NOTE The usage of the MSE: RESTORE command has not been included in the command flows of the services specified in this Technical Specification. This may lead to interoperability problems. The Security Environments that are

supported by an application as well as the corresponding access conditions for a service are listed by the Cryptographic Information Application.

Table A.26 — MSE:SET command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'22'
P1-P2	See Table A.27 — MSE:SET command variants
L _C field	Length of command data field
Data field	Control Reference Template, see Table A.27 — MSE:SET command variants
L _e field	Absent

Response Parameter	Meaning
Data field	Absent
SW1-SW2	See ISO/IEC 7816-4

Table A.27 — MSE:SET command variants lists the P1-P2 parameter combinations for the MSE:SET command as well as the data objects present in the command data field for the different services. The order of the data objects in the command data field is arbitrary. Every data object shall be present in the command data field exactly once and further data objects shall not be present. Whether the support of an MSE:SET command variant is mandatory or not depends on the service, see Clause 6.

Table A.27 — MSE:SET command variants (1 of 2)

P1	P2	Data Objects	Description
'41'	'A4'	'80' – Algorithm reference '84' – Private key reference	Client/Server Authentication: SET for INTERNAL AUTHENTICATE
'41'	'A4'	'80' – Algorithm reference '83' – Public key reference	Device authentication with privacy protection: SET for GENERAL AUTHENTICATE
'41'	'A4'	'80' – Algorithm reference '84' – Private key reference	Device authentication with privacy protection: SET for INTERNAL AUTHENTICATE
'41'	'A6'	'80' – Algorithm reference '83' – Public key reference '91' – Random number	Device authentication with privacy protection (Alternative Execution Flow ^a): SET for GET DATA (key agreement)
'41'	'A4'	'80' – Algorithm reference '84' – Private key reference	Modular EAC (mEAC) protocol family: SET for GENERAL AUTHENTICATE
'41'	'AA'	'80' – Algorithm reference	Digital signature Service: SET for PSO:HASH

Table A.27 — MSE:SET command variants (2 of 2)

'41'	'B6'	['80' – Algorithm reference] '84' – Private key reference	Digital signature Service: SET for PSO: COMPUTE DIGITAL SIGNATURE
'41'	'B6'	'80' – Algorithm reference '84' – Private key reference	Client/Server Authentication (Alternative Execution Flow): SET for INTERNAL AUTHENTICATE
'41'	'B8'	'80' – Algorithm reference '84' – Private key reference	Encryption key decipherment Service: SET for PSO: DECIPHER
'81'	'A4'	'80' – Algorithm reference '83' – Public key reference	Role authentication, Asymmetric role authentication (asymmetric): SET for EXTERNAL AUTHENTICATE
'81'	'A4'	'83' – Public key reference ['67' – Auxiliary data]	Device authentication with privacy protection: SET for EXTERNAL AUTHENTICATE
'81'	'A4'	'80' – Algorithm reference '83' – Public key reference '91' – hash value of the public key of the IFD ['67' – Auxiliary data]	Modular EAC (mEAC) protocol family: SET for EXTERNAL AUTHENTICATE
'81'	'A4'	'80' – Algorithm reference '83' – Secret key reference	Role authentication, Symmetric : SET for EXTERNAL AUTHENTICATE
'81'	'B6'	'83' – Public key reference	Modular EAC (mEAC) protocol family: SET for PSO: VERIFY CERTIFICATE
'81'	'B6'	'83' – Public key reference	Device authentication with privacy protection, SET for PSO: VERIFY CERTIFICATE for CA root key
'81'	'B6'	'80' – Algorithm reference '83' – Public key reference	Device authentication with privacy protection, : SET for PSO: VERIFY CERTIFICATE of IFD's CA key
'81'	'B6'	'80' – Algorithm reference '83' – Public key reference	Key transport protocol: SET for PSO: VERIFY CERTIFICATE
'C1'	'A4'	'80' – Object identifier value for PBM '83' – Reference of secret key, representing the selected password '84' – Reference of domain parameters ['7F4C'] – CHAT for following terminal authentication	Modular EAC (mEAC) protocol family: SET for GENERAL AUTHENTICATE for performing a password- based authentication (PBM)
'C1'	'A4'	'80' – Algorithm reference '83' – Public key reference '84' – Private key reference	Key transport protocol: SET for INTERNAL and EXTERNAL AUTHENTICATE

^a This execution flow is also used in the field for future cards it is recommended to use GENERAL AUTHENTICATE. The IFD has to support both methods.

A.7.6 Commands for security operations

A.7.6.1 PSO:COMPUTE DIGITAL SIGNATURE

The PSO:COMPUTE DIGITAL SIGNATURE command initiates the computation of a digital signature. The algorithm may be either a digital signature algorithm or a combination of a hash algorithm and a digital signature algorithm.

The support of the PSO:COMPUTE DIGITAL SIGNATURE command is mandatory. The European Citizen Card shall support at least one of the following two cryptographic algorithms for digital signature generation:

- RSA;
- ECDSA in accordance with TR-03111. The signature format shall be returned in plain format, i.e. $r || s$.

Table A.28 — PSO:COMPUTE DIGITAL SIGNATURE command-response pair

Command Parameter	Meaning
CLA	'0x', see Table A.1 — Interindustry values of CLA
INS	'2A'
P1	'9E'
P2	'9A'
L_C field	Length of command data field for off-card hashing, Absent otherwise
Data field	Hash value for off-card hashing, Absent otherwise
L_e field	Present for encoding $N_e > 0$

Response Parameter	Meaning
Data field	Digital Signature
SW1-SW2	See ISO/IEC 7816-8

A.7.6.2 PSO:HASH

The PSO:HASH command initiates the computation of a hash-code by performing either the complete computation inside the card or a partial computation inside the card (e.g. last round of hashing).

The support of the PSO:HASH command variant that performs the last round of the hash computation is mandatory. The resulting hash value is stored in the ICC and used by a PSO:COMPUTE DIGITAL SIGNATURE command that shall immediately follow the PSO: HASH command. Command flows that are not compliant with this order and the reset of the stored hash value are out of the scope of this Technical Specification.

If an implementation supports further variants of the PSO:HASH command for national applications that are not subject to European interoperability, it is highly recommended that the implementation is compliant with EN 14890-1:2008.

Table C.5 defines which hash algorithms are mandatory and optional to support.