

BS EN 419212-1:2014



BSI Standards Publication

Application Interface for smart cards used as Secure Signature Creation Devices

Part 1: Basic services

bsi.

...making excellence a habit.
AVA Documents™
Document Sharing Hub

National foreword

This British Standard is the UK implementation of EN 419212-1:2014. It supersedes BS EN 14890-1:2008 which is withdrawn.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 2014. Published by BSI Standards Limited 2014

ISBN 978 0 580 77109 5

ICS 35.240.15

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

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 31 December 2014.

Amendments issued since publication

Date	Text affected
------	---------------

EUROPEAN STANDARD
NORME EUROPÉENNE
EUROPÄISCHE NORM

EN 419212-1

December 2014

ICS 35.240.15

Supersedes EN 14890-1:2008

English Version

Application Interface for smart cards used as Secure Signature Creation Devices - Part 1: Basic services

Interface applicative des cartes à puces utilisées comme
dispositifs de création de signature numérique sécurisés -
Partie 1 : Services de base

Anwendungsschnittstelle für Chip-Karten, die zur
Erzeugung qualifizierter elektronischer Signaturen
verwendet werden - Teil 1: Allgemeine Dienste

This European Standard was approved by CEN on 27 September 2014.

CEN members are bound to comply with the CEN/CENELEC Internal Regulations which stipulate the conditions for giving this European Standard the status of a national standard without any alteration. Up-to-date lists and bibliographical references concerning such national standards may be obtained on application to the CEN-CENELEC Management Centre or to any CEN member.

This European Standard exists in three official versions (English, French, German). A version in any other language made by translation under the responsibility of a CEN member into its own language and notified to the CEN-CENELEC Management Centre has the same status as the official versions.

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

CEN-CENELEC Management Centre: Avenue Marnix 17, B-1000 Brussels

Contents

Page

Foreword.....	7
Introduction	9
1 Scope	10
2 Normative references	10
3 Terms and definitions	11
4 Symbols and abbreviations	15
5 Signature application	18
5.1 Application Flow	18
5.2 Trusted environment versus untrusted environment	22
5.3 Selection of E-SIGN application	22
5.3.1 General.....	22
5.3.2 Exceptions for Secure Messaging	23
5.4 Selection of cryptographic information application	23
5.5 Concurrent usage of signature applications	24
5.5.1 General.....	24
5.5.2 Methods of channel selection	24
5.5.3 Security issues on multiple channels	24
5.6 Security environment selection	24
5.7 Key selection.....	25
5.8 Security Services	25
6 User verification.....	26
6.1 General.....	26
6.2 Knowledge based user verification	26
6.2.1 General.....	26
6.2.2 Explicit user verification	27
6.2.3 Password based mechanisms.....	28
6.2.4 Presentation formats	28
6.2.5 Retry and Usage counters	28
6.2.6 Password Change.....	29
6.2.7 Reset of RC and setting a new password	29
6.3 Biometric user verification	30
6.3.1 General.....	30
6.3.2 Retrieval of the Biometric Information Template	31
6.3.3 Performing the biometric user verification	32
6.3.4 Reset of RC.....	34
7 Digital Signature Service	34
7.1 General.....	34
7.2 Signature generation algorithms	35
7.3 Activation of digital signature service.....	35
7.4 General aspects	36
7.5 Signature Generation	37
7.5.1 General.....	37
7.5.2 No hashing in Card	37
7.5.3 Partial hashing	38
7.5.4 All hashing in ICC	39
7.6 Selection of different keys, algorithms and input formats	40
7.6.1 General.....	40
7.6.2 Restore an existing SE.....	41
7.6.3 Setting the Hash Template (HT) of a current Security Environment (SE).....	42
7.6.4 Modify the Digital Signature Template (DST) of a current Security Environment (SE)	42
7.7 Read certificates and certificate related information.....	43

7.7.1	General	43
7.7.2	Read certificate related CIOs.....	43
7.7.3	Read signer's certificate from ICC	44
7.7.4	Retrieval of the signer's certificate from a directory service.....	44
8	Device authentication	45
8.1	General	45
8.2	Asymmetric Authentication introduction.....	46
8.3	Certification authorities and certificates.....	46
8.3.1	Certificate chains.....	46
8.3.2	Usage of link certificates	47
8.4	Authentication environments.....	48
8.4.1	General	48
8.4.2	SCA in trusted environment.....	48
8.4.3	SCA in untrusted environment	48
8.4.4	Specification of the environment.....	49
8.4.5	Display message mechanism	49
8.4.6	Additional authentication environments.....	49
8.5	Key transport and key agreement mechanisms	49
8.6	Key transport protocol based on RSA	50
8.6.1	General	50
8.6.2	Authentication Steps.....	52
8.6.3	Session Key creation	62
8.7	Device authentication with privacy protection.....	63
8.7.1	General	63
8.7.2	Authentication steps	63
8.8	Privacy constrained Modular EAC (mEAC) protocol with non-traceability feature.....	82
8.8.1	General	82
8.8.2	Example for traceability case	83
8.8.3	Notation	83
8.8.4	Authentication steps	84
8.8.5	Unlinkability Mechanism with individual private keys	99
8.9	Symmetric authentication scheme	108
8.9.1	General	108
8.9.2	Authentication steps	108
8.9.3	Session Key creation	112
8.10	Compute Session keys from key seed $K_{IFD/ICC}$	113
8.10.1	General	113
8.10.2	Generation of key data.....	113
8.10.3	Partitioning of the key data	113
8.10.4	Algorithm and method specific definition for key derivation	113
8.10.5	Key derivation from passwords.....	116
8.11	Compute send sequence counter SSC	118
8.12	Post-authentication phase.....	118
8.13	Ending the secure session	119
8.13.1	General	119
8.13.2	Example for ending a secure session	119
8.13.3	Rules for ending a secure session	119
8.14	Reading the Display Message.....	119
8.15	Updating the Display Message	122
9	Password-based authentication protocols.....	123
9.1	General	123
9.2	Notation	123
9.3	Authentication steps	124
9.3.1	General	124
9.3.2	Step 1 — Reading the protocol relevant public parameters	125
9.3.3	Step 2 — Set PBM parameters and generate blinding point.....	127
9.3.4	Step 3 — Get encrypted nonce	128
9.3.5	Step 4.1 — Map nonce and compute generator point for generic mapping	129
9.3.6	Step 4.2 — Map nonce and compute generator point for integrated mapping.....	130
9.3.7	Step 5 — Generate session keys	133

9.3.8	Step 6 — Explicit key authentication	134
10	Secure Messaging	135
10.1	General.....	135
10.2	CLA byte	135
10.3	TLV coding of command and response message	135
10.4	Treatment of SM-Errors.....	136
10.5	Padding for checksum calculation	136
10.6	Send sequence counter (SSC)	136
10.7	Message structure of Secure Messaging APDUs.....	136
10.7.1	Cryptograms.....	136
10.7.2	Cryptographic Checksums	139
10.7.3	Final command APDU construction	143
10.8	Response APDU protection	143
10.9	Use of TDES and AES	150
10.9.1	TDES/AES encryption/decryption	150
10.9.2	CBC mode.....	151
10.9.3	Retail MAC with TDES	151
10.9.4	EMAC with AES.....	152
10.9.5	CMAC with AES.....	154
11	Key Generation	155
11.1	General.....	155
11.2	Key generation and export using PrK.ICC.AUT	155
11.3	Key generation and export with SM	155
11.4	Write certificates	156
12	Key identifiers and parameters	156
12.1	General.....	156
12.2	Key identifiers (KID).....	156
12.2.1	General.....	156
12.2.2	Secret and private keys.....	156
12.3	Public Key parameters	156
12.3.1	General.....	156
12.3.2	RSA public key parameters	157
12.4	Diffie-Hellman key exchange parameters	157
12.5	Authentication tokens in the protocols mEACv2 and PCA.....	157
12.5.1	General.....	157
12.5.2	TDES	157
12.5.3	AES.....	157
12.5.4	Ephemeral Public Key Data Object	157
12.6	The compression function Comp()	158
12.7	DSA with ELC public key parameters.....	158
12.7.1	General.....	158
12.7.2	The plain format of a digital signature	159
12.7.3	The uncompressed encoding.....	159
12.8	ELC key exchange public parameters	160
13	Data structures.....	160
13.1	CRTs.....	160
13.1.1	CRT AT for the selection of internal private authentication keys.....	160
13.1.2	CRT AT for selection of internal authentication keys	161
13.1.3	CRT for selection of IFD's PuK.CA _{IFD} .CS_AUT	161
13.1.4	CRT for selection of IFD's PuK.IFD.AUT	162
13.1.5	CRT AT for selection of the public DH / ECDH key parameters.....	162
13.1.6	CRT AT for selection of the PBM key parameters.....	162
13.1.7	GENERAL AUTHENTICATE DH key parameters used by the Privacy Protocol.....	163
13.1.8	CRT AT for selection of ICC's private authentication key	163
13.1.9	CRT for selection of IFD's PuK.IFD.AUT	164
13.1.10	CRT for selection of PrK.ICC.KA.....	164
13.2	Key transport device authentication protocol	164
13.2.1	EXTERNAL AUTHENTICATE	165
13.2.2	INTERNAL AUTHENTICATE.....	166

13.3	Privacy device authentication protocol	166
13.3.1	EXTERNAL AUTHENTICATE (DH case)	167
13.3.2	EXTERNAL AUTHENTICATE (ECDH case)	168
13.3.3	INTERNAL AUTHENTICATE (DH case).....	169
13.3.4	INTERNAL AUTHENTICATE (ECDH case).....	170
14	AlgIDs, Hash- and DSI Formats	171
14.1	Algorithm Identifiers and OIDs	171
14.2	Hash Input-Formats.....	172
14.2.1	PSO:HASH without command chaining.....	172
14.2.2	PSO:HASH with command Chaining.....	173
14.3	Formats of the Digital Signature Input (DSI).....	173
14.3.1	DSI according to ISO/IEC 14888-2 (scheme 2).....	174
14.3.2	DSI according to PKCS #1 V 1.5	175
14.3.3	Digest Info for SHA-X.....	176
14.3.4	DSI according to PKCS #1 V 2.x	178
14.3.5	DSA with DH key parameters	179
14.3.6	Elliptic Curve Digital Signature Algorithm - ECDSA.....	179
15	CV_Certificates and Key Management.....	180
15.1	Level of trust in a certificate	180
15.2	Key Management	180
15.3	Certificate types.....	181
15.3.1	Card Verifiable Certificates.....	181
15.3.2	Signature-Certificates	181
15.3.3	Authentication Certificates.....	181
15.4	Use of the public key extracted from a CV-certificate	181
15.5	Validity of the key extracted from a CV-certificate	182
15.6	CVC structure	183
15.6.1	Non-self-descriptive certificates	183
15.6.2	Self-descriptive certificates	183
15.7	Certificate Content	184
15.7.1	CPI-Certificate Profile Identifier	184
15.7.2	CAR-Certification Authority Reference DO.....	185
15.7.3	CHR-Certificate Holder Reference DO.....	186
15.7.4	CHA-Certificate Holder Authorization Data Object (CHA-DO)	187
15.7.5	Role identifier specifications.....	189
15.7.6	CHAT-Certificate Holder Authorization Template (CHAT).....	192
15.7.7	OID — Object identifier	192
15.7.8	CEDT — Certificate Effective Date Template.....	192
15.7.9	CXDT — Certificate Expiration date Template	192
15.8	Certificate signature	193
15.8.1	Non self-descriptive certificates	193
15.8.2	Self-descriptive certificates	194
15.9	Coding of the certificate content	194
15.9.1	Non self-descriptive certificates	194
15.9.2	Self-descriptive certificates.....	195
15.9.3	Self-descriptive certificates for elliptic curve cryptography.....	195
15.10	Steps of CVC verification.....	199
15.10.1	First round: CVC verification from a Root PuK.....	200
15.10.2	Subsequent round(s)	201
15.11	Commands to handle the CVC	201
15.12	C_CV.IFD.AUT (non self-descriptive)	201
15.13	C_CV.CA.CS-AUT (non self-descriptive).....	203
15.14	C.ICC.AUT.....	204
15.15	Self-descriptive CV Certificate (Example).....	204
15.15.1	Public Key	205
15.15.2	Certificate Holder Authorization Template	205
15.15.3	Certificate Extension.....	205
15.15.4	ECDSA Signature	206
16	Files.....	207

16.1	File structure	207
16.2	File IDs	208
16.3	EF.DIR	208
16.4	EF.SN.ICC	208
16.5	EF.DH	209
16.6	EF.ELC	209
16.7	EF.C.ICC.AUT	210
16.8	EF.C.CA _{ICC} .CS-AUT	211
16.9	EF.C_X509.CH.DS	211
16.10	EF.C_X509.CA.CS (DF.ESIGN).....	212
16.11	EF.DM.....	212
17	Cryptographic Information Application	213
17.1	ESIGN cryptographic information layout example.....	214
17.1.1	EF.CIAInfo.....	215
17.1.2	EF.AOD	216
17.1.3	EF.PrKD	219
17.1.4	EF.PuKD.....	221
17.1.5	EF.CD	222
17.1.6	EF.DCOD	223
Annex A (normative) Algorithm Identifiers — Coding and specification.....		226
Annex B (informative) Device authentication Protocol Properties		234
Annex C (informative) Personalization scenarios		236
Annex D (informative) OID values		238
D.1	OIDs for certificate signatures	238
D.2	OIDs for key transport protocol	239
D.3	OIDs for device authentication with privacy	239
D.4	OIDs for password based mechanisms	240
D.5	OIDs for mEAC protocol	241
D.5.1	OIDs for Chip Device Authentication	241
D.5.2	OIDs for Terminal Device Authentication.....	241
D.6	OIDs for privacy protocols.....	242
D.6.1	OIDs for Restricted Identification.....	242
D.6.2	OIDs for Restricted Identification.....	243
D.7	OIDs for mEAC based eServices.....	243
D.7.1	OIDs for Terminal Device Authentication in mEAC-based eServices	243
D.8	OIDs for the PCA mechanism.....	244
Annex E (informative) Build scheme for object identifiers defined by EN 14890		245
Bibliography		247

Foreword

This document (EN 419212-1:2014) 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.

This European Standard shall be given the status of a national standard, either by publication of an identical text or by endorsement, at the latest by June 2015 and conflicting national standards shall be withdrawn at the latest by June 2015.

This document supersedes EN 14890-1:2008.

This document has been prepared under a mandate given to CEN by the European Commission and the European Free Trade Association.

EN 419212, *Application Interface for smart cards used as Secure Signature Creation Devices*, consists of two parts:

- *Part 1: Basic services*; [the present document] which describes the specifications for IAS based services on smart cards to be used in compliance to the requirements of Article 5.1 of the Electronic Signature Directive; and
- *Part 2: Additional services* which describes other services that may be used in conjunction with all, some or none of the services described in Part 1.

This standard supports services in the context of Identification, Authentication and Electronic Signature (IAS) services, as well as other services.

In this Part 1 of EN 419212, the standard allows support of implementations of the European legal framework for electronic signatures, defining the functional and security features for a smart card intended to be used as a Secure Signature Creation Device according to the Terms of the European Directive on Electronic Signature 1999/93/EC. A card compliant to the standard will be able to produce a "Qualified electronic signature" that fulfils the requirements of Article 5.1 of the Electronic Signature Directive and therefore can be considered equivalent to a hand-written signature.

EN 419212-2 specifies mechanisms to support other services like generic identification, authentication, confidentiality and signature verification services.

EN 419212 defines a set of services that will enable the development of interoperable cards issued by any card industry sector. The standard describes an application interface and behavior of the SSCD, i.e. it should be possible to implement it on native and interpreter based cards.

Compared with the 2008 versions of EN 14890, the following broad change has been made:

The scope of the standard was enhanced through new mechanisms in the field of password based mechanisms and privacy.

Regarding EN 419212-1, the most significant technical changes that have been made are the following ones:

- new algorithms added to device authentication protocols (e.g. AES, ELC);
- added AES to secure messaging;
- introduced password based mechanisms (PACEv2);
- updating references to their latest releases;

- algorithm Identifier coding;
- recommendation for making best use of device authentication protocols.

Regarding EN 419212-2, the most significant technical changes that have been made are the following ones:

- a) added privacy services including:
 - 1) anonymity and pseudonymity services;
 - 2) auxiliary data transmission e.g. for Age verification;
 - 3) e-Services with trusted third party;
 - 4) e-Services with 2-parties.

According to the CEN-CENELEC Internal Regulations, the national standards organizations of the following countries are bound to implement this European Standard: 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 the United Kingdom.

Introduction

The European Committee for Standardization (CEN)] draws attention to the fact that it is claimed that compliance with this document may involve the use of a patent concerning the mapping function given in 9.3.6 "Step 4.2 — Map nonce and compute generator point for integrated mapping".

The patent relates to "Sagem, MorphoMapping Patents FR09-54043 and FR09-54053, 2009".

CEN takes no position concerning the evidence, validity and scope of this patent right.

The holder of this patent right has assured CEN that he/she is willing to negotiate licences under reasonable and non-discriminatory terms and conditions with applicants throughout the world. In this respect, the statement of the holder of this patent right is registered with CEN. Information may be obtained from:

Morpho

11, boulevard Galliéni

92445 Issy-les-Moulineaux Cedex - France

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights other than those identified above. CEN shall not be held responsible for identifying any or all such patent rights."

1 Scope

This European Standard specifies mechanisms for smart cards to be used as secure signature creation devices covering:

- signature creation;
- user verification;
- password based authentication;
- device authentication;
- establishment of a secure channel.

The specified mechanisms are suitable for other purposes like services in the context of IAS.

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 419212-2:2014, *Application Interface for smart cards used as Secure Signature Creation Devices — Part 2: Additional services*

ISO 3166 (all parts), *Codes for the representation of names of countries and their subdivisions*

ISO 11568-2, *Financial services — Key management (retail) — Part 2: Symmetric ciphers, their key management and life cycle*

ISO/IEC 7816-3, *Identification cards — Integrated circuit cards — Part 3: Cards with contacts -- Electrical interface and transmission protocols*

ISO/IEC 7816-4:2013, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*

ISO/IEC 7816-6, *Identification cards — Integrated circuit cards — Part 6: Interindustry data elements for interchange*

ISO/IEC 7816-8:2004, *Identification cards — Integrated circuit cards — Part 8: Commands for security operations*

ISO/IEC 7816-11:2004, *Identification cards — Integrated circuit cards — Part 11: Personal verification through biometric methods*

ISO/IEC 7816-15:2004, *Identification cards — Integrated circuit cards — Part 15: Cryptographic information application*

ISO/IEC 8859 (all parts), *Information technology — 8-bit single-byte coded graphic character sets*

ISO/IEC 9796 (all parts), *Information technology — Security techniques — Digital signature schemes giving message recovery*

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

ISO/IEC 14888-2, *Information technology — Security techniques — Digital signatures with appendix — Part 2, Integer factorization based mechanisms*

ISO/IEC 14888-3, *Information technology — Security techniques — Digital signatures with appendix — Part 3: Discrete logarithm based mechanisms*

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

3 Terms and definitions

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

NOTE These definitions are in compliance with those given in the revision of ISO/IEC 7816-4.

3.1

anonymity

assurance that a user may use a resource or service without disclosing their user identity

3.2

anonymization

process that removes the association between an identifying data set and a data subject

3.3

anonymized data

data that was once linked to an individual but can now no longer be related to them

3.4

anonymous data

data that cannot be linked to a specific individual

3.5

answer-to-Reset file

elementary file which indicates operating characteristics of the card

3.6

a priori trusted

operating environment which by definition can be trusted without further device authentication

EXAMPLE An example of this is the use within a company, where any available access point is connected to a trusted network.

3.7

authentication

verification that an entity is the claimed one [56]

3.8

command-response pair

set of two messages: a command followed by a response

3.9

confidentiality protection

prevention of information disclosure to unauthorized individuals, entities or systems [56]

3.10

data unit

smallest set of bits which can be unambiguously referenced

3.11

data element

item of information seen at the interface for which are defined a name, a description of logical content, a format and a coding

3.12

data object

information seen at the interface which consists of a tag, a length and a value (i.e. a data element)

Note 1 to entry: In this specification, data objects are referred to as BER-TLV data objects. Refer to ISO/IEC 7816-4:2013, 6.2 and E.1.

3.13

dedicated file

file containing file control information and, optionally, memory available for allocation. It may be the parent of EFs and/or DFs

3.14

device authentication

process of validating the credentials of a device.

3.15

DF name

string of bytes which uniquely identifies a dedicated file in the card

3.16

elementary file

set of data units or records which share the same file identifier. It cannot be the parent of another file

3.17

file control parameters

logical, structural and security attributes of a file

3.18

file identifier

2-bytes binary value used to address a file

3.19

forward secrecy

security property of a protocol that guarantees that the disclosure of long-term private key does not enable an opponent to compromise the secrecy property of the executions of the protocol made in the past, for example, by re-computing previously derived keys

3.20

identification

unique association of a set of descriptive parameters to an individuum within a given context

3.21

integrity protection

mechanism ensuring that data cannot be modified undetectably

3.22

master file

mandatory unique dedicated file representing the root of the file structure

3.23

message

string of bytes transmitted by the interface device to the card or vice-versa, excluding transmission-oriented characters as defined in ISO/IEC 7816-3

3.24

mutual authentication

authentication where both parties (ICC and IFD) are authenticated to each other

3.25

non-traceability

refer to traceless authentication

3.26

parent file

dedicated file immediately preceding a given file within the hierarchy

3.27

password

data which may be required by the application to be presented to the card by its user and which, in the context of this specification, is a string of numbers and/or ASCII characters

3.28

path

concatenation of file identifiers without delimitation.

Note 1 to entry: If the path starts with the identifier of the master file, it is an absolute path.

3.29

privacy

claim of individuals, groups, or institutions to determine for themselves when, how, and to what extent information about them is communicated to others [43]

3.30

private key

secret part of an asymmetric key pair e.g. signature creation data as specified in the EU directive for electronic signatures

3.31

pseudonymisation

particular type of anonymization that both removes the association with a data subject and adds an association between a particular set of characteristics relating to the data subject and one or more pseudonyms

3.32

pseudonymised data

data that can only be linked to such a person if one has possession of a decoding “key”

3.33

pseudonymity

assurance that a user may use a resource or service without disclosing its user identity, but can still be accountable for its use

3.34

public key

public part of an asymmetric key pair

3.35

record

string of bytes which can be handled as a whole by the card and referenced by a record number

3.36

record number

sequential number assigned to each record which uniquely identifies the record within its elementary file

3.37

retry counter

counter being used to count the number of erroneous usages of a related (security) object

Note 1 to entry: If the object (e.g. password entry) was used correctly (correct password entered) then the retry counter is reset to its initial value. A typical value of a retry counter is 3.

3.38

secret key

either a symmetric key or private part of a public key pair

3.39

traceless authentication

authentication which does not give any persistent cryptographic proof that the authentication has occurred

3.40

trusted environment

operating environment that avoids tampering with data seen at the communication interface by the definition of its physical location, physical security protection or physical access conditions or other measures

Note 1 to entry: See 8.4.4 "Specification of the environment".

3.41

trusted channel

means by which a TSF and a remote trusted IT product can communicate with the necessary confidence to support the TSP

3.42

trusted path

means by which a user and a TSF can communicate with the necessary confidence to support the TSP

3.43

unlinkability

assurance that a user may make multiple uses of resources or services without others being able to link these uses together [44]

3.44

untrustable environment

operating environment that does not prevent tampering and intrusion of data available at the communication interface

Note 1 to entry: An environment which is not a priori trustable and which cannot prove trustability through a device authentication, will be considered as untrustable. No further operations should be done by the card holder on an untrustable environment.

3.45

untrusted environment

operating environment which is not known to be trustable

Note 1 to entry: Until a proof of confidence is given (typically through a device authentication) the environment should not be sent any sensitive data from the cardholder, in particular a PIN.

After a successful device authentication, *untrusted environment* turns into *trusted environment*. Therefore the quality 'untrusted' does not imply 'untrustable' but the requirement of trust verification.

3.46

usage counter

counter updated each time a related object is used. Unlike retry counters, usage counters are not reset to their initial value after the object was successfully used

3.47

user consent

expression of the act-of-will to a card related service

3.48

usage of expressions in this standard

use of the key words "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document as to be interpreted as described in [37]

Note 1 to entry: See the following list of key-words:

SHALL:	This word, or the terms " REQUIRED " or " SHALL ", means that the definition is an absolute requirement of the specification.
SHALL NOT:	This phrase, or the phrase " SHALL NOT ", means that the definition is an absolute prohibition of the specification.
SHOULD:	This word, or the adjective " RECOMMENDED ", means that there may exist valid reasons in particular circumstances to ignore a particular item, but the full implications will be understood and carefully weighed before choosing a different course.
SHOULD NOT:	This phrase, or the phrase " NOT RECOMMENDED " means that there may exist valid reasons in particular circumstances when the particular behaviour is acceptable or even useful, but the full implications should be understood and the case carefully weighed before implementing any behaviour described with this label .
MAY:	This word, or the adjective "OPTIONAL", means that an item is truly optional. One vendor may choose to include the item because a particular marketplace requires it or because the vendor feels that it enhances the product while another vendor may omit the same item. An implementation which does not include a particular option will be prepared to interoperate with another implementation which does include the option, though perhaps with reduced functionality.
[...]	Square brackets indicate a freedom of choice. This can be either a: - <i>choice of values</i> in a given range ['00' .. '03'] – typical for fixed length fields, - or a choice whether to present a value or not – ... '80' L80 AlgID ['83' L83 KeyID] whereas the set of data in square brackets is the <i>optional</i> part (= at the discretion of the implementation) or <i>conditional</i> (= depending on a condition given in the standard).
CONDITIONAL:	The application of a specification depends on one or more conditions and shall only be applied if the condition(s) are met. The associated condition(s) are always described if a <i>conditional</i> attribute is made with a specification. If the conditions are met, the specification may be <i>normative</i> or <i>informative</i> depending on the context in which the specification is made.

4 Symbols and abbreviations

For the purposes of this document, the following symbols and abbreviations apply.

AES	Advanced Encryption Standard
AID	Application Identifier
APDU	Application protocol data unit
AKI	Authority Key Identifier
AT	Authentication Template, CRT for Authentication
AUT	Authentication
BCD	Binary Coded Decimal
BER	Basic Encoding Rules
C	Certificate
CA	Certification Authority
CAR	CA Reference
CC	Cryptographic Checksum

CDA	Card Device Authentication
CG	Cryptogram
CH	Cardholder
CHA	Certificate Holder Authorization
CHN	Cardholder Name
CHR	Certificate Holder Reference
CIA	Cryptographic Information Application
CIO	Cryptographic Information Object
CLA	Class byte
Comp()	The public key compression function <i>Comp(...)</i> is defined in 12.6 "The compression function <i>Comp()</i> ".
CPI	Certificate Profile Identifier
CRT	Control Reference Template
CS	CertSign, certificate containing the public part of a signature key pair
CT	CRT for confidentiality
CV	Card Verifiable
CVC	Card Verifiable Certificate
CWA	CEN Workshop Agreement
C/S	Client/Server
D[key](msg)	Decipherment of <msg> with <key>
DE	Data Element
DES	Data Encryption Standard
DF	Dedicated File
DH	Diffie-Hellman
DH.P	DH parameters, $DH.P = g p q$
DIR	Directory
DO	Data Object
DS[key](msg)	Digital Signature of <msg> with <key>
DSA	Digital Signature Algorithm
DSI	Digital Signature Input
DST	Digital Signature Template, CRT for DS
E[key](msg)	Encipherment of <msg> with <key>
ECC	European Citizen Card
ECDH	DH key agreement based on elliptic curve cryptography (ELC)
ECDH.P	ECDH parameters based on elliptic curve cryptography. $ECDH.P = p a b G n$
ECDSA	Signature Scheme based on elliptic curve cryptography (ELC)
ELC	Elliptic Curve Cryptography
EDFB	Encrypted Data Formatted Block
FCI	File Control Information
FID	File Identifier

h(msg)	hash-code of <msg>
HT	CRT for hash code
ICC	Integrated Circuit(s) Card
ICV	Initial Chaining Value
ID	Identifier
IFD	Interface Device
INS	Instruction byte
KE	Key Encipherment
KID	Key Identifier
L.F	Length of elliptic curve parameter
MAC[key](msg)	Message Authentication Code of <msg> built with <key>
MF	Master File
MSE	MANAGE SECURITY ENVIRONMENT
OAEP	Optimal Asymmetric Encryption Padding
OID	Object Identifier
P1-P2	Parameter bytes
PBM	Password based mechanism
PCA	Privacy based Chip Authentication
PrK	Private Key
PuK	Public Key
PKI	Public Key Infrastructure
PKCS	Public Key Cryptography Standards
PI	Padding Indicator
PIN	Personal Identification Number
PP	Protection Profile
PRND	Padding Random Number
PSO	PERFORM SECURITY OPERATION
PSS	Probabilistic Signature Scheme
QES	Qualified Electronic Signature
RC	Retry Counter
RCA	RootCA
RD	Reference Data
RFU	Reserved for Future Use
RID	Registered application provider identifier
RND	Random number
RSA	Cryptographic algorithm invented by Ronald Rivest, Adi Shamir and Leonard Adleman
SCA	Signature-Creation Application
SHA	Secure Hash Algorithm
SE	Security Environment
SIG()	Signature of

SK	Secret Key
SN	Serial Number
SM	Secure Messaging
SSC	Send Sequence Counter
SSCD	Secure Signature Creation Device
SW1-SW2	Status bytes
TDA	Terminal Device Authentication
TDES	Triple-DES, this standard only considers the 2-key variant
TOE	Target Of Evaluation
TPDU	Transmission Protocol Data Unit
TLS	Transport Layer Security
TLV	Tag, Length, Value
TSF	TOE Security Functions
TSP	TOE Security Policy
x_P	x-coordinate of elliptic curve point P
y_P	y-coordinate of elliptic curve point P

5 Signature application

5.1 Application Flow

Figure 1 shows an execution flow for a signature creation as it is mandated in Bibliographical Reference [3].

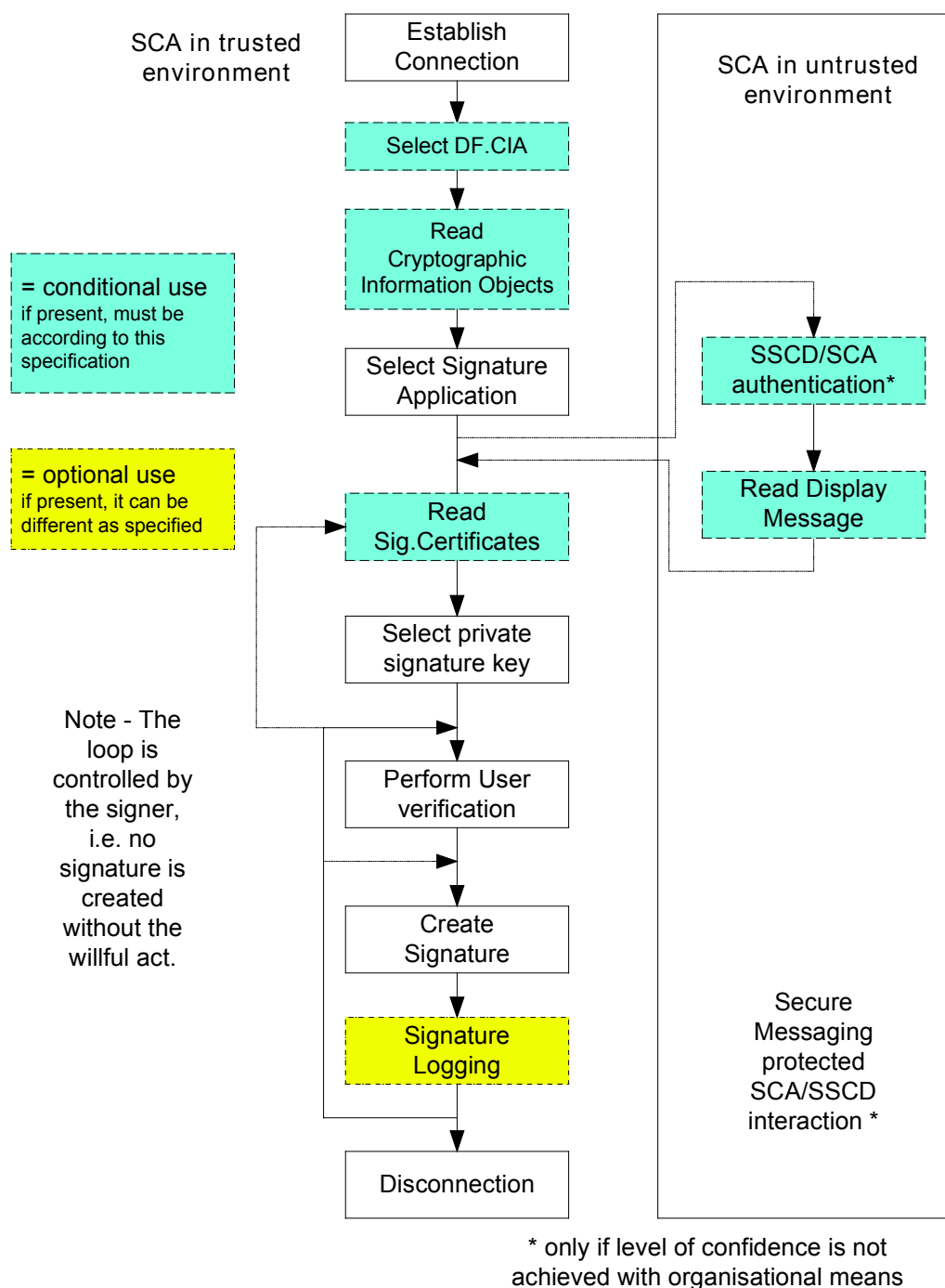


Figure 1 — Interaction sequences between SCA and SSCD in compliance with EU directive

The corresponding technical implementation is given in this document and as various mechanisms are specified for device authentication and user verification with a number of resulting combinations, Figure 2 gives some example execution flows for typical signature cards. Each card implements a signature application and they differ in the combination of privacy and security features.

A card with a contact interface that is used in a trusted environment may not require any device authentication. In contrast, contactless cards are vulnerable to skimming and eavesdropping attacks and hence require protection.

Reading SN.ICC would be a possible start for a non-privacy protecting device authentication (e.g. 8.6) whereas the card is addressed through contacts. For privacy protection (e.g. 8.7/8.8) reading SN.ICC would not be desirable. Instead a password based mechanism may be involved which prevents the card being addressed unless a password, e.g. secret PIN or Card Access Number printed on the card is entered first. This is suitable for contactless cards which shall not allow communication without either the explicit act-of-will of the card holder or proof of legitimate physical possession of the card.

In multi-application environments device authentication may be performed on card (e.g. MF level) prior to application selection.

Whether for basic or additional services, a device authentication is always mandatory if the environment where the signature device is being used, is not known to be trustable. This is typical for public environment (e.g. airport, merchandise, POS). Only in trusted environments (e.g. company, campus) can device authentication be skipped.

The user can decide to abort the operation when (s)he is prompted to enter the PIN/password during user verification. Once the IFD has performed a device authentication, a display message (Personal Assurance Message – PAM) can be shown on the terminal screen to inform the cardholder that the environment is safe and s(he) may continue entering the PIN/password since the environment is safe. More information about this mechanism can be found in 8.14 "Reading the Display Message".

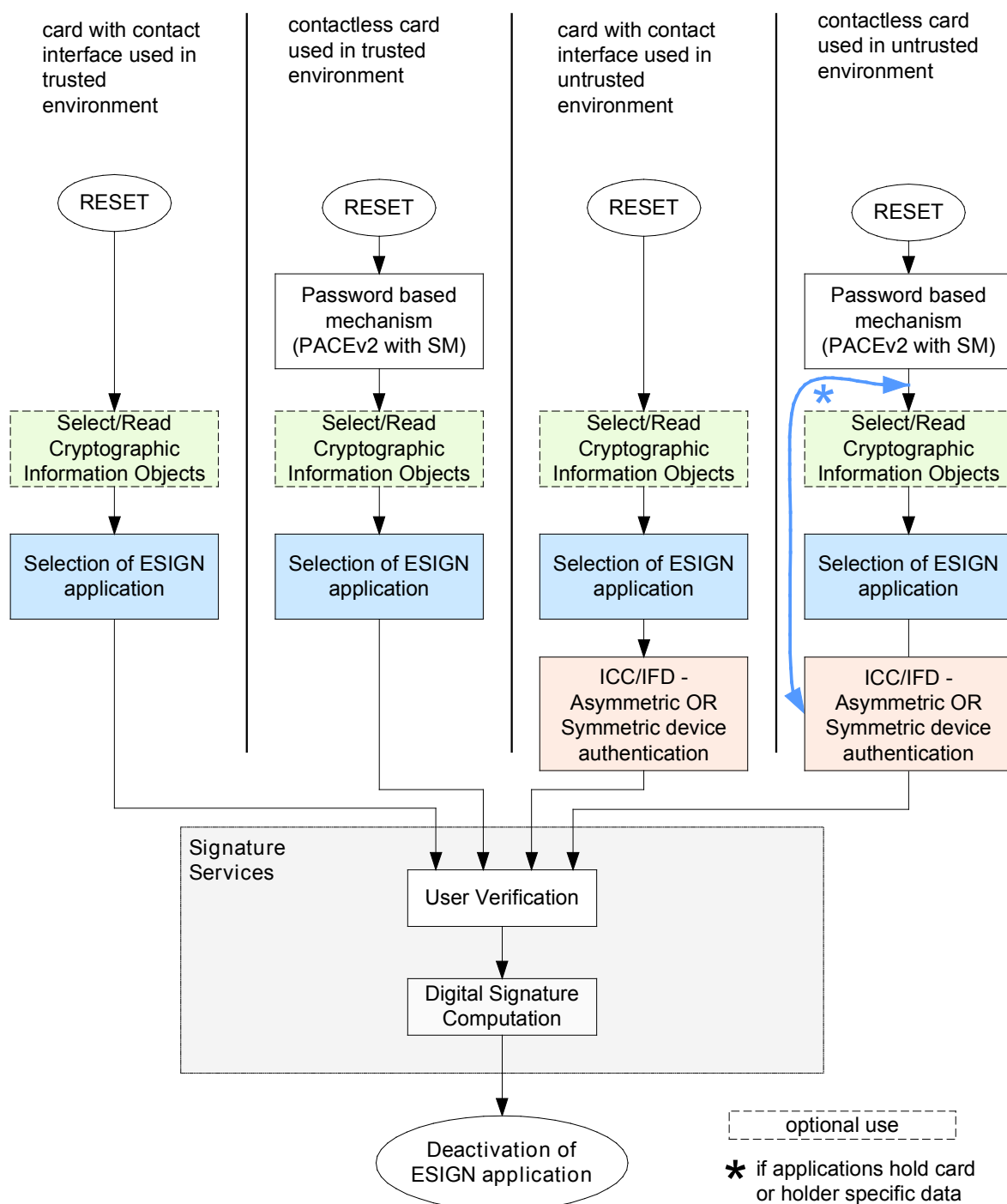


Figure 2 — Execute signature services

Figure 2 shows the selection of additional services in the context of the ESIGN application. User verification might be required for some of the additional services. The detailed access conditions are described in the appropriate security environments.

* For contactless case in untrusted environment, two choices are possible. Either the reading of CIA file (refer to Clause 17) and the selection of application are done before device authentication, in conformity with Figure 1, or the device authentication is done first. For privacy reasons the cryptographic information objects shall not reveal any information, otherwise the device authentication shall be done prior to the reading. To protect the

leakage of user information from CIA file, refer to PKCS#15 (thanks to EF.DIR, DF.CIA files may be hosted within an application and not readable without security measures).

5.2 Trusted environment versus untrusted environment

According to the definitions in "trusted environment" and "untrusted environment", this specification describes additional mechanisms in order to accomplish the required security as claimed in the protection profile [2]. The additional mechanisms according to this specification are:

- Device authentication
- Secure messaging

The following table shows which authentication method is to be used for digital signatures in various privacy contexts, e.g. for contactless cards compared to contact cards.

Table 1 – Using authentication protocols in different context to secure signatures

Digital Signatures according to Clause 7 can be performed after ...	Trusted Environment (private) to be secured by organisational means	Untrusted Environment (public) to be secured by cryptographic means
Contact communication channel has been established	As in generic case no authentication protocol necessary (see 5 "Signature application")	Chip and Terminal Authentication , see device authentication protocols according to 8 "Device authentication".
Contactless communication channel has been established	Password Based authentication (PACEv2) with provision of secure channel by secure Messaging (restricted to specific types of passwords e.g. biometric passwords or passwords stored as keys inside the ICC (see 9 "Password-based authentication protocols")	1. Password based authentication (PACEv2) with provision of secure channel by Secure Messaging 2. Chip and Terminal authentication e.g. see privacy and /or mEAC device authentication protocol according to 8 "Device authentication". NOTE The RSA transport protocol [8.6] is not suitable, unless the SN.ICC is set to a "dummy value" for privacy reasons.

5.3 Selection of ESIGN application

5.3.1 General

An ESIGN application is selected by its Application Identifier (AID) [51]. The RID is registered by the ISO registration authority and has the following value:

A0 00 00 01 67.

The first 10 bytes of the AID have the following value

AID = A0 00 00 01 67 || "ESIGN" = A0 00 00 01 67 45 53 49 47 4E

with

Category = 'A.....' (international)

PIX = 'ESIGN' = 5 bytes

The maximum possible size of the PIX is 11 bytes (refer to ISO/IEC 7816-4:2013, 12.2.3). The remaining 6 bytes may be used by CEN to distinguish between different implementations in the scope of this standard.

Table 2 — SELECT ESIGN application — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'A4' — SELECT
P1	'04' select by DF Name
P2	'00' or '02' return FCI info (see remark below)
Lc field	Length of command data field
Data field	AID.ESIGN — application identifier
Le field	'xx' expected response length

For the description of the SELECT command refer to ISO/IEC 7816-4:2013, 11.1.1.

Remarks

P2 It is left to the particular application to use other specified values from ISO/IEC 7816-4, this specification does not mandate the retrieval of the file control information. JAVA cards, however, require P2 = '00' or P2 = '02'.

data field contains the AID.ESIGN as described above.

Table 3 — SELECT ESIGN application — response APDU

Response Parameter	Meaning
Data field	data according to P2. An FCI shall at least contain the DO '84' (DF Name)
SW1-SW2	Refer to ISO/IEC 7816-4

5.3.2 Exceptions for Secure Messaging

In case of an ICC without a MF, the SELECT by AID of an application shall always be issued by the terminal in clear, even if a secure messaging session is currently active. For such ICCs this SELECT by AID in clear shall not break the current secure messaging session (if any). For such ICCs this behavior is necessary in order to allow the association of one applet instance to one application in a JAVA Card. Refer to 8.13.3 "Rules for ending a secure session".

5.4 Selection of cryptographic information application

In compliance with ISO/IEC 7816-15 the cryptographic information application (CIA) belonging to the ESIGN application has the following AID:

AID.CIA_ESIGN = 'E8 28 BD 08 0F A0 00 00 01 67 || "ESIGN" (15 bytes)

The selection of the DF.CIA(ESIGN) is done with a DF Name as shown above.

5.5 Concurrent usage of signature applications

5.5.1 General

Concurrent usage of signature applications can be realized using the mechanisms of logical channel as defined in ISO/IEC 7816-4:2013, 5.4.2

5.5.2 Methods of channel selection

The mechanism for channel selection is optional.

A channel may be selected by

- implicit selection with CLA byte of the SELECT command
- explicit selection with the MANAGE CHANNEL command

Both methods are described in ISO/IEC 7816-4 and will not be considered in this document.

5.5.3 Security issues on multiple channels

In order to access the resources of the ICC from another channel the 'shareable' bit shall be set in the FCI of the application DF.

Each invocation of an ESIGN-Application has its own security status.

5.6 Security environment selection

A signature card may contain more than one authentication- or signature key and one or more user reference data respectively. In this case the IFD needs to select the appropriate keys to be involved in authentication or signature operations and/or user verification (e.g. PIN, Password or biometric data) respectively. The MANAGE SECURITY ENVIRONMENT (SET) command (refer to ISO/IEC 7816-4:2013, 11.5.11.) is used in order to specify the desired keys to be involved in the next operation.

A security environment selection by means of an MSE: RESTORE command, is applicable in the context of the following operations:

- User verification (PIN, Password or biometric).
- Signature creation as described in Clause 7 "Digital Signature Service".
- Device authentication as described in Clause 8 "Device authentication".
- Key Generation as described in Clause 11 "Key Generation".
- Verification of card verifiable certificates as described in Clause 15 "CV_Certificates and Key Management".
- Authentication services as described in Clause 6 "Client/Server Authentication" in EN 419212-2:2014.
- Authentication services as described in Clause 7 "Role Authentication" in EN 419212-2:2014.
- Key decipherment as described in Clause 8 "Symmetric key transmission between a remote server and the ICC" in EN 419212-2:2014.
- Signature verification as described in Clause 9 "Signature verification" in EN 419212-2:2014.

The actual application of the MANAGE SECURITY ENVIRONMENT is discussed in the appropriate clauses.

5.7 Key selection

The selection of keys is performed by the MSE:SET command if not specified otherwise (e.g. in the parameters of the command header). An appropriate keyID is submitted and stored in the ICC in order to specify the key to be used with the next command(s). Key selection shall be done prior to a security operation unless the key was previously implicitly, or explicitly, selected.

A keyID is typically a number represented by one or more bytes describing a non-standardized value which is application- or operating system specific. The IFD may retrieve the link between a certificate and a corresponding KeyID from the DF.CIA application data as described in 17 "Cryptographic Information Application".

5.8 Security Services

A security service is a set of SSCD functions that serve the purpose of signature generation, verification, authentication or key en-/decipherment.

Extended security services comprise the installation of data and keys, card life cycle management and other functionalities that extend the scope described in EU Directive 1999/93/EC [1]. Extended services are out of the scope of this standard.

The two volumes of this document specify the following security services.

- **User verification** service, refer to Clause 6 "User verification"
- **Key generation**, refer to Clause 11 "Key Generation"
- **Device authentication** service including secure messaging, refer to Clause 8 "Device authentication"
- **Digital signature service** for the advanced electronic signatures services of the cardholder, refer to Clause 7 "Digital Signature Service"
- **Authentication service** as required for specific authentication procedure, refer to Clause 6 "Client/Server Authentication" in EN 419212-2:2014
- **Role authentication**, refer to Clause 7 "Role Authentication" in EN 419212-2:2014.
- **Encryption key decipherment** service, refer to Clause 8 "Symmetric key transmission between a remote server and the ICC" in EN 419212-2:2014
- **Signature verification** service, refer to 9 "Signature verification" in EN 419212-2:2014.
- **Auxiliary data** comparison, refer to 11.3 "Auxiliary Data Verification " in EN 419212-2:2014.
- **Anonymity services** refer to 11.5 "mERA-based eServices with trusted third party protocol" in EN 419212-2:2014.
- **Pseudonymity services**
 - **Two party** service, refer to 11.4 "Restricted Identification" in EN 419212-2:2014.
 - **Three party** service, refer to 11.5 "mERA-based eServices with trusted third party protocol" in EN 419212-2:2014.

6 User verification

6.1 General

User verification is part of the application flow as described in Figure 2.

Two general methods of user verification are available:

- *Knowledge based*
- *Biometric based*

Each method provides the following verification mechanisms

- *Explicit* user verification
- *Implicit* user verification (Password Based Mechanisms (PBM))

Verification data, dedicated to protect the use of the private key for signature generation shall be applied in the ICC. If several private keys for signature generation are in the ICC, their use may optionally be protected by different verification data.

The protection of the client/server authentication function and the protection of the key decipherment function shall use separate (global or local according to the security policy) verification data.

The following commands, as defined in ISO/IEC 7816-4:2013, 11.5.6 ff are referenced:

- VERIFY
- CHANGE REFERENCE DATA
- RESET RETRY COUNTER
- GENERAL AUTHENTICATE

Regardless of the usage of these commands, at no time shall the signature key be used, unless it is protected by a preceding user verification

If knowledge based or biometric user verification is part of a Common Criteria evaluation a vulnerability analysis (AVA_VAN) shall be performed. The assessment of a knowledge based mechanism is a relatively simple and well established probabilistic analysis. For biometric authentication mechanisms the analysis is more complex for example proof that a claimed False Acceptance Rate (FAR) is met might need analysis of comprehensive fingerprint data bases.

6.2 Knowledge based user verification

6.2.1 General

Knowledge based user verification requires a password to be entered by the user. The explicit verification is done with the VERIFY command in contrast to Password based mechanisms

To avoid brute-force attacks by an exhaustive search attack on the password the guessing probability should be bounded. The guessing probability depends on the minimal password length, the coding of the password (numerical or alpha-numerical password) and any retry counter that may be assigned to the password.

Choosing a retry count of 3 for a password of 6 numerical digits guarantees a guessing probability of $3/10^6$ if the numerical password is selected randomly. Selecting a so called trivial password dramatically lowers the security of the user verification.

Using a longer or a randomly chosen alpha-numerical password is more secure and can allow a larger retry counter for the same (or smaller) $3/10^6$ guessing probability. This depends on the range where the password

is selected from, e.g. for randomly chosen alpha-numerical passwords of length 4 the key space contains 1.5×10^7 elements (refer to [48], Chapter 8.1). For such a password a retry count of 10 is comparable with a 6 digit numerical password.

6.2.2 Explicit user verification

Applications in the European domain frequently restrict passwords to numbers only sometimes known as 'PIN'. This standard does not restrict passwords to numbers – specifications using this standard have the freedom to restrict the range of possible input characters to "numbers only" on their own.

Execution Flow 1 of 1

Table 4 — VERIFY command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'20' VERIFY
P1	'00'
P2	<KeyRef> password reference
Lc field	Length of command data field
Data field	<verification data>

The structure and coding of the VERIFY command refer to ISO/IEC 7816-4:2013, 11.5.6.. The parameter P2 codes the reference value of the password.

If a password is referenced the reference is called "reference data qualifier" (refer to ISO/IEC 7816-4:2013, 11.5.1)

This standard specifies the following KIDs

KID = '8x' Key reference to local reference data for protection of the use of a private signature key (min. 6 digits or characters); the value of the KID and the purpose of the signature password shall be indicated in the cryptographic information objects.

Also other passwords may be of the type "local".

KID = '0x' Key reference to global reference data; the value of the KID and the purpose of the password shall be indicated in the cryptographic information objects.

In order to allow the co-existence of EMV-applications and ESIGN-applications on the same ICC, only the three least significant bits should be used to differentiate keys. Refer to Chapter 6.5.12.2 – Table 23 of Bibliographical Reference [11].

Password Based Mechanisms should implement PIN error- or usage counters in order to avoid exhaustive retries with the PIN or password.

Table 5 — VERIFY — response APDU

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

6.2.3 Password based mechanisms

Password based mechanisms provide an implicit user verification by performing a multi-step authentication protocol, see PACEv2 in Clause 9 "Password-based authentication protocols". In contrast to the application as device authentication protocol the corresponding password is not printed on the card but known by the user only. A reset counter is recommended to be assigned to the password. A password change and a reset of the Reset Counter shall be done applying the RESET RETRY COUNTER command, see 6.2.7 "Reset of RC and setting a new password".

6.2.4 Presentation formats

The data field codes the verification data in one of the following formats.

The password to be coded is '12345'

BCD coding	<table><tr><td>12</td><td>34</td><td>5F</td></tr></table>	12	34	5F						
12	34	5F								
ASCII coding	<table><tr><td>31</td><td>32</td><td>33</td><td>34</td><td>35</td></tr></table>	31	32	33	34	35				
31	32	33	34	35						
UTF8	<table><tr><td>31</td><td>32</td><td>33</td><td>34</td><td>35</td></tr></table>	31	32	33	34	35	special (e.g. language specific) characters will be different from ASCII coding			
31	32	33	34	35						
Half nibble bcd	<table><tr><td>F1</td><td>F2</td><td>F3</td><td>F4</td><td>F5</td></tr></table>	F1	F2	F3	F4	F5				
F1	F2	F3	F4	F5						
PIN in Format 2 PIN Block (see ISO 9564-1)	<table><tr><td>25</td><td>12</td><td>34</td><td>5F</td><td>FF</td><td>FF</td><td>FF</td><td>FF</td></tr></table>	25	12	34	5F	FF	FF	FF	FF	
25	12	34	5F	FF	FF	FF	FF			

Figure 3 — Example of coding '12345' in different presentation formats

Figure 3 demonstrates the coding when the different presentation formats are used. The actual presentation format may be retrieved from information presented in DF.CIA (refer to 17 "Cryptographic Information Application"), unless implicitly known.

National regulations may specify additional security requirements.

6.2.5 Retry and Usage counters

In order to avoid exhaustive search (brute force) attacks, the password should be protected by an appropriate retry or usage counter. For a password of 6 numerical digits the retry counter is typically 3 to achieve a guessing probability of $3/10^6$. For longer or randomly chosen alpha-numerical passwords a greater value of the retry counter can be selected with a similar or smaller guessing probability.

This depends on the range where the password is selected from, e.g. if the password is a randomly chosen string of 32 bytes, then a retry counter may even be not necessary. Passwords protecting the usage of signature keys shall consist, unless otherwise specified, of at least 6 digits or characters.

In order to avoid an exhaustive search attack on the password under a known resetting code, the resetting code should be protected by an appropriate usage counter with a usage counter limit with a low value (typically $U = 3$ for a resetting code of numerical digits).

After successful presentation, the appropriate retry counter is automatically reset to its initial value.

For the definition of the counter types refer to 3 "Terms and definitions".

6.2.6 Password Change

The CHANGE REFERENCE DATA command for changing the global password may be used under the same access conditions defined for the VERIFY command.

The change of DF-specific (local) password can only be performed after selection of the appropriate DF.

Execution Flow 1 of 1

Table 6 — Password Change — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'24' CHANGE REFERENCE DATA
P1	'00' no algorithm specified
P2	['0x' '8x'] KID of referenced password
Lc field	Length of command data field
Data field	<password> <new password>

For the description of the command refer to ISO/IEC 7816-4:2013, 11.5.7.

The length of the existing reference data is known in the ICC, so that neither a delimiter nor padding for filling up fixed formats is necessary. The length of the new password therefore computes $L_{\text{new}} = L_c - L_{\text{old}}$.

Table 7 — Password Change — response APDU

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

6.2.7 Reset of RC and setting a new password

After N (N as specified by application) subsequent false presentations of the password, the password is locked and does not allow further usage of the protected functions.

With the ISO/IEC 7816-4 command RESET RETRY COUNTER, the cardholder can initiate the reset of the RC to its initial value N. The Resetting Code shall have a minimum length of 6 digits¹⁾.

It is also possible to define a new password with the RESET RC command.

The reset of DF-specific (local) retry counter can only be performed after selection of the appropriate DF.

1) 'digit' can also be a character as part of a password.

Table 8 — RESET RETRY COUNTER — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2C' RESET RETRY COUNTER
P1	['00' .. '03']
P2	['0x' '8x'] KID of referenced password
Lc field	Length of command data field
Data field	see below

For the coding of the RESET RETRY COUNTER command refer to ISO/IEC 7816-4:2013, 11.5.10.

The content of the data field depends on the value of P1. The RESET RETRY COUNTER shall not change the security status, i.e. for using the signature creation key a VERIFY command is still required.

P1 = '00' Resetting code followed without delimitation by new reference data

P1 = '01' Resetting code

P1 = '02' New reference data

P1 = '03' Data field absent

Table 9 — RESET RETRY COUNTER — response APDU

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

After successful presentation of the Resetting Code, the retry counter of the referenced password is automatically reset to its initial value (typically N = 3 for a 6 numerical digit password).

6.3 Biometric user verification

6.3.1 General

The support of a biometric user verification (on-card matching) is optional. User verification shall always be done with a password. If biometric user verification is available in the IFD and ICC then the biometric user verification can be performed instead of password verification.

The following general methods of biometric user verification are available:

1. The **sensor is off-card**, whereas the following cases apply:
 - a. The biometric verification data is sent from the interface device to the ICC and is compared by a match on card mechanism.
 - b. The biometric **verification and reference data are transformed** by a suitable application specific mechanism as e.g. Fuzzy Vault [38] to password / PIN format.

The verification mechanism is then performed as used for passwords / PINs by means of the VERIFY command or a password based mechanism

2. The **sensor is on-card**, so there is no need to transmit the biometric verification data from the interface device to the ICC.

Transforming biometric data to a unique value (b.) has the consequence that the biometric algorithm is performed completely in the offcard application and the ICC only has to perform a user verification by means of direct comparison with yes / no response.

The users have the benefit that they do not have to remember their password because it is linked to their biometrics and usually to an additional randomizer. The security of this mechanism is, however, only as sound as the underlying biometric algorithm i.e. it is in general weaker than a password / PIN Comparison based on random passwords.

If biometric user verification is supported and the sensor is off-card, the Finger Minutiae Matcher compliant to 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 to ISO/IEC 7816-11 that shall be encoded as presented in Table 15. The BIT shall not contain a data object for minutiae order referenced by tag '82', i.e., the ICC shall not rely on an ordering procedure performed by the IFD.

6.3.2 Retrieval of the Biometric Information Template

Prior to the verification, the SCA may need information about:

- AlgID of the matching algorithm and key reference of the biometric reference data;
- biometric type (e.g. fingerprint);
- biometric type instance (e.g. left pointer finger enrolled);
- format owner of the biometric data;
- format type of the biometric data;
- on-/off-card sensor.

This information may be found in the DF.CIA description (refer to 17 "Cryptographic Information Application").

Alternatively this information may be provided in the Biometric Information Template (BIT), refer to ISO/IEC 7816-11. The BIT may be present in the DF.CIA or be retrieved with the GET DATA command.

The enrolment process is out of the scope of this standard.

Table 10 — Retrieve Biometric Information Template(s) — command APDU (optional)

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'CA' GET DATA
P1-P2	'7F60' Example: Biometric Information Template
Le field	'00'

For the coding of the command GET DATA refer to ISO/IEC 7816-4:2013, 11.4.3.

Table 11 — Retrieve Biometric Information Template(s) — response APDU (optional)

Response Parameter	Meaning
Data field	BIT (Value field of Biometric Information Template(s))
SW1-SW2	Refer to ISO/IEC 7816-4

For the format of the biometric information template refer to 6.3.3.3 "Biometric Templates".

6.3.3 Performing the biometric user verification

6.3.3.1 General

Two situations have to be distinguished:

- Sensor **off**-card;
- Sensor **on**-card.

Secure messaging (refer to 9.7 "Message structure of Secure Messaging APDUs" is required when executing commands on biometric user verification with Sensor off-card.

6.3.3.2 Sensor off-card

The sensor is outside the ICC, i.e. the captured biometric data needs to be sent to the ICC for on-card matching.

The IFD retrieves the Biometric Information Template by means of a GET DATA command according to 6.3.2. The corresponding path attribute in the Cryptographic Information Application denotes the DF that shall be selected prior to this GET DATA command.

Then the IFD uses a VERIFY command to process the biometric data structured according to the biometric template.

Table 12 — Biometric user verification — command APDU (Biometric verification)

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'21' VERIFY
P1	'00'
P2	'xx' reference data qualifier according to 6.2.2.
Lc field	Length of command data field
Data field	<biometric data template> refer to Table 14

For the structure and content of the APDU refer to ISO/IEC 7816-4:2013, 11.5.6.

Table 13 — Biometric user verification 2 of 2- response APDU

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

6.3.3.3 Biometric Templates

Table 14 — Biometric data template

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

Table 15 — Biometric Information Template

Tag	Length	Value					Presence			
'7F 60'	Var.	Biometric Information Template (BIT)								
		Tag	Length	Value						
		'83'	'01'	Reference data qualifier according to 6.2.2.					Mandatory	
		'A1'	Var.	Biometric Header Template (BHT) in compliance with CBEFF					Mandatory	
				Tag	Length	Value				
				'81'	'01'	'08' (fingerprint)			Mandatory	
				'82'	'01'	Biometric subtype (refer to ISO/IEC 7816-11:2004, Table C.3)			Mandatory	
				'87'	'02'	'0101' (ISO/IEC JTC 1/SC37)			Mandatory	
				'88'	'02'	'00 05'			Mandatory	
				'B1'	Var.	Biometric matching algorithm parameters				Optional
						Tag	Length	Value		
						'81'	'02'	Minimum number of minutiae required (1 byte, binary coding) Maximum number of minutiae accepted (1 byte, binary coding)		

6.3.3.4 Sensor on-card

For the on-card verification, the VERIFY command selects the biometric data reference through P2.

Execution Flow

Table 16 — Biometric verification — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'21' VERIFY
P1	'00' no algorithm specified
P2	['0x' '8'x] reference data qualifier acc. to 6.2.2.
Lc field	Length of command data field
Data field	'5F2E' '00' '4D' '00' empty verification DO and extended. header list DO

For the description of the command refer to ISO/IEC 7816-4:2013, 11.5.6.

The length of the existing reference data is known in the ICC, so that neither a delimiter, nor padding for filling up fixed formats is necessary.

Table 17 — Biometric verification — response APDU

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

For the structure and content of the APDU refer to ISO/IEC 7816-4:2013, 11.5.6.

6.3.4 Reset of RC

For resetting the RC refer to 6.2.6 "Reset of RC and setting a new password".

With respect to biometric reference data only the RESET RETRY COUNTER command variants that make use of P1 = '01' and P1 = '03' are specified in this document. The usage of the command variants with P1 = '00' and P1 = '02' is outside the scope of the document as the enrolment process is not defined in this document.

7 Digital Signature Service

7.1 General

This clause is related to technical aspects of electronic signatures according to Article 5 of EU Directive 1999/93/EC [1].

NOTE For QES additional organizational aspects need to be considered like evaluation (e.g. according to the PP SSCD (e.g. CWA 14169 [2]), certification, CA accreditation).

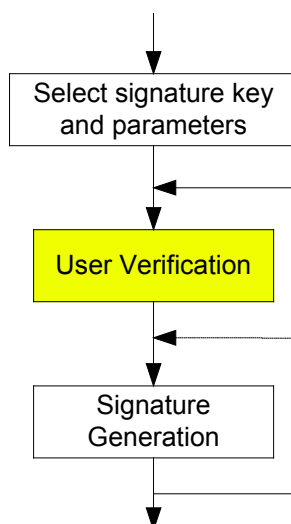
The technical requirements are fulfilled by the specifications of this standard.

7.2 Signature generation algorithms

For signature generation, algorithms as referenced in the current European and national algorithm catalogs shall be applied. Examples of existing documents are given in Bibliographical References [4] [5] [6] [7] and [8].

7.3 Activation of digital signature service

In order to enable the digital signature service a user verification shall be applied prior to signature generation.



NOTE The loop is controlled by the signer, i.e. no signature is created without a willful act.

Figure 4 — User Verification of the Signer

Refer to Figure 1 for the overview of the signature generation flow.

A knowledge based interaction always expresses the explicit will of the card holder to sign a document whereas a biometric identification does not necessarily imply the intention to sign.

For the description of the user verification refer to 6 "User verification".

7.4 General aspects

A digital signature process consists of several steps. Some of them are done outside the ICC, some of them are done inside the ICC, and some could be done on either side.

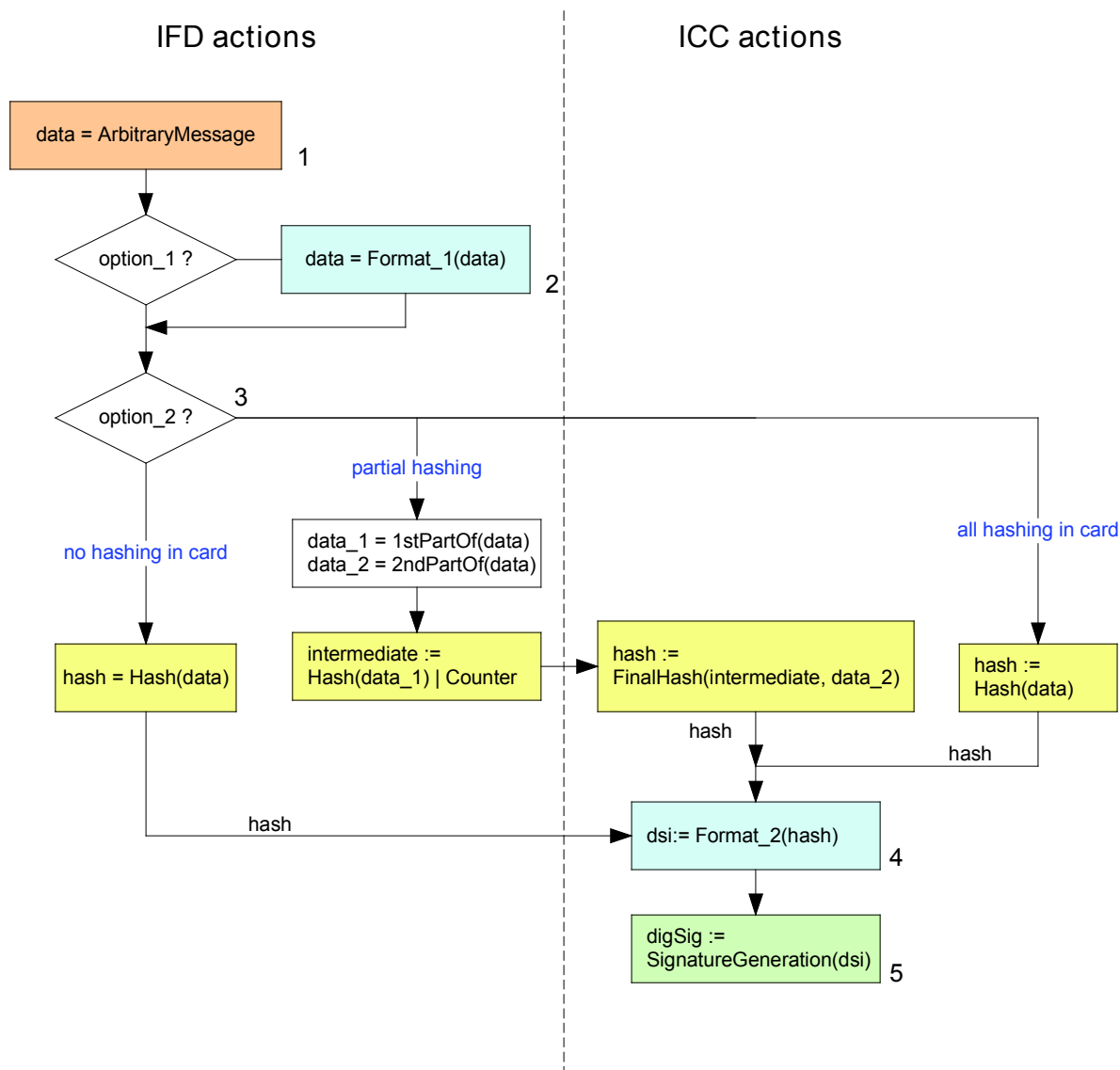


Figure 5 — Data-Flow chart for signature generation process (RSA Example)

Figure 5 shows an example of the alternatives to process a hash data signing. Signature input formatting aspects are not covered by Figure 5. The numbers 1..5 refer to the steps described below. The intermediate hash-code is the digest of preceding hash rounds followed by the big endian bit count. Only fully hashed blocks are transmitted, therefore the length of *data_1* (on partial hashing) is always a multiple of the digest size, related to the applied algorithm.

In principle, the signature generation process works as follows:

1. The signature generation process starts with an arbitrary message.
2. Optionally the original message is formatted in an application specific way. Typically, signature attributes and other format information besides the actual document to be signed, are added to the arbitrary message. The process of pre-formatting the data to be hashed is expressed as **Format_1** mechanism.

The specification of the optional format mechanism **Format_1** is outside the scope of this document. It may require, e.g., also additional data flow from or to the ICC.

3. The data shall be hashed by applying a hash function which is on the list of the applicable hash functions. The hash-code could be considered as a check value of the data to be signed. Depending on *option_2* the hashing will be done either
 - completely outside the ICC,
 - partly outside and partly inside the ICC, thus data is split into two parts,
 - completely inside the ICC depending on the card functionality and the software organization.
4. Depending on the signature algorithm the *hash-code* shall be formatted by a format mechanism **Format_2**. Typically this is a *padding mechanism* as described in 14.3 "Formats of the Digital Signature Input (DSI)". The result of this formatting is a *Digital Signature Input (dsi)*. Several format mechanisms have been defined, some of them are evaluated as "weak" and should not be used. The algorithm catalogues cover these aspects and propose formatting schemes which are approved until a specified date.
5. The final *dsi* (Digital Signature Input) is the string to be transformed by the signature algorithm. The result is the Digital Signature DS[<key>](*dsi*).

7.5 Signature Generation

7.5.1 General

Based on the example provided by Figure 5 different processes may be used to compute the hash value to be signed. The selection of the process may be based on security reasons, especially in case of RSA. If the ECDSA algorithm is used the typical flow will be "no hashing in card".

7.5.2 No hashing in Card

The data to be signed is completely hashed outside the ICC. Depending on the algorithm the *hash-code* is transferred to and signed by the ICC using the PSO:COMPUTE DIGITAL SIGNATURE command.

Execution Flow 1 of 1

Table 18 — Signature computation (no hashing in ICC) — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2A' PERFORM SECURITY OPERATION
P1	'9E' COMPUTE DIGITAL SIGNATURE
P2	'9A' 'data to be signed'
Lc field	Length of command data field
Data field	data to be signed
Le field	'00'

For the structure and content of the APDU refer to ISO/IEC 7816-8:2004, 5.2.

The response data field contains the *plain signature* without further TLV coding.

Table 19 — Signature computation (no hashing in ICC) — response APDU

Response Parameter	Meaning
Data field	plain signature without further TLV coding
SW1-SW2	Refer to ISO/IEC 7816-4

7.5.3 Partial hashing

The first part of the data to be signed is hashed outside the ICC. The intermediate hash-code is transferred to the ICC by the PSO:HASH command together with the remaining part of the data. Afterwards the signature is generated by the PSO:COMPUTE DIGITAL SIGNATURE command.

Execution Flow 1 of 2

Table 20 — Signature computation 1 of 2 (transfer intermediate hash-code) — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2A' PERFORM SECURITY OPERATION
P1	'90' hash
P2	'A0' input template for hash computation
Lc field	Length of command data field
Data field	'90' L ₉₀ <intermediate hash-code followed by big endian bit counter> '80' L ₈₀ <data to be hashed>

For the structure and content of the APDU refer to ISO/IEC 7816-8:2004, 5.2.

For the content of the command data field or DO 'A0' refer to 14.2 "Hash Input-Formats".

The response data field is absent.

Table 21 — PSO-Hash response

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

Execution Flow 2 of 2

The second command performs a digital signature on the hashed value.

Table 22 — Signature computation 2 of 2 (compute signature) — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2A' PERFORM SECURITY OPERATION
P1	'9E' COMPUTE DIGITAL SIGNATURE
P2	'9A' data to be signed
Le field	'xx'

For the structure and content of the APDU refer to ISO/IEC 7816-8:2004, 5.2.

The response data field contains the plain signature without further TLV coding.

Table 23 — PSO-COMPUTE DIGITAL SIGNATURE response

Response Parameter	Meaning
Data field	plain signature
SW1-SW2	Refer to ISO/IEC 7816-4

7.5.4 All hashing in ICC

The data to be signed is hashed inside the ICC. This is done by the PSO:HASH command. Finally the signature is generated by the PSO:COMPUTE DIGITAL SIGNATURE command.

Execution Flow 1 of 2**Table 24 — Signature computation 1 of 2 (all hashing in ICC) — command APDU**

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2A' PSO:HASH
P1	'90' hash operation
P2	'80' plain value
Lc field	'xx'
Data field	plain data to be hashed

For the structure and content of the APDU refer to ISO/IEC 7816-8:2004, 5.2.

The response data field is absent.

Table 25 — Signature computation 1 of 2 (all hashing in ICC) — response APDU

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

Execution Flow 2 of 2

The second command computes the digital signature.

Table 26 — Signature computation 2 of 2 (compute signature) — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'2A' PERFORM SECURITY OPERATION
P1	'9E' COMPUTE DIGITAL SIGNATURE
P2	'9A' data to be signed
Le field	'00'

For the structure and content of the APDU refer to ISO/IEC 7816-8:2004, 5.2.

The response data field contains the plain signature without further TLV coding.

Table 27 — PSO-COMPUTE DIGITAL SIGNATURE response

Response Parameter	Meaning
Data field	plain signature
SW1-SW2	Refer to ISO/IEC 7816-4
NOTE In the case of ECDSA the digital signature is calculated according to ISO/IEC 15946-5 [57]; for the format, see Bibliographical Reference [23], 5.3.	

7.6 Selection of different keys, algorithms and input formats

7.6.1 General

If the ICC supports more than one key and an implicit selection is not available, then the private signature key needs to be selected.

If the ICC supports several digital signature algorithms or different Digital Signature Input formats (refer to 14 "AlgIDs, Hash- and DSI Formats"), then the algorithm or the DSI format respectively shall be selectable. However, it is possible that either (e.g. an algorithm), is implicitly specified if a key is selected.

It is further possible that an algorithm, a key or a signature input format is not implicitly known, nor uniquely specified.

In all of the above cases the appropriate parameter shall be selected explicitly with a MANAGE SECURITY ENVIRONMENT command as described in ISO/IEC 7816-4:2013, 11.5.11.

This can be achieved by

- restoring an existing SE (refer to 7.6.2)
- modifying a KeyID/[AlgID] in a template (refer to 7.6.4)

7.6.2 Restore an existing SE

Execution Flow 1 of 1

Table 28 — MANAGE SECURITY ENVIRONMENT — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'22' MANAGE SECURITY ENVIRONMENT
P1	'F3' RESTORE
P2	SE# security environment number

For the structure and content of the APDU refer to ISO/IEC 7816-4:2013, 11.5.11.

The response data field is absent.

Table 29 — MANAGE SECURITY ENVIRONMENT response

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

NOTE The actual SE# is described in EN 419212-2.

7.6.3 Setting the Hash Template (HT) of a current Security Environment (SE)

If a hash algorithm is not implicitly known by the ICC, it shall be selected with the MANAGE SECURITY ENVIRONMENT command.

Execution Flow 1 of 1

Table 30 — Set hash template — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'22' MANAGE SECURITY ENVIRONMENT
P1	'41' SET (hash template acc. to p2)
P2	'AA' CRT for HT (hash template)
Lc field	'xx' length of the following data
Data field	'80' L ₈₀ yy AlgID

For the structure and content of the APDU refer to ISO/IEC 7816-4:2013, 11.5.11.

The response data field is absent.

Table 31 — Set hash template — response

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

7.6.4 Modify the Digital Signature Template (DST) of a current Security Environment (SE)

If a signing key is not implicitly known by the ICC, it shall be selected with the MANAGE SECURITY ENVIRONMENT command. Optionally an algorithm identifier can be supplied in order to specify variations of the algorithm (e.g. for padding formats).

Execution Flow 1 of 1

Table 32 — Selection of signature key — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'22' MANAGE SECURITY ENVIRONMENT
P1	'41' SET for signature creation
P2	'B6' CRT for DST
Lc field	'xx' length of the following data
Data field	'84' L ₈₄ xx ['80' L ₈₀ yy] KeyID [AlgID]

For the structure and content of the APDU refer to ISO/IEC 7816-4:2013, 11.5.11.

For the algorithm ID refer to Annex A.

The response data field is absent.

Table 33 — Selection of signature key — response

Response Parameter	Meaning
Data field	absent
SW1-SW2	Refer to ISO/IEC 7816-4

7.7 Read certificates and certificate related information

7.7.1 General

If more than one signature key is present in the ICC, the user may need to select the signature key for a certain signature creation. In the user certificates are fields indicating whether it is a private or professional certificate. Therefore the user may want to evaluate these fields in order to select the proper certificate and with it the associated private key. In the CIOs there is a link between a certificate and the associated private key and an indication where the certificates are to be found (in the ICC or on a server), in the latter case the ICC will provide an URL and optionally the certificate serial number.

7.7.2 Read certificate related CIOs

CIOs related to certificates denote which certificates are present in the ICC or are to be found on a server and which private key corresponds to the public key present in the certificate (refer to 17 "Cryptographic Information Application").

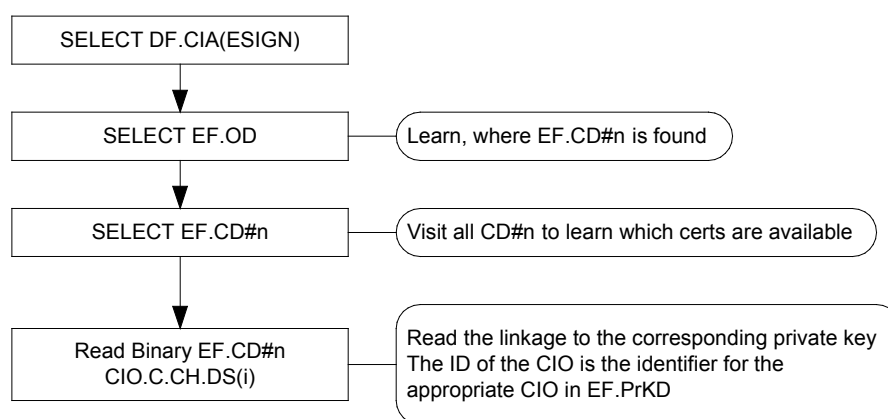


Figure 6 — Reading certificate related information (certificate is present in the ICC)

The certificate related information contains the ID of the appropriate CIO in EF.PrKD which contains the KeyID of the private key in question.

7.7.3 Read signer's certificate from ICC

If present, the corresponding certificate file(s) may be read from the ICC. Prior to reading the appropriate file needs to be selected. The location (e.g. path or (short) FileID) is denoted in the corresponding CIO in EF.CD#n.

Certificate related information is stored in EF.C.CH.DS(i). After selection of EF.C.CH.DS(i) according to the path information found in the CIO.C.CH.DS(i), either the certificate or related information may be read from the EF.C.CH.DS(i).

The actual content to be expected in EF.C.CH.DS(i) is described in DF.CIA. If no other description is available, then EF.C.CH.DS(i) contains the certificate.

Execution Flow 1 of 1

Table 34 — Read signer's certificate — command APDU

Command Parameter	Meaning
CLA	according to ISO/IEC 7816-4
INS	'B0' READ BINARY
P1-P2	'xxxx' Offset
Le field	'00' read all data

The response data field codes the data content of EF.C.CH.DS(i).

Table 35 — Read signer's certificate — response APDU

Response Parameter	Meaning
Data field	Data.EF.C.CH.DS(i)
SW1-SW2	Refer to ISO/IEC 7816-4

The command and response APDUs are described in ISO/IEC 7816-4:2013, 11.2.3.

The structure and coding of EF.C.CH.DS(i) will typically be an X.509 certificate format, but it could also be some other format. The actual format is application specific and out of the scope of this standard.

NOTE If the expected length exceeds the length of the APDU response data field then the READ BINARY might have to be called several times with an appropriate offset.

7.7.4 Retrieval of the signer's certificate from a directory service

In order to retrieve the signer's certificate through a directory service the card holder shall identify the certificate uniquely. This reference value information may be retrieved by reading the appropriate CIO.C.CH.DS from the DF.CIA(ESIGN) according to Figure 6.

According to ISO/IEC 7816-15:2004, 8.7.2 "X.509 certificate attributes" the appropriate data object CIO.C.CH.DS contains the following fields.

```
X509CertificateAttributes ::= SEQUENCE
{
  value ObjectValue { Certificate },
  subject Name OPTIONAL,
  issuer [0] Name OPTIONAL,
  serialNumber CertificateSerialNumber OPTIONAL,
  ... -- For future extensions
}
```

The interpretation of the components shall be as follows:

X509CertificateAttributes.value: The value shall be a *Referenced Value* either identifying a file containing a DER encoded certificate at the given location, or a URL pointing to some location where the certificate can be found (see ISO/IEC 7816-15).

8 Device authentication

8.1 General

This clause assumes that device authentication has to be performed as required in 8.4 "Authentication environments".

Device authentication requires mandatory steps in order to provide a secure authentication. A device authentication is mutual and combines two mechanisms:

- an ICC verifies the external world (TDA) and itself verified by the external world (CDA);
- the two devices negotiate or exchange information to establish common symmetric session keys for subsequent operations.

After negotiation of the symmetric keys, a secure session is established. A secure session is a cryptographic protection of the messages from both sides. The cryptographic protection can be:

- a cryptographic checksum on a plain text for integrity protection and/or
- an encrypted message text for confidentiality protection (mandates cryptographic checksum).

Refer to 10 "Secure Messaging".

For performance reasons, the secure messaging keys are symmetric keys. Therefore this document describes the establishment of symmetric session keys only, and does not consider an option for asymmetric session keys.

Once the session keys are established, a trusted channel is available to protect or conceal the information transmitted over the interface. The application of Secure Messaging (SM) is **mandatory** for **every subsequent operation** to ensure the provision of an entirely trusted channel. For exceptions refer to 5.3.2 "Selection of ESIGN application". The conditions to end a secure messaging session are given in 8.13 "Ending the secure session".

This clause describes the following device authentication mechanisms

- asymmetric session key transport mechanism based on RSA;
- asymmetric session key agreement mechanism with privacy protection;
- asymmetric session key agreement mechanism with non-traceability and privacy protection;

- symmetric authentication mechanism

in order to perform a mutual authentication protocol between IFD and ICC. The presentation of the asymmetric schemes is sequentially staged in order to trade off the security features with the required complexity of the authentication protocols.

This document also introduces password based mechanisms as PACEv2.

Authentication in general requires a defined order of steps to be processed. Violation of this order may result in the ICC aborting the process. The mechanism to control the proper order of execution steps are out of the scope of this standard.

In order to distinguish which device authentication is the most appropriate for a given situation refer to Annex B.

For the use of certificates the following table indicates the correct certificate type for each device authentication protocol.

Table 36 — Certificate type for use with device authentication protocols

Certificate Device Auth.	CV Certificate self-descriptive	CV Certificate non self-descriptive	Attribute certificates self-descriptive
8.6 Key transport protocol	-	x	-
8.7 Privacy protocol RSA	-	x	-
8.7 Privacy protocol ELC	x	-	x
8.8 mEAC	x	-	x

8.2 Asymmetric Authentication introduction

The above steps result in a high level of security for device authentication. They contain some mandatory aspects of the device authentication protocols considered to comply with the claims of the ESign-G1 [3] document.

- The public key of the IFD shall not be used until successfully verified by a certificate.
- The device to be authenticated shall submit its identity relevant token to the authenticator in a secure way, typically this can be achieved implicitly if the identity is part of the certificate.
- For both sides of the authentication it is mandatory to have evidence (after authentication), that the authentication was made with the same entity that shares the session keys for the secure messaging.
- After device authentication a trusted channel has been established. Relevant subsequent commands shall use the secure messaging session.
- A display message may be used to indicate successful device authentication in order to inform the card holder of the presence of a secure channel. The application of the display message is described in 8.14 "Reading the Display Message".

8.3 Certification authorities and certificates

8.3.1 Certificate chains

Asymmetric authentication requires the presentation of public keys in certificates. A chain of certificates might have to be presented. To establish the 'starting point of trust' the first certificate sent to the ICC shall always be verifiable with a public key already existing in the ICC. The most general of such keys available in the ICC

can be a Root CA public key. Each certificate introduces another public key, while the presented certificate is always signed with a key, trusted by the ICC.

As the key pair of the Root CA changes over time, link certificates provide a link between expired and valid key pairs (refer to 8.3.2 "Usage of link certificates").

The Root CA provides

- certificates for the subordinate CAs
- link certificates in order to convey the public key(s).

In order to use an ESIGN application a mutual authentication between ICC and customer terminal is required by the security policy. As PK algorithms are to be employed ICC and IFD authentication certificates are required.

The ICC is the carrier of the ICC authentication certificate; carrier of the IFD authentication certificate could be a plug-in ICC as an example for the realization of an authentication module inside the Signature Creation Application (SCA).

The handling of the DOs "CEDT — Certificate Effective Date Template" and "CXDT — Certificate Expiration date Template" are out of the scope of this standard.

8.3.2 Usage of link certificates

A typical use of link certificates is to provide a link between expired and valid Root-CA key pairs.

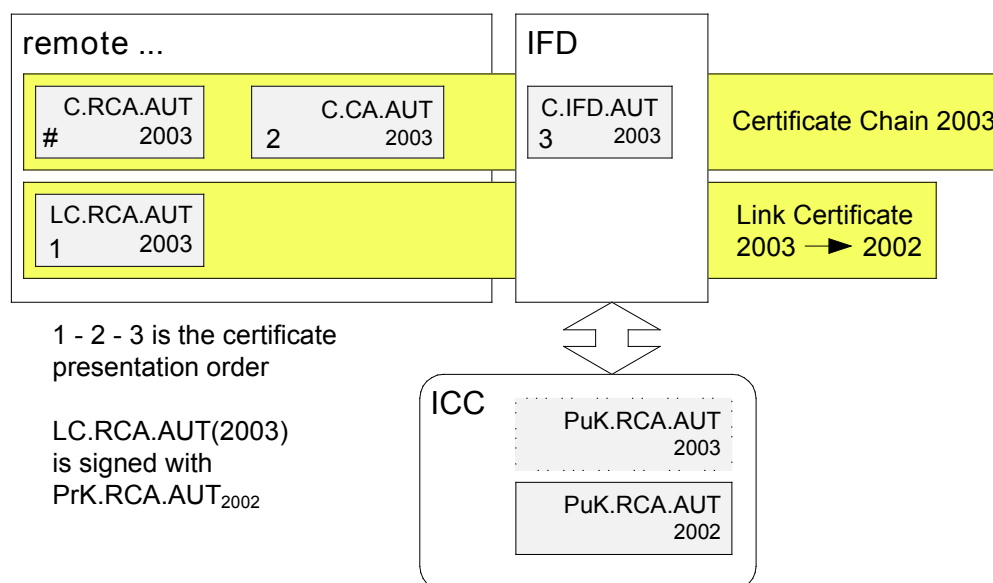


Figure 7 — Usage of link certificates for different key versions

The example above shows PuK.CA.AUT(2002) available at the ICC. This key was brought to the ICC in 2002, however in 2003 the root key was changed and another key chain

C.RCA.AUT(2003) → C.CA.AUT(2003) → C.IFD.AUT(2003)

was used when generating the IFD's certificate C.IFD.AUT(2003).

In order to allow authentication of the IFD, in the above example the ICC might have to receive a link certificate LC.RCA.AUT(2003) first. This certificate is signed with the PrK.RCA.AUT(2002) and it transports the PuK.RCA.AUT(2003).

After having verified the link certificate LC.RCA.AUT(2003) the ICC may now receive the certificates of the 2003-chain up to the C.IFD.AUT which carries the final public key required for authentication.

If it can verify the incoming certificate with an existing public key, the ICC accepts the transported public key and may use it for the next instance of the chain. The 'start of trust' is always established with the first certificate, being verified with a public key that is available in the ICC.

The example in Figure 7 considers a static Root CA key stored in the ICC. Hence, an IFD shall present a proper link certificate in every device authentication session.

Alternatively, an ICC may provide the possibility to permanently store the new Root CA key, which would require the import of a link certificate only once. If the trust anchor has been permanently updated within the ICC, the 'start of trust' for subsequent device authentications is the new Root CA key. The certificate chain to be presented by the IFD is shortened.

When verifying a certificate, the verifier shall verify the correct role of the signer. Only those roles being allowed to sign certificates may be accepted for the presentation of a certificate chain. The role of a certificate holder is specified in 15.7.4 "CHA-Certificate Holder Authorization Data Object (CHA-DO)".

8.4 Authentication environments

8.4.1 General

In Bibliographical Reference [3] two environments are distinguished with respect to signature creation applications.

8.4.2 SCA in trusted environment

The environment is considered to be trusted by the user. Device authentication is not required as the card holder knows the trusted environment that s(he) will apply the card to.

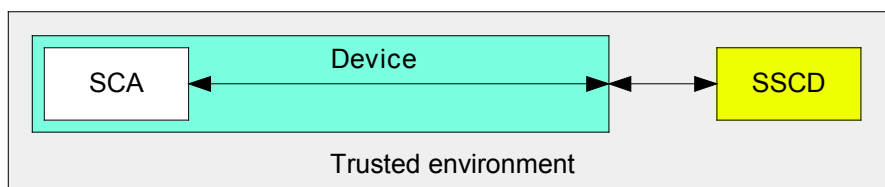


Figure 8 — Trust of the environment

8.4.3 SCA in untrusted environment

A device authentication shall be used if the operating environment of the ICC cannot be entirely trusted. This can be the case in public signature terminals or other devices, that cannot provide an a-priori secure channel.