

eEpoch

(eEurope Smart Card Charter proof of concept and
holistic solution)

Deliverable D3.1: Requirements of IOP specifications

Giesecke & Devrient

Deliverable D3.1: Requirements of IOP specifications

Document Control Sheet	
Responsible Author(s):	Jens Urmann
Organisation:	Giesecke & Devrient
Subject / Title of Document:	Deliverable D3.1: Requirements of IOP specifications
Related Task('s):	
Deliverable No.	
Save Date of File:	12/05/2003
Version Number:	1.0
Ref./File Name	Deliverable_D31_v10.doc
Number of Pages	6

Document Distribution			
Membertype	Organisation	Name	Distributed
Webpage	[Project Web Site]	Internet	DD/MM/YY
Contractors Partners			
European Commission			
Additional			

Contents

1. Introduction.....	5
1.1 Version List.....	5
1.2 References	5
2. Report Deliverable D3.1	6

List of Tables

<i>Table 1 Version List</i>	5
-----------------------------------	---

1. Introduction

This report lists the documents that have been prepared in the context of work package 3 of the eEpoch project according to the Detailed Workplan, cf.[Workplan]. The documents specify an IAS service with a workable solution for electronic signatures according to the EU Directive on electronic signature and compliant with the requirements of ESIGN.

1.1 Version List

<i>Version / Date</i>	<i>Brief Description of Changes, Name</i>
0.1 / 09.05.2003	Initial Document, Jens Urmann (uj)
1.0 / 12.05.2003	Document delivered to the EC, uj

Table 1 Version List

1.2 References

- [E-SIGN K, Part 1] Application Interface for smart cards used as Secure Signature Creation Devices, Version 1 Release 1, 08.04.2003, CEN/ISSS WS/E-Sign Draft CWA Group K; Part 1 – Basic requirements
- [E-Sign K, Part 2] Application Interface for SmartCards used as Secure Signature Creation Devices, Version 0 Release 10, 01.06.2002, CEN/ISSS WS/E-Sign Draft CWA Group K; Part 2 – Optional features
- [Workplan] D0.2 Detailed workplan, Version 1.2, 06.02.2003, ETRA Investigacion y Desarrollo, S.A

2. Report Deliverable D3.1

The Deliverable D3.1 comprises besides the present report the following documents:

1. Functional Mapping of GIF on E-Sign K, Version 1.3, 12.05.2003, Giesecke & Devrient
2. Generic IAS Application / System package based on E-Sign K, Version 1.0, 12.05.2003, Giesecke & Devrient

Please note: Both documents are based upon the E-Sign K specification, cf. [E-SIGN K, Part 1] and [E-Sign K, Part 2]. The work on the E-Sign K specification is not yet finalised but ongoing. A stable version for the first part will be available earliest in June 2003. The final eEpoch IAS application should be based upon this stable version in order to provide a high level of quality and in order to support interoperability. For this reason changes in the IAS application specification, i.e. in the document "Generic IAS Application / System package based on E-Sign K", may be necessary.

eEpoch

(eEurope Smart Card Charter proof of concept and
holistic solution)

Functional Mapping of GIF on E-Sign K

Giesecke & Devrient

Functional Mapping of GIF on E-Sign K

Document Control Sheet	
Responsible Author(s):	Nigol Martin, Dr. Jens Urmann
Organisation:	Giesecke & Devrient
Subject / Title of Document:	Functional Mapping of GIF on E-Sign K
Related Task('s):	
Deliverable No.	
Save Date of File:	09/05/2003
Version Number:	1.3
Ref./File Name	Functional_Mapping_GIF_ESign_K_v1.3.doc
Number of Pages	20

Document Distribution			
Membertype	Organisation	Name	Distributed
Webpage	[Project Web Site]	Internet	DD/MM/YY
Contractors Partners			
European Commission			
Additional			

Contents

1.	Introduction	6
1.1	Revision Log	6
1.2	References.....	6
1.3	Notations	6
1.4	Abbreviations	7
1.5	Brief description	8
2.	File structure	10
3.	Requirements for IAS.....	11
3.1	Identification	11
3.1.1	Global Interoperability Framework	11
3.1.2	E-Sign K Specification.....	11
3.1.3	Mapping.....	11
3.2	Authentication	12
3.2.1	Global Interoperability Frame work	12
3.2.2	E-Sign K Specification.....	12
	Device Authentication	12
3.2.3	Mapping.....	13
	Device Authentication	13
3.3	Signature.....	16
3.3.1	Global Interoperability Framework	16
3.3.2	E-Sign K Specification.....	16
3.3.3	Mapping.....	18
3.4	Secure Messaging	19
3.4.1	Global Interoperability Framework	19
3.4.2	E-Sign K Specification.....	19
3.4.3	Mapping.....	20

List of Figures

Figure 1: <i>Signer's User Verification</i> [E-SIGN K, Part 1]	17
Figure 2: <i>Data Flow chart for a signature generation process</i> [E-SIGN K, Part 1].....	18

List of Tables

Table 1: Revision Log	6
Table 2: Identification - Functional Mapping	12
Table 3: Device Authentication Flow (retrieval of public keys and MANAGE SECURITY ENVIRONMENT commands have been omitted).....	14
Table 4: Device Authentication – Functional Mapping.....	15
Table 5: User Authentication – Functional Mapping	16
Table 6: Signature – Functional Mapping	19
Table 7: Signature – Additional Features.....	19

1. Introduction

1.1 Revision Log

Version / Date	Brief Description of Changes, Name
0.1 / 10.12.02	Initial document, Nigol Martin
0.2 / 20.12.02	New Annex A
1.0 / 20.12.02	Document sent to the pilot sites, Nigol Martin
1.1 / 04.02.03	Adjusted to version 0.13 of E-Sign K specification, Jens Urmann
1.2 / 03.03.03	Considered comments, Jens Urmann
1.3 / 12.05.03	Adjusted to latest versions of GIF and version 1 release 1 of E-Sign K part 1, document delivered to the EC, uj

Table 1: Revision Log

1.2 References

- [E-SIGN K, Part 1] Application Interface for smart cards used as Secure Signature Creation Devices, Version 1 Release 1, 08.04.2003, CEN/ISSS WS/E-Sign Draft CWA Group K; Part 1 – Basic requirements
- [GIF, Part 2] Open Smart Card Infrastructure for Europe v2, Volume 3: Global Interoperability Framework for identification, authentication and electronic signature (IAS) with smart cards, Part 2: Requirements for IAS functional interoperability, Version 3.10, March 2003, eESC GIF Expert Group
- [GIF, Part 3] Open Smart Card Infrastructure for Europe v2, Volume 3: Global Interoperability Framework for identification, authentication and electronic signature (IAS) with smart cards, Part 3: Recommendations for interoperability specifications, Version 1.00, March 2003, eESC GIF Expert Group
- [ISO/IEC 7816] ISO/IEC FCD 7816, “Information technology – Identification cards- Integrated circuit(s) cards with contact”
 Part 4: Interindustry commands for interchange”, FCD2003
 Part 8: Security related interindustry commands, FCD 2003
 Part 11: Personal verification through biometric methods, FCD 2002
 Part 15: Cryptographic information application, FDIS 2003

1.3 Notations

The following simplified Backus-Naur notation applies to keys and certificates:
`<object descriptor> ::= <key descriptor> | <certificate descriptor>`

```
<key descriptor> ::=
<key> . <keyholder> . <usage>

<key> ::= <secret key> | <public key> | <group key> | <individual key> |
<pin>
<secret key> ::= SK
<public key> ::= PK
<group key> ::= GK
<individual key> ::= IK
<pin> ::= PIN

<keyholder> ::= <cardholder> | <certification authority> | <integrated
circuit(s) card> | <interface device> | <smart card manufacturer>
<cardholder> ::= CH
<certification authority> ::= CA | RCA
<integrated circuit(s) card> ::= ICC
<interface device> ::= IFD
<smart card manufacturer> ::= GD

<usage> ::= <digital signature> | <authentication> | <CertSign> | <key
encipherment> | <personalisation> | <termination> | <secure messaging>
<digital signature> ::= DS
<authentication> ::= AUT
<CertSign> ::= CS <usage> or <usage>
<key encipherment> ::= KE
<personalisation> ::= PERS
<termination> ::= TER
<secure messaging> ::= SM

<certificate descriptor> ::=
    <certificate> . <certholder> . <usage>
<certificate> ::= C

<certholder> ::= <cardholder> | <certification authority> | <integrated
circuit(s) card> | <interface device>
```

For subsequent data items, the following notation is used:

|| = Concatenation of data

1.4 Abbreviations

AID	Application Identifier
APDU	Application Protocol Data Unit
AUT	Authentication
BIT	Biometric Information Template

CA	Certification Authority
CBC	Cipher Block Chaining
CIA	Cryptographic Information Application
DES	Data Encryption Standard
DF	Dedicated File
DSI	Digital Signature Input
DTBS	Data To Be Signed
E()	Encipherment of
GIF	Global Interoperability Framework
h()	Hash value of
IAS	Identification, Authentication, electronic Signature
ICC	Integrated Circuit(s) Card
ID	Identifier or Identity
IFD	Interface Device, e.g. terminal
PK	Public Key
PRND	Padding Random Number
PSO	Perform Security Operation
RND	Random number
RSA	Rivest Shamir Adleman
SK	Secret Key (equiv. to Private Key)
SN	Serial Number
SSC	Send Sequence Counter
TDES	Triple DES
URL	Uniform Resource Locator

1.5 Brief description

This document presents a mapping of the required functions for an interoperability (IOP) of the core functionalities Identification, Authentication and Signature (IAS) as defined in the Global Interoperability Framework (GIF) documents onto a more technical specification based on [E-SIGN K, Part 1].

In order to map the functionality tables are used that oppose both specifications (on the left side the GIF functions are found and on the right side the specification based on E-Sign K is presented). If a requirement of the GIF framework is not covered by the E-Sign K specification, the table contains only the GIF requirement and the fundamentals of the specification are outlined.

In this document the term identity token is used for GIF's subjects handled by the IAS application.

2. File structure

The file system of the E-Sign K specification is compliant to ISO/IEC 7816-15 (PKCS#15), see reference [ISO/IEC 7816]. This standard shall also be used for the IAS application.

3. Requirements for IAS

3.1 Identification

3.1.1 Global Interoperability Framework

According to document [GIF, Part 3] Identification is defined as (see p.20):

A process through which the smart card provides descriptive data to an off-card application about any of the subjects (identity token) managed by the on-card IAS application.

The cited document points out that “Identification may be a public function”. The access conditions can be adjusted to the needs of the system.

3.1.2 E-Sign K Specification

The information about the subjects (identity tokens) managed by the ICC is stored in the file system. The corresponding files can be accessed by means of ISO/IEC 7816-4 GET DATA command. For all ISO/IEC 7816 commands please refer to document [ISO/IEC 7816].

Using access conditions one can restrict the access to the files, i.e. to the identification data. Therefore one can distinguish between an identification process using public data and a process using private data. In the last case access to the card holder’s identity is only granted after e.g. a successful mutual device authentication. Using Secure Messaging the retrieval of the card holder’s identity can be protected. The device authentication is treated in the next section.

3.1.3 Mapping

GIF	E-Sign K
<p>IAS availability [GIF, Part 3], section 4.4.1, page 25: “The smart card should be equipped with a standard function enabling an off-card application to determine whether or not the card is equipped with a generic IAS application.”</p>	<p>Application Selection [E-SIGN K, Part 1], section 5.3, page 5-16: “An E-SIGN application is selected by its Application Identifier (AID).” I.e. the SELECT command specified in ISO/IEC 7816-4 provides the required functionality.</p>
<p>IAS_SubList() [GIF, Part 3], section 4.4.2, page 25: “The IAS application handles a minimum of two mandatory subjects but may manage more. In order to provide an off-card application with a referenced list of these subjects (which does not mean that</p>	<p>The file containing the list of referenced list of subjects can be read after selection using the ISO/IEC command GET DATA.</p>

<p>the off card application has rights to access them) a specific function is required.”</p>	
<p>IAS_Sel(Sub) [GIF, Part 3], section 4.4.3, page 25: “This function selects one subject which then becomes the default subject for all following IAS operations. Alternatively this function can be replaced by adding a Sub parameter to all the following functions.”</p>	<p>Each identity token's settings are stored in different Security Environments which are restored when required using the ISO/IEC 7816-4 command MANAGE SECURITY ENVIRONMENT.</p>
<p>IAS_GetID() [GIF, Part 3], section 4.4.4, page 25: “This function is required for an off-card application to obtain identification data from the card or from a URL about a given subject. The Identification data is returned.”</p>	<p>The file containing the identification data of an identity token (subject) as data objects can be read after selection using the ISO/IEC command GET DATA.</p>

Table 2: Identification - Functional Mapping

3.2 Authentication

3.2.1 Global Interoperability Framework

According to document [GIF, Part 3] Authentication is defined as (see p.20):

“A process through which the smart card provides an off card application with strong and verifiable electronic evidence of identity (and possibly attributes) for any of the subjects managed by the on-card IAS application.”

The Global Interoperability Framework requires that the off card application (IFD) authenticates an identity token (subject) on the card by means of a challenge response mechanism. No mutual authentication is required (compare the functional definition of the method IAS_IntAuth(Chal) [GIF, Part 3], section 4.4.5).

The implementation of the IAS application depends on asymmetric cryptography and related Public Key Infrastructures ([GIF, Part 3], section 4.5.1).

3.2.2 E-Sign K Specification

In the context of the E-Sign K specification one has to distinguish between a device authentication (for the smart card subject according to GIF) and a user authentication/verification (for the card holder's public identity according to GIF). These two different authentication schemes are covered in the next subsections.

Device Authentication

This specification distinguishes two environments with respect to the signature creation application: In a trusted environment a device authentication between the card and the off

card application or IFD (device authentication) is optional, in an untrusted environment device authentication is mandatory. The user has to decide about the trustworthiness of the environment. In contrast to the Global Interoperability Framework device authentication according to E-Sign K is mutual. Therefore after a successful authentication symmetric session keys are available on both sides, so that secure messaging can be used in subsequent transmissions.

The E-Sign K specification provides three reference authentication schemes:

1. a key agreement protocol based upon symmetric TDES cryptography ([E-SIGN K, Part 1], section 8.7,
2. a key transport protocol based upon asymmetric RSA cryptography ([E-SIGN K, Part 1], section 8.4
3. and a key negotiation protocol based upon asymmetric RSA cryptography with ICC privacy protection ([E-SIGN K, Part 1], section 8.5.

User Authentication/Verification

The E-Sign K specification considers two methods for user authentication/verification:

- a knowledge based user authentication, e.g. by means of a PIN, that is mandatory
- a biometric user authentication that is an optional feature

The verification of biometric data mandates a device authentication and the usage of secure messaging in order to avoid replay attacks. I.e. the environment will always be considered as untrusted environment.

3.2.3 Mapping

Device Authentication

With respect to the mapping of the E-Sign K functionality to the GIF requirements the device authentication schemes 1 and 3 are not taken into consideration for the following reasons: The first scheme is based upon symmetric cryptography. The third scheme provides privacy protection of the ICC, i.e. the identity of the subject (identity token) of the IAS application is not revealed prior to a successful authentication of the IFD. This feature is not required by the GIF framework.

The second authentication protocol provides a mutual authentication based upon RSA cryptography. If this protocol is applied, session keys are generated so that Secure Messaging – an additional feature according to the GIF framework – can also be used.

This protocol is outlined in the following:

1. The IFD and the ICC exchange card verifiable certificates to get the public authentication key (denoted by PK.ICC.AUT and PK.IFD.AUT) of the other side. If the corresponding certification authority key is not known by the IFD or ICC a certificate chain up to the root-CA key has to be verified.
2. After the private authentication key SK.ICC.AUT of the ICC and the public authentication key of the IFD have been selected using the ISO/IEC MANAGE SECURITY

ENVIRONMENT command, a challenge RND.IFD and a serial number SN.IFD is sent to the ICC. The ICC generates a random number K.ICC used to derive session keys later and a padding random number PRND1. As shown in table [Table 3] this data is concatenated and a hash is calculated. Then a signature is generated using SK.ICC.AUT. This signature is encrypted using PK.IFD.AUT and sent to the IFD.

3. The IFD verifies the response using SK.IFD.AUT, PK.ICC.AUT and the same hash method.
4. The IFD request a challenge RND.ICC from the ICC using the ISO/IEC command GET CHALLENGE.
5. Then the IFD generates a random number K.IFD used to derive session keys later and a padding random number PRND2. As shown in table [Table 3] this data is concatenated and a hash is calculated. Then a signature is generated using SK.IFD.AUT. This signature is encrypted using PK.ICC.AUT and sent to the ICC.
6. After the corresponding keys have been selected the ICC verifies the signature.

IFD	Transmission	ICC
INTERNAL AUTHENTICATE C = RND.IFD SN.IFD Decrypt with SK.IFD.AUT and verify response from ICC with PK.ICC.AUT	 	Application calculates: E.PK.IFD(Sig _{SK.ICC.AUT} (PRND1 K.IFD h[PRND1 K.IFD C]))
IFD has now authenticated the card.		
GET CHALLENGE	 	RND.ICC
EXTERNAL AUTHENTICATE IFD calculates: E.PK.ICC(Sig _{SK.IFD.AUT} (PRND2 K.ICC h[PRND2 K.ICC S])) with S = RND.ICC SN.ICC	 	verify the signature OK
ICC has now authenticated the IFD, i.e. mutual authentication is now complete.		

Table 3: Device Authentication Flow (retrieval of public keys and MANAGE SECURITY ENVIRONMENT commands have been omitted)

GIF	E-Sign K
IAS_AuthGetData() [GIF, Part 3], section 4.4.5, page 25: “This function is required for an off-card application to obtain:	Read cryptographic token information Prior to authentication the IFD might require parameters on how to proceed with, and where to find resources relevant

<ul style="list-style-type: none"> • The data which is to be authenticated (often identical to identification data) • The operational means through which to perform this authentication (supported algorithms etc)” 	<p>for, the authentication.</p> <p>The IFD may issue commands to read files from DF.CIA to retrieve information for the authentication parameters. The files in this DF contain data objects that describe the structure and identification tags (file IDs). In particular, it may retrieve information on:</p> <ul style="list-style-type: none"> • the authentication algorithm • format of certificates • presence of specific certificates • key related information (key ID’s, key length etc.) <p>This can be done using the ISO/IEC 7816-4 SELECT and READ BINARY commands.</p>
<p>IAS_IntAuth(Chal)</p> <p>[GIF, Part 3], section 4.4.6, page 25:</p> <p>“This function enables an off card application to authenticate a subject by sending a challenge (random value) to the IAS application. The challenge is then operated upon by the IAS application and sent back to the off card application which can verify it using the operational data/algorithms returned by IAS_AuthGetData().”</p>	<p>The key transport protocol ([E-SIGN K, Part 1], section 8.4) outlined above comprises a mutual authentication with session key generation, so that secure messaging can be used for following commands.</p>

Table 4: Device Authentication – Functional Mapping

User Authentication/Verification

GIF	E-Sign K
<p>IAS_GetCstDta()</p> <p>[GIF, Part 3], section 4.4.9, page 25/26:</p> <p>“This function returns the formats and protocols to be used to provide a proof of user consent from the current subject to the IAS application.</p> <p>The returned data needs further formalisation and must be able to indicate for example what type of data will be used (PIN code, biometric), with what format and algorithms, etc.”</p>	<p>Biometric User Verification</p> <p>[E-SIGN K, Part 1], section 6.2.1, page 7-22</p> <p>Information about the applied matching algorithm, biometric type, biometric type instance and so on are stored in a Biometric Information Template (BIT) as defined in ISO/IEC 7816-11. These BITs can be retrieved using the GET DATA command or using ISO 7816-15.</p>

<p>IAS_UsrCstVerif(Data) [GIF, Part 3], section 4.4.10, page 26: “In line with the consent protocol enquiry function above, this function requires further formalisation. It is mandatory that the data is sent to the card by secure channel.”</p>	<p>Biometric user verification and knowledge based user verification (PIN): The ISO/IEC command VERIFY is used for this purpose. In the case of biometric user verification secure messaging has to be used.</p>
---	--

Table 5: User Authentication – Functional Mapping

3.3 Signature

3.3.1 Global Interoperability Framework

According to document [GIF, Part 3] a Signature is defined as (see section 4.1.2, p.20):

“A process through which the smart card - triggered by the cardholder – performs a digital signature on an object presented by an off card application on behalf of one of the subjects managed by the IAS application.”

The framework distinguishes between a signature generated by the Smart Card subject and a trusted signature generated by the Public ID subject:

- A signature generated by the Smart Card subject will be used as part of security processes, has no legally binding and for this reason can be generated without any prerequisites (in the following denoted as un-trusted signature).
- A signature generated by the Public ID subject is legally binding and thus must only be generated under strict control by the card holder him/her-self. A trusted Signature is performed by an authenticated Public ID subject.

The implementation of the IAS application depends on asymmetric cryptography and related Public Key Infrastructures ([GIF, Part 3], section 4.5.1).

3.3.2 E-Sign K Specification

The E-Sign K specification does not explicitly distinguish between trusted and un-trusted signatures. Nevertheless signatures are used without any prerequisites and without any legally binding in the authentication schemes (see above) corresponding to an un-trusted signature in the Global Interoperability Framework.

The Digital Signature Service described in the E-Sign K specification is under strict control of the user as is the trusted signature of the GIF as shown in the following figure:

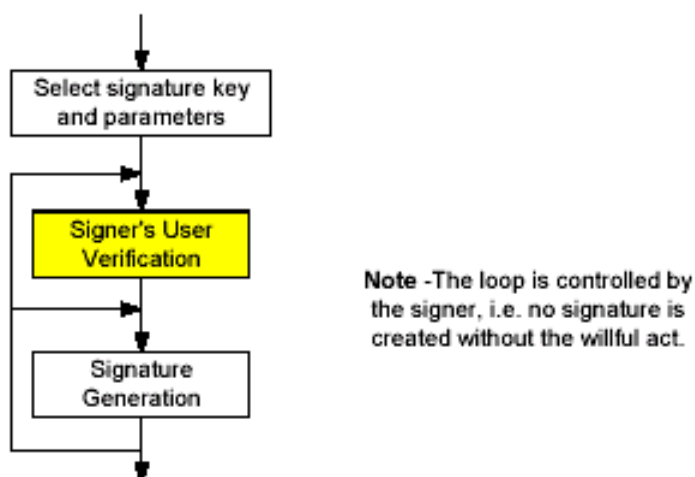


Figure 1: *Signer's User Verification* [E-SIGN K, Part 1]

In E-Sign K different keys (associated to different identity token or subjects) are used in order to generate the signatures in the authentication process and the Digital Signature Service corresponding to the GIF requirements.

The signature process itself 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. An overview is given in **Figure 2**:

An arbitrary message (1) is optionally formatted in an application specific way (2). This format mechanism is outside the scope of the E-Sign K specification. The data has to be hashed by applying a hash function (3). This can be done either completely outside the ICC, partly outside and partly inside the ICC or completely inside the ICC. Depending on the signature algorithm the hash value has to be formatted by a format mechanism (4) The final Digital Signature Input (DSI) is the string to be transformed by the signature algorithm. The result is the Digital Signature digSig (5).

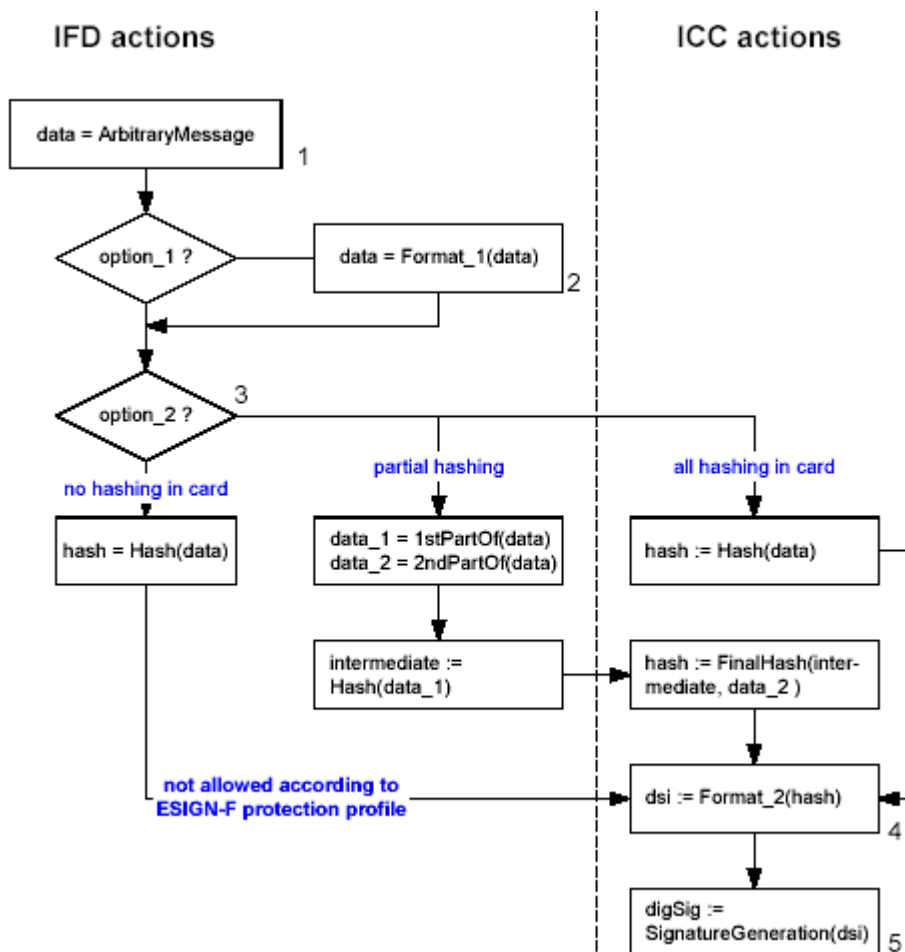


Figure 2: Data Flow chart for a signature generation process[E-SIGN K, Part 1]

3.3.3 Mapping

GIF	E-Sign K
IAS_SignGetData() [GIF, Part 3], section 4.4.7, page 25: “This function is required for an off card application to obtain the operational means through which to perform the signature (supported algorithms etc)”	Selection of different algorithms and input formats [E-SIGN K, Part 1], section 7.5, page 7-28 “If the card supports several digital signature algorithms or different Digital Signature Input formats ... then the algorithm or the DSI format respectively has to be selectable.” The ISO/IEC 7816-4 command MANAGE SECURITY ENVIRONMENT is used for this purpose.

	It is also possible that e.g. an algorithm is implicitly specified if a key is selected.
IAS_GenSign(Data) [GIF, Part 3], section 4.4.8, page 25: “This function actually performs the signature of the data (ID + hash DTBS) provided as a parameter.”	[E-SIGN K, Part 1], section 7.4, page 7-27 The data to be hashed or the final digest are sent to the ICC with the PERFORM SECURITY OPERATION: HASH (PSO:HASH) command as described in ISO/IEC 7816-8. The digital signature is calculated by means of the PSO:COMPUTE DIGITAL SIGNATURE command as described in ISO/IEC 7816-8. The digital signature input format is specified through the appropriate signature algorithm and computed by the ICC.

Table 6: Signature – Functional Mapping

GIF
Digital Signature Service According to [GIF, Part 2], section 2.2.3, page 14 one of the Primary IAS Processes is to “Sign an information object (if requested/required by the e-service)”. This object can be presented by an off-card application (see Table 6), but can also be an ICC file holding identification data. This feature is not covered by E-Sign K and has to be specified. This specification shall be compliant to ISO/IEC 7816 if possible. For the signature generation the command PSO:COMPUTE DIGITAL SIGNATURE shall be used. The hashing of data stored in the ICC is currently not covered by ISO/IEC 7816.

Table 7: Signature – Additional Features

3.4 Secure Messaging

Secure Messaging is used in order to protect the integrity and/or confidentiality of the information transmitted between ICC and IFD and vice versa.

3.4.1 Global Interoperability Framework

According to GIF Secure Messaging between the IAS application and the off card application is an additional concept, see ([GIF, Part 3], subsection 4.3.2). This framework contains no requirements concerning secure messaging.

3.4.2 E-Sign K Specification

Secure Messaging is not mandated for every command of the E-Sign K specification, but highly recommended.

Secure Messaging as specified by E-Sign K is compliant to ISO/IEC 7816-4. The two required triple-DES session keys are derived using the two 32byte long random numbers (denoted as K_{IFD} and K_{ICC}) that have been exchanged encrypted during the mutual authentication between the IFD and ICC. One session key is used to encrypt (decrypt) the data field of the APDU (triple-DES, CBC-mode), the other key is used to calculate a checksum including the APDU-header and the encrypted data field by means of a retail-MAC. A Secure Sequence Counter (SSC) is also included in the checksum calculation. The initial value of the SSC is also derived from the random numbers K_{IFD} and K_{ICC} . Every time a checksum is calculated the value of the SSC is incremented so that the transmitted APDUs are linked. For security reasons the data field is encrypted first and then the checksum is calculated.

3.4.3 Mapping

As mentioned above the GIF framework contains no special requirements concerning Secure Messaging. For this reason the schema specified by E-Sign K can be used for the IAS application without any changes

eEpoch

(eEurope Smart Card Charter proof of concept and
holistic solution)

**Generic IAS Application / System package based on
E-Sign K**

Giesecke & Devrient

Generic IAS Application / System package based on E-Sign K

Document Control Sheet	
Responsible Author(s):	Jens Urmann
Organisation:	Giesecke & Devrient
Subject / Title of Document:	Generic IAS Application / System package based on E-Sign K
Related Task('s):	
Deliverable No.	
Save Date of File:	09/05/2003
Version Number:	1.0
Ref./File Name	Generic-IAS-Application_v10.doc
Number of Pages	45

Document Distribution			
Member type	Organisation	Name	Distributed
Webpage	[Project Web Site]	Internet	DD/MM/YY
Contractors / Partners			
European Commission			
Additional			

Contents

1. Introduction	7
1.1 Version List	7
2. References.....	8
3. Abbreviations and notation.....	9
3.1 Abbreviations	9
3.2 Notation and coding conventions	9
4. Definitions	10
5. Selection of IAS-application	11
5.1 Trusted environment versus untrusted environment.....	11
5.2 Application Flow	11
5.3 Selection of IAS application	11
5.4 Selection of cryptographic information application	11
5.5 Concurrent usage of signature applications	11
5.6 Security environment selection	11
5.7 Key selection.....	11
5.8 Basic Security Services.....	11
6. User verification	12
6.1 Knowledge based user verification	12
6.2 Biometric user verification	12
7. Digital Signature Service.....	13
8. Device Authentication	14
8.1 Certification authorities and certificates	14
8.2 Authentication environments.....	14
8.3 Key transport and key agreement mechanisms.....	14
8.4 Key transport protocol	14
8.5 Device authentication with privacy protection	14
8.6 Authentication summary.....	14
8.7 Symmetric authentication scheme	14
8.8 Compute session keys from key seed $K_{\text{FD/ICC}}$	14
8.9 Compute send sequence counter SSC.....	15
8.10 Post-authentication phase.....	15
8.11 Reading the Display Message	15
8.12 Updating the Display Message	15

9. Secure Messaging	16
10. Key Generation	17
10.1 Key generation and export using SK.ICC.AUT	17
10.2 Key generation and export with dynamic SM.....	17
10.3 Key import with externally generated key	17
10.4 Keys in Static Secure Messaging.....	17
11. Key identifiers and parameters	18
11.1 Key identifiers (KID)	18
11.2 Public Key Parameters.....	18
11.3 DSA with ECC public key parameters.....	18
11.4 Diffie-Hellman key exchange parameters	18
12. APDU data structures	19
12.1 CRTs	19
12.2 Key transport device authentication protocol	19
12.3 Privacy device authentication protocol.....	19
13. AlgIDs, Hash- and Digital Signature Input Formats	20
13.1 Algorithm Identifiers and OIDs	20
13.2 Hash-Input-Formats	20
13.3 Formats of the Digital Signature Input (DSI)	20
14. CV_Certificates and Key Management	21
15. Files.....	22
16. Cryptographic Information Application	23
17 Additional Functionality	24
17.1 Perform Hash of File	24
17.1.1 Description	25
17.1.2 Command Message	25
17.1.3 Response Message.....	25
17.2 Perform Hash of Data Object	26
17.2.1 Description	26
17.2.2 Command Message	26
17.2.3 Response Message.....	27
17.3 Definition of Security Environments	27
17.3.1 CRT for Authentication (AT)	28
17.3.2 CRT for Cryptographic Checksum (CCT).....	29
17.3.3 CRT for Digital Signature (DST).....	29
17.3.4 CRT for Confidentiality (CT)	30

17.3.5 Security Environments	30
17.3.6 Security Environment #1	31
17.3.7 Security Environment #2	31
17.3.8 Security Environment #3	31
17.3.9 Security Environment #4	32
17.3.10 Security Environment #5	32
17.3.11 Security Environment #6	33
17.4 Definition of File Control Information Templates	33
17.4.1 File Control Parameter	33
17.4.2 File Management Data	34
17.4.3 File Control Information	34
17.4.4 Data Objects in FCP, FMD and FCI Templates	35
17.5 Mapping of E-Sign K Access Conditions on IAS Application File Control Parameters.....	36
17.5.1 Access Conditions.....	37
17.5.2 Security Attributes in expanded Format	37
17.6 Key Generation and Export using SK.ICC.AUT	39
17.6.1 Key Generation	40
17.6.2 Key Export by means of the READ BINARY command	41
17.6.3 Key Export by means of the GET DATA command	41
Annex 1: Features of E-Sign K Part 2	45

List of Tables

<i>Table 1: Version List</i>	7
<i>Table 2 Command sequence for downloading signed data files</i>	25
<i>Table 3 PSO: PERFORM HASH OF FILE: Command Message</i>	25
<i>Table 4 Perform Hash of File: Status words</i>	26
<i>Table 5 Perform Hash of File: Command Message</i>	26
<i>Table 6 Control Reference Templates for Security Environments</i>	27
<i>Table 7 Data Objects in CRTs</i>	27
<i>Table 8 Security Environments</i>	31
<i>Table 9 File Control Parameter data objects</i>	34
<i>Table 10 File Management Data data objects</i>	34
<i>Table 11 File Control Information data objects</i>	34
<i>Table 12 Short File Identifier encoding</i>	35
<i>Table 13 Access Mode data object</i>	36
<i>Table 14 Access Mode data object</i>	36
<i>Table 15 Access Conditions of IAS Files</i>	37
<i>Table 16 EF.ARR</i>	39
<i>Table 17 ARR as function of SE and File</i>	39
<i>Table 18 Key generation execution flow 1 of 3</i>	40
<i>Table 19 Key generation execution flow 2 of 3</i>	40
<i>Table 20 Key generation execution flow 3 of 3</i>	41
<i>Table 21 GET DATA command for key export</i>	42
<i>Table 22 Extended header list for public key export</i>	42
<i>Table 23 GET DATA command message, step 1 of 3</i>	42
<i>Table 24 GET DATA response message, step 1 of 3</i>	43
<i>Table 25 GET DATA command message, step 2 of 3</i>	43
<i>Table 26 GET DATA response message, step 2 of 3</i>	43
<i>Table 27 GET DATA command message, step 3 of 3</i>	43
<i>Table 28 GET DATA response message, step 3 of 3</i>	44

1. Introduction

The aim of the Global Interoperability Framework (GIF) for Identification, Authentication and electronic Signature (IAS) with Smart Cards for Internet applications is to facilitate interoperability between various IAS schemes. Therefore a Generic IAS application has been specified from a high level perspective in document [GIF, Part 3]. These requirements have been mapped to the functionality of the E-Sign K Smart Card specification, cf. [E-SIGN K, Part 1] and [E-Sign K, Part 2], in document [Mapping]. The E-Sign K standard is currently developed in order to support the EU-directive on electronic signatures and the key issue of the specification is to enable interoperability.

The present document defines a subset of the E-Sign K specification that shall be used for the Generic IAS application. Therefore this document has the same structure (i.e. chapters and sections) as the first part of the E-Sign K specification. The required features of the second part of the E-Sign K specification are considered in the Annex. In chapter 17 Additional Functionality features that enhance the specification of E Sign K are taken into consideration.

In order to define the subset the results of the questionnaire, cf [Questionnaire], are regarded. This questionnaire has been answered by pilot sites intending to use the Generic IAS application.

The specification of the IAS/IOP interface will be based upon the system package defined in this document.

1.1 Version List

<i>Version / Date</i>	<i>Brief Description of Changes, Name</i>
0.1 / 06.03.2003	Initial Document, Jens Urmann (uj)
0.2 / 29.03.2003	first draft of additional functionality, uj
0.3 / 31.03.2003	included comments of Gisela Meister, updated to version 0 release 16 of E-Sign K, uj
0.4 / 06.04.2003	added section "Mapping of E-Sign K Access Conditions on IAS Application File Control Parameters", uj
0.5 / 15.04.2003	updated to Version 1, Release 1 of E-Sign K included comments of Mourad Faher and R. Rosset of Schlumberger, uj
0.6 / 08.05.2003	moved section key generation/export from chapter 10 to 17.6.1 and 17.6.2 added section 17.6.3: key export using a GET DATA command, uj
1.0 / 12.05.2003	Document delivered to the EC, uj

Table 1: Version List

2. References

- [E-SIGN K, Part 1] Application Interface for smart cards used as Secure Signature Creation Devices, Version 1 Release 1, 08.04.2003, CEN/ISSS WS/E-Sign Draft CWA Group K; Part 1 – Basic requirements
- [E-Sign K, Part 2] Application Interface for SmartCards used as Secure Signature Creation Devices, Version 0 Release 10, 01.06.2002, CEN/ISSS WS/E-Sign Draft CWA Group K; Part 2 – Optional features
- [GIF, Part 3] Open Smart Card Infrastructure for Europe v2, Volume 3: Global Interoperability Framework for identification, authentication and electronic signature (IAS) with smart cards, Part 3: Recommendations for interoperability specifications, Version 1.00, March 2003, eESC GIF Expert Group
- [ISO/IEC 7816] ISO/IEC FCD 7816, "Information technology – Identification cards- Integrated circuit(s) cards with contact"
Part 4: Interindustry commands for interchange", FCD2003
Part 8: Security related interindustry commands, FCD 2003
Part 11: Personal verification through biometric methods, FCD 2002
Part 15: Cryptographic information application, FDIS 2003
- [Mapping] Functional Mapping of GIF on E-Sign K, Version 1.3, 12.05.2003, Giesecke & Devrient
- [Questionnaire] Questionnaire for Pilots, Version 1.2, 10.03.2003, Schlumberger, Giesecke & Devrient
- [LoadApplets] LOAD of APPLETS protocol, Draft 1.0, 17.04.2003, Schlumberger
- [Tachograph] Commission Regulation (EC) No 1360/2002, 13.06.2002, Official Journal of the European Communities L207

3. Abbreviations and notation

3.1 Abbreviations

The following abbreviations are used throughout this document:

AC	Access Condition
AID	Application Identifier
ARR	Access Rule Reference
AT	Authentication Template
CCT	Cryptographic Checksum Template
CT	Confidentiality Template
CIA	Cryptographic Information Application
CRT	Control Reference Template
DST	Digital Signature Template
EF	Elementary File
FCI	File Control Information
FCP	File Control Parameter
FMD	File Management Data
HT	Hash Template
IAS	Identification, Authentication and electronic Signature
ICC	Integrated Circuit Card
IFD	Interface Device
IOP	Interoperability
GIF	Global Interoperability Framework
KID	Key Identifier
PSO	Perform Security Operation
SE	Security Environment
SEID	Security Environment Identifier
SFI	Short File Identifier
SM	Secure Messaging
UQB	Usage Qualifier Byte

3.2 Notation and coding conventions

The same notation and coding conventions as in the E-Sign K specification are used. Please refer to document [E-SIGN K, Part 1].

4. Definitions

For a list of definitions please refer to the corresponding chapter of the E-Sign K specification [E-SIGN K, Part 1].

5. Selection of IAS-application

5.1 Trusted environment versus untrusted environment

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

5.2 Application Flow

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

5.3 Selection of IAS application

The IAS application is selected as described in this chapter. Therefore an application identifier (AID) registered e.g. by the ISO registration authority is required.

5.4 Selection of cryptographic information application

The AID of the cryptographic information application (CIA) has to be redefined for the IAS application.

5.5 Concurrent usage of signature applications

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

5.6 Security environment selection

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

5.7 Key selection

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

5.8 Basic Security Services

There are three basic security services defined in the E-Sign K specification. With respect to the IAS application

- the digital signature service shall be considered as mandatory,
- the authentication service (Client-Server Authentication) shall be optional as this service is required by one pilot site only according to the answers to the questionnaire,
- the key decipherment service shall not be considered for the IAS application as it is not required.

6. User verification

6.1 Knowledge based user verification

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

6.2 Biometric user verification

According to the answers of the pilot sites to the questionnaire this feature is required by one pilot site only (as a PUK). For this reason biometric user verification shall be – as in the case of the E Sign K specification - an optional feature for the IAS application.

7. Digital Signature Service

This chapter of the E-Sign K specification shall be used for the IAS specification without any changes.

Please refer to chapter 17 for additional functionality with respect to the Digital Signature Service!

8. Device Authentication

Please note: For the load of applets protocol a symmetric authentication scheme is required that is specified in document [LoadApplets]. The Secure Messaging scheme used in the context of the load of applets protocol is also specified in document [LoadApplets].

8.1 Certification authorities and certificates

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.2 Authentication environments

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.3 Key transport and key agreement mechanisms

As defined below only the key transport protocol described in section 8.4 of [E-SIGN K, Part 1] shall be taken into consideration for the IAS application.

8.4 Key transport protocol

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.5 Device authentication with privacy protection

Since privacy protection is not required for the IAS application, the protocol described in this section of the E-Sign K specification shall not be used for the IAS specification.

8.6 Authentication summary

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.7 Symmetric authentication scheme

According to the Global Interoperability Framework [GIF, Part 3] the IAS application shall be based upon asymmetric cryptography. For this reason the symmetric protocol described in this section shall not be used for the IAS specification.

8.8 Compute session keys from key seed $K_{IFD/ICC}$

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.9 Compute send sequence counter SSC

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.10 Post-authentication phase

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.11 Reading the Display Message

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

8.12 Updating the Display Message

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

9. Secure Messaging

This chapter of the E-Sign K specification shall be used for the IAS specification without any changes.

Please note: For the load of applets protocol a symmetric authentication scheme is required that is specified in document [LoadApplets]. The Secure Messaging scheme used in the context of the load of applets protocol is also specified in document [LoadApplets].

10. Key Generation

10.1 Key generation and export using SK.ICC.AUT

This section of the E-Sign K specification shall be used as basis for the IAS specification, but some clarification with respect to the public key export and the execution flow is necessary, please refer to section 17.6 of the present document.

10.2 Key generation and export with dynamic SM

As an online registration process is not required for the IAS application this feature shall not be supported.

10.3 Key import with externally generated key

As this feature does not provide the same security level as on card key generation this feature shall not be taken into consideration for the IAS application.

10.4 Keys in Static Secure Messaging

This feature shall not be taken into consideration for the IAS application as it is not required.

11. Key identifiers and parameters

11.1 Key identifiers (KID)

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

In addition the optional Client-Server authentication key SK.CH.AUT shall be considered.

11.2 Public Key Parameters

For the device authentication only the key transport protocol specified in section 8.4 of [E-SIGN K, Part 1] shall be used. Therefore only RSA keys are used, i.e. the subsections 11.2.3 of [E-SIGN K, Part 1] shall not be considered with respect to the IAS application. All other subsections shall be used for the IAS specification without any changes.

11.3 DSA with ECC public key parameters

This section of the E-Sign K specification shall not be considered for the IAS specification as it refers to a device authentication protocol that is not supported by the IAS application, see section 8.5.

11.4 Diffie-Hellman key exchange parameters

This section of the E-Sign K specification shall not be considered for the IAS specification as it refers to a device authentication protocol that is not supported by the IAS application, see section 8.5.

12. APDU data structures

12.1 CRTs

This section of the E-Sign K specification shall be used for the IAS specification, but subsection 12.1.3 shall not be considered as it refers to a device authentication protocol that is not supported by the IAS application, see section 8.5.

12.2 Key transport device authentication protocol

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

12.3 Privacy device authentication protocol

This section shall not be taken into consideration as it refers to a device authentication protocol that is not supported by the IAS application, see section 8.5.

13. AlgIDs, Hash- and Digital Signature Input Formats

13.1 Algorithm Identifiers and OIDs

This section of the E-Sign K specification shall be used for the IAS specification. Algorithm Identifier that are used in device authentication protocols not supported by the IAS application shall not be taken into consideration.

13.2 Hash-Input-Formats

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

13.3 Formats of the Digital Signature Input (DSI)

This section of the E-Sign K specification shall be used for the IAS specification, but subsections 13.3.3 and 13.3.4 shall not be taken into consideration as they refer to a device authentication protocol that is not supported by the IAS application, see section 8.5.

14. CV_Certificates and Key Management

This section of the E-Sign K specification shall be used for the IAS specification without any changes.

15. Files

This section of the E-Sign K specification shall be used for the IAS specification (the file structure is only an example). Further files are required e.g. in order to store the identification data while the file EF.DH is not required as it is related to a device authentication protocol that is not supported by the IAS application.

16. Cryptographic Information Application

This chapter of the E-Sign K specification shall be used for the IAS specification. Furthermore the CIA application shall match the following requirements:

- the Security Environments shall be listed in the CIAInfo object using the selInfo element.
- all keys shall be referenced including all relevant information, e.g. access conditions, PIN format, kind of biometric template
- all files (accessible from the IFD) should be referenced including all relevant information, e.g. access conditions

17 Additional Functionality

In this chapter features that are required either by the Global Interoperable Framework GIF, cf. [GIF, Part 3] or by the pilot sites (according to the answers to the questionnaire) are specified.

17.1 Perform Hash of File

As described in document [Mapping] there is also the requirement to read signed data stored on the ICC. If this data is stored in a transparent EF, the command `PSO: PERFORM HASH OF FILE` as specified in the context of the tachograph system [Tachograph] shall be used to compute the hash value. There is one enhancement: In the tachograph system the hash algorithm is prescribed while in the IAS Application the hash algorithm shall be selectable (from the algorithms E-Sign K supports). The resulting hash value is used by the command `PSO: COMPUTE DIGITAL SIGNATURE` in order to generate the signature.

In the following table the command sequence for downloading a signed data file is depicted. The presentation stays close to document [Tachograph]. The MSE commands and the user verification (depending on the private key used) are not depicted.

ICC	Direction	IFD	Description
	←	Select File	Select a transparent EF
OK	→		
	←	Perform Hash of File	Calculates the hash value over the data content of the selected file using the prescribed hash algorithm by means of a proprietary command.
Calculate hash of file and store hash value temporarily			
OK	→		
	←	Read Binary	Repeat the command until the complete file is read.
File data, OK	→	Store received data	Store data transparent (i.e. preserve order of bits and bytes) in one file.
	←	PSO: COMPUTE DIGITAL SIGNATURE	
Compute digital signature using the temporarily stored			

hash value			
Signature, OK	→	Append Signature to the previous stored data of the file	Store data transparent (i.e. preserve order of bits and bytes).

Table 2 Command sequence for downloading signed data files

17.1.1 Description

The command `PERFORM HASH OF FILE` is used to hash the data area of the currently selected transparent EF. The resulting hash value is stored in the ICC and can be used by a `PSO: COMPUTE DIGITAL SIGNATURE` command in order to generate the digital signature of the data in the corresponding EF. The hash value remains available for the `PSO: COMPUTE DIGITAL SIGNATURE` command until the next hash value is generated successfully and stored in the ICC.

The command `PERFORM HASH OF FILE` is not ISO/IEC 7816-4 compliant, please refer to [ISO/IEC 7816]. For this reason the CLA byte of the command indicates that there is a proprietary use of the ISO/IEC command `PERFORM SECURITY OPERATION: HASH`.

17.1.2 Command Message

Byte	Value	Description
CLA	'80'	Proprietary command
INS	'2A'	Perform Security Operation
P1	'90'	Hash
P2	'00'	Hash the data of the currently selected transparent file

Table 3 PSO: `PERFORM HASH OF FILE`: Command Message

17.1.3 Response Message

The command returns no data, but the status words listed in the following table. The description in brackets follows ISO/IEC 7816-4, see [ISO/IEC 7816].

Status Words	Description
'90 00'	Normal processing
'69 85'	No application is selected (conditions of use not satisfied)
'64 00 '	Selected EF is considered corrupted (execution error)
'65 81 '	Selected EF is considered corrupted (memory failure)

'69 86'	Selected file is not a transparent file (command not allowed, no current EF)
---------	--

Table 4 Perform Hash of File: Status words

17.2 Perform Hash of Data Object

If the data that shall be retrieved from the ICC and signed by the ICC is stored in a data object the command `PSO: PERFORM HASH OF DATA OBJECT` shall be used in order to generate the hash value. The hash algorithm shall be selectable (from the algorithms E-Sign K supports) and the resulting hash value is used by the command `PSO: COMPUTE DIGITAL SIGNATURE` in order to generate the signature.

17.2.1 Description

The command `PERFORM HASH OF DATA OBJECT` is used to hash a data object stored in the ICC. Therefore an extended header list is provided in the command data field that specifies the relevant data object(s). The resulting hash value is stored in the ICC and can be used by a `PSO: COMPUTE DIGITAL SIGNATURE` command in order to generate the digital signature of the corresponding data object. The hash value remains available for the `PSO: COMPUTE DIGITAL SIGNATURE` command until the next hash value is generated successfully and stored in the ICC.

The command `PERFORM HASH OF DATA OBJECT` is not ISO/IEC 7816-4 compliant, please refer to [ISO/IEC 7816]. For this reason the CLA byte of the command indicates that there is a proprietary use of the ISO/IEC command `PERFORM SECURITY OPERATION: HASH`.

17.2.2 Command Message

<i>Byte</i>	<i>Value</i>	<i>Description</i>
CLA	'80'	Proprietary command
INS	'2A'	Perform Security Operation
P1	'90'	Hash
P2	'AC'	data object to be hashed
Lc	present	length of extended header list in data field
Data field	{'4D' – L _{4D} ' ..}	extended header list according to ISO/IEC 7816-4
Le	absent	

Table 5 Perform Hash of File: Command Message

The extended header list provided in the command data field specifies the relevant data object(s) according to ISO/IEC 7816-4, cf. [ISO/IEC 7816] section 9.3.1. The value of P2 denotes whether the byte string resulting from the extended header list

consists of a concatenation of data elements or data objects. This byte string serves as input for the hash operation.

Case1: P2 = 'AC', the byte string consists of the value fields of the data objects, possibly truncated according to the indicated lengths.

17.2.3 Response Message

The response message is empty. Interindustry values of SW1-SW2 for warning and error condition according to ISO/IEC 7816-4 shall be returned, cf. [ISO/IEC 7816].

17.3 Definition of Security Environments

For the IAS application the following interindustry types of Control Reference Templates shall be used in the context of Security Environments:

<i>Tag</i>	<i>Value</i>
'A4'	CRT valid for authentication (AT)
'B4'	CRT valid for Cryptographic Checksum (CCT)
'B6'	CRT valid for Digital Signature (DST)
'B8'	CRT valid for Confidentiality (CT)

Table 6 Control Reference Templates for Security Environments

The contents of the CRT DOs referenced by these tags consists of DOs from the following list:

<i>Tag</i>	<i>Length</i>	<i>Value</i>
'95'	'01'	Usage Qualifier Byte (UQB)
'83' or '84'	variable	Key Reference
'80'	variable	Algorithm ID

Table 7 Data Objects in CRTs

The Usage Qualifier Byte is optional but if present shall be the first DO. The UQB shall be encoded according to ISO/IEC 7816-4.

The DO with tag '83' references a secret or a public key, the tag '84' references a private key, see ISO/IEC 7816-4. Key identifiers are not prescribed by the E Sign K specification. The key identifiers are specified in the CIA application.

If the algorithm ID is present it shall be compliant to the E Sign K specification.

A CRT shall neither be empty nor contain only a Usage Qualifier Byte DO.

In the following sections the CRTs for the IAS application are specified.

17.3.1 CRT for Authentication (AT)

The UQB is optional as well as the algorithm ID. The key reference is mandatory.

The following authentication templates are required:

- AT_1 for knowledge based user authentication using the PIN PIN.CH.ADMIN
- AT_2 for knowledge based user authentication using the PIN PIN.CD.DS
- AT_3 for external device authentication
- AT_4 for internal device authentication
- AT_5 for client-server authentication

These templates are encoded as follows:

```
AT_1 := ('A4' - L_A4' - //CRT authentication (AT)
        ('95' - '01' - '08') //UQB: user authentication, knowledge based
        ('83' - L_83 - reference to PIN.CH.ADMIN)
        )
AT_2 := ('A4' - L_A4' - //CRT authentication (AT)
        ('95' - '01' - '08') //UQB: user authentication, knowledge based
        ('83' - L_83 - reference to PIN.CH.DS)
        )
AT_3 := ('A4' - L_A4' - //CRT authentication (AT)
        ('95' - '01' - '80') //UQB: external authentication
        ('83' - L_83 - reference to PK.IFD.AUT)
        ('84' - L_84 - reference to SK.ICC.AUT)
        ('80' - '01' - '1A') //algorithm ID for device authentication
        )
AT_4 := ('A4' - L_A4' - //CRT authentication (AT)
        ('95' - '01' - '40') //UQB: internal authentication
        ('83' - L_83 - reference to PK.IFD.AUT)
        ('84' - L_84 - reference to SK.ICC.AUT)
        ('80' - '01' - '1A') //algorithm ID for device authentication
        )
AT_5 := ('A4' - L_A4' - //CRT authentication (AT)
        ('95' - '01' - '40') //internal authentication
        ('84' - L_84 - reference to SK.CH.AUT)
        ('80' - '01' - 'xx') //algorithm reference, see [E-SIGN K, Part 1]
        //table 13.1
```

)

17.3.2 CRT for Cryptographic Checksum (CCT)

The UQB is optional, the key reference is mandatory, the algorithm ID shall not be used.

The following CCT templates are required:

- CCT_1 for Secure Messaging in the command data fields
- CCT_2 for Secure Messaging in the response data fields

These templates are encoded as follows:

```
CCT_1 := ('B4' - L_B4' - //CRT cryptographic checksum (CCT)
          ('95' - '01' - '10') //UQB: SM in command data field
          ('84' - L_84' - reference to macing session key K(MAC))
        )
```

```
CCT_2 := ('B4' - L_B4' - //CRT cryptographic checksum (CCT)
          ('95' - '01' - '20') //UQB: SM in response data field
          ('84' - L_84' - reference to macing session key K(MAC))
        )
```

17.3.3 CRT for Digital Signature (DST)

The UQB is optional as well as the algorithm ID. The key reference is mandatory.

The following DST template is required:

- DST_1 for a digital signature using SK.CH.DS
- DST_2 for certificate verification (in the context of device authentication) using the root CA key PK.RCA.AUT
- DST_3 for certificate verification (in the context of device authentication) using a public key of the certificate chain

These templates are encoded as follows:

```
DST_1 := ('B6' - L_B6' - //CRT Digital Signature
          ('95' - '01' - 'C0') //UQB: Computation (DST)
          ('84' - L_84' - reference to SK.CH.DS)
          ('80' - '01' - 'xx') //algorithm reference, see [E-SIGN K, Part 1]
          //table 13.1
        )
```

```
DST_2 := ('B6' - L_B6' - //CRT Digital Signature
          ('95' - '01' - '80') //UQB: Verification (DST)
          ('83' - L_83' - reference to PK.RCA.AUT)
          ('80' - '01' - 'xx') //algorithm reference, see [E-SIGN K, Part 1]
          //table 13.1
        )

DST_3 := ('B6' - L_B6' - //CRT Digital Signature
          ('95' - '01' - '80') //UQB: Verification (DST)
          ('83' - L_83' - reference to public key in certificate chain)
          ('80' - '01' - 'xx') //algorithm reference, see [E-SIGN K, Part 1]
          //table 13.1
        )
```

17.3.4 CRT for Confidentiality (CT)

The UQB is optional, the key reference is mandatory, the algorithm ID shall not be used.

The following CT templates are required:

- CT_1 for Secure Messaging in the command data fields
- CT_2 for Secure Messaging in the response data fields

These templates are encoded as follows:

```
CT_1 := ('B8' - L_B8' - //CRT confidentiality (CT)
          ('95' - '01' - '10') //UQB: SM in command data field
          ('84' - L_84' - reference to encryption session key K(ENC))
        )

CT_2 := ('B8' - L_B8' - //CRT confidentiality (CT)
          ('95' - '01' - '20') //UQB: SM in response data field
          ('84' - L_84' - reference to encryption session key K(ENC))
        )
```

17.3.5 Security Environments

The following Security Environments are specified:

SE#	Meaning
SE#1	empty, no restrictions

SE#2	knowledge based user verification (PIN)
SE#3	mutual device authentication, secure messaging
SE#4	mutual device authentication, secure messaging, user verification (PIN)
SE#5	mutual device authentication, secure messaging, signature (using PSO: PERFORM HASH OF FILE or PSO: PERFORM HASH OF DO)
SE#6	Mutual device authentication, secure messaging, client-server authentication

Table 8 Security Environments

17.3.6 Security Environment #1

This SE is empty.

17.3.7 Security Environment #2

Knowledge based user verification

```
( '7B' - L7B - //interindustry template
    ('80' - '01' - '02') //SE#2
    AT_1 //user authentication, knowledge based
)
```

17.3.8 Security Environment #3

```
( '7B' - L7B - //interindustry template
    ('80' - '01' - '03') //SE#3
    DST_2 //verify certificate using PK.RCA.AUT
    DST_3 //verify certificate using any other key in
           //certificate chain
    AT_3 //external authentication
    AT_4 //internal authentication
    CT_1 //SM in command data field
    CT_2 //SM in response data field
    CCT_1 //SM in command data field
    CCT_2 //SM in response data field
)
```

17.3.9 Security Environment #4

```
('7B' – L7B –  
('80' – '01' – '04')  
DST_2 //verify certificate using PK.RCA.AUT  
DST_3 //verify certificate using any other key in  
//certificate chain  
AT_3 //external authentication  
AT_4 //internal authentication  
AT_1 //user authentication, knowledge based  
CT_1 //SM in command data field  
CT_2 //SM in response data field  
CCT_1 //SM in command data field  
CCT_2 //SM in response data field  
)
```

17.3.10 Security Environment #5

In this Security Environment the client-server authentication key SK.CH.AUT protected by the global PIN PIN.CH.ADMIN is used in order to create a signature. For the SK.CH.AUT the same CRT as in the case of the client-server authentication is used. This SE is used in order to read signed data from the ICC. The key SK.ICC.AUT (not protected by any PIN, using AT_4) or the signature key SK.CH.DS (using DST_1) can be used alternatively.

```
('7B' – L7B –  
('80' – '01' – '05')  
DST_2 //verify certificate using PK.RCA.AUT  
DST_3 //verify certificate using any other key in  
//certificate chain  
AT_3 //external authentication  
AT_4 //internal authentication  
AT_1 //user authentication, knowledge based  
CT_1 //SM in command data field  
CT_2 //SM in response data field  
CCT_1 //SM in command data field  
CCT_2 //SM in response data field  
AT_5 //Digital Signature using SK.CH.AUT  
)
```

17.3.11 Security Environment #6

According to E-Sign K Part 2 an Authentication Template as CRT is used in the context of Client-Server-Authentication:

```
(‘7B’ – L7B – //interindustry template
    (‘80’ – ‘01’ – ‘06’) //SE#6
    DST_2 //verify certificate using PK.RCA.AUT
    DST_3 //verify certificate using any other key in
           //certificate chain
    AT_3 //external authentication
    AT_4 //internal authentication
    AT_1 //user authentication, knowledge based
    CT_1 //SM in command data field
    CT_2 //SM in response data field
    CCT_1 //SM in command data field
    CCT_2 //SM in response data field
    AT_5 //client-server authentication
)
```

17.4 Definition of File Control Information Templates

In this chapter templates for file control information are specified using a subset of the interindustry templates according to ISO/IEC 7816-4. The IAS application shall return a template according to this specification in response to the SELECT command for every file.

The following interindustry templates for file information are used:

- File Control Parameter (FCP) template denoted by tag ‘62’
- File Management Data (FMD) template denoted by tag ‘64’
- File Control Information (FCI) template denoted by tag ‘6F’

17.4.1 File Control Parameter

The following data objects shall be used in the context of the File Control Parameter:

<i>Tag</i>	<i>Length</i>	<i>Value</i>	<i>Applies to</i>
‘80’	‘02’	Number of data bytes in the file, excluding structural information	EFs
‘82’	‘01’	File descriptor byte	DFs, transparent EFs

	'05'	File descriptor byte, data coding byte, maximum record size on two bytes, number of records on one byte	EFs of record structure
'83'	'02'	File identifier	any file
'84'	up to '10'	DF name (AID)	DFs
'88'	'00' or '01'	Short EF identifier	EFs
'8B'	'01', '03' or $n*2, n \geq 2$	Security attribute referencing the expanded format	any file
'A0'	var.	Security attribute template for data objects	DFs

Table 9 File Control Parameter data objects

The data objects referenced by tag '88', '8B' and 'A0' are optional.

17.4.2 File Management Data

By means of the File Management Data the ICC provides information about the EFs that belong to the IAS application DF. If the SELECT command is applied to this DF for every EF that belongs to the IAS application a data object with the following structure is included in the FMD:

Tag	Length	Value
'85'	variable	SFI and path of the EF

Table 10 File Management Data data objects

The FMD of an EF and of DFs other than the IAS application DF the FMD consists of the two bytes '64 00'.

17.4.3 File Control Information

The FCI of a DF comprises the following data objects:

Tag	Length	Value	Applies to
'84'	'01' to '10'	DF name (AID)	DFs
'A5'	variable	proprietary information	DFs

Table 11 File Control Information data objects

The proprietary information provided in the DO referenced by tag 'A5' is specified in the next section.

The FCI of an EF is given by the two bytes '6F 00'.

17.4.4 Data Objects in FCP, FMD and FCI Templates

17.4.4.1 Tag '80'

The value of this DO contains the number of data bytes in the file. A binary encoding applies to this value.

17.4.4.2 Tag '82'

The value of this DO comprises either 1 or 5 bytes, see **Table 9 File Control Parameter data objects**:

- The first byte is the file descriptor byte that is encoded according to ISO/IEC 7816-4.
- The second byte is the data coding byte according to ISO/IEC 7816-4.
- The third and the fourth byte contain the maximum record size. The third byte shall be '00' the fourth byte shall have a value '01' to 'FE'.
- The fifth byte denotes the maximum number of records of the EF. The value shall be '01' to 'FE'.

17.4.4.3 Tag '83'

The value of this DO contains the file ID. A binary encoding applies to this value.

The file ID shall be compliant to ISO/IEC 7816-4.

17.4.4.4 Tag '84'

The value of this DO contains the DF name that consists of 1 to 16 bytes. The value has to be unique in the context of the ICC and a binary encoding applies. The DF name can be an AID according to ISO/IEC 7816-5.

17.4.4.5 Tag '85'

The value of this DO contains in the first byte the SFI and in the following bytes the absolute path to the EF according to ISO/IEC 7816-4 without the file ID of the MF. A binary encoding applies to the value of the SFI that is encoded in the bits b8 to b4 of this byte. The values '00' and '1F' shall not be used:

<i>b8</i>	<i>b7</i>	<i>b6</i>	<i>b5</i>	<i>b4</i>	<i>b3</i>	<i>b2</i>	<i>b1</i>	<i>Meaning</i>
x	X	x	x	x	0	0	0	SFI, b8 to b4 not all set to 0, b8 to b4 not all set to 1

Table 12 Short File Identifier encoding

17.4.4.6 Tag '88'

The value of this DO contains a Short File Identifier encoded as described in **Table 12 Short File Identifier encoding**. This data object is optional.

17.4.4.7 Tag '8B'

The value of this DO contains the security attribute DO referencing expanded format as specified in ISO/IEC 7816-4. The access mode DOs and Security Condition DOs shall be compliant to ISO/IEC 7816-4. For interoperability reasons the access mode DO shall only contain the following interindustry data objects:

Tag	Length	Value
'80'	'01'	access mode byte
'81' to '8F'	variable	List of [part of] command headers

Table 13 Access Mode data object

This means the tag '9C' denoting a proprietary state machine description is not supported.

The expanded format requires the presence of an Access Rule Reference File, normally EF.ARR, or another file that is referenced by means of the expanded format, cf. [ISO/IEC 7816]. This file has the following structure: Every record contains one or more access rules. Every access rule consists of an access mode data object and one or more security condition data objects according to ISO/IEC 7816-4.

17.4.4.8 Tag 'A0'

The value of this DO contains the following pair of DOs:

Tag	Length	Value
'8B'	'01', '03' or 2*n, n >= 2	security attribute referencing the expanded format
'5C'	variable	tag list according to ISO/IEC 7816-6

Table 14 Access Mode data object

The DO referenced by tag '8B' shall always be the first DO. The structure of this DO is defined in the previous section. The DO referenced by tag '5C' contains a list of tags according to ISO/IEC 7816-6 that denotes the DO the access mode applies to.

17.5 Mapping of E-Sign K Access Conditions on IAS Application File Control Parameters

In this section the Access Conditions (ACs) with respect to files presented in chapter 15 of the E Sign K specification are mapped to the Security Environments and File

Control Parameters of the IAS application presented in the previous sections. Additional files required for the IAS application are currently not considered.

17.5.1 Access Conditions

In the following table the ACs of the E Sign K specification are summarised:

<i>Filename</i>	<i>Access Condition</i>
EF.DIR	Read: always, Write: never
EF.SN.ICC	Read: always, Write: never
EF.C.ICC.AUT	Read: always, Update: never
EF.C.CA _{ICC} .CS-AUT	Read: always, Update: never
EF.C_X509.CH.DS	Read: always, Update: never
EF.C_X509.CA.CS	Read: always, Update: never
EF.DM	Read: successful device authentication using SM Update: after successful presentation of PIN.CH.ADMIN

Table 15 Access Conditions of IAS Files

Please note that not all files are mandated, cf. [E-SIGN K, Part 1]. Access (write, update) to the certificate files and the EF.DIR file is granted for administrative purposes and not in the scope of the present document.

17.5.2 Security Attributes in expanded Format

In order to link these Access Conditions to Security Environments the FCP Security Attributes are used in expanded format. In general these Security Attributes in connection with SEs have the following format:

('8B' – L_{8B}' – file ID || SEID byte || ARR byte || SEID byte || ARR byte || ...)

The file ID (2 byte) is optional and only required in the case that the file EF.ARR is not used. For every SE denoted by the corresponding Security Environment Identifier (SEID, 1 byte) the Security Attribute is referenced by means of the Access Rule Reference byte (ARR, 1 byte) that denotes a record in the EF.ARR file (or the file given by the file ID).

In order to represent the ACs in all SEs for all files of **Table 15 Access Conditions of IAS Files** but the EF.DM file just the following record in EF.ARR is required:

('80' - '01' - '01') || ('90' - '00') || ('80' - '01' - '7E') || ('97' - '00')

The first access mode data object (tag '80') denotes read access. The following security condition data object (tag '90') denotes that this access is always permitted. The second access mode DO denotes write and update access (as well as file-management commands, see DELETE FILE, TERMINATE EF, ACTIVATE FILE and DEACTIVATE FILE), the following security condition DO (tag '97') denotes that this access is never permitted. In the following it is assumed that this access rule is stored in the first record of the EF.ARR file.

In the case of the file EF.DM the access rules depend on the Security Environment. Using the same abbreviations for the CRTs as in the specification of the SEs, see section 17.3, this leads to the following EF.ARR records (the numbering of the records is up to the implementation):

ARR byte (record number)	Record content (one or more access rules)	Meaning
1	('80' - '01' - '01') ('90' - '00') ('80' - '01' - '7E') ('97' - '00')	Read: always Write/Update: never File-management commands: never
2	('80' - '01' - '01') ('90' - '00') ('80' - '01' - '02') AT_1 ('80' - '01' - '7C') ('97' - '00')	Read: always Update: User verification using PIN.CH.ADMIN File-management commands: never
3	('80' - '01' - '01') AT_3 CCT_1 CT_1 CCT_2 CT_2 ('80' - '01' - '02') AT_3 CCT_1 CT_1 CCT_2 CT_2 AT_1 ('80' - '01' - '7C') ('97' - '00')	Read: device authentication (external authentication), SM (encryption and macing of command and response message) Update: device authentication (external authentication), SM (encryption and macing of command and response message), user verification using PIN.CH.ADMIN File-management commands: never

4	('80' – '01' – '01') AT_3 CCT_1 CT_1 CCT_2 CT_2 ('80' – '01' – '02') AT_3 CCT_1 CT_1 CCT_2 CT_2 ('80' – '01' – '7C') ('97' – '00')	Read: device authentication (external authentication), SM (encryption and macing of command and response message) Update: device authentication (external authentication), SM (encryption and macing of command and response message) File-management commands: never
---	---	--

Table 16 EF.ARR

These records are referenced in the expanded format of the security attributes for every file and SEID as specified in the following table:

Filename	SEID byte					
	1	2	3	4	5	6
EF.DIR	1	1	1	1	1	1
EF.SN.ICC	1	1	1	1	1	1
EF.C.ICC.AUT	1	1	1	1	1	1
EF.C.CA _{ICC} .CS-AUT	1	1	1	1	1	1
EF.C_X509.CH.DS	1	1	1	1	1	1
EF.C_X509.CA.CS	1	1	1	1	1	1
EF.DM	3	4	2	1	1 or 2	1

Table 17 ARR as function of SE and File

In the case of the file EF.DM and the SE#5 it depends on the key used for the digital signature of the data stored in the card. If the usage of the key is already protected by PIN.CH.ADMIN ARR byte 1 is used, if the key is not protected by this PIN, ARR byte 2 is used.

17.6 Key Generation and Export using SK.ICC.AUT

The signature key pair is generated by means of the GENERATE ASYMMETRIC KEY PAIR command. In order to export the public signature key PK.CH.DS a data object is generated that includes a digital signature generated by means of the device authentication key SK.ICC.AUT, as described in [E-SIGN K, Part 1], section 10.1. The problem arises to return or retrieve this data object from the ICC as the length extends – for reasonable key length - 256 bytes. Therefore two procedures are specified in this section, the first procedure stores the data object in a file and

makes use of the READ BINARY command in order to retrieve it, while the second makes use of a data object and the GET DATA command.

17.6.1 Key Generation

The following execution flow shall be used for the generation of the signature key pair SK.CH.DS/PK.CH.DS:

1. SK.ICC.AUT is selected by means of a MANAGE SECURITY ENVIRONMENT command:

<i>Byte</i>	<i>Value</i>	<i>Description</i>
CLA	Cf. [ISO/IEC 7816]	
INS	'22'	MSE
P1	'41'	SET
P2	'A4'	CRT AT
Lc	Length of data field	
Data Field	'95' – '01' – '40' '84'- L ₈₄ – 'xx' '80' – '01' – 'xx'	Usage qualifier byte (optional), key reference to SK.ICC.AUT, Algorithm ID (optional)
Le	Absent	

Table 18 Key generation execution flow 1 of 3

The value of the optional usage qualifier byte may vary.

2. SK.CH.DS is selected by means of a MANAGE SECURITY ENVIRONMENT command.

<i>Byte</i>	<i>Value</i>	<i>Description</i>
CLA	Cf. [ISO/IEC 7816]	
INS	'22'	MSE
P1	'41'	SET
P2	'B6'	CRT DST
Lc	Length of data field	
Data Field	'95' – '01' – 'C0' '84'- L ₈₄ – 'xx' '80' – '01' – 'xx'	Usage qualifier byte (optional), Key reference to SK.CH.DS, Algorithm ID (optional)
Le	Absent	

Table 19 Key generation execution flow 2 of 3

The value of the optional usage qualifier byte may vary. (Alternatively a DST specifying the public key PK.CH.DS may be used.)

3. Generate the key pair SK.CH.DS/PK.CH.DS and the data object presented in table 10-2 of document [E-SIGN K, Part 1]. Store this object using the following GENERATE ASYMMETRIC KEY PAIR command:

<i>Byte</i>	<i>Value</i>	<i>Description</i>
CLA	Cf. [ISO/IEC 7816]	
INS	'46'	GENERATE ASYMMETRIC KEY PAIR
P1	'02'	Generate new key pair
P2	'00'	No information provided
Lc	Absent	
Data Field	Absent	
Le	Absent	

Table 20 Key generation execution flow 3 of 3

This command description implies that all necessary parameters for the key generation are already known by the IAS application. If this is not the case the data field may be used to specify the necessary parameters.

17.6.2 Key Export by means of the READ BINARY command

The command READ BINARY, cf. [ISO/IEC 7816], is used in order to retrieve the data object referenced above. The file identifier is known implicitly.

17.6.3 Key Export by means of the GET DATA command

The command GET DATA, cf. [ISO/IEC 7816], is used in order to retrieve the data object presented in table 10-2 of document [E-SIGN K, Part 1] in several steps. The command is used as described in the following table. The data field is described for every step below.

<i>Byte</i>	<i>Value</i>	<i>Description</i>
CLA	Cf. [ISO/IEC 7816]	
INS	'CB'	GET DATA, odd INS code
P1	'3F'	'3FFF' indicates the current DF
P2	'FF'	
Lc		Length of data field
Data Field		Described below in this section
Le	'00'	'00'

Table 21 GET DATA command for key export

The data field of every GET DATA command contains an extended header list followed by tags that indicate the primitive data objects to be retrieved.

The following extended header list is used:

Extended Header List for Public Key Export				
'4D'	L _{4D}	T-L pair to indicate an extended header list		
	'A8'	L _{A8}	T-L pair to indicate a template for digital signature verification	
		'B6'	L _{B6}	T-L pair to indicate a DST
			'83'	L ₈₃ T-L pair to indicate a key reference to PK.CH.DS
		'7F49'	L _{7F49}	T-L pair to indicate a public key data object
			'81'	L ₈₁ T-L pair to indicate the modulus
			'82'	L ₈₂ T-L pair to indicate the public exponent
		'9E'	L _{9E}	T-L pair to indicate a digital signature

Table 22 Extended header list for public key export

Please note: The context of the digital signature data object referenced by tag '9E' is given by the MSE:SET command described in the key generation execution flow above, see section 17.6.1.

The following execution flow shall be used to retrieve the data object presented in table 10-2 of document [E-SIGN K, Part 1]:

1. The command GET DATA is used in order to retrieve the DST data object and – dependent on the key length - the first part or the complete modulus The remainder of the modulus – if present - can be read in the next step.

Therefore the GET DATA command is used with the following command data field:

GET DATA command message				
'4D'	L _{4D}	extended header list, see Table 22 Extended header list for public key export		
'B6'	'00'	T-L pair that indicates a DST data object		
		'83'	'00'	T-L pair that indicates a public key reference
'7F49'	'00'	T-L pair that indicates the public key data object		
		'81'	'00'	T-L pair that indicates the modulus (first part, dependent on the key length)

Table 23 GET DATA command message, step 1 of 3

The response data is shown in the following table:

<i>GET DATA response message</i>		
'83'	L ₈₃	Key reference PK.CH.DS
'81'	L ₈₁	Modulus (first part, dependent on the key length)

Table 24 GET DATA response message, step 1 of 3

- This step is used to read the public key remainder, i.e. the missing part of the modulus in the case that the modulus could not be retrieved completely in step 1, and the public exponent. The data field of the GET DATA command is presented in the following table:

<i>GET DATA command message</i>			
'4D'	L _{4D}	extended header list, see Table 22 Extended header list for public key export	
'7F49'	'00'	T-L pair that indicates public key data object	
	'5F38'	'00'	T-L pair that indicates the public key remainder
	'82'	'00'	T-L pair that indicates the public exponent

Table 25 GET DATA command message, step 2 of 3

If the modulus could be retrieved completely in step 1 the tag '5F38' may be absent.

The response message is presented in the following table:

<i>GET DATA response message</i>		
'5F38'	L _{5F38}	Public key remainder
'82'	L ₈₂	Public exponent

Table 26 GET DATA response message, step 2 of 3

The public key remainder data object is only present in the response message if the tag '5F38' is present in the command message.

- In the third step the digital signature data object is retrieved from the ICC. The command message is given as follows:

<i>GET DATA command message</i>		
'4D'	L _{4D}	extended header list, see Table 22 Extended header list for public key export
'9E'	'00'	T-L pair that indicates digital signature data object

Table 27 GET DATA command message, step 3 of 3

The response message is given as follows:

<i>GET DATA response message</i>		
'9E'	L _{9E}	Digital signature

Table 28 GET DATA response message, step 3 of 3

Annex 1: Features of E-Sign K Part 2

According to the answers to the questionnaire client server authentication is required. For this reason chapter 5 of Part 2 of the E-Sign K specification shall be used for the IAS application without any changes, i.e. client server authentication is an optional feature in the context of the IAS application.