Liberté • Égalité • Fraternité
RÉPUBLIQUE FRANÇAISE
Premier ministre
ANSSI

Bundesamt
für Sicherheit in der
Informationstechnik

Technical report

Signature creation and administration for eIDAS token

Version 1.0 Release Candidate 6

Date: 2014/03/17

Foreword

This technical report specifies an autonomous signature creation application embedded on an eIDAS token. Unless explicitly mentioned, this specification is compliant with all referenced standards.

This document has been written by ACSIEL (www.acsiel.fr) in close relation with ANSSI (www.ssi.gouv.fr) and ANTS (www.ants.interieur.gouv.fr).

TABLE OF CONTENT

Glossary

ADF	Application dedicated file
AES	Advanced encryption standard
AID	Application identifier
APDU	Application protocol data unit
BIO	Biometric data
CAN	Card access number
CHAT	Cardholder authorization template
DTBS	Data to be signed
EC	Elliptic curve
ECDSA	Elliptic curve digital signature algorithm
EFID	Elementary file identifier
GAP	General authentication procedure
IFD	Interface Device
MF	Master file
PACE	Password authenticated connection establishment
PCSE	Partial context security environment
PIN	Personal identification number
PUK	PIN unlocking key
RSA	Rivest Shamir Adleman
SCA	Signature creation application
SFI	Short file identifier
SHA	Secure hashing algorithm
SMT	Signature management terminal
SSCA	Secure signature creation application
SSCD	Secure signature creation device
UD	User device

1. Definitions

This chapter contains definition of concepts discussed all along this document.

1.1. User

Natural person holding the eIDAS token.

1.2. eIDAS token

The eIDAS token is a device based on a secure element that may have various form factors (smartcard, µSD,...). It MAY contain several applications (electronic ID, travel document,…), and in the scope of this document contains an application enabling the creation of electronic signature according to [EC_Regulation].

1.3. Electronic signature key

Private portion of an asymmetric key, used to create electronic signature in the sense of the [EC_regulation]. This key is identified in the signature application with the [ISO/IEC 7816-15] structure.

The key usage shall be compliant with the certificate policy of the issuer.

1.4. General Authentication procedure (GAP)

General authentication procedure as defined in [TR 03110v2 part2].

1.5. Global Personal Identification Number (PIN)

The global PIN is a user credential global to the whole eIDAS token. Depending on the configuration, it may be absent. When present, it is shared by the [eSign application] and the other applications.

1.6. Local PIN/BIO

A local PIN/BIO is a user credential that is local to the [eSign application]. Depending on the configuration, it may be absent.

1.7. Password Unblocking Key (PUK)

The PUK may be used to unblock a user credential: signature PIN/BIO, or a global PIN.

1.8. Qualified electronic signature key

Private portion of an asymmetric key, used to create qualified electronic signature in the sense of the [EC_regulation]. This key is identified in the signature application with the [ISO/IEC 7816-15] structure.

A qualified electronic signature key is associated with a qualified electronic certificate issued by a qualified authority.

The key usage shall be compliant with the certificate policy of the issuer.

1.9. Secure signature creation application (SSCA)

The secure signature creation application ([eSign application]) is the signature application contained in the eIDAS token enabling the creation of electronic signature. More details about the secure signature creation application can be found in §5.

1.10. Signature creation application (SCA)

The signature creation application ([SCA]) is the application requesting the creation of a digital signature from the eIDAS token. An external application (text editor, mail software,…) – executed on a service provider, local computer,…- authorized to request a signature creation and for which the user has expressed its consent.

1.11. Signature management terminal (SMT)

Terminal entitled to interact with the [eSign application] embedded on the eIDAS token. The signature management terminal is an authentication terminal (AT) as defined in [TR 03110v2] whose certificate contains the extensions laid down in §3.1.

1.12. User credential

User credentials are defined in [TR 03110v2 Part2]. Unless otherwise specified, the ‚user credentials are the PIN, PUK, Bio and CAN.

1.13. User device (UD)

The user device is the device the user directly interacts with. It is always local, hence physically accessible by the user. This device is used to enter a user credential or PUK (PIN/BIO) and provides an IFD (interface device) to communicate with the eIDAS token. It may be a computer, a mobile phone or a pad.

2. Introduction

Using the [eSign application], the following services are provided:

a. The signature creation can be performed either via contact or contactless interface;

b. The signature creation is under control of the holder independently of the communication interface;

c. The received data to be signed only comes from the intended signing application;

d. The user consent with user authentication is done in a secure way, to ensure protection of the authentication data. User authentication via PIN code or Biometric data ensures authentication of the given agreement;

The [eSign application] is designed to be embedded in an eIDAS token (smartcard, SIM, µSD,…). It is able to operate alone: it does not require the presence of any other applications (e.g. eID application).

3. Environment of the eIDAS token

The eIDAS token SHALL

a. Manage the authorization of an external entity;

b. Manage on card key generation and keep the private key secret;

c. Ensure user consent through the dedicated signature password (PIN code) or biometric data (BIO);

d. Compute the digital signature on card;

To compute an electronic signature, three entities are involved:

e. **Signature Creation Application [SCA]**: application (that can be local or remote) requesting a signature, which use [UD] to send/receive APDU;

f. **User Device [UD]**: manages the local transmission and reception of APDUs with the eIDAS token;

g. **The eIDAS token**, holding the embedded electronic signature application named the [eSign application]

This specification focuses on end to end communications between the [eSign application] and:

h. The [SCA] through a dedicated protected channel [CH_SCA];

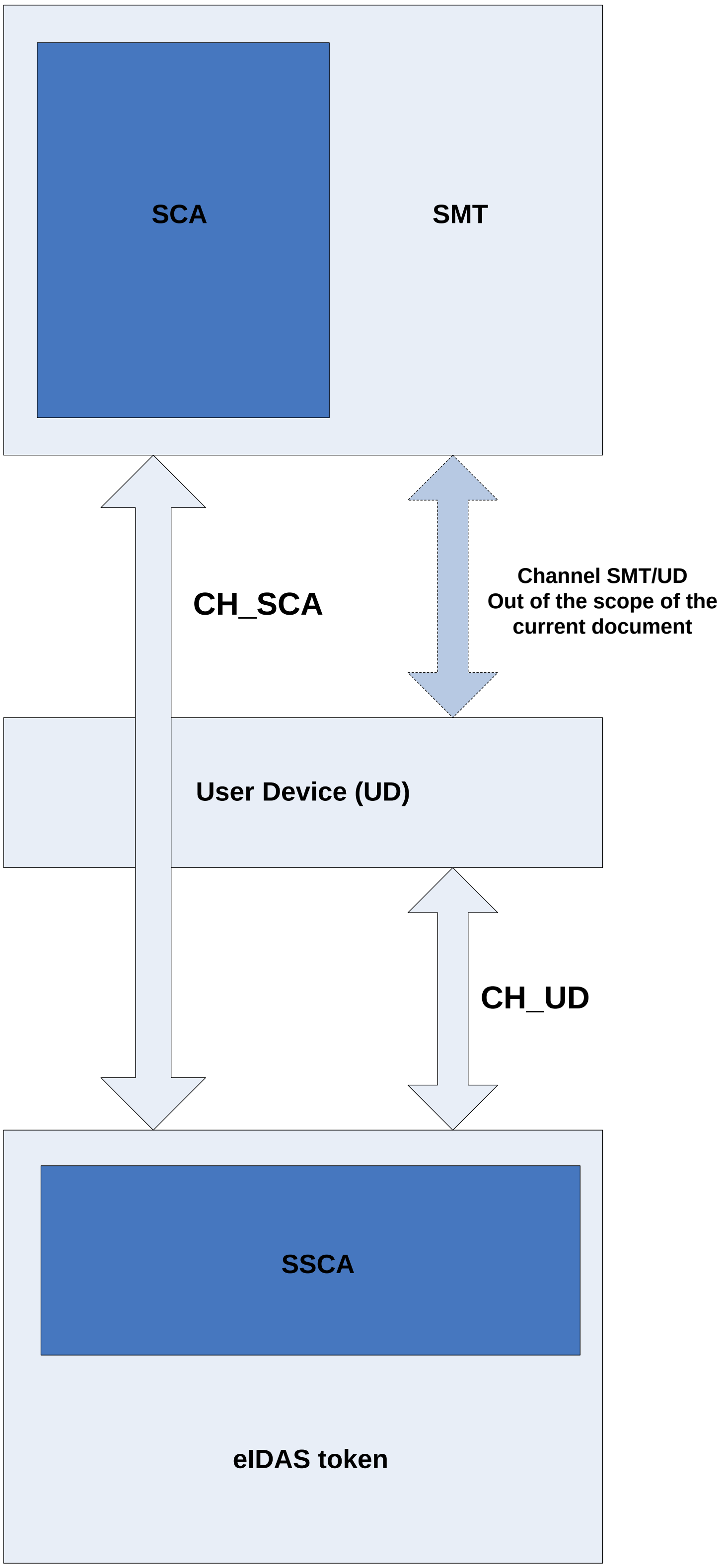
i. The [UD] through a dedicated protected channel [CH_UD];

The communication channel between the [SCA] and [UD] is out of the scope of this specification.

The [UD] includes an [IFD] in charge of the communication with the eIDAS token. The [UD] MAY operate with a keypad and/or display, with a local or remote [SCA].

The [SCA] and [UD] are complementary sub-parts of an eIDAS compliant terminal. Thus, this terminal is able to authenticate with the eIDAS token and the underlying [eSign application]. The [SMT] (see §3.1) SHALL use the General Authentication Procedure. The [SMT] MAY use other authentication procedures.

The following figure summarizes the environment of the eIDAS token:



Environment of the eIDAS token

3.1. Signature management terminal

Depending on the rights the [SMT] has been granted, it can request the [eSign application] to:

a. create electronic signature;

b. manage its elements (PIN code, Biometric data, keys and files);

c. manage the [eSign application] itself;

As such, the [SMT] MAY be used by the following entities:

d. [SCA] when the [SMT] is entitled to create electronic signature;

e. Certificate provider for the signature service when the [SMT] is entitled to generate signature keys and upload certificates;

f. Identity provider of the user when the [SMT] is entitled to update the content of the file EF.INFO4CERT (if present);

g. Administrator of the [SCA] when the [SMT] is entitled to [eSign application] management;

The [SMT] is an authentication terminal as defined in [TR 03110v2] whose certificate contains the following certificate extensions (structured as defined in [TR 03110v2]):

The following Object Identifier SHALL be used to indicate the authorization extensions related to the [SMT]:

Id-SMT OBJECT IDENTIFIER :: = {1.2.250.1.223.1001.1.1}

The following context specific data SHALL be used in any certificate of the chain during GAP:

0x80 : Relative authorization of the [SMT];

0x81 : Maximum number of consecutive signatures;

For more information about certificate extensions management, refer to [TR 03110 v2 part3] §C.3.

The effective authorization of the [SMT] is computed as described in [TR 03110 v2 part3] §2.7.

3.1.1. Authorization encoding

Note: The proposed description of the mechanism is provided as a first draft and is likely to change in further versions.

This field SHALL be present in the certificate extensions. The authorization of [SMT] is encoded as described below:

General			**Services**		**Role description**
1st byte			2rd byte	3rd byte	
8 7	6 5	4 3 2 1	8 7 6 5 4 3 2 1	8 7 6 5 4 3 2 1	
1 1					C VCA
1 0					DV (Accreditation Body)
0 1					DV (Certification Service Provider)
0 0					Signature And Management Terminal
	1 0				Local only
	0 1				Remote only
		x x x x			RFU
			x x x x x x x x	x x x x x x x x	Access Rights
			1- - - - - - -	- - - - - - - - -	Generate Electronic signature • PSO CDS on a key • Read EF.CD, EF.cert associated
			- 1 - - - - - -	- - - - - - - - -	Generate Qualified Electronic signature • PSO CDS on a key • Read EF.CD, EF.cert associated
			- -1 - - - - -	- - - - - - - - -	Install electronic Certificates • Read EF.INFO4CERT • Generate keys • UpdateEF.CD[x] (dedicated) • [optional] write EF.cert associated
			- - -1 - - - -	- - - - - - - - -	Install Qualified electronic Certificates

					• Read EF.INFO4CERT • Generate keys • UpdateEF.CD[y] (qualified one) • [optional] write EF.cert associated
			- - - - 1 - - -	- - - - - - - - -	SSCA Management • Life Cycle
			- - - - - 1 - -	- - - - - - - - -	Signature Keys management • Life Cycle • Update [ISO/IEC 7816-15] files
			- - - - - - - 1 -	- - - - - - - - -	File EF.INFO4CERT Management • Update
			- - - - - - - - 1	- - - - - - - - -	User credential management • Unblock PIN/BIO
			- - - - - - - - -	1 - - - - - - -	User credential management • Change PIN/BIO value
			- - - - - - - - -	- 1 - - - - - -	User credential management • Life Cycle • Update [ISO/IEC 7816-15] file
			- - - - - - - - -	- - 1 - - - - -	User credential management • Initialization

3.1.2. Maximum number of consecutive signatures

The effective maximum number of consecutive signature the [eSign application] is allowed to perform is computed by taking the smallest value amongst all the value received in the certificate chain within the DO “Maximum number of consecutive signatures”.

The [eSign application] SHALL not exceed the effective maximum number of consecutive signature obtained during the GAP. Once the number of consecutive signature has been met, the [eSign application] SHALL require new user consent.

4. Communication, protocol and security mechanisms

In contact and contactless, the [eSign application] SHALL mandate secure messaging ensuring integrity, authenticity and confidentiality for the communication with the [UD], once PACE authentication is established.

4.1. Communication protocols

The eIDAS token SHALL support the communication protocols defined in [TR 03110v2 Part2].

4.2. Security protocols

The [eSign application] mandates global authentication services to be provided in the master file (MF) of the eIDAS token. These global authentication services are:

a. PACE as defined in [TR 03110v2 part2]. In the scope of this technical report, PACE SHALL be performed under the MF, and before selecting the [eSign application];

b. GAP as defined in [TR 03110v2 part2]; This protocol allows to update internal date and to validate the effective authorization of the remote terminal through the CHAT on the [eSign application];

4.3. Secure messaging

After a successful authentication using PACE or GAP (as defined in [TR 03110v2 part2]), the secure messaging SHALL comply with the one specified in [TR 03110v2 part3].

4.4. User consent

Depending of the configuration, the global PIN and/or local PIN/BIO are considered as the user consent.

4.4.1. [eSign application]User consent to access to the eIDAS token

The holder must express his agreement for the physical use of the eIDAS token. This operation is performed using PACE with a global credential.

PACE SHALL be performed in the MF as described in [TR 03110v2 part1] and [TR 03110v2 part2] with the global user credentials available in the eIDAS token.

The different PACE configurations supported are indicated in the file EF.CardAccess (as defined in [TR 03110v2 part3]). This file SHALL be present.

4.4.2. User consent to access to the [eSign application]

Note: The proposed description of the mechanism is provided as a first draft and is likely to change in further versions.

In order to use the [eSign application], the user needs to express his agreement. This operation is performed by using a user credential. Depending on the configuration of the [eSign application], it SHALL be one of the following:

a. a local PIN/BIO(s) stored in the [eSign application];

b. the global PIN stored in the MF (global PIN of the eIDAS token);

The verification of this user credential SHALL be performed using:

c. VERIFY command for validation of local PIN/BIO(s) or global PIN. It allows submitting the PIN/BIO. It MAY be performed in the [eSign application], to submit the local PIN/BIO, or the global PIN. This command SHALL be executed under secure messaging;

d. PACE for validation of global PIN;

Notes

When the global PIN is also the signature PIN, the user consent for the physical access to eIDAS token, and logical access to [eSign application] is merged. As a result, a successful PACE performed with the global PIN grant access to both the eIDAS token and the [eSign application].

4.4.3. User consent and operations

The table below indicates for each use case the user credential that is mandated to get access to the said service.

Operation	Type of data required	Mode of operation
Generate electronic signature Generate qualified electronic signature	It SHALL be the user credential protecting the signature creation function of the key. Depending on the configuration of the eIDAS token, one of the following user credential SHALL be used: -the global PIN(1) -the local PIN/BIO (2) Note : the user credential (Global PIN & local PIN/BIO) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	**Case 1** -PACE + VERIFY Global PIN -PACE with Global PIN In case the user consent is lost (see §7.2.1 and §7.2.2), additional VERIFY Global PIN may be performed. **Case 2** -PACE + VERIFY local PIN/BIO
Install electronic certificate Install qualified electronic certificate	It SHALL be the user credential protecting the signature key generation. Depending on the configuration of the eIDAS token, one of the following user credential SHALL be used: -the global PIN(1) -the local PIN/BIO (2) Note : the user credential (Global PIN & local PIN/BIO) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	**Case 1** -PACE + VERIFY Global PIN -PACE with Global PIN In case the user consent is lost (see §7.2.1 and §7.2.2), additional VERIFY Global PIN may be performed. **Case 2** -PACE + VERIFY local PIN/BIO
[eSign application] management	CAN, PUK or Global PIN Note : the user credential (Global	PACE

User credential management (Change)	PIN) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	
Signature keys management EF.INFO4CERT management User credential management (Initialization)	Global PIN Note : the user credential (Global PIN) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	-PACE with Global PIN -PACE + VERIFY Global PIN
User credential management (Unblock)	PUK Note : the user credential (PUK) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	PACE with PUK
User credential management (Life cycle)	Global PIN or PUK Note : the user credential (Global PIN & PUK) to use is the one protecting the access to such operations (several user credentials MAY be present in the eIDAS token).	-PACE with Global PIN -PACE + VERIFY Global PIN -PACE with PUK

5. Secure signature creation application [eSign application]

The [eSign application] is an Application Dedicated File (ADF) as defined in [ISO/IEC 7816-4], located under the master file (MF) of the eIDAS token.

5.1. Prerequisite

The [eSign application] does not require any other application to be present on the eIDAS token. It can be the sole application present on the eIDAS token.

The eIDAS token MAY have an MRZ, but the [eSign application] SHALL NOT be accessible using PACE with MRZ.

5.2. Selection of the application

The selection of the [eSign application] SHALL be made under secure messaging. The AID of [eSign application] used for selection SHALL be the following, as defined in [EN419212]:

A0 00 00 01 67 45 53 49 47 4E

Notes

For privacy reason, the [eSign application] selection MAY be restricted to Authentication terminals. The eIDAS token SHALL return “file not found” for unauthorized selection.

5.3. Life cycle of the [eSign application]

[eSign application] has a life cycle compliant with [ISO/IEC 7816-9] and supports the following states:

a. Operational state – activated : in this state the [eSign application] is usable. It contains all the data objects and files as defined in §6;

b. Operational state – deactivated: in this state the [eSign application] is created and selectable. In response to the selection, it SHALL indicate that its state is deactivated. The [eSign application] services allowed are switching to Operational state – activated or Termination state.

c. Termination state: in this state, the [eSign application] is irreversibly unusable;

The [eSign application] may be in state “Operational state-activated” or “Operational state – deactivated” at issuance.

5.4. Initialization after issuance

The [eSign application] SHALL manage all elements required for signature creation, including the global PIN, PUK, and local PIN/BIO. [eSign application] SHALL be able to handle empty objects and initialized ones.

The signature creation service of the [eSign application] is not available if:

a. the private key(s) for electronic signature or the corresponding user credential has not been initialized : the content of the object is empty and SHALL be assigned first;

b. the private key(s) for electronic signature, or the corresponding user credential is not in state “operational – activated”;

c. the user credential is blocked (the number of tries has reached the maximum number authorized);

6. Data elements of the [eSign application]

The [eSign application] manages the following types of data:

a. PACE credentials

   These data are credentials that can be used in a PACE

   i. Global PIN;

   ii. CAN;

   iii. PUK;

b. User credentials

   These data are credentials used to authenticate the user and unlock access to the [eSign application]. Depending on the configuration of the [eSign application], they MAY unlock the access to the application (global PIN), some function of it, or unlock the usage of the signature creation function;

   i. [eSign application] PIN (either a local PIN or global PIN);

   ii. [eSign application] BIO;

   They may be used either through a VERIFY command, or with a PACE protocol. These credentials are described in §6.1.

c. Signature keys

   They are cryptographic keys that may be of RSA or EC type depending on the configuration of the [eSign application]. They may be used to create qualified electronic signature or electronic signature in the sense of [EC_Regulation]. The usage of the signature key is indicated in the [ISO/IEC 7816-15] structure;

   i. private key(s) for electronic signature;

   ii. private key(s) for qualified electronic signature;

   These credentials are described in §6.3.

d. Transparent files as defined in [ISO/IEC 7816-4]. These files are used to store the [ISO/IEC 7816-15] structure as well as other information. They are described in §6.4.

All the elements listed above SHALL be identified in the [ISO/IEC 7816-15] structure as described in §9.

6.1. User credentials

User credentials are PIN/BIO data used by eIDAS token to:

a. Authenticate the user;

b. Express the consent of user before creating an electronic signature;

c. allow the holder to express his consent and to authenticate himself prior any operation he wishes to perform with the eIDAS token;

When delivered, the object(s) stored in the [eSign application] SHALL be in one of the following state:

d. Initialized. The user credential is already loaded.

e. Uninitialized. The user SHALL initialize the user credential prior any usage.

Notes

The current technical report does not limit the number of user credential contained in the eIDAS token. Depending on the configuration of the eIDAS token, one or several user credential may be present (local and/or global), and each of them may protect the access to one or several operations.

6.1.1. Available operations

The following operations may be performed on a user credential:

a. **Verification**: this operation submits a candidate user credential to the eIDAS token that compares it against the reference user credential. Upon success the following actions are performed

   - the user credential verification status is set;
   - the corresponding access rights are granted;
   - the retry counter is restored to its initial value;

   Upon failure, the following actions are performed:

   - the user credential verification status is reset;
   - the corresponding access rights are denied;
   - the retry counter is decremented by one;

b. **Change**: this operation changes the reference user credential values stored in the eIDAS token. If successful, the user credential verification status is reset"

c. **Devalidation**: this operation resets the user credential verification status.

d. **Unblocking** : this operation consists in unblocking the user credential, namely restoring its retry counter to the initial value (described below), resetting its verification status, and changing its reference value.

Notes

In order to identify the user credential on which the operation SHALL be performed, the identifier of the user credentials SHALL be provided in the field "reference data" of the command. All user credentials share the same range of identifiers whether they are PIN or BIO objects. For example, if a PIN credential has the identifier #1, no BIO credential SHOULD have the identifier #1. Identifiers SHALL be in range 1 to 31 included.

6.1.2. Life cycle state

User credentials have a life cycle compliant with [ISO/IEC 7816-9] and support the four following states:

a. **Initialization state**: in this state the data container is created, but the data (PIN/BIO) has not been initialized yet. The user credential usage is restricted. The global PIN MAY be used for PACE. Other restrictions are described in §8.

b. **Operational state – activated** : in this state the data container is created, filled with a PIN/BIO, and usable;

c. **Operational state – deactivated**: in this state the data container is created, filled with a PIN/BIO, and its usage is restricted;

d. **Termination state**: in this state, the data container is irreversibly unusable;

eIDAS token allows transition between these states for each of the object it contains. The transitions and the command used to perform these transitions are compliant with [ISO/IEC 7816-9].

6.1.3. Attributes

User credentials SHALL contain the two following attributes:

a. **Retry counter**: counter indicating the number of remaining tries for the verification of the user credential. This counter is persistent, meaning it is not reset upon reset or eIDAS token selection. It is decremented on a wrong verification, and restored to its initial value (described below) upon successful verification of the user credential. When this counter has reached ’00’, the user credential becomes unusable. Prior any other use, it SHALL be unblocked (see §7.3.5).

b. **Initial value of retry counter**: value indicating the maximum number of incorrect verification allowed by the user credential. The retry counter is reset to the initial value in case of successful verification/unblock/change.. The initial value may take any value between ’1’ (decimal) and ‘15’ (decimal) and its value is indicated in the [ISO/IEC 7816-15] structure. No modification SHALL be possible once this value is set.

6.2. PIN unblocking keys (PUK)

PIN unblocking key (PUK) aims at unblocking user credentials once their retry counter has reached zero. A PUK may be either a PIN or BIO object and SHALL be located in the MF. The PUK may be a blocking or unblocking PIN or BIO.

Its usage MAY be limited thanks to a usage counter.

Notes

The current technical report does not limit the number of PUK contained in the eIDAS token. Depending on the configuration of the eIDAS token, one or several PUK may be present, and each of them may unlock a given user credential.

6.2.1. Available operations

The following operation may be performed on PUK:

a. **Verification**: this operation submits a candidate PUK to the eIDAS token that compares it against the reference PUK.

   Upon success the following actions are performed

   - the PUK verification status is set;
   - the corresponding access rights are granted;
   - if exists, the retry counter is restored to its initial value;
   - if exists, the usage counter is decreased by one;

   Upon failure, the following actions are performed:

   - the PUK verification status is reset;
   - the corresponding access rights are denied;
   - if exists, the retry counter is decremented by one;
   - if exists, the usage counter is unchanged;

Notes

In order to identify the PUK on which the operation SHALL be performed, the identifier of the PUK SHALL be provided in the field “reference data” of the command.

6.2.2. Life cycle state

PUK has a life cycle compliant with [ISO/IEC 7816-9] and supports the following state:

e. **Operational state – activated** : in this state the data container is created, filled with a value, and usable;

f. **Termination state**: in this state, the data container is irreversibly unusable;

eIDAS token allows transition between these states for each of the object it contains. The transitions and the command used to perform these transitions are compliant with [ISO/IEC 7816-9].

6.2.3. Attributes

PUK MAY contain the two following attributes:

c. **Retry counter**: counter indicating the number of remaining tries for the verification of the PUK. This counter is persistent, meaning it is not reset upon reset or eIDAS token selection. It is decremented on a wrong verification, and restored to its initial value (described below) upon successful verification of the PUK. When this counter has reached ’00’, the PUK becomes unusable.

d. **Initial value of retry counter**: value indicating the maximum number of incorrect verification allowed by the PUK. The retry counter is reset to the initial value in case of successful verification. The initial value may take any value between ’1’ (decimal) and ‘15’ (decimal) and its value is indicated in the [ISO/IEC 7816-15] structure. No modification SHALL be possible once this value is set.

Moreover, a PUK MAY contain the following attributes:

e. **Usage counter**: counter indicating the number of time the PUK can be successfully verified. This counter is persistent, meaning it is not reset upon reset or eIDAS token selection. It is set at creation with an initial value, and decremented after a successful verification. When this counter has reached ‘00’, the PUK becomes unusable.

f. **Initial value of usage counter**: Number of time the PUK can be successfully verified.

6.3. Signature keys

Signature keys are cryptographic keys that may be of RSA or EC type depending on the configuration of the [eSign application].

Notes

The current technical report does not limit the number of signature keys contained in the [eSign application]. Depending on the configuration, one or several signature key may be present, each of them having its own usage: qualified electronic signature or electronic signature.

The [ISO/IEC 7816-15] structure MAY declare the signature keys present in the eIDAS token in the following ways:

- the certificate(s) associated to the signature keys should be stored in EF.CD[x] files;
- the attribute “key usage” of a signature key used for non qualified signature should be set to “sign”;
- the attribute “key usage” of a signature key used for qualified signature should be set to “non repudiation”;

6.3.1. Available operations

In the scope of the current technical report, signature key import is not considered. The signature keys MAY be generated on board, several times over the life time of the [eSign application].

The signature keys can be used to create qualified electronic signature or electronic signature in the sense of [EC_Regulation]. The usage of the signature key is indicated in the [ISO/IEC 7816-15] structure. For security reasons, a signature key SHALL have a unique usage, i.e. it SHALL not be usable to create a qualified electronic signature and an electronic signature.

When delivered, the object(s) stored in the [eSign application] SHALL be in one of the following state:

a. Initialized. The key value is present.

b. Uninitialized. The user SHALL initialize the key prior any usage through a key generation.

<u>Notes</u>

In order to identify the signature key on which the operation SHALL be performed, the identifier of the signature key SHALL be provided. Signature key share the same range of identifier whether they are RSA or EC keys. For example, if a RSA signature key has the identifier #1, no EC signature key should have the identifier #1. Identifiers SHALL be in range from 1 to 31 included.

6.3.2. Life cycle state

Signature keys have a life cycle compliant with [ISO/IEC 7816-9] and support the four following states:

a. **Initialization state**: in this state the data container is created, but the data (signature key) has not been initialized yet. The signature key is not usable;

b. **Operational state – activated** : in this state the data container is created, filled with a signature key, and usable;

c. **Operational state – deactivated**: in this state the data container is created, filled with a signature key, and its usage is restricted;

d. **Termination state**: in this state, the data container is irreversibly unusable;

The [eSign application] manages transitions between these states for the objects it contains. The transitions and the command used to perform these transitions are compliant with [ISO/IEC 7816-9].

6.4. Files

The [eSign application] supports elementary files with transparent structure as defined in [ISO/IEC 7816-4].

Notes

The current technical report only considers the [eSign application] which is an ADF as defined in [ISO/IEC 7816-4], and elementary files stored under this ADF. The support of other types of files is out of the scope of the current technical report.

6.4.1. File attributes

The transparent files may be selected either by EFID, or SFI. While the support of EFID is mandatory, the support of SFI is optional.

a. The EFID SHALL be indicated in the [ISO/IEC 7816-15] structure for each file.

b. The SFI MAY be indicated in the [ISO/IEC 7816-15] structure for file.

6.4.2. Available operations

The [eSign application] supports the following operations on files:

a. Selection : the current technical report only envisions file selection by EFID or SFI as defined in [ISO/IEC 7816-4];

b. Reading : the file reading is compliant with [ISO/IEC 7816-4];

c. Updating : the file updating is compliant with [ISO/IEC 7816-4];

Notes

When importing a certificate (at signature key initialization or regeneration), the file dedicated to store the certificate in the [ISO/IEC 7816-15] structure, SHALL be large enough to allow storage of the data. The management of file resizing, in case the size of the certificate exceeds the size of the file is out of the scope of this technical report. Therefore, it is recommended to create the file dedicated to store the signature key certificate large enough.

6.4.3. Life cycle state

Files only have one life cycle compliant with [ISO/IEC 7816-9]:

a. Operational state – activated;

6.5. File Structure

When delivered, the [eSign application] SHALL contain a minimum set of files.

The files are also used to store the [ISO/IEC 7816-15] structure that allows the [UD] to discover the content of the [eSign application] and its features.

Four types of information can be present in the eIDAS token:

- Configuration information ;
- Public information ;
- Private information ;
- Other information ;

Configuration information

The following files are used to store the configuration information:

a. The file EF.DIR SHALL be present;

b. The files EF.CardAccess SHALL be present. For more details refer to [TR 03110v2 part3].

c. The files EF.CardSecurity and EF.ChipSecurity MAY be present. For more details refer to [TR 03110v2 part3].

d. The file EF.ATR/INFO (defined in [ISO/IEC 7816-4]) MAY be present.

This information is public information that is used by the [UD] to discover the capabilities of the eIDAS token. As such, it does not hinder the privacy of the holder. Any other data that could hinder the user privacy SHALL not be stored within these files.

Public information

The following files are used to store the public information:

e. The file EF.CIAInfo SHALL be present;

f. The file EF.OD SHALL be present;

g. The file EF.AOD for user credentials and PUK SHALL be present;

h. The file EF.PrKD for electronic signature keys SHALL be present;

i. The file EF.CD[x] for signature certificate SHALL be present;

j. The file EF.DCOD for the description of other files and objects MAY be present. This file MAY reference EF.INFO4CERT flagged using the string “EF.INFO4CERT”, or contain an URL.

This information is public information that is used to discover the content of the eIDAS token. As such, it may hinder the privacy of the holder. In order to protect the privacy of the holder, it SHALL be readable after access to [eSign application] is granted and MAY require further access conditions (user consent or dedicated role).

Notes

The eIDAS token SHALL hold one EF.CD[x] file per certificate.

The file EF.CD[x] MAY contain the signature certificate or a reference to it (e.g. URL).

Private information

The following files are used to store the private information:

k. The file EF.INFO4CERT, which is also a transparent file, MAY be present. It is intended to store all the data related to the user that are needed to generate signature key certificate (name, surname,…). This file can be omitted if this information can be found somewhere else, or if no signature key generation (certificate generation step) is required during the eIDAS token life.

l. The certificates files MAY be present

This information is private information that contains information about the eIDAS token and/or its holder. In order to protect the privacy of the holder, it SHALL be readable after access to [eSign application] is granted.

Other information

The [eSign application] MAY also contain other files. In such case, a great attention SHALL be paid to the type of information stored in these files, and the access conditions applied to them, in order not to endanger the privacy of the user.

For more information about the [ISO/IEC 7816-15] structure, refer to Annex B: Example of [ISO/IEC 7816-15] structure.

Notes

Warning: the files protected by the [eSign application] user credentials SHOULD be read first before signature creation

6.6. PIN and Biometric data format

PIN and Biometric data SHALL be used for the user credentials and the PUK. This chapter provides information about their format and structure.

6.6.1. Format

The PIN SHALL be encoded as an octet string. [ISO/IEC 7816-15] structure MAY provide information about PIN in the file EF.AOD (maximum PIN size,…)

BIO may be of any type of biometric data (fingerprint, iris,..). The type of biometric data, as well as its format SHALL be described in the [ISO/IEC 7816-15] structure.

Notes

The current technical report does not impose any specific format or encoding for the PIN. It SHALL be an octet string.

The PIN length, the PIN policy, the PIN format, and the FAR when using PIN/BIO is out of the scope of the current technical report. It is up to the issuer to set up its own recommendations.

The technical report is neutral and does not consider any specific biometry technology. As such, any type of biometry can be used.

6.6.2. Specific interoperability concerns for biometric

Biometric data SHALL be handled in compliance with [ISO/IEC 7816-11].

The BIT/BITg corresponding to the Biometric data SHALL be made available in a file declared in the [ISO/IEC 7816-15] structure. These data structure SHALL be protected in a manner ensuring privacy protection (PACE).

7. Available [eSign application] services

The terminal should read the EF.DIR from the [ISO/IEC 7816-15] structure to discover which security mechanism is applied on the [eSign application]. Execute the GAP process to open the access condition.

7.1. Life cycle management of the [eSign application]

The [eSign application] enables to manage its life cycle, namely by locking/unlocking (suspension of its usage) and termination. The process flow is the following:

a. GAP;

b. Selection of the [eSign application];

c. Manage the life cycle of the [eSign application](through commands ACTIVATE, DEACTIVATE and TERMINATE);

PACE SHALL be performed according to [TR 03110v2 part2] and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege "[eSign application] management" as defined in §.3.1.1

In the state "Operational – Deactivated", the [eSign application] SHALL NOT execute any of the operational command except "ACTIVATE" or "TERMINATE".

The state "Operational – Deactivated" is reversible.

The state "Terminated" is irreversible as described in [ISO/IEC 7816-8].

Notes

Information about the state of the [eSign application] is indicated by the status word returned by the SELECT command, as defined in [ISO/IEC 7816-4].

7.2. Signature creation

A digital signature operation can only be executed provided the following conditions are met:

a. The user has given his consent;

b. There is an active secure messaging;

User consent may be expressed to the [eSign application] as described in §4.4.3.

The secure messaging results from the protocols described in §4.2

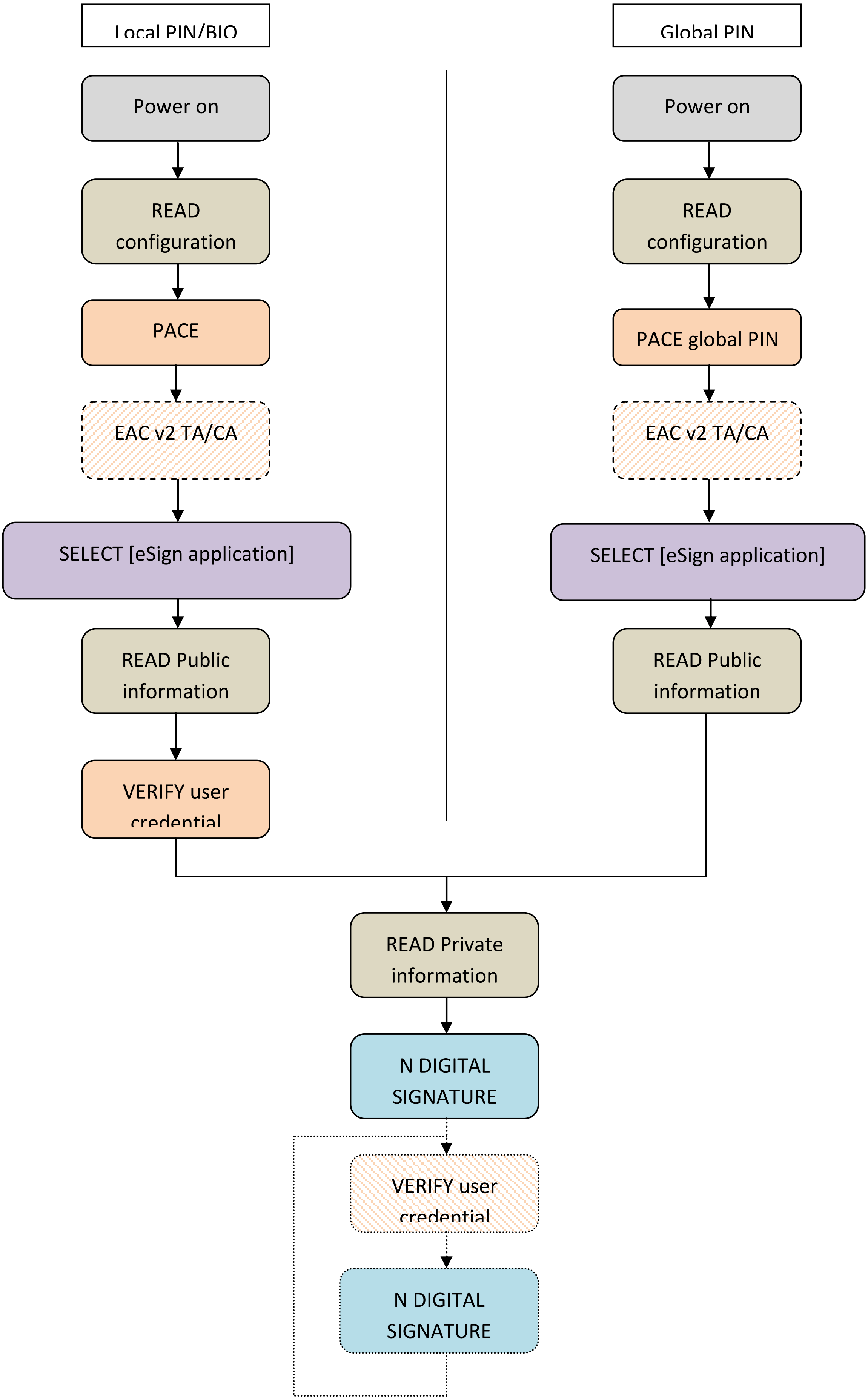
Once the user has given the consent for creating a signature, the [eSign application] can process the data and return the computed signature.

The signature is created using the following APDU commands:

c. MSE SET – DST to select the signature key, and implicitly the hashing and signature algorithm;

d. PSO - HASH (conditional) to compute the message digest;

e. PSO - COMPUTE DIGITAL SIGNATURE to compute digital signature;



Local PIN/BIO
Global PIN
Power on
READ configuration
PACE
EAC v2 TA/CA
SELECT [eSign application]
READ Public information
VERIFY user credential
Power on
READ configuration
PACE global PIN
EAC v2 TA/CA
SELECT [eSign application]
READ Public information
READ Private information
N DIGITAL SIGNATURE
VERIFY user credential
N DIGITAL SIGNATURE

7.2.1. Discovery mechanism

When accessed, the eIDAS token provides an [ISO/IEC 7816-15] based card discovery mechanism.

The access conditions shall be fulfilled before accessing any [ISO/IEC 7816-15] data containing privacy related data

As the configurations of [eSign application] requiring or not authentication of the [SMT] are exclusive, the terminal SHALL discover the configuration of the [eSign application] in order to request an electronic signature creation.

The terminal SHALL apply the following discovery mechanism:

a. Read the EF.DIR from the [ISO/IEC 7816-15] structure. It SHALL require PACE or GAP protocol to be performed using the global PIN or the CAN (if present). After successful completion of the protocol, the terminal SHALL select [eSign application].

b. After successful selection of the [eSign application], the [SMT]/terminal SHALL explore the [ISO/IEC 7816-15] structure, and:

   i. Discover the list and location of signature key available in the [eSign application], as well as their life cycle and usage

   ii. Choose one signature key among the list that is in operational-activated state, matching the required usage;

   iii. Identify the user credential required to create electronic signature with the signature key it has chosen, and check its life cycle;

At this stage, three cases (case 1a, case 1b and case 2) SHALL be sorted out, depending on whether the global PIN is the signature PIN or not.

7.2.2. Case 1: the user credential signature is the global PIN

Case 1a): If PACE with the global PIN has been performed, under secure messaging the [SMT]/[UD]:

a. SHALL select the signature key to be used;

b. SHALL send the data to be signed to the [eSign application];

For each subsequent signature creation required, under secure messaging the [SMT]/[UD]:

c. SHALL submit the global PIN using the VERIFY command; [CONDITIONAL] to usage of consecutive signature mode (see §3.1.2).

d. MAY select the signature key to be used;

e. SHALL send the data to be signed to the [eSign application];

Case 1b): If PACE with the CAN has been performed, under secure messaging the [SMT]/[UD]:

a. SHALL submit the global PIN using the VERIFY command;

b. SHALL select the signature key to be used;

c. SHALL send the data to be signed to the [eSign application];

For each subsequent signature creation required, under secure messaging the [SMT][UD]:

d. SHALL submit the global PIN using the VERIFY command;[CONDITIONAL] to usage of consecutive signature mode (see §3.1.2).

e. MAY select the signature key to be used;

f. SHALL send the data to be signed to the [eSign application];

7.2.3. Case 2: the user credential signature is NOT the global PIN

Under secure messaging, the [SMT]/[UD]:

a. SHALL submit the local PIN/BIO using the VERIFY command;

b. MAY select the signature key to be used;

c. SHALL send the data to be signed to the [eSign application];

For every subsequent signature, under secure messaging the [SMT]/[UD]:

d. SHALL submit the user credential (local PIN/BIO) using the VERIFY command; [CONDITIONAL] to usage of consecutive signature mode (see §3.1.2).

e. MAY select the signature key to be used;

f. SHALL send the data to be signed to the [eSign application];

Notes

The [SMT]/terminal SHALL iterate the two last operations to create signature as long as required and allowed by the [eSign application] (before the user credential status is not verified)

7.2.4. Signature creation with authentication of the [SMT]

The sequence of operation described in §7.2.1 SHALL be applied. In this case, the eIDAS token SHALL require a successful GAP before signature creation.

Depending on the configuration of the [eSign application], the signature creation is controlled by a user credential that is either the Global PIN, or the local PIN/BIO.

In any case, PACE SHALL be performed according to [TR 03110v2 part2] and SHALL contain the confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege “Generate qualified electronic signature” or “Generate electronic signature” as defined in §3.1.

The terminal SHALL have the privilege matching the usage of the electronic signature key it intends to select to create an electronic signature:

a. The [SMT] SHALL have the privilege “Generate qualified electronic signature” to use a qualified electronic signature key;

b. The [SMT] SHALL have the privilege “Generate electronic signature” to use an electronic signature key;

7.2.5. Signature creation without authentication of the [SMT]

This mode of operation is available only if the [eSign application] has been configured to create signature without the authentication of the [SMT]. As such, the [eSign application] SHALL be used in a trusted environment. The user and the issuer SHALL take all appropriate measures to ensure a sufficient security level is met.

> ***Notes***
>
> This configuration is exclusive with the one described in §7.2.4

The sequence of operation described in §7.2.1 SHALL be applied. In this case, the eIDAS token SHALL require a successful PACE before signature creation.

Depending on the configuration of the [eSign application], the signature creation is controlled by a user credential that is either the Global PIN, or the local PIN/BIO.

PACE with a global PIN SHALL be performed. In any case, PACE SHALL be performed according to [TR 03110v2 part2] and SHALL contain confined authorization (in the CHAT),, ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

7.2.6. Management of user consent’s internal state

The user’s consent for signing a document is materialized by the submission (and authentication) of the [eSign application] user credentials.

The management of the user’s consent internally stored in [eSign application] differs depending on the type of key used for the signature key:

a. Electronic signature key;

b. Qualified electronic signature key;

By default, [SMT] can produce only one signature before the user’s consent is reset. However, the [SMT] can create up to the number of signatures specified in the certificate extension before the user’s consent is reset.

Once the maximum number of consecutive signature(s) has been met, the user consent SHALL be set through the VERIFY command.

7.3. Data management

7.3.1. Local user credential vs. global user credential

eIDAS token MAY handle global and local credentials:

- global user credentials are located in the MF;
- local user credentials are located in the ADF;

Both local and global user credentials SHALL be managed according to the current document.

Global user credentials that grant access to the signature creation function and local user credentials SHALL only be managed according to the security policies defined herein.

In case the eIDAS token also contains other applications, the global user credentials MAY also be managed according to other security policies defined by these supplemental application, provided they do not grant access to the signature creation function. In particular, if the eIDAS token also supports an eID application as defined in [TR 03110v2 part2], global user credentials SHALL also be managed as described in [TR 03110v2 part2].

7.3.2. Specific issues for global user credentials and PUK

The suspension mechanism as stated in [TR 03110v2] SHALL be supported for the PUK(s) and global user credential(s) (PIN/BIO). As such, the support of CAN is mandatory.

7.3.3. User credential initialization

The [eSign application] enables to initialize the user credential (PIN/BIO) used to access the signature function, whether it is global or local. The process flow is the following:

a. GAP;

b. Selection of the [eSign application];

c. Retrieval of the [ISO/IEC 7816-15] data;

d. Initialization of the user credential;

e. Update of [ISO/IEC 7816-15] data

PACE SHALL be performed according to [TR 03110v2 part2] and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The [SMT], after successful completion of GAP is allowed to initialize the user credential and to update the content of [ISO/IEC 7816-15] files in the [eSign application]. In particular the [SMT] is responsible for updating accordingly the [ISO/IEC 7816-15] structure.

The terminal performing the GAP SHALL be a [SMT] with the privilege “User credential initialization” as defined in §3.1. to initialize a user credential;

Prior the user credential initialization, the [SMT] SHALL retrieve the [ISO/IEC 7816-15] structure to discover the entire user credential available in the eIDAS token as well as their internal state: “initialization”, “operational – activated”, “operational – deactivated” or “terminated” (see Life cycle state).The [eSign application] SHALL only accept initialization on user credential whose internal life cycle state is set to “initialization state”.

The user credential SHALL be initialized using CHANGE REFERENCE DATA command with P1=’01’, P2 containing the identifier of the user credential and the data field containing the PIN/BIO. This command is available only once. After a successful completion, the user credential life cycle status is automatically set to “operational – activated”.

The [SMT] SHALL also update the [ISO/IEC 7816-15] structure to reflect the change of the life cycle state of the user credential. The [SMT] SHALL use the UPDATE BINARY command to update the [ISO/IEC 7816-15].

7.3.4. User credential update

The [eSign application] enables to change the value of the user credential, whether it is global or local. The process flow is the following:

a. GAP;

b. Selection of the [eSign application];

c. Retrieval of the [ISO/IEC 7816-15] data;

d. Change the user credential value;

PACE SHALL be performed according to [TR 03110v2 part2] and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege “User credential value change” as defined in §4.1.

Prior the user credential value change, the [SMT] SHALL retrieve the [ISO/IEC 7816-15] structure to discover the entire user credential available as well as their internal state: “initialization”, “operational – activated”, “operational – deactivated” or “terminated” (see Life cycle state).The [eSign application] SHALL only accept update on user credential whose internal life cycle state is set to “operational – activated”.

Case of PIN credential update

The old PIN, and the new PIN SHALL be submitted together in the same incoming CHANGE REFERENCE DATA command with P1=’00’ and reference indicated in P2.

Case of BIO credential update

The old template SHALL be submitted first with VERIFY command, and then new template SHALL be sent with CHANGE REFERENCE DATA command with P1=’01’ and reference indicated in P2. The update is effective only if first step of verification with the indicated current value succeeded. Previous validation status is lost.

7.3.5. User credential unblocking

Once the retry counter of the user credential has reached zero, it is locked and does not allow further usage. The [eSign application] enables to unblock and modify the value of the user credential, whether it is global or local. The process flow is the following:

a. GAP;

b. Selection of the [eSign application];

c. Retrieval of the [ISO/IEC 7816-15] data;

d. Unblock the user credential;

PACE SHALL be performed according to [TR 03110v2 part2], and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege “User credential unblock” as defined in §4.1.

Prior the user credential unblocking, the [SMT] SHALL retrieve the [ISO/IEC 7816-15] structure to discover the entire user credential available as well as their internal state: “initialization”, “operational – activated”, “operational – deactivated” or “terminated” (see Life cycle state).The [eSign application] SHALL only unblock user credential whose internal life cycle state is set to “operational – activated”.

A user credential is unblocked with RESET RETRY COUNTER command with ‘02’ set in P1, reference indicated in P2 and new value set in data field. Retry counter is reset to its initial value n in case of success.

7.3.6. Signature key and certificate generation

The [eSign application] enables to generate an electronic signature key on board, whether it is a qualified electronic signature key or not. The process flow is the following:

a. GAP;

b. Selection of the [eSign application];

c. Retrieval of the [ISO/IEC 7816-15] data;

d. Generation of a signature key;

e. Certificate building;

f. Update EF.CD[x] in [ISO/IEC 7816-15] structure;

PACE SHALL be performed according to [TR 03110v2 part2], and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege “install certificate” or “install qualified certificate” as defined in §4.1. The [SMT] SHALL have the privilege:

g. “install qualified certificate” to

    i. generate a qualified electronic signature key;

    ii. update the content of EF.CD[x] in [ISO/IEC 7816-15] structure;

h. “install certificate” to

    iii. generate an electronic signature key;

    iv. update the content of EF.CD[x] in [ISO/IEC 7816-15] structure;

Prior the key generation, the [SMT] SHALL retrieve the [ISO/IEC 7816-15] structure to discover all the electronic signature keys available as well as their internal state: “initialization”, “operational – activated”, “operational – deactivated” or “terminated” (see Life cycle state). The [eSign application] SHALL only generate keys on electronic signature keys whose internal life cycle is set to “initialization state” or “operational – Activated”.

The [SMT], after successful completion of GAP is allowed to perform the electronic signature key generation AND to update the content of [ISO/IEC 7816-15] files in the [eSign application]. In particular the [SMT] is responsible for loading the certificate in the [eSign application] and updating accordingly EF.CD[x] in the [ISO/IEC 7816-15] structure.

Key generation allow to generate EC and RSA key pair. It is performed with the GENERATE ASYMETRIC KEY PAIR command containing:

During the generation of the electronic signature key the following operations are performed by the [eSign application]:

i. the private and the public portion are generated and stored into the [eSign application];

j. the public portion is exported so that the [SMT] can generate the signature key certificate;

Notes

The public portion of the signature key is returned only once, by the GENERATE ASSYMETRIC KEY PAIR command.

Key generation can be used for electronic signature key initialization and renewal. Upon key generation, signature keys are generated using the attributes associated to the data container:

k. signature algorithm;

l. key type;

m. key length;

n. domain parameters for EC;

In case of successful renewal the key initially saved to this location is replaced by the newly generated one. In case of failure during renewal while the usage of the command was allowed, the key initially saved to this location is erased. A new generation SHALL be launched.

The [eSign application] does not require the user credential protecting the electronic signature key to be initialized before signature key generation.

The holder submits its [eSign application] user credentials to unlock the signature creation function in the [eSign application]. The user's consent internal state is reset after electronic signature keys have been generated, whether the key is a qualified electronic signature key or not.

In order to generate the signature key certificate, the [SMT] SHALL retrieve the following information:

o. attributes of the electronic signature key. Thanks to [ISO/IEC 7816-15] structure, the [SMT] can retrieve all the keys attributes (Electronic signature algorithms, Key type and size,…);

p. attributes of the user (name, surname,…). The [SMT] can either retrieve it in another application present in the eIDAS token (e.g. an eID application), or in the file EF.INFO4CERT if present.

Once the signature key certificate has been generated, the [SMT] shall update the [ISO/IEC 7816-15] structure in order to store the newly generated signature key certificate. In case the signature key was in "initialization state", the [SMT] SHALL also update the [ISO/IEC 7816-15] structure to reflect the change of the life cycle state of the signature key. The [SMT] SHALL use the UPDATE BINARY command to update the [ISO/IEC 7816-15].

7.3.7. Life cycle management

The [eSign application] enables to manage the life cycle of the electronic signature key (whether it is qualified or not) and user credential, namely by locking/unlocking (suspension of its usage) and termination (the object becomes irreversibly unusable). The process flow is the following:

d. GAP;

e. Selection of the [eSign application];

f. Retrieval of the [ISO/IEC 7816-15] data;

g. Manage the life cycle of the signature key or user credential;

h. Update of [ISO/IEC 7816-15] data;

PACE SHALL be performed according to [TR 03110v2 part2], and SHALL contain confined authorization (in the CHAT), ensuring the user validates the operations to be performed on the [eSign application]. Once PACE has been successfully performed, all the subsequent communication SHALL be protected using secure messaging.

The terminal performing the GAP SHALL be a [SMT] with the privilege “object management” as defined in §.3.1.1.

The [SMT], after successful completion of GAP is allowed to manage the lifecycle of the electronic signature keys and user credential, and to update the content of [ISO/IEC 7816-15] files in the [eSign application]. In particular the [SMT] is responsible for updating accordingly the [ISO/IEC 7816-15] structure.

While the state “Operational – Deactivated” is reversible, the state “Terminated” is irreversible as described in [ISO/IEC 7816-8].

After having successfully managed the life cycle of an object, the [SMT] SHALL update the [ISO/IEC 7816-15] structure to modify the life cycle on the said object:

i. Operational – Activated

j. Operational – Deactivated

k. Terminated

7.4. User authentication devalidation

The [eSign application] can devalidate the authentication status of user credential.

7.4.1. Global user credentials

The verification status of the global PIN is reset after the following events:

a. a reset of the eIDAS token;

b. a CHANGE REFERENCE DATA command on the PIN;

c. a RESET RETRY COUNTER on the PIN;

d. failing to establish a new secure channel using PACE with the global PIN;

e. a VERIFY command with P1 set to ‘FF’, P2 containing the identifier of the PIN and an empty data field;

Moreover, if the Global PIN is used for electronic signature (qualified or not), its verification status SHALL be reset after the following events:

f. a signature key update, whatever the procedure succeed or not;

g. the n-th signature creation, where n indicates the maximum number of electronic signature indicated in the [SMT]’s certificate extension (see §3.1) validated during GAP (n = 1 if the field is absent from the certificate extension);

h. one signature creation when the terminal has not been authenticated;

7.4.2. Local user credentials

The verification status of the local user credentials SHALL be is reset after the following events:

a. a reset of the eIDAS token;

b. a VERIFY command with P1 set to ‘FF’, P2 containing the identifier of the user credential and an empty data field;

c. a CHANGE REFERENCE DATA command on the user credential;

d. a RESET RETRY COUNTER on the user credential;

e. Selection of the MF or [eSign application];

Moreover, if the user credential is used for electronic signature (qualified or not), its verification status SHALL be reset after the following events:

f. a signature key update, whatever the procedure succeed or not;

g. the n-th signature creation, where n indicates the maximum number of electronic signature indicated in the [SMT]’s certificate extension (see §3.1) validated during GAP (n = 1 if the field is absent from the certificate extension);

h. one signature creation when the terminal has not been authenticated;

8. ISO/IEC 7816 mapping

8.1. GENERATE ASYMMETRIC KEY PAIR

The generate public key pair command initiates the generation and storing of a key pair, i.e. the private portion and the public portion are stored in the eIDAS token. The public portion SHALL be returned by the eIDAS token in order to generate a dedicated certificate and use for checking the signature.

This command is operational for ECC and RSA key type.

COMMAND PARAMETER	MEANING
CLA	ISO
INS	'47'
P1	'82'
P2	'00'
L_c field	Data Length
Data field	’B6’-L-{ {‘83’- L-PublicKeyRef} {‘4D’- L-{‘7F49’-’80’}} {[‘E2’ L ‘82’- L –V]} } , for RSA key , with [] optional (default value ‘10001’) ’B6’-L-{ {‘83’- L-PublicKeyRef} {‘4D’- L-{‘7F49’-‘04’-‘06’-‘00’-‘86’-‘00’}} }, for EC key
L_e field	Present

RESPONSE PARAMETER	MEANING
Data field	Return only public elements in order to generate a dedicated certificate 7F49’-L – {‘81’L <modulus> - 82 L <exponent>}} for RSA key 7F49’-L – {‘06’L <OID> - ‘86’ L <public point> } for ELC key
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.2. MSE SET for CRT DST

This command sets or replaces the signature CRT DST hash and scheme algorithm in the current Security Environment.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '22' '41' – Set for signature creation 'B6' – CRT of DST
L_c field	Data Length
Data field	’84’-‘01’ -{ Private key Reference}
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.3. PSO: HASH

The off-card entity is responsible of computing the intermediate hash over the first part of the data to be signed. The intermediate hash-code is transferred to the eIDAS token by the PSO:HASH command together with the remaining part of the data, the eIDAS token performs the last round hash computation.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '2A' '90' 'A0'
L_c field	Data Length
Data field	'90' L90 <intermediate hash-code followed by a bit counter encoded with most significant bit first> '80' L80 <data to be hashed>, limited to one block
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.4. PSO: Compute Digital Signature

The PSO Compute Digital Signature operation 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.

To compute a digital signature, the data to be signed or integrated in the signing process are transmitted in the command data field or provided through a previous PSO: HASH command.

Note: in case of PSO: HASH used command, the eIDAS token SHALL guarantee this DTBS only will be signed.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '2A' '9E' '9A'
L_c field	Absent if PSO:HASH occurred before Data Length if Hash value to transfer
Data field	Absent (PSO:HASH occurred before) Data To be Signed (Hash value computed off card)
L_e field	Present

RESPONSE PARAMETER	MEANING
Data field	Digital signature
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.5. VERIFY PIN

Knowledge based user verification requires the user to enter a password.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '20' '00' reference data qualifier
Lc field	Data Length
Data field	<Password>
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.6. VERIFY based on biometric user verification

Knowledge based user verification requires a user to provide a biometric template.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '21' '00' reference data qualifier
L_c field	Data Length
Data field	< biometry template>
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.7. DEVALIDATION User authentication PIN.

The command resets the user authentication status.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '20' 'FF' reference data qualifier
L_c field	Absent
Data field	Absent
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.8. DEVALIDATION User authentication BIOMETRY.

The command resets the user authentication status.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '21' 'FF' reference data qualifier
L_c field	Absent
Data field	Absent
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.9. CHANGE REFERENCE DATA for PIN update

This command intends to replace a current password value with a new one. It requires a successful comparison between the reference password and the verification data sent from the interface device. The reference data replacement can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '24' '00' reference data qualifier
L_c field	Data Length
Data field	<old password> || <new password>
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.10. CHANGE REFERENCE DATA for PIN initialization

The command initializes the PIN value It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '24' '01' reference data qualifier
L_c field	Data Length
Data field	<new password>
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.11. CHANGE REFERENCE DATA for BIOMETRY initialization

The command initializes the new biometry template sent from the interface device. It can be performed only if the security status satisfies the security attributes for this command.

This command SHALL be used in order to activate the BIOPIN if the life cycle of the BIOMETRY is in created state, or if the biometry already activated, the command is used to change the current biometry with the new biometric template.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '25' '01' reference data qualifier
L_c field	Data Length
Data field	<new Biometric template>
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.12. CHANGE REFERENCE DATA for BIOMETRY update

This command intends to replace a current biometry template value with a new one. A VERIFY command on the current biometry template SHALL grant the access rights of the change It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '25' '01' reference data qualifier
L_c field	Data Length

Data field	<new Biometric template>
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.13. RESET RETRY COUNTER for PIN

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. It can be performed only if the security status satisfies the security attributes for this command.

With the [ISO/IEC 7816-4] command RESET RETRY COUNTER, the user can initiate the reset of the RC to its initial value N by providing a new password.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '2C' '02' reference data qualifier
L_c field	Data Length
Data field	<new password>
Le field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.14. RESET RETRY COUNTER for BIOMETRY

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. It can be performed only if the security status satisfies the security attributes for this command.

With the [ISO/IEC 7816-4] command RESET RETRY COUNTER, the user can initiate the reset of the RC to its initial value N.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '2D' '02' reference data qualifier
L_c field	Data Length
Data field	<new biometric template>
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.15. MANAGE DATA : OPERATIONAL CHANGE LCS User credential and Keys

The function of this command is to set the LCS (see [ISO/IEC 7816-4] Table 14) of User credential or keys by the LCS indicated in P2. It can be performed only if the security status satisfies the security attributes for this command.

The Life Cycle State MAY be changed to operational-state activated, operational-state deactivated or terminated

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'CF' '00' LCS to be set by the command (see [ISO/IEC 7816-4] Table 14)
L_c field	Data Length

Data field	‘7F71’- L – { ‘7F70’ – L – {‘83’ –L - <User credential Id>}}
	‘7F71’- L – { ‘7F70’ – L – {‘84’ –L - <Private key Id>}}

L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.16. SELECT

This command selects a file (EF) or an application (MF or ADF). After a successful selection the file selected becomes the current file. After the reset the current DF is the MF and no EF is selected. It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'A4' '00' - Select MF, EF '02' - Select EF under current DF '04' - Select by DF name (ADF) '0C' - no data in response field
L_c field	Data Length
Data field	If P1=‘00’, MF or EF Identifier If P1=‘02’, EF Identifier If P1=‘04’, AID
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.17. SELECT MF

This command selects the MF as define in [ICAO 9303]. After a successful selection the file selected becomes the current file. It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'A4' '00' '00'
L_c field	Absent
Data field	Absent
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.18. READ BINARY

This command reads the content of a transparent EF (on a current selected file or with implicit SFI selecting). It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'B0' As indicated in Table below Binary P1-P2 coding As indicated in Table below Binary P1-P2 coding
L_c field	Absent
Data field	Absent
L_e field	Number of bytes to read

RESPONSE PARAMETER	MEANING
Data field	Data read
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

<table>
<tr><th colspan="8">P1</th><th colspan="8">P2</th></tr>
<tr><th>B8</th><th>B7</th><th>B6</th><th>B5</th><th>B4</th><th>B3</th><th>B2</th><th>B1</th><th>B8</th><th>B7</th><th>B6</th><th>B5</th><th>B4</th><th>B3</th><th>B2</th><th>B1</th></tr>
<tr><td>0</td><td colspan="15">Offset in the currently selected file over 15 bits
'00' ≤ Offset ≤ '7FFF'</td></tr>
<tr><td>1</td><td>0</td><td>0</td><td colspan="5">Short File Identifier
1 ≤ SFI ≤ 30</td><td colspan="8">Offset in the file over 8 bits</td></tr>
</table>

Table Binary P1-P2 coding

8.19. UPDATE BINARY

This command updates the content of a transparent EF (on a current selected file or with implicit SFI selecting). It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'D6' As indicated in Table Binary P1-P2 coding As indicated in Table Binary P1-P2 coding
L_c field	Data Length
Data field	Data to write
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.20. ACTIVATE [eSign application]

This command is used to turn the [eSign application] to the activated state. No error occurs if [eSign application] was already activated. The command applies when [eSign application] is selected. It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '44' '00' '00'
L_c field	Absent
Data field	Absent
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.21. DEACTIVATE [eSign application]

This command is used to turn the [eSign application] to the deactivated state. No error occurs if the file was already deactivated. The command applies when [eSign application] is selected. It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO '04' '00' '00'
L_c field	Absent
Data field	Absent
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

8.22. TERMINATE [eSign application]

This command is used to turn the [eSign application] to the terminated state. No error occurs if the file was already terminated. The command applies when [eSign application] is selected. It can be performed only if the security status satisfies the security attributes for this command.

COMMAND PARAMETER	MEANING
CLA INS P1 P2	ISO 'E6' '00' '00'
L_c field	Absent
Data field	Absent
L_e field	Absent

RESPONSE PARAMETER	MEANING
Data field	Absent
SW1-SW2	See [ISO/IEC 7816-4], Tables 5 and 6 where relevant

9. Annex B: Example of [ISO/IEC 7816-15] structure

Remark : to be completed

This annex is informative and proposes an illustration of [ISO/IEC 7816-15] applied in the context of the current technical report.

10. Normative references

ISO/IEC 7816-4 - 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 - Identification cards -- Integrated circuit cards -- Part 8: Commands for security operations

ISO/IEC 7816-9 - Identification cards -- Integrated circuit cards -- Part 9: Commands for card management

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 -- Part 15: Cryptographic information application

ISO/IEC 2382-37 - Information technology -- Vocabulary -- Part 37: Biometrics

EN 14890-1 - Application Interface for smart cards used as Secure Signature Creation Devices - Part 1: Basic services (EN 419212)

EN 14890-2 - Application Interface for smart cards used as Secure Signature Creation Devices - Part 2: Additional Services (EN 419212)

FIPS PUB 180-3, Secure Hash Standard - October 2008 , National Institute of Standards and Technology

Technical Guideline TR-03111 - Elliptic Curve Cryptography - Version 2.0

Technical Guideline TR-03110 v2.10 part 1 - Advanced Security Mechanisms for Machine Readable Travel Documents - Part 1 – eMRTDs with BAC/PACEv2 and EACv1 - Version 2.10 part 1 - 20. March 2012

Technical Guideline TR-03110 v2.10 part 2 - Advanced Security Mechanisms for Machine Readable Travel Documents - Part 2 – Extended Access Control Version 2 (EACv2), Password Authenticated Connection Establishment (PACE),and Restricted Identification (RI) - Version 2.10 part 2 - 20. March 2012

ICAO 9303 , Machine Readable Travel Documents - Part 1: Machine Readable Passport, Specifications for electronically enabled passports with biometric identification capabilities (including supplement), ICAO Doc 9303, 2006

11. Other references

EC_Regulation - COM(2012) 238/2 - REGULATION OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL on electronic identification and trust services for electronic transactions in the internal market