CARD CENTRIC LIMITED

C2 SIM OS

Functional **Specification**

CARD CENTRIC LIMITED

CARD CENTRIC LIMITED

2 Table of Contents

3 Abbreviations and Definitions

ADF	Application Dedicated File
AID	Application identifier
AK	Anonymity key
APDU	Application protocol data unit
API	Application programming interface
CAT	Card application toolkit
DEK	Data Encryption Key
DF	Dedicated File
ENC	Encryption
EF	Elementary File
GP	Global Platform
IND	Index value used in the SQN array scheme
K	Subscriber key
KEK	Key encryption key
KIC	Key and algorithm Identifier for ciphering
KID	Key and algorithm Identifier for RC/CC/DS
MAC	Message Authentication Code
MSL	Minimum Security Level
NAA	Network access application
OP	A 128-bit Operator Variant Algorithm Configuration Field that is a component of the functions f1, f1*, f2, f3, f4, f5 and f5*
OPc	A 128-bit value derived from OP and K and used within the computation of the functions
OS	Operating system
OTA	Over the air

PIN	Personal identification number
RAM	Remote application management
RFM	Remote file management
SQN	Sequence number
SIM	Subscriber identity module
TAR	Toolkit Application Reference
USIM	Universal Subscriber Identity Module

4 Introduction

This document gives a description of C2 SIM OS- smart cards operation system. C2 OS is a component-oriented high-performance smart-card operating system targeted at various telecom and internet applications fully compatible with most international and industry standards such as ISO, Oracle, Global Platform, 3GPP and ETSI. Targeted as a firmware for subscriber identity devices and based on the widely adopted UICC standards, C2 SIM covers a broad range of telecom and IoT applications.

5 Configuration Parameters

5.1 Configuration Parameters (changeable)

The following chapter describes different parameter sets which contain parameters or flags to configure the UICC.

This chapter contains all of the parameters which can be changed during card initialization (see 7.3.1 INITIALIZE CARD) or in runtime via SET FLAG command (see 7.3.5 SET FLAG). Default values are defined on compile-time definitions. Default values may vary for different OS configuration variants (see Release Notes for particular OS configuration). Values here are most commonly used.

5.1.1 OTA initialization data

OTA initialization data (Tag '02') structure described in the following table.

Table 1. OTA Initialization data

Length	Default Value	Description
'02'	770	C-SMS buffer size
'02'	1	Count of C-SMS buffers
'02'	158	C-CBP buffer size
'02'	1024	BIP buffer size
'02'	4	Count of BIP buffers

C-SMS buffer size should be equal 135 bytes multiplied to amount of C-SMS parts plus 70 bytes for header.

For example: to support 5 C-SMS buffer should be 135 * 5 + 70= 745 bytes.

Several C-SMS buffers are used to support several parallel C-SMS streams.

C-CBP buffer size should be equal 82 bytes multiplied to amount of C-CBP plus 70 bytes for header.

BIP buffer size should be 1024 bytes. It's not recommended to change this value.

For OS variant without CAT_TP support BIP buffer size and BIP buffers count should be '00000000'.

5.1.2 Toolkit initialization data

Toolkit initialization data (Tag '03') structure described in the following table.

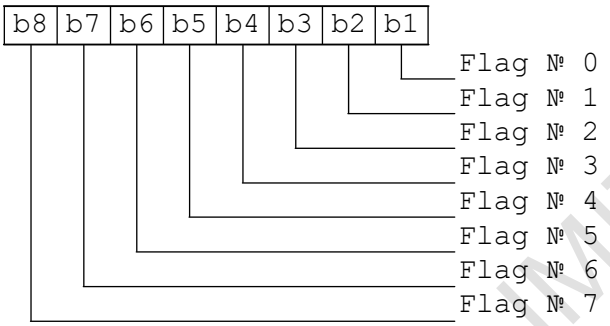
Table 2. Toolkit Initialization data

Length	Default Value	Description
'02'	255	Size of the EnvelopeHandler buffer
'02'	256	Size of the EnvelopeResponseHandler buffer
'02'	256	Size of the ProactiveHandler buffer
'02'	255	Size of the ProactiveResponseHandler buffer

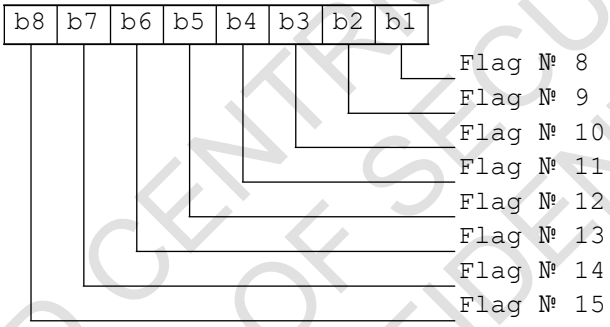
5.1.3 UICC Configuration flags

These parameters can be used to configure UICC framework behavior. UICC configuration flags array must contain default values for all flags. UICC configuration flags data (Tag '04') have the following structure:

First byte



Second byte



etc.

Table 3. UICC Configuration Flags

Byte №	Flag №	Default value	Description
1	0	false	RFU
	1	true	MULTI_CHANNEL_UICC
	2	true	SUPPORTS_CONCATENATED_SM
	3	false	FIRST_ENVELOPE_BUFFER_IN_RAM
	4	false	SECOND_ENVELOPE_BUFFER_IN_RAM
	5	false	BIG_ENVELOPE_BUFFER_IN_RAM
	6	true	OTA_9EXX_ONLY_IF_SECURITY_ERROR
	7	false	RELAXED_DELETE_CHECK
2	8	true	SUPPORTS_CONCATENATED_CB
	9	false	TAGS_IN_GET_DATA_VIA_RAM
	10	false	BIG_CB_BUFFER_IN_RAM
	11	false	BIG_BIP_BUFFER_IN_RAM
	12	false	RFM_REFRESH_MODE_FOR_MENU

	13	false	NO_0348_SMS_IN_PROACTIVE_SESSION
	14	false	DISPLAY_TOOLKIT_ISO_EXCEPTIONS
	15	false	RELAXED_UDH_CHECK
3	16	true	SEND_MORE_TIME
	17	true	DO_CONCATENATION_FOR_UNFORMATTED_SMS_CB
	18	true	DO_CLEANUP_UNUSED_FILE_VIEWS
	19	false	RELAXED_CPI_CHECK
	20	false	RFU
	21	false	RFU
	22	false	FS_EF_CACHE_DISABLED
	23	false	PERMANENT_OTA_SM_PROPS
4	24	false	STRICT_CHECK_POR_SUBMIT_BUSY
	25	false	STRICT_TAR_DUPLICATION_CHECK
	26	false	RFU
	27	false	OTA_SEC_ENH_NO_POR_SECURITY_FOR_UNTRUSTED_CP
	28	false	OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_UNTRUSTED_CP
	29	false	OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_INSECURE_CP
	30	false	OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_CNTR_PROBLEMS
	31	false	RFU
5	32	false	FLAG_OTA_EXP_OMIT_BAD_FORMATTED_TLV
	33	false	FLAG_OTA_RETURN_GET_STATUS_SW_IN_GET_RESPONSE
	34	false	FLAG_OTA_INCL_RPDU_WITH_WARNING_SW
	35	false	FLAG_OTA_EXP_ERROR_FOR_UNKNOWN_FORMAT
	36	false	RFU
	37	false	FLAG_SIM_BLOCK_DF_GSM_SELECTION_BY_7F21
	38	false	FLAG_GET_RESPONSE_EXT_9900_AVAILABLE
	39	false	FLAG_ALLOW_ZERO_LENGTH_ACCESS_DOMAIN
6	40	false	FLAG_PROACTIVE_BACKUP_BUFFER_IN_RAM
	41	false	RFU
	42	false	RFU
	43	false	RFU
	44	false	RFU
	45	true	FLAG_SUPPORTS_CONCATENATED_SM_POR
	46	true	FLAG_SUPPORTS_DEFAULT_TOOLKIT_PARAMS

MULTI_CHANNEL_UICC: flag is indicating if UICC supports more than one basic channel.

SUPPORTS_CONCATENATED_SM: flag is indicating whether the SIM supports concatenated Short Messages or not.

FIRST_ENVELOPE_BUFFER_IN_RAM: flag is indicating if the EnvelopeHandler small buffer is in RAM. If the flag is set to true the UICC Framework attempts to allocate the EnvelopeHandler buffer as a transient byte array which saves 255 bytes of EEPROM.

SECOND_ENVELOPE_BUFFER_IN_RAM: flag is indicating if the second EnvelopeHandler small buffer is in RAM. If the flag is set to true the UICC Framework attempts to allocate the EnvelopeHandler buffer as a transient byte array which saves 255 bytes of EEPROM.

BIG_ENVELOPE_BUFFER_IN_RAM: flag is indicating if the EnvelopeHandler big buffer is in RAM. If the flag is set to true the UICC Framework attempts to allocate the EnvelopeHandler buffer as a transient byte array which saves a lot of EEPROM space.

OTA_9EXX_ONLY_IF_SECURITY_ERROR: flag is indicating that the response to a Data Download shall only be issued with '9Exx' if a security error has occurred.

RELAXED_DELETE_CHECK: flag is indicating that the applet deletion condition must be relaxed as they were in JavaCard 2.1.x (static references don't prevent an applet from being deleted).

SUPPORTS_CONCATENATED_CB: flag is indicating whether the SIM supports concatenated Cell-Broadcast pages or not.

TAGS_IN_GET_DATA_VIA_RAM: flag is indicating that SIM returns TLV-structures in the GET DATA responses.

BIG_CB_BUFFER_IN_RAM: flag is indicating that the 1st big CB buffer is in RAM. If the flag is set to true the GSM Framework attempts to allocate the EnvelopeHandler buffer as a transient byte array which saves a lot of bytes of E^2.

BIG_BIP_BUFFER_IN_RAM: flag is indicating that the 1st big BIP buffer is in RAM. If the flag is set to true the GSM Framework attempts to allocate the EnvelopeHandler buffer as a transient byte array which saves a lot of bytes of E^2.

NO_0348_SMS_IN_PROACTIVE_SESSION: flag is indicating that a 03.48-formatted SMS is rejected with SIM_TOOLKIT_BUSY while a proactive session is pending. This will disable reentrant mode.

DISPLAY_TOOLKIT_ISO_EXCEPTIONS: flag is indicating whether an ISOException thrown by an applet shall be seen by the ME or not.

RELAXED_UDH_CHECK: flag is indicating whether we need to parse TP-UD instead of looking for UDHI flag.

SEND_MORE_TIME: flag is indicating that MORE TIME will be sent if needed.

DO_CONCATENATION_FOR_UNFORMATTED_SMS_CB: flag is indicating whether we need to call toolkit applet only after all part of C-CB was received

DO_CLEANUP_UNUSED_FILE_VIEWS: flag is indicating that we'll try to release some FileView objects after triggering any toolkit applet with a toolkit event.

RELAXED_CPI_CHECK: flag is indicating whether we need to wink at presence Command Packet Identifier ('7000') in subsequent Short Message's in the concatenated series.

FS_EF_CACHE_DISABLED: flag is indicating that file system cache should be disabled.

PERMANENT_OTA_SM_PROPS: flag is indicating whether we need to create OTA related properties in persistence memory.

STRICT_CHECK_POR_SUBMIT_BUSY: flag is indicating whether we need to check that if PoR via SMS SUBMIT and Proactive handler is unavailable, return SW = CAT busy (0x9300).

STRICT_TAR_DUPLICATION_CHECK: flag is indicating whether we need to check that if TAR of newly loaded STK applet, which is received by default from Instance AID (and it is not directly present in the TAR List in specific parameters) is already present in system registry.

OTA_SEC_ENH_NO_POR_SECURITY_FOR_UNTRUSTED_CP: flag is indicating whether PoR will not be secured if the Command Packet is not trusted.

OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_UNTRUSTED_CP: flag is indicating whether PoR via SMS SUBMIT will not be sent for untrusted Command Packets.

OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_INSECURE_CP: flag is indicating whether PoR via SMS SUBMIT will not be sent for Command Packets not secured with a key (ciphering, RC or CC).

OTA_SEC_ENH_NO_POR_VIA_SMS_FOR_CNTR_PROBLEMS: flag is indicating whether PoR via SMS SUBMIT will not be sent for Command Packets with problems related to counter processing.

FLAG_OTA_EXP_OMIT_BAD_FORMATTED_TLV: flag is indicating whether we need to omit the incorrectly formatted TLV in incoming OTA message in expanded format from calculation of the Number of executed C-APDUs.

FLAG_OTA_RETURN_GET_STATUS_SW_IN_GET_RESPONSE: flag indicating whether we need to return SW '6310' (related to GET STATUS) as GET RESPONSE SW in OTA session.

FLAG_OTA_INCL_RPDU_WITH_WARNING_SW: flag indicating whether we need to include R-APDUs for Case 1/Case 3 commands with warning SWs into card responses for expanded OTA packages.

FLAG_OTA_EXP_ERROR_FOR_UNKNOWN_FORMAT: flag indicating whether we need to generate R-APDUs for Case 1/Case 3 commands with warning SWs into card responses for expanded OTA packages.

FLAG_SIM_BLOCK_DF_GSM_SELECTION_BY_7F21: flag indicating whether we need to block selection of DF GSM by FID = 7F21.

FLAG_GET_RESPONSE_EXT_9900_AVAILABLE: flag indicating whether we need to use proprietary extended format (99XX) of GET RESPONSE.

FLAG_ALLOW_ZERO_LENGTH_ACCESS_DOMAIN: flag indicating whether we need to allow usage of zero-length Access Domains for backward compatibility.

FLAG_PROACTIVE_BACKUP_BUFFER_IN_RAM: flag indicating whether we need to backup in RAM a current proactive command during sending of implicit PoR via SMS SUBMIT.

FLAG_SUPPORTS_CONCATENATED_SM_POR: Flag indicating whether the SIM supports concatenated PoR via SMS-Submit or not.

FLAG_SUPPORTS_DEFAULT_TOOLKIT_PARAMS: Flag indicating whether the SIM allows the installation of Toolkit applets without toolkit parameters (with default one) or not. In case of this flag is set installing toolkit applet without toolkit parameters will not lead to installation error. Default toolkit parameters will apply. Default toolkit parameters are:

Access domain: 0xFF (No access to the File System)
Priority level: 0xFF
Number of timers: 0x00
Maximum text length for a menu entry: 0x20
Number of menu entries: 0x00
Number of channels: 0x00
Length of MSL: 0x02
MSL type: 0x01 (Minimum SPI1)
Minimum SPI: 0x16

RFU flags shall be set to '0'.

5.1.4 GP Configuration Flags

These parameters can be used to configure Global Platform framework behavior. GP configuration flags array must contain default values for all flags. GP configuration flags data (Tag 'C5') have the following structure:

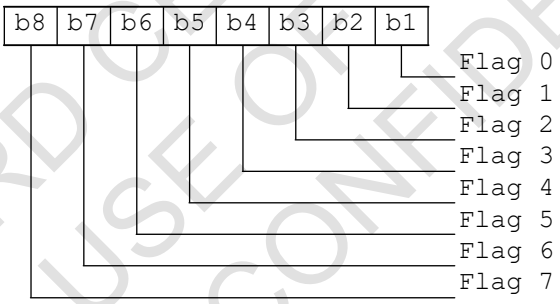


Table 4. GP Configuration Flags

Byte №	Flag №	Default value	Description
1	0	false	FLAG_RELAXED_C9_ORDER_CHECK
	1	false	FLAG_ALLOW_MULT_OBJ_DELETION
	2	false	FLAG_STRONG_CNTR_CHECK
	3	false	FLAG_STRONG_PCNTR_CHECK
	4	false	FLAG_CHECK_KIC_KID_VERSION
	5	false	FLAG_USE_SAME_KEYSET_FOR_KIC_KID
	6	false	FLAG_SEC_ENH_NO_SINGLE_DES
	7	true	FLAG_ALLOW_NO_SEC_IN_OP_READY

2	8	false	FLAG_ALLOW_NON_ELF_IN_GET_STATUS
	9	false	FLAG_REMOVE_CONCAT_FROM_GET_STATUS
	10	false	FLAG_HIDE_MASK_PACKAGES
	11	false	FLAG_ALLOW_PROACTIVE_COMMANDS
	12	false	FLAG_SEC_ALLOW_ANY_SPI2_B4B3
	13	false	FLAG_SEC_ALLOW_ANY_SPI2_B5
	14	false	FLAG_ALLOW_SD_PERSO_FOR_ALL_TAGS
	15	false	FLAG_HANDLE_NO_STRUCTURE_DATA_AS_BER_IN_STORE_DATA
3	16	false	FLAG_USE_BLOBS_WITH_PROTECTION_WHERE_POSSIBLE
	17	false	FLAG_DO_NOT_REGISTER_CVM_GLOBAL_PIN_FOR_ISD
	18	false	FLAG_UICC_STACK_OBJECTS_SUPPORT

FLAG_RELAXED_C9_ORDER_CHECK: flag indicating whether we need to check that tag C9 (Application Specific Parameters) is before tag EA (UICC System Specific Parameters constructed field) or tag CA (SIM file access and toolkit application specific parameters field) in INSTALL FOR INSTALL command.

FLAG_ALLOW_MULT_OBJ_DELETION: flag which indicates that the OS can delete multiple objects (packages) in one command.

FLAG_STRONG_CNTR_CHECK: flag indicating whether we need to check that CNTR in the Command Packet is zero (if CNTR check isn't determined in the SPI).

FLAG_STRONG_PCNTR_CHECK: flag indicating whether we need to check that PCNTR in the Command Packet is zero (if ciphering check isn't determined in the SPI).

FLAG_CHECK_KIC_KID_VERSION: flag indicating whether checking of version of KIC and KID is activated. According appendix A.1 of ETSI 102 225: keyset versions must be identical in KIC and KID or it must be zero (if e.g. KIC or KID is not used), otherwise the “unidentified security problem” code must be returned.

FLAG_USE_SAME_KEYSET_FOR_KIC_KID: flag which indicates that the OS must use the same keyset version for KIC and KID when one of them is zero and both are to be used. If SPI indicates that both KIC and KID are to be used, the actual keyset version may be stored just in one of them (e.g. in KIC), and in the other one (e.g. in KID) the keyset version nibble will be zero. In this case the OS must use the same keyset version for both.

FLAG_SEC_ENH_NO_SINGLE_DES: flag indicating whether attempt to use single DES in CP and/or PoR for ciphering and/or CC will lead to the '0A' error.

FLAG_ALLOW_NO_SEC_IN_OP_READY: flag indicating whether we allow GP commands which require Secure Channel Session without it in OP_READY.

FLAG_ALLOW_NON_ELF_IN_GET_STATUS: flag indicating whether we allow to return non-ELF entries in GET STATUS.

FLAG_REMOVE_CONCAT_FROM_GET_STATUS: flag indicating that concatenation shall be removed from GET STATUS.

FLAG_HIDE_MASK_PACKAGES: flag indicating whether we forbid displaying Mask packages.

FLAG_ALLOW_PROACTIVE_COMMANDS: flag indicating whether we allow proactive commands on GP APDUs.

FLAG_SEC_ALLOW_ANY_SPI2_B4B3: flag indicating whether we need to skip checking of the next requirements: ETSI 102.225 Rel 12: If SPI2.b4b3 is not set to 00, then the RC or CC requested for the Response Packet, i.e. SPI2.b4b3, shall be the same as the Command Packet, i.e. SPI1.b2b1.

FLAG_SEC_ALLOW_ANY_SPI2_B5: flag indicating whether we need to skip checking of the next requirement: ETSI 102.225 Rel 12: Ciphering of the Response Packet, i.e. SPI2.b5 set to '1', is only allowed if the Command Packet has been successfully authenticated (e.g. using CC) and if ciphering was applied to the Command Packet, i.e. SPI1.b3 is set to '1'.

FLAG_ALLOW_SD_PERSO_FOR_ALL_TAGS: allow SD personalization with all tags.

FLAG_HANDLE_NO_STRUCTURE_DATA_AS_BER_IN_STORE_DATA: allow SD personalization with 'No general data structure information' in P1.

FLAG_USE_BLOBS_WITH_PROTECTION_WHERE_POSSIBLE: create SecBlob objects with protection where possible.

FLAG_DO_NOT_REGISTER_CVM_GLOBAL_PIN_FOR_ISD: do not register CVM Global PIN service in GP Services Registry for ISD (if not set - register it).

FLAG_UICC_STACK_OBJECTS_SUPPORT: flag indicating whether we support allocation on stack.

5.1.5 Simple Journal File Storage Configuration

SJFS configuration parameters (Tag 'FE') define configuration of Simple Journal File Storage for M2M solutions. M2M extension should be enabled on compile time.

Table 5. Simple Journal File System configuration

Length	Default Value	Description
'03'	'14'	number of supported records
	'03'	size of Main Storage in sectors (e.g. 03 == 12K)
	'01'	size of TMP Storage in sectors (e.g. 01 == 4K)

Records here are the high updatable Transparent EFs and records in high updatable Line Fixed and Cyclic EFs.

TMP Storage size should be equal to size of all high updatable EFs.

Main Storage size should be equal at least TMP Storage size multiplied to 2.

5.2 Configuration files

The following chapter describes different proprietary DFs and EFs which contain parameters to configure the UICC. These files are creating in process of MNO profile loading. Description here is for information mostly.

5.2.1 Configuration Files under MF

This proprietary DF may be present as child directory of MF. The following DF is defined:

- DF_{AUTH} ('7FCC') contains proprietary files used for handling of the current authentication options.

5.2.2 Configuration Files under DF_{AUTH}

The EFs in the Dedicated File DF_{AUTH} contain authentication related information.

5.2.2.1 EF_{AUTH} (Authentication Algorithms)

This EF contains the codes of currently used authentication algorithms in 2G/3G modes.

Identifier: '6F00'	Structure: transparent	Optional
File size: 2 bytes	Update activity: Low	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		

Bytes	Description	M/O	Length
1	Identifier of algorithm for 2G	M	1 byte
2	Identifier of algorithm for 3G	M	1 byte

Accepted values are enumerated in Table 6.

Table 6. Id values for authentication algorithms

Algo	ID
Milenage	'01'
CAVE	'02'

COMP128v1	'03'
COMP128v2	'04'
COMP128v3	'05'
TUAK	'06'
XOR 3G	'07'
XOR 2G	'08'
CIS-B	'09'
EAP-AKA	'0A'

5.2.2.2 Default Values

If this file isn't present on the card the default values are:

- '01' (Milenage) for 2G mode;
- '01' (Milenage) for 3G mode.

5.2.3 Configuration Files under DF GSM

The EFs under the Dedicated File DF_{GSM} contain network related information.

5.2.3.1 EF_{NAP} (Network Authentication Parameters)

This EF contains LV stuctures with value of 128-bit Operator Variant Algorithm Configuration (OP) or 128-bit value derived from OP and K (OPc) and Milenage Constants C1 – C5 , R1 – R5. These values used within the computation of the functions f1, f1*, f2, f3, f4, f5 and f5*.

Table 7. EFNAP structure in case of Milenage

Identifier: '00F2'	Structure: transparent		Mandatory
File size: 20 + X + Y bytes		Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED YES TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length

1	OPx block length (=17)	M	1
2	OPx block: OP usage flag	M	1
3 – 18	OPx block: OP or OPc value	M	16
19	Ci block length (X=0/80)	M	1
20 – 99	Constants C1-C5	O	80
100	Ri block length (Y=0/5)	M	1
101 – 105	Constants R1-R5	0	5
NOTE: This file is mandatory if and only if the selected algorithm in 2G or 3G mode is 'Milenage'.			

Coding: OP usage flag coded as follows

- 0x00 – bytes 2 - 17 contains OP value;
- 0x01 - bytes 2 - 17 contains OPc value;
- all other value - RFU (will lead to 3G AKA failure).

If C1 –C5 and R1 – R5 values aren't present on the card the default values are:

C1	'00000000000000000000000000000000'
C2	'00000000000000000000000000000001'
C3	'00000000000000000000000000000002'
C4	'00000000000000000000000000000004'
C5	'00000000000000000000000000000008'

R1	'40'
R2	'00'
R3	'20'
R4	'40'
R5	'60'

Table 8. EFNAP structure in case of Tuak

Identifier: '00F2'	Structure: transparent	Mandatory
File size: 34 bytes	Update activity: Low	

Access Conditions:		Special Attributes:	
READ	NEVER	SECURED	YES
UPDATE	ADM	TRANSIENT	NO
DEACTIVATE	ADM	TRANSACTED	NO
ACTIVATE	ADM		

Bytes	Description	M/O	Length
1	TOPx block length (=33)	M	1
2	TOPx block: TOP usage flag	M	1
3 – 34	TOPx block: TOP or TOPc value	M	32
NOTE: This file is mandatory for USIM/ISIM FS if the selected algorithm is 'TUAK'.			

Coding: TOP usage flag coded as follows

- '00' – bytes 3 - 34 contains TOP value;
- '01' – bytes 3 - 34 contains TOPc value;
- all other value - RFU (will lead to 3G AKA failure).

5.2.3.2 EF_{NAKS} (Network Authentication Key Sizes)

This EF contains definitions for networks key sizes.

This EF have not be linked with the same EF under any ADF.

Identifier: '00F4'	Structure: transparent		Mandatory
File size: 6 bytes		Update activity: Low	
Access Conditions:		Special Attributes:	
READ	NEVER	SECURED	YES
UPDATE	ADM	TRANSIENT	NO
DEACTIVATE	ADM	TRANSACTED	NO
ACTIVATE	ADM	LINKABLE	NO

Bytes	Description	M/O	Length
1	Length of K	M	1 byte
2	Length of RES	M	1 byte
3	Length of CK	M	1 byte
4	Length of IK	M	1 byte

5	Length of KECCAK	C (see NOTE1)	1 byte
6	Length of MAC	M	1 byte
NOTE1: This byte is meaningful for TUAK algo only.			

Possible values for Milenage 2G (hex):

Length of K	'10'
Length of RES	'04'
Length of CK	'00'
Length of IK	'00'
Length of KECCAK	'FF'
Length of MAC	'08'

Possible values for TUAK 2G(hex):

Length of K	'10'
Length of RES	'04'
Length of CK	'00'
Length of IK	'00'
Length of KECCAK	TBD
Length of MAC	'08'

5.2.3.3 EF_{AC} (Authentication counter)

This EF contains the value of authentication counter.

Identifier: '00FE'	Structure: transparent	Mandatory
File size: 4 bytes		Update activity: High
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO

Bytes	Description	M/O	Length
1 - 4	Authentication counter	M	4 bytes

The range of valid values: '00000000' - 'FFFFFFFE'.

- '00000000' - the counter is locked - authentication is locked.
- 'FFFFFFFE' - maximum value - 4'294'967'294 authentication attempts is available.
- 'FFFFFFFF' - authentication counter mechanism is turned off.

5.2.3.4 EF_{KI} (Subscriber Key)

This EF contains the value of a subscriber key (KI).

Identifier: '00FF'	Structure: transparent		Mandatory
File size: 16 bytes		Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED YES TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1 - 16	Subscriber key	M	16 bytes

5.2.4 Configuration Files under USIM ADF

The EFs in the USIM ADF contain service and network related information.

5.2.4.1 EF_{NAP} (Network Authentication Parameters)

This EF contains LV stuctures with value of 128-bit Operator Variant Algorithm Configuration (OP) or 128-bit value derived from OP and K (OPc) and Milenage Constants C1 – C5 , R1 – R5. These values used within the computation of the functions f1, f1*, f2, f3, f4, f5 and f5*.

Table 9. EFNAP structure in case of Milenage

Identifier: '00F2'	Structure: transparent	Mandatory
File size: 20 + X + Y bytes	Update activity: Low	

Access Conditions:		Special Attributes:	
READ	NEVER	SECURED	YES
UPDATE	ADM	TRANSIENT	NO
DEACTIVATE	ADM	TRANSACTED	NO
ACTIVATE	ADM		

Bytes	Description	M/O	Length
1	OPx block length (=17)	M	1
2	OPx block: OP usage flag	M	1
3 – 18	OPx block: OP or OPc value	M	16
19	Ci block length (X=0/80)	M	1
20 – 99	Constants C1-C5	O	80
100	Ri block length (Y=0/5)	M	1
101 – 105	Constants R1-R5	0	5
NOTE: This file is mandatory if and only if the selected algorithm in 2G or 3G mode is 'Milenage'.			

Coding: OP usage flag coded as follows

- '00' – bytes 2 - 17 contains OP value;
- '01' – bytes 2 - 17 contains OPc value;
- all other value - RFU (will lead to 3G AKA failure).

If C1 –C5 and R1 – R5 values aren't present on the card the default values are:

C1	'00000000000000000000000000000000'
C2	'00000000000000000000000000000001'
C3	'00000000000000000000000000000002'
C4	'00000000000000000000000000000004'
C5	'00000000000000000000000000000008'

R1	'40'
R2	'00'
R3	'20'
R4	'40'
R5	'60'

Table 10. EFNAP structure in case of Tuak

Identifier: '00F2'	Structure: transparent		Mandatory
File size: 34 bytes		Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED YES TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1	TOPx block length (=33)	M	1
2	TOPx block: TOP usage flag	M	1
3 – 34	TOPx block: TOP or TOPc value	M	32
NOTE: This file is mandatory for USIM/ISIM FS if the selected algorithm is 'TUAK'.			

Coding: TOP usage flag coded as follows

- '00' – bytes 3 - 34 contains TOP value;;
- '01' - bytes 3 - 34 contains TOPc value;
- all other value - RFU (will lead to 3G AKA failure).

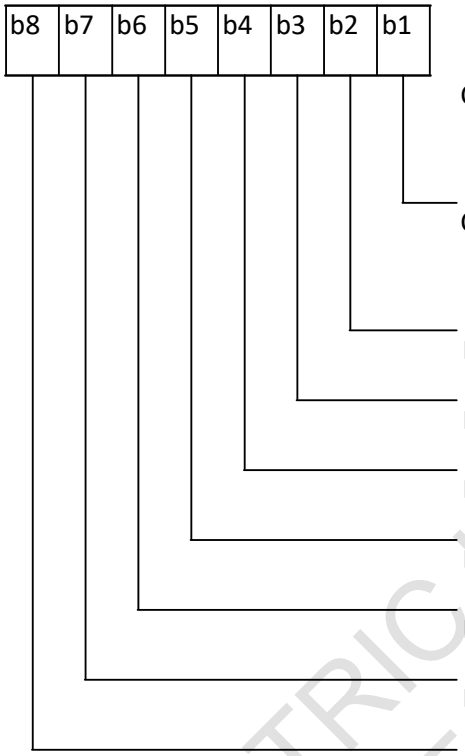
5.2.4.2 EF NAEF (Network Authentication Expanded Flags)

This file contains a set of different algo-specific flags.

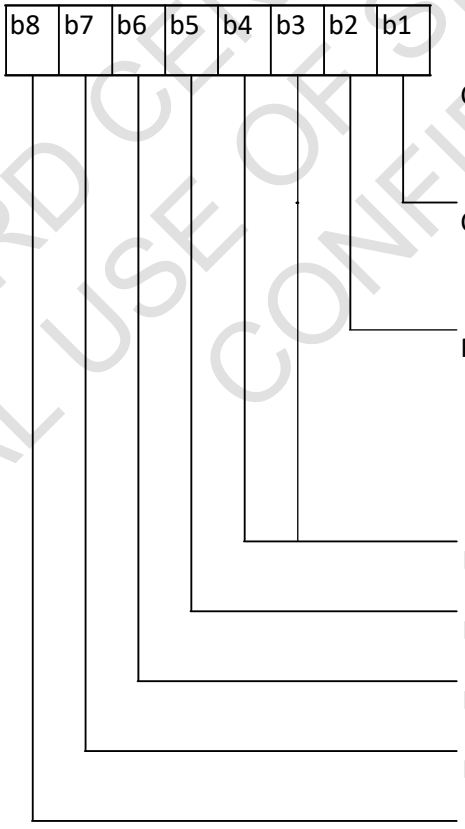
Identifier: '00F3'	Structure: transparent		Conditional (see Note)
File size: 1 byte		Update activity: Low	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1	Flags	M	1 byte
NOTE: This file is mandatory for USIM/ISIM FS.			

Flags: this byte is coded as below:

- For Milenage



- For TUAK



Here, AK is an anonymity key used to conceal the sequence number as the latter may expose the identity and location of the user. The concealment of the sequence number is to protect against passive attacks only.

See **Ошибка! Источник ссылки не найден.** for additional information.

5.2.4.3 EF_{NAKS} (Network Authentication Key Sizes)

This EF contains definitions for networks key sizes.

This file have not be linked.

Identifier: '00F4'	**Structure: transparent**	**Mandatory**
File size: 6 bytes	Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM	Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	

Bytes	Description	M/O	Length
1	Length of K	M	1 byte
2	Length of RES	M	1 byte
3	Length of CK	M	1 byte
4	Length of IK	M	1 byte
5	Number of iterations for Keccak	C (see NOTE1)	1 byte
6	Length of MAC	M	1 byte
NOTE1: This byte is meaningful for TUAK algo only.			

Possible values for Milenage (hex):

Length of K	'10'
Length of RES	'08'
Length of CK	'10'
Length of IK	'10'
Number of iterations for Keccak	'FF'
Length of MAC	'08'

Possible values for TUAK (hex):

Length of K	'10'/'20'

Length of RES	'08'/'10'/'20'
Length of CK	
Length of IK	'10'/'20'
Number of iterations for Keccak	N>=1
Length of MAC	'08'/'10'/'20'

5.2.4.4 EF_{SNAP} (Secondary Network Authentication Parameters)

This EF contains the same LV-structures as EF_{NAP} (Network Authentication Parameters). This file should be placed at the same level as the main set of parameters (EF_{NAP}), but will be used only with the Secondary Algorithm in case it's configured.

Identifier: '00F5'	Structure: transparent	Mandatory
File size: Z bytes	Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM	Special Attributes: SECURED YES TRANSIENT NO TRANSACTED NO	

File content is absolutely the same as in EF_{NAP}. So, please refer to the section 5.2.4.1 for more details about file content.

5.2.4.5 EF_{LAUTH} (Local Authentication Algorithm)

This EF contains the codes of currently used authentication algorithms in 3G mode.

Identifier: '00F8'	Structure: transparent	Optional
File size: 1 byte	Update activity: Low	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM	Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	

Bytes	Description	M/O	Length
1	Indicator of algorithm for 3G	M	1 byte

NOTE 1: If this file is present on the card its value overrides value from the basic EFAUTH (Authentication Algorithms).
NOTE 2: If this file isn't present on card then basic EFAUTH (Authentication Algorithms) is used.

Coding: accepted values are enumerated in Table 6.

5.2.4.6 EF_{SQNC} (SQN Configuration)

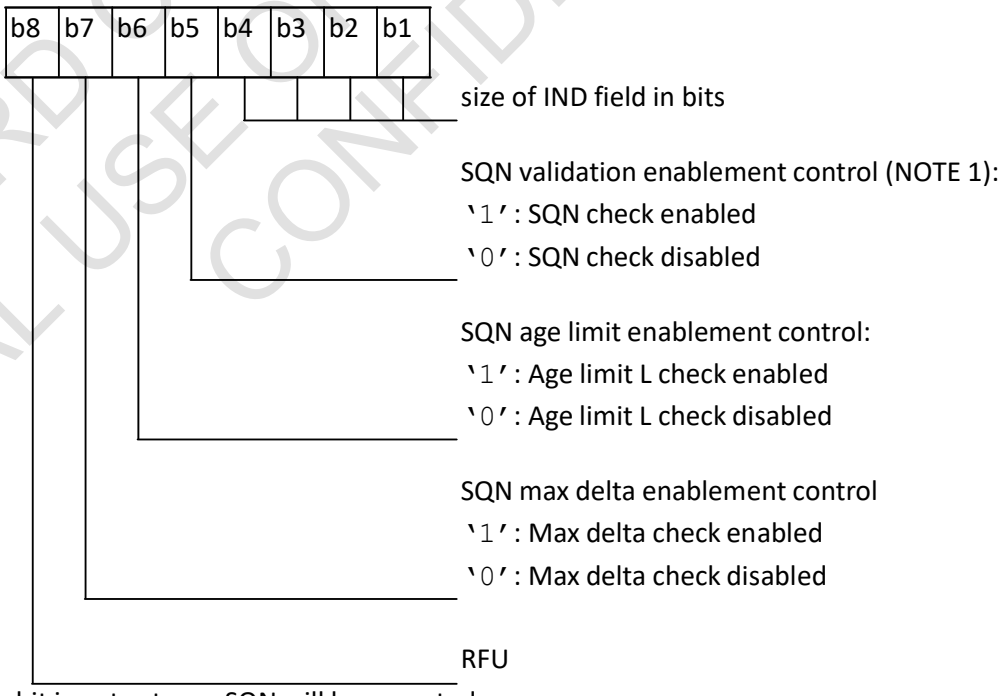
This file contains basic parameters of SQN processing.

C2 OS supports management of sequence numbers which are not time-based. Please refer Annex C 3.2

Identifier: '00FB'	Structure: transparent		Conditional (see Note)
File size: 15 bytes		Update activity: High	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	

Bytes	Description	M/O	Length
1	Flags	M	1 byte
2-3	SQNms offset	M	2 bytes
4-9	Max delta	M	6 bytes
10-15	Age limit	M	6 bytes
NOTE: This file is mandatory if and only if the selected algorithm in 3G mode is 'Milenage'.			

Flags: this byte is coded as below:



NOTE 1: If this bit is not set, any SQN will be accepted.

```
SQNms offset: these bytes are offset to SQNms inside First byte value (SQN
flags): 0x75
bit 8: RFU
bit 7: Max delta control (enabled)
bit 6: Age limit control (enabled)
bit 5: SQN validation control (enabled)
bit 4: ---|
bit 3: ---|
bit 2: ---| - size of IND field in bits
bit 1: ---|
(MAX_DELTA << IND_shift) = ('000010000000'<<5) = '000200000000'.
(AGE_LIMIT << IND_shift) = ('0000010000000'<<5) = '000200000000'.
File content: '75 0000 000200000000 000200000000'.
```

EF_{SQNA} (SQN Array).

Max delta: coded as hex value. Max delta must be stored in file with the IND-part set to zeros.

Age limit: coded as hex value. Age limit must be stored in file with the IND-part set to zeros.

Default coding scheme for EF_{SQNC}

The Age Limit and Max Delta values must be pre-scaled to match the SEQ number as it is embedded in an SQN: they must be shifted to the left by the amount of bits allocated for the IND part.

For the initial values:

```
Max delta control – enabled;
Age limit control – enabled;
SQN validation control – enabled;
IND=5;
Max delta = 0x10000000; (2^28)
Age limit (L) = 0x10000000; (example value; value for Age limit shall be
provided by Operator, if not Age limit control should be disabled)
```

EF_{SQNC} content:

```
First byte value (SQN flags): 0x75
bit 8: RFU
bit 7: Max delta control (enabled)
bit 6: Age limit control (enabled)
bit 5: SQN validation control (enabled)
bit 4: ---|
bit 3: ---|
bit 2: ---| - size of IND field in bits
bit 1: ---|
(MAX_DELTA << IND_shift) = ('000010000000'<<5) = '000200000000'.
(AGE_LIMIT << IND_shift) = ('0000010000000'<<5) = '000200000000'.
File content: '75 0000 000200000000 000200000000'.
```

5.2.4.7 EF_{SQNA} (SQN Array)

This file contains a list of SQN.

Identifier: '00FA'	Structure: transparent		Conditional (see Note1)
File size: 6 * (2^IND) bytes (see Note 2)		Update activity: High	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1-6	SQN (SEQ at index=0)	M	6 bytes
…			
(6n - 5) – 6n	SQN (SEQ at index=n)	M	6 bytes
NOTE 1: This file is mandatory if and only if the selected algorithm in 3G mode is 'Milenage'. NOTE 2: For IND = 5 file size is 192 bytes.			

See **Ошибка! Источник ссылки не найден.** for additional information.

5.2.4.8 EF_{AC} (Authentication counter)

This EF contains the value of authentication counter.

Identifier: '00FE'	Structure: transparent		Mandatory
File size: 4 bytes		Update activity: High	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1 - 4	Authentication counter	M	4 bytes

The range of valid values: '00000000' - 'FFFFFFFE'.

- '00000000' – the counter is locked – authentication is locked;
- 'FFFFFFFE' – maximum value – 4'294'967'294 authentication attempts is available;
- 'FFFFFFFF' - authentications counter mechanism is turned off.

5.2.4.9 EF_K (Subscriber Key)

This EF contains the value of subscriber key (K).

Identifier: '00FF'	Structure: transparent		Mandatory
File size: 16 bytes		Update activity: Low	
Access Conditions: READ NEVER UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED YES TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1 - 16	Subscriber key	M	16 bytes

5.2.5 Configuration Files under ISIM ADF

The EFs in the ISIM ADF contain service and network related information. ISIM application must contain files described in 5.2.4, if required.

5.2.5.1 EF_{LAUTH} (Local Authentication Algorithm)

This EF contains the codes of currently used authentication algorithms in 4G (LTE) mode.

Identifier: '00F8'	Structure: transparent		Optional
File size: 1 byte		Update activity: Low	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1	Indicator of algorithm for 4G	M	1 byte
NOTE 1: If this file is present on the card its value overrides value from the basic EFAUTH (Authentication Algorithms).			

Coding: accepted values are enumerated in Table 6.

5.2.5.2 $EF_{KS\_INT\_NAF}$ (Ks_int_NAF)

This file contains Ks_int_NAF keys stored in GBA security context (NAF Derivation Mode) authentication (see **Ошибка! Источник ссылки не найден.**).

Identifier: '00F0'	Structure: linear fixed	Conditional (see Note 1)

Record length: 32 bytes Records count: see Note 2		Update activity: High	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1-32	Ks_int_NAF key	M	32 bytes
NOTE 1: This file is mandatory if and only if GBA authentication is required. NOTE 2: Records count should be equal to records count in EF 6FD7 (GBA NAF List) under ADF ISIM.			

5.2.6 Configuration Files under EAP-AKA ADF

The EFs in the EAP-AKA ADF contain service and network related information. EAP-AKA application must contain files described in 5.2.4, if required.

5.2.6.1 EF_{LAUTH} (Local Authentication Algorithm)

This EF contains the codes of currently used authentication algorithms in 3G mode.

Identifier: '00F8'	**Structure: transparent**		**Mandatory**
File size: 1 byte		Update activity: Low	
Access Conditions: READ ADM UPDATE ADM DEACTIVATE ADM ACTIVATE ADM		Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO	
Bytes	Description	M/O	Length
1	Indicator of algorithm for 3G	M	1 byte
NOTE 1: If this file is present on the card its value overrides value from the basic EFAUTH (Authentication Algorithms).			

Coding: accepted values are enumerated in Table 6.

5.2.6.2 Contents of Files at the DF_{EAP} Level

The EFs in the Dedicated File DF_{EAP} contain authentication related information.

5.2.6.2.1 EF EAPKEYS (EAP Derived Keys)

This EF contains the key material derived after a successful EAP authentication (see **Ошибка! Источник ссылки не найден.**).

Identifier: '4F01'	Structure: transparent	Mandatory

File size: >=132 bytes	Update activity: High
Access Conditions: READ PIN UPDATE ADM/NEVER DEACTIVATE ADM ACTIVATE ADM	Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO

Bytes	Description	M/O	Length
1	Master Session Key (MSK) Tag = '80'	M	1 byte
2	Master Session Key (MSK) Length = '40'	M	1 byte
3 - 66	Master Session Key (MSK)	M	64 bytes
67	Extended Master Session Key (EMSK) Tag = '81'	M	1 byte
68	Extended Master Session Key (EMSK) Length = '40'	M	1 byte
69 - 132	Extended Master Session Key (EMSK)	M	64 bytes

5.2.6.2.2 EF EAPSTATUS (EAP Authentication Status)

This EF contains the authentication status corresponding to the EAP client supported by the application.

Identifier: '4F02'	Structure: transparent	Mandatory

File size: 1 byte	Update activity: High
Access Conditions: READ PIN UPDATE ADM/NEVER DEACTIVATE ADM ACTIVATE ADM	Special Attributes: SECURED NO TRANSIENT NO TRANSACTED NO

Bytes	Description	M/O	Length
1	Authentication Status: '00': No authentication started '01': Authenticating '02': Authenticated '03': Held (Authentication failure)	M	1 byte

5.2.6.3 Contents of EAP-AKA Related Data Objects in EF_{DIR}

DF_{EAP} FID is defined in EF_{DIR} (see **Ошибка! Источник ссылки не найден.**). Example of EAP-AKA related Data Objects to be added after Application template TLV in EF_{DIR}:

Table 11. Example of Coding of EAP related DOs in EFDIR

Length	Description	Value
1	Discretionary template tag	'73'
1	Length of the discretionary template	'12'
1	EAP Application service specific data content tag	'A0'
1	EAP Application service specific data content length	'10'
1	Application EAP supported types list tag	'80'
1	Length of the Application EAP supported types list	'01'
1	Application EAP supported types list (EAP-AKA)	'17'
1	Application EAP Dedicated file list tag	'81'
1	Length of Application EAP Dedicated file list	'02'
2	Application EAP Dedicated File list	'5F40'
1	Application EAP Label tag	'82'
1	Length of the Application EAP Label	'07'
7	Application EAP Label = "EAP AKA"	'45415020414B41'

6 System Applets

System applets are applets providing different services for both the configuration of the UICC and remote file and applet management. The packages of this applets are located in ROM. Therefore it is possible to install these applets at any time of the card life cycle. System applets list is a compile time (not changeable) configuration parameter. System applet's packages can't be deleted via Card Manager commands. System applets are:

- SIM application
- USIM application (optionally)
- ISIM application (optionally)
- OTA RAM/RFM applications
- CAT-TP application (optionally)
- TLS application (optionally)
- BIP application (optionally)
- HTTP Admin application (optionally)

6.1 System Applets List

The table below contains a list of pre-loaded applications and their AIDs.

Table 12. Default AIDs of pre-loaded applications

Application	PID/AID
SIM application	PID: 'A0000000090001FFFFFFFF8900' AID: 'A0000000090001FFFFFFFF8900000000'
USIM/ISIM application	PID: 'A0000000871002FF49FFFF8900' AID: 'A0000000871002FF49FFFF89040B0000'
TLS application	PID: 'FF434E5258010107060000001000000FF' AID: 'FF434E525801010706000000010000000'
BIP application	PID: 'FF434E5258010107070000001000000FF' AID: 'FF434E525801010707000000010000000'
HTTP Admin application	PID: 'FF434E525801010704000000100000FF' AID: 'FF434E52580101070400000010000000'

6.2 SIM Application

This application must be installed as a default selected application.

Table 13. SIM Application install parameters

Parameter	Value

Applet Type	JavaCard Applet
Package ID	'A00000000090001FFFFFFFF8900'
Applet ID	'A00000000090001FFFFFFFF8900000000'
Instance ID	'A00000000090001FFFFFFFF8900000000'
Application Privileges	Card Terminate '04'
Applet Specific Parameters	None
Toolkit Applet Specific Parameters	None

6.3 USIM Application

This application should be installed to provide 3G functionality. This application are installing in process of MNO profile loading. Description here is for information mostly.

Table 14. USIM Application install parameters

Parameter	Value
Applet Type	JavaCard Applet
Package ID	'A00000000871002FF49FFFF8900'
Applet ID	'A00000000871002FF49FFFF89040B0000'
Instance ID	'A00000000871002FF49FFFF89040B00FF'
Applet Specific Parameters	None
Application Privileges	None
Toolkit Applet Specific Parameters	None

6.4 ISIM Application

This application should be installed to provide LTE functionality. This application are installing in process of MNO profile loading. Description here is for information mostly.

Parameter	Value
Applet Type	JavaCard Applet
Package ID	'A00000000871002FF49FFFF8900'
Applet ID	'A00000000871002FF49FFFF89040B0000'
Instance ID	'A00000000871004FFFFFFFF8903020000'
Applet Specific Parameters	None
Application Privileges	None
Toolkit Applet Specific Parameters	None

6.5 TLS application

TLS application is responsible for processing of TLS-level packets.

Table 15. TLS application install parameters

Parameter	Value
Applet Type	CAT Applet

Package ID	'FF434E525801010706000000100000FF'
Applet ID	'FF434E5258010107060000001000000'
Instance ID	'FF434E5258010107060000001760000'
Application Privileges	None
Applet Specific Parameters	'03010420' Coding see Table 16 below
UICC System Specific Parameters	None

TLS application specific parameters contain TLS buffer size and PSK structures coded as LV structure. Max TLS fragment size have to be equal to max size of TLS fragment.

Table 16. TLS application specific parameters

Parameter	Size	Description
TLS version	2 bytes	'0303' - TLS 1.2 (default value) '0302' - TLS 1.1 '0301' - TLS 1.0
Max TLS fragment size	1 byte	'00' - 16k (default value) '04' - 4k '03' - 2k '02' - 1k '01' - 0.5k
Max PSK ID length	1 byte	'40' by default

6.6 BIP application

BIP application is responsible for processing of BIP-level packets.

Table 17. BIP application install parameters

Parameter	Value
Applet Type	CAT Applet
Package ID	'FF434E525801010707000000100000FF'
Applet ID	'FF434E5258010107070000001000000'
Instance ID	'FF434E5258010107070000001770000'
Application Privileges	None
Applet Specific Parameters	None
UICC System Specific Parameters	None

6.7 HTTP Admin application

HTTP Admin is responsible for processing of HTTP packets.

Table 18. HTTP Admin application install parameters

Parameter	Value
Applet Type	CAT Applet
Package ID	'FF434E525801010704000000100000FF'
Applet ID	'FF434E5258010107040000001000000'
Instance ID	'FF434E5258010107040000001740000'

Application Privileges	Global Service '000100'
Applet Specific Parameters	'01000005DC09C4' Coding see Table 19 below
SIM File Access and Toolkit Application Specific Parameters	
Access Domain Parameter	No access to the File System
Priority	'FF'
Timers	2 see NOTE1
Maximum text length for a menu entry	'20'
Menu entries	'00'
Max number of channels	'02' see NOTE1
MSL	'00' see NOTE2
TAR List	None
NOTE1: Maximum number of timers must be greater or equal maximum number of HTTP channels. NOTE2: If SCP81 protocol is activated for ISD MSL have to be set to not NULL and one OTA key set have to be personalsed.	

Table 19. HTTP Admin application specific parameters

Parameter	Size	Description
Channels number	1 byte	Number of HTTP channels processed in parallel.
Max URL length supported by the App	2 bytes	'0000' - use default, 128 bytes by default
Input buffer size	2 bytes	If 0 - default size. Default size is 1024 bytes '05DC' - 1500 bytes
Output buffer size	2 bytes	If 0 - default size. Default size is 641 bytes '09C4' - 2500 bytes

6.8 Supplementary Security Domains

Security Domains support security services such as key handling, encryption, decryption, digital signature generation and verification for their owner's applications.

Table 20. Security Domain install parameters

Parameter	Value
Applet Type	Java Applet
Package ID	'A0000000035350'
Applet ID	'A000000003535041'
Instance ID	'A000000003535041'
Applet Specific Parameters	'45' This parameter means Security Domain can accept extradition
Application Privileges	'80' – Security Domain in addition it's possible to set Application Privileges DAP Verification and Mandated DAP Verification

7 Description of Commands

7.1 Generic Commands

7.1.1 SELECT

Command described in /ETSI102221/ . This command selects a file. After a successful selection the record pointer and the current tag pointer are undefined.

Input:

- file ID, application ID, path or empty.

Output:

- if the selected file is the MF, a DF or an ADF:
  - file ID, total file size, PIN status and other application specific data;
- if the selected file is an EF:
  - file ID, file size, total file size, access conditions, invalidated/not invalidated indicator, structure of EF , length of the records in case of linear fixed structure or cyclic structure and reserved and maximum file size in case of BER-TLV structure.

As described in /ETSI102221/ the with some additional TLVs in the “Proprietary Information” TLV.

7.1.1.1 Proprietary information

This is a constructed TLV object.

Byte(s)	Description	Length
1	Tag = 'A5'	1
2	Length	1
3-(2+X)	Proprietary data, constructed	X

The /ETSI102221/ specification describes the standard TLV objects for the proprietary template (tag 'A5'). This specification describes additional private TLV objects for the proprietary template.

7.1.1.2 Local PIN Map

This TLV contains a PIN map for the current context. Internally each FS context maintains its own local PIN map gathered from the current navigation context. This PIN map takes up 13 bytes and contains only local PIN instances. PIN IDs are determined implicitly from the array element index (0 - 7 are mapped to 0x81 - 0x88, local PINs, and 8 - 12 are mapped to local ADMs). In FCP we just return this local PIN map array.

Byte(s)	Description	Length
1	Tag = 'CB'	1
2	Length	1
3-15	Local PIN map data	13

NOTE 1: The design glitch is that in CREATE FILE command the same TLV uses a bit different format: pairs of PIN ID, PIN instance.

This TLV is available only for DF.

7.1.1.3 Free Memory

This TLV contains size of free data space. This is the size of the largest free NVRAM memory block in our realization.

Byte(s)	Description	Length
1	Tag = '83'	1
2	Length	1
3-4	Free memory size	2

This TLV is available only for DF.

7.1.1.4 Link Count

This TLV describes the maximum number of links to this EF and the current number of links.

Byte(s)	Description	Length
1	Tag = 'CD'	1
2	Length	1
3	Maximum number of links	1
4	Current number of links	1

This TLV is available only for linkable EFs.

7.1.1.5 Target EF

This TLV contains a path to the target EF. If it is present in the proprietary TLV object of an EF, the EF was created as a link to the provided target EF.

Byte(s)	Description	Length
1	Tag = '9B'	1
2	Length	1
3-(3+X)	Link count	X

This TLV is available only for linked EF.

7.1.1.6 Special Info

This TLV contains a special info for current EF.

Byte(s)	Description	Length
1	Tag = 'C0'	1
2	Length	1
3	Special info	1

This TLV is available only for EF.

Special info coded as a bitmask. See coding scheme below:

Meaning	b8	b7	b6	b5	b4	b3	b2	b1
Not readable or updatable when deactivated	0	0	0	0	0	0	0	0
Readable and updatable when deactivated	0	1	0	0	0	0	0	0
RFU	Any other value							

7.1.2 STATUS

This function returns information concerning the current directory or current application.

In addition, according to the application specification, it may be used to indicate to the application in the UICC that its session activation procedure has been successfully executed or that its termination procedure will be executed.

Input:

- none.

Output:

- one of the following:
  - FCP of the current directory;
  - the DF name TLV Data Object of the currently selected application;
  - no data returned.

In case STATUS command returns FCP of the current directory some proprietary TLVs are included into the response. See 7.1.1.1 for details.

7.2 Administrative Commands

7.2.1 CREATE FILE

This command allows the creation of a new file under the current directory. The access condition for the CREATE FILE function of the current directory shall be fulfilled.

When creating an EF with linear fixed or cyclic structure, the UICC shall directly create as many records as allowed by the requested file size.

After the creation of a DF, the current directory shall be on the newly created file. In case of an EF creation, the current EF shall be on the newly created file and the current directory is unchanged. After creation of an EF with linear fixed structure, the record pointer is not defined. After creation of an EF with cyclic structure, the current record pointer is on the last created record. After creation of an EF with BER TLV structure, the current tag pointer is undefined.

The memory space allocated shall be reserved for the created file.

If an ADF is created, some instance should take care of the administration of the application, e.g. updating the EF DIR with the application ID. The CREATE FILE command does not take care of this administration by its own. The DF Name tag shall only be provided in the command if an ADF is created.

The CREATE FILE command shall initialize newly created EFs with 'FF'. The content of the whole newly created EF shall consist of bytes of this value. If any other default values are required for another application, this default behavior can be overwritten by specifying an appropriate TLV in the application dependent data TLV (tag '85' or 'A5') of the CREATE FILE command.

CREATE FILE command can be sent via OTA.

7.2.1.1 Command message

The Command message is coded according to the following table.

Table 21. CREATE FILE command message

Code	Value
CLA	'00', '01', '02', or '03'
INS	'E0'
P1	'00'
P2	'00'
Lc	Length of the subsequent data field
Data field	Data sent to the UICC
Le	Not present

P1 and P2 are set to '00' indicating: File ID and file parameters encoded in data.

7.2.1.1.1 Creating a DF/ADF

If a DF/ADF is to be created the content data field codded as follows:

Value	M/O/C	Description	Length
'62'	M	Tag: FCP Template	1 byte
LL		Length of FCP Template (next byte to the end)	1 or 2 bytes
'82'	M	Tag: File Descriptor File Descriptor byte followed by Data Coding Byte	1 byte
'02'		Length of File Descriptor	1 byte
		File Descriptor Byte indicating DF or ADF	1 byte
'21'		Data Coding Byte	1 byte
'83'	M	Tag: File ID	1 byte
'02'		Length of File ID	1 byte
		File ID	2 bytes
'84'	C (see note)	Tag: DF Name (AID)	1 byte
LL		Length of DF Name (AID)	1 byte
		DF Name (AID)	1 byte to 16 bytes
'8A'	M	Life Cycle Status Information (LCSI)	1 byte
'01'		Length of the LCSI	1 byte

		Life Cycle Status Information	1 byte
'8C' 'AB' '8B'	M	Tag: Security Attributes: one of the following: Compact Expanded Referenced	1 byte
LL		Length of Security Attributes	1 byte
		Data for the Security Attributes	W bytes
'81'	M	Tag: Total File Size	1 byte
LL, LL ≥ 2		Length of Total File Size	1 byte
		Total File Size	X bytes
'C6'	M	Tag: PIN Status Template DO	1 byte
LL		Length of PIN Status Template DO	1 byte
		PIN Status Template DO	Y bytes
'85' or 'A5'	O	Tag: Proprietary, application dependent	1 byte
LL		Length of application dependent data	1 byte
		Application dependent data (see 0)	Z bytes

Tag '84' shall only be present for an ADF, otherwise it is not present.

Tag '8C', Tag 'AB' or Tag '8B': Security Attributes

Exactly one of the tags shall be present.

At least the key references that are used to allow access during the operational phase of the UICC shall be supplied in the Security Attributes.

Tag '81': Total File Size

Amount of physical memory allocated for the DF or ADF. The amount of memory specifies how much memory will be available within the currently created DF or ADF to create EFs or other DFs. It shall include the memory needed for structural information for these EFs and DFs. The size of the structural information for the created DF/ADF shall not be included.

Some card implementations support dynamic allocation of memory (memory is allocated for the whole UICC), and therefore will ignore this TLV object.

By specifying a value other than '0000', it is possible to indicate the requested amount of physical memory for the content of a DF or an ADF. This amount is taken from the memory allocated for the current DF.

The behavior of the UICC for a value equal to '0000' is for a further study.

Tag '82': File Descriptor with Data Coding Byte

The File Descriptor Byte shall be coded as defined in the following table.

Table 22. Structure of the file descriptor

Meaning	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
File accessibility	0	X	-	-	-	-	-	-
Not shareable file	0	0	-	-	-	-	-	-
Shareable file	0	1	-	-	-	-	-	-
File type	0	-	X	X	X	-	-	-
Working EF	0	-	0	0	0	-	-	-
Internal EF	0	-	0	0	1	-	-	-
RFU	0	-	0	1	0	-	-	-
RFU	0	-	0	1	1	-	-	-
RFU	0	-	1	0	0	-	-	-
RFU	0	-	1	0	1	-	-	-
RFU	0	-	1	1	0	-	-	-
DF or ADF	0	-	1	1	1	-	-	-
EF structure	0	-	-	-	-	X	X	X
No information given	0	-	-	-	-	0	0	0
Transparent	0	-	-	-	-	0	0	1
Linear fixed	0	-	-	-	-	0	1	0
RFU	0	-	-	-	-	0	1	1
RFU	0	-	-	-	-	1	0	0
RFU	0	-	-	-	-	1	0	1
Cyclic	0	-	-	-	-	1	1	0
RFU	1	X	X	X	X	X	X	X

The data coding byte can be used differently according to table 86 in / ISO7816_4/. For the present document, value '21' (proprietary) shall be used and shall not be interpreted by the UICC.

Tag '84': DF Name

The DF Name is a string of bytes which is used uniquely to identify an application dedicated file (ADF) in the card.

Tag '8A': Life Cycle Status Information (LCSI)

The Life Cycle Status Information shall be coded according to Table 23.

This TLV specifies the status of the file after creation.

The initialization state can be used to set the file into a specific security environment for administrative purposes. See ACTIVATE FILE command.

Table 23. Coding of the LCSI

Meaning	Bit7..4	Bit3	Bit2	Bit1	Bit0
No information given	0	0	0	0	0
Creation state	0	0	0	0	1
Initialization state	0	0	0	1	1

Operational state - activated	0	0	1	-	1
Operational state - deactivated	0	0	1	-	0
Termination state	0	1	1	-	-
Proprietary	≠ 0	X	X	X	X

Tag 'C6': PIN Status Template DO

The PIN Status Template DO shall be coded according to /ETSI UICC/.

7.2.1.1.2 Creating an EF

If a EF is to be created the content data field codded as follows:

Value	M/O/C	Description	Length
'62'	M	Tag: FCP Template	1 byte
LL		Length of FCP Template (next byte to the end)	1 byte or 2 bytes
'82'	M	Tag: File Descriptor File Descriptor Byte followed by Data Coding Byte or File Descriptor Byte followed by Data Coding Byte and record length, coded on 2 bytes	1 byte
'02' or '04'		Length of File Descriptor	1 byte
		File Descriptor Byte	1 byte
'21'		Data Coding Byte	1 byte
	C (see note 1)	Record length	2 bytes
'83'	M	Tag: File ID	1 byte
'02'		Length of File ID	1 byte
		File ID	2 bytes
'8A'	M	Life Cycle Status Information (LCSI)	1 byte
'01'		Length of the LCSI	1 byte
		Life Cycle Status Information	1 byte
'8C' 'AB' '8B'	M	Tag: Security Attributes: one of the following: Compact Expanded Referenced	1 byte
LL		Length of Security Attributes related data	1 byte
		Data for the Security Attributes	X bytes
'80'	M	Tag: File Size (Reserved File Size)	1 byte
LL		Length of File Size	1 byte
		File Size	Y bytes
'88'	O	Tag: Short File Identifier	1 byte
'00' or '01'		Length of Short File Identifier	1 byte
		Short File Identifier	0 or 1 byte
'A5' or '85'	C (see note 2)	Tag proprietary, application dependent	1 byte

LL		Length of application dependent data	1 byte or 2 bytes
		Application dependent data (see below for tag 'A5')	Z bytes

NOTE 1: Mandatory for linear fixed and cyclic files, otherwise it is not applicable.
NOTE 2: Tag 'A5' is mandatory for BER TLV structured EFs, otherwise it is optional.

Tag '80': File Size (Reserved File Size)

File Size indicates the number of bytes allocated for the body of the file (i.e. it does not include structural information) and cannot be allocated by any other entity.

In case of an EF with linear or cyclic structure, it is the record length multiplied by the number of records of the EF.

In case of a BER-TLV structured EF, the File Size indicates the number of bytes allocated for the body of the file. The value shall include administrative overhead (if any) required to store TLV objects, but not the structural information for the file itself. This value shall be returned in the FCP information provided in a response to a SELECT APDU command and be labeled as "Reserved File Size".

Tag '82': File Descriptor

The File Descriptor Byte shall be coded as defined in Table 22.

The data coding byte can be used differently according to table 86 in / ISO7816_4/. For the present document, the value '21' (proprietary) shall be used and shall not be interpreted by the UICC.

The record length shall be present if a record structured file (i.e. for linear fixed or cyclic files) is created. In this case it indicates the length of the records coded on 2 bytes. Most significant byte comes first in the value field.

Tag '8A': Life Cycle Status Information (LCSI)

The Life Cycle Status Information shall be coded as defined Table 23.

This TLV specifies the status of the file after creation.

The initialization state can be used to set the file into a specific security environment for administrative purposes. See ACTIVATE FILE command.

Tag '88': Short File Identifier

The following 3 cases shall be supported by the UICC if the ATR indicates that the UICC supports selection by SFI:

- Tag '88' is missing in the CREATE FILE command: The lower five bits of the file ID are used as the Short File Identifier by the EF;

- Tag '88' is available in the CREATE FILE command, there is no value part in the TLV: Short File Identifier is not supported by the EF;
- Tag '88' is available in the CREATE FILE command, there is a Short File Identifier value in the TLV: The Short File Identifier is coded from bits b8 to b4. Bits b3,b2,b1 = 000.

Tag 'A5': Proprietary, application dependent

This is a constructed TLV object.

There's a set of custom TLVs recognized in Create File command implementation (see /ETSI UICC/).

The following TLV objects are defined for the proprietary template (tag 'A5').

Table 24. Proprietary TLV data field coding

Value	M/O/C	Description	Length
'C0'	O	Tag: Special File Information (File Status Byte)	1 byte
'01'		Length of Special File Information	1 byte
		Special File Information (File Status Byte), see	1 byte
'CA'	O	Tag: Flags	1 byte
'01'		Length of Flags	1 byte
		Flags TLV. Defines various proprietary flags affecting the EFs	1 byte
'CB'	C	Tag: Local PIN Map	1 byte
2*n		Length of Local PIN Map	1 byte
		Local PIN Map TLV can be included for a DF, ADF or the MF, and defines how local PINs are used. n pairs of bytes, each defines a mapping of a local PIN reference to a corresponding local PIN instance number. Example: CB 04 81 01 8D 02 Defines two local PIN mappings: 2APIN1 is mapped to its instance #1, local ADM9 is mapped to its instance #2.	2*n bytes
'CC'	C	Tag: SIM Characteristics	1 byte
'01'		Length of SIM Characteristics	1 byte
		SIM Characteristics TLV can only be included for the MF. It defines SIM characteristics, which will later be returned in 2G FCI. Value: SIM characteristics byte	1 byte
'80'	C	Tag: UICC Characteristics	1 byte
'01'		Length of UICC Characteristics	1 byte
		UICC Characteristics TLV can only be included for the MF. It defines UICC characteristics, which will later be returned in 3G FCP. Value: UICC characteristics byte	1 byte
'81'	C	Tag: Application Power	1 byte
'03'		Length of Application Power	1 byte
		Application Power TLV can only be included for an ADF. It defines application power requirements, which will later be returned in 3G FCP. Value: application power requirements	3 bytes

'82'	C	Tag: Minimum Application Frequency	1 byte
'01'		Length of Minimum Application Frequency	1 byte
		Minimum Application Frequency TLV can only be included for an ADF. It defines minimum application frequency requirement, which will later be returned in 3G FCP. Value: minimum application frequency requirements	1 byte
'9B'	C	Target EF	1 byte
2*n		Length of Target EF	1 byte
		Target EF TLV can only be included for an EF. It defines the target EF path and forces creation of a link instead of creating the vanilla EF. Value: path to the target EF	2*n bytes
'CD'	C	Link count	1 byte
'01'		Length of 'Link Count' TLV	1 byte
		Link Count TLV can only be included for a linkable EF. It defines the max number of links to this EF.	2*n bytes

The following table defines various proprietary flags affecting the EFs.

Table 25. Structure of flags

Description of flags	Bit7	Bit6	Bit5	Bit4	Bit3	Bit2	Bit1	Bit0
OTA accessible (if not set - the file will not be accessible over-the-air, e.g. via RFM)	X	-	-	-	-	-	-	-
Transaction secured, only for EFs (if set - updates to file's data will be guarded by JavaCard transaction mechanism)	-	X	-	-	-	-	-	-
Transient, only for EFs (if set - file's data will be allocated in clear-on-reset RAM)	-	-	X	-	-	-	-	-
Checksum secured, only for EFs (if set - file's data will be protected by a checksum, which is calculated during updates and verified during reads)	-	-	-	X	-	-	-	-
File is inconsistent (internal flag, must not be set from outside world)	-	-	-	-	X	-	-	-
Linkable, only for EFs (if not set - the file will not allow any links to itself, e.g. it will be not shareable)	-	-	-	-	-	X	-	-
Skip initialization, only for EFs (if set - the file's data will not be explicitly filled with 0xFF bytes during file creation, which can considerably speed up personalization process if the file is to be fully updated anyway later)	-	-	-	-	-	-	X	-
Increasable, only for Cyclic EFs (if not set - there will be no way to run the INCREASE command on this file in 2G mode; 3G INCREASE ignores this flag)	-	-	-	-	-	-	-	X

Tag 'C0': Special File Information

Tag 'C0': Special File Information (File Status Byte) within the proprietary TLV (tag 'A5') is coded as described in the following table.

Meaning	b8	b7	b6	b5	b4	b3	b2	b1
Not readable or updatable when deactivated	0	0	0	0	0	0	0	0
Readable and updatable when deactivated	0	1	0	0	0	0	0	0
RFU	Any other value							

7.2.1.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by UICC.

Table 26. CREATE FILE status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'67'	'XX'	Incorrect length field, XX provides expected length
'69'	'81'	Command incompatible with file structure
'69'	'82'	Security status not satisfied
'69'	'85'	Conditions of use not satisfied
'6A'	'80'	Incorrect parameters in the data field
'6A'	'82'	File not found
'6A'	'84'	Not enough memory space
'6A'	'89'	File ID already exists
'6B'	'00'	Incorrect parameter P1 or P2
'6D'	'00'	Instruction code not supported or invalid
'6E'	'00'	Class not supported
'6F'	'00'	Technical problem with no diagnostic given

7.2.2 DELETE FILE

This command initiates the deletion of a referenced EF immediately under the current DF, or a DF with its complete subtree.

If a file is indicated as not-shareable as defined in Table 22 and is the current file of one application, then another application cannot delete it.

If a file is indicated as shareable as defined in Table 22, then it can be deleted by one application regardless of whether or not the file is the current file of any other application.

NOTE 1: If any other application concurrently uses the deleted file, the processing by the application may fail.

NOTE 2: If a DF is shareable and an application having the appropriate rights requests to delete it, the whole DF including all EFs can be deleted whatever shareable status they have.

If an EF is to be deleted, the access condition "DELETE FILE" of the EF to be deleted shall be fulfilled. After successful completion the current directory is unchanged and no EF is selected. If the EF that is successfully deleted is the current file of another application, the current directory of this application is unchanged and no EF is selected.

If a DF is to be deleted, the access condition "DELETE FILE (self)" of the DF to be deleted shall be fulfilled. After successful completion the parent directory is selected and no EF is selected.

If an ADF is to be deleted, the access condition "DELETE FILE (self)" of the ADF to be deleted shall be fulfilled and the ADF is not currently selected on another logical channel. After successful completion the MF is selected and no EF is selected.

The access conditions "DELETE FILE" and "DELETE FILE (self)" shall be coded as specified in ISO7816. The access condition "DELETE FILE (child)" shall not be used.

After successful completion of this command, the deleted file can no longer be selected. The resources held by the file shall be released and the memory used by this file shall be set to the logical erased state. It shall not be possible to interrupt this process in such a way that the data can become recoverable.

If an ADF is deleted, some instance has to take care of the administration of the application, e.g. deleting the application ID entry in the EF DIR. The DELETE FILE command does not take care of this administration by its own.

DELETE FILE command can be sent via OTA.

7.2.2.1 Command message

The **Command APDU** is coded according to the following table:

Code	Value
CLA	'00', '01', '02', or '03' (according to /ISO7816_4/, clause 5.4.1)
INS	'E4'
P1	'00'
P2	'00'
Lc	Length of the subsequent data field
Data field	Data sent to the UICC
Le	Not present

P1 and P2 are set to '00', indicating the selection by file identifier as defined in /ISO7816_4/ for SELECT FILE command.

Data field sent in the command message is coded according following table

Value	M/O/C	Description	Length
	M	File ID	2 bytes

7.2.2.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 27. DELETE FILE status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'67'	'XX'	Incorrect length field, XX provides expected length
'69'	'82'	Security status not satisfied
'6A'	'82'	File not found
'6A'	'87'	Incorrect length of file ID
'6B'	'00'	Incorrect parameter P1 or P2
'6D'	'00'	Instruction code not supported or invalid
'6E'	'00'	Class not supported
'6F'	'00'	Technical problem with no diagnostic given

7.2.3 TERMINATE DF

The TERMINATE DF command initiates the irreversible transition of the currently selected DF/ADF into the termination state (see LCSI coding in Table 23).

Following a successful completion of the command, the DF/ADF is in terminated state and the functionality is available from the DF/ADF and its subtree is reduced. The DF/ADF shall be selectable and if selected, the warning status SW1/SW2='6285' (selected file in termination state) shall be returned.

Further possible actions are not defined.

The intention of DF/ADF termination is to make the application unusable by the cardholder.

The command can be performed only if the security status satisfies the Security Attributes defined for this command.

An appropriate security rule is to be setup and fulfilled in order to execute this command.

If a DF is indicated as not-shareable as defined in Table 22 and is the current DF of one application, then another application cannot terminate it. If a DF is indicated as shareable as defined in Table 22, then it can be terminated by an application regardless of whether or not the DF is the current file of any other application.

If another application concurrently uses the terminated DF, the processing by this application may fail.

7.2.3.1 Command message

The TERMINATE DF command message is coded according to the following table.

Table 28. TERMINATE DF command message

Code	Value
CLA	'00', '01', '02', or '03' (according to / ISO7816_4/, clause 5.4.1)
INS	'E6'
P1	'00'
P2	'00'
Lc	Not present
Data field	Not present
Le	Not present

P1 and P2 are set to '00'.

The data field of the command message is not present.

7.2.3.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 29. TERMINATE DF status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'62'	'85'	Selected file terminated
'67'	'XX'	Incorrect length field, XX provides expected length
'69'	'81'	Command incompatible with file structure
'69'	'82'	Security status not satisfied
'69'	'84'	Referenced data invalid
'69'	'85'	Conditions of use not satisfied - not-shareable file selected by another application
'6B'	'00'	Incorrect parameter P1 or P2
'6D'	'00'	Instruction code not supported or invalid
'6E'	'00'	Class not supported
'6F'	'00'	Technical problem with no diagnostic given

7.2.4 TERMINATE EF

The TERMINATE EF command initiates the irreversible transition of the currently selected EF to the termination state (see LCSI coding in Table 23).

The command can be performed only if the security status satisfies the Security Attributes defined for this command.

If an EF is indicated as not-shareable as defined in Table 22 and is the current EF of one application, then another application cannot terminate it. If an EF is indicated as shareable as defined in Table 22, then it can be terminated by an application independently of whether or not the EF is the current file of any other application.

If any other application concurrently uses the terminated EF, the processing by this application may fail.

7.2.4.1 Command message

The TERMINATE EF command message is coded according to the following table.

Table 30. TERMINATE EF command message

Code	Value
CLA	'00', '01', '02', or '03' (according to ISO7816_4, clause 5.4.1)
INS	'E8'
P1	'00'
P2	'00'
Lc	Not present
Data field	Not present
Le	Not present

P1 and P2 are set to '00'.

The data field of the command message is not present.

7.2.4.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 31. TERMINATE EF status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'62'	'85'	Selected file terminated
'67'	'XX'	Incorrect length field, XX provides expected length
'69'	'81'	Command incompatible with file structure
'69'	'82'	Security status not satisfied
'69'	'84'	Referenced data invalid
'69'	'85'	Conditions of use not satisfied - not-shareable file selected by another application
'69'	'86'	Command not allowed (no EF selected)

'6B'	'00'	Incorrect parameter P1 or P2
'6D'	'00'	Instruction code not supported or invalid
'6E'	'00'	Class not supported

7.2.5 RESIZE FILE

This command allows modifying of the memory space allocated for a transparent file or a linear fixed file under the current directory (MF, DF/ADF). This command shall not be allowed for a cyclic file.

MF or DF/ADF resizing is not allowed, the error status SW1/SW2=’6981' (command incompatible with file structure) shall be returned.

The access condition for the RESIZE FILE command shall be fulfilled for the file to be resized.

The RESIZE FILE access condition is indicated in the access rules using AM_DO tag '84'. Tag '84' indicates that the INS code for the RESIZE FILE command is indicated as the value in the TLV object (instruction code 'D4'). The RESIZE FILE command can only be used on files that refer to an access rule where this INS code is indicated as part of the rule.

In case of a successful execution of the command, the current file or directory on which the command was applied is selected. If the RESIZE FILE command was performed on a linear fixed file the record pointer shall be undefined.

After an unsuccessful execution of the command, the current selected file and directory shall remain the same as prior to the execution. If in this case the RESIZE FILE command was performed on a linear fixed file the record pointer shall not be changed.

After a successful execution of the command, the Total File Size, if applicable, and the File Size TLVs defined in the FCP template of the modified file shall be updated accordingly.

The allocated memory space is updated according to the new data size.

For a linear fixed file, the RESIZE FILE command modifies the number of records but doesn't change the record length.

In case the size of a linear fixed or transparent EF is increased:

- the extension data shall be appended to the end of the existing data (e.g. if 3 records are added to a linear fixed EF with 2 records, these 2 records remain the record 1 and 2 in the increased file, and the 3 new records will become the records 3, 4 and 5); and
- the data contained in the previously allocated memory space shall not be modified by the RESIZE FILE command (e.g. if 3 records are added to a linear fixed EF with 2 records, the content of the 2 initially allocated records shall be identical before and after the execution of the RESIZE FILE command); and
- the newly allocated memory space shall be initialized with 'FF'. The content of the whole newly allocated memory space shall consist of bytes of this value.

In case the size of a linear fixed or transparent EF is decreased:

- the removed data shall be deleted and removed from the end of the existing data (e.g. if 3 bytes are removed from a transparent EF with 5 bytes, the bytes 3, 4 and 5 shall be removed from the file); and
- the remaining data already contained in the previously allocated memory space shall not be modified by the RESIZE FILE command (e.g. if 3 bytes are removed from a transparent EF with 5 bytes, the content of the 2 remaining bytes shall be identical before and after the execution of the RESIZE FILE command).

RESIZE FILE command can be sent via OTA.

7.2.5.1 Command message

The RESIZE FILE command message is coded according to the following table.

Table 32. RESIZE FILE command message

Code	Value
CLA	'80', '81', '82', or '83' (according to /ISO7816_4/, clause 5.4.1)
INS	'D4'
P1	'00'
P2	'00'
Lc	Length of the subsequent data field
Data field	Data sent to the UICC
Le	Not present

P1 and P2 are set to '00'.

7.2.5.2 Data field sent in the command message

Table 33. Coding of the data field of the RESIZE FILE command

Value	Length		M/O/C	Description
'62'	1 byte		M	Tag: FCP Template
LL	1 or 2 bytes			Length of FCP Template (next byte to the end)
'83'	1 byte		M	Tag: File ID
'02'	1 byte			Length of File ID
	2 bytes			File ID
'80'	1 byte		M	Tag: File Size (Reserved File Size)
LL	1 byte			Length of the File Size
	X bytes			File Size (New File Size)

There is at least one occurrence of the following Tags.

Tag '83': File ID

Contains the FID of the EF to be resized.

Tag '80': File Size (Reserved File Size)

This TLV shall only be provided if an EF is resized. It contains the New File Size for this EF.

This size is the new number of bytes allocated for the body of the EF (i.e. it does not include structural information).

In the case of an EF with linear fixed structure, the File Size shall be the record length multiplied by the number of EF records; otherwise the command is rejected. The New File Size shall contain at least one record.

For transparent files, if this size is set to '00', all the content of the EF is removed but the EF is not deleted (it is then exactly as if the EF was created with a size set to '00') and the structural information is still available.

7.2.5.3 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 34. RESIZE FILE status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'67'	'XX'	Incorrect length field, XX provides expected length
'69'	'82'	Security status not satisfied
'69'	'84'	Referenced data invalid
'69'	'85'	Conditions of use not satisfied
'6A'	'82'	File not found
'6B'	'00'	Incorrect parameter P1 or P2
'6D'	'00'	Instruction code not supported or invalid
'6E'	'00'	Class not supported
'6F'	'00'	Technical problem with no diagnostic given

7.3 Proprietary Commands

7.3.1 INITIALIZE CARD

This command initializes internal OS structures and sets card to CLEAR_CARD state. In this state card is ready to create profile (create file system, install applets, etc).

7.3.1.1 Command message

The command message is coded according to the following table.

Table 35. INITIALIZE CARD command message

Code	Value
CLA	'80'
INS	'00'
P1	'01'
P2	'00'
Lc	Length of data field
Data	Not present. Card will be initialized with default parameters.
Le	Not present.

It is possible to change default initialization parameters. In such case data field will contain TLV coded groups of parameters. Each TLV are optional. See the following tables for details. Default values of initialization parameters can be found in Release Notes for every operation system configuration.

The data field of the command message is coded according to the following table.

Table 36. INITIALIZE CARD command message data field

Tag	Length	Presence	Value and Description
'02'	'0A'	O	OTA initialization data (see 5.1.1 OTA initialization data)
'03'	'08'	O	Toolkit initialization data (see 5.1.2 Toolkit initialization data)
'04'	'06'	O	UICC Configuration flags (as array) (see 5.1.3 UICC Configuration flags)
'C5'	'01'	O	GP Configuration flags (see 5.1.4 GP Configuration Flags)
'FE'	'03'	O	SJFS Configuration (see 5.1.5 Simple Journal File Storage Configuration)
'06'	'00'	M	Multiple MNO profile zones supported
'E4'	'00'-'7F' or '8180'–'81FF'	M	ISD Configuration parameters
'C1'	'04'–'0A'	M	Active protocols and amount of entities '0204'- SCP02, 4 entities '0304' – SCP03, 4 entities 'F301'- SCP03t, 1 entity '8002'- SCP80, 2 entities (Mandatory for M2M case) '8102'- SCP81, 2 entities (Mandatory for M2M case)
'81'	'02'	M	Secure Channel Protocol Identifier and Implementation Option "i". Several records possible, one for each protocol. '0255'– SCP02 '0270'– SCP03 '8000'– SCP80 '8104'– SCP81 'F300'– SCP03t
'EF'	'00'-'7F'	M	System Specific Parameters for ISD initialization. Coding see Table 37 below

'FD'	'00'-'7F'	C	RFU

Table 37. System Specific Parameters for ISD initialization

Tag	Length	Value	Description
'EF'			System Specific Parameters for ISD initialization
'C8'	'02'	'0000'	Non volatile data space limit
'C7'	'02'	'0000'	Volatile data space limit
'CA'	'0B'–'0D'		SIM File Access and Toolkit Application Specific Parameters
		'01FF'	No access to the File System
		'FF'	Priority level of the Toolkit application instance
		'01'	Maximum number of timers allowed for this application instance
		'20'	Maximum text length for a menu entry
		'01'	Maximum number of menu entries allowed for this application Instance.
		'0000'	Position and Identifier of the first menu entry ('00' means do not care)
		'00'	Maximum number of channels for this application instance
		'02010A'	Minimum Security Level (MSL)
		'00'	Length of TAR Value(s) field

7.3.1.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 38. INITIALIZE CARD status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
		All other values.

7.3.2 VIRGINIZE CARD

This command erases all data from card except OS and set card to PRE_INIT mode.

7.3.2.1 Command message

The command message is coded according to the following table.

Table 39. VIRGINIZE CARD command message

Code	Value
CLA	'D0'
INS	'00'

P1	'02'
P2	'00'
Lc	'00'
Data	Not present.
Le	Not present.

The data field of the command message is not present.

7.3.2.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 40. VIRGINIZE CARD status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
		All other values.

7.3.3 CHANGE CARD ATR

This command is optional for 3G and LTE configurations. But it's highly recommended to change UICC ATR (3G or "long" ATR) to SIM ATR (2G ATR or "short" ATR).

CHANGE ATR command changes ATR bytes, full scope or partially.

7.3.3.1 Command message

The command message is coded according to the following table.

Code	Value
CLA	'80'
INS	'02'
P1	'01' - change bytes in ATR specific part '02' - change Historical bytes only '03' - change whole ATR '04' - restore default ATR '05' - set offset for ATR informing bytes
P2	'00'
Lc	the actual length of specific part of ATR which will be merged with Historical bytes '00' for P1= '04' '02' for P1= '05'
Data field	Data sent to the UICC (optional)
Le	Not present

7.3.3.1.1 Data field sent in the command message

P1	Data Field

'01'	ATR specific bytes. The T0 byte will be updated for containing the length of Historical bytes, LRC byte will be recalculated and updated
'02'	Historical bytes. The T0 byte will be updated for containing the length of Historical bytes, LRC byte will be recalculated and updated
'03'	Whole ATR including LRC byte
'04'	No data available
'05'	1st byte - offset from the beginning of historical ATR bytes for informing; 2nd byte - flags indicating what information should be displayed in ATR bytes (according to requested data it could be 1 o 2 bytes)

7.3.3.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 41. CHANGE ATR status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
'6B'	'00'	Wrong P1 or P2
'67'	'00'	Wrong length
'6A'	'80'	P1= '03' only: Invalid command data, *LRC* verifying was failed

7.3.4 CLEAR FLAG

This command clears the configuration flag specified in P2.

CLEAR FLAG command shall be sent to RAM applet via OTA.

The command message is coded according to the following table.

7.3.4.1 Command message

Code	Value
CLA	'80'
INS	'00'
P1	'00' – UICC configuration flag group '01' – GP configuration flag group
P2	Flag number as specified in 5.1.3 and 5.1.4
Lc	Not present
Data	Not present
Le	Not present

The data field of the command message is not present.

7.3.4.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 42. CLEAR FLAG status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
'6B'	00	P1 or P2 are invalid

7.3.5 SET FLAG

This command sets the configuration flag specified in P2.

SET FLAG command shall be sent to RAM applet via OTA.

The command message is coded according to the following table.

7.3.5.1 Command message

Code	Value
CLA	'80'
INS	'01'
P1	'00' – UICC configuration flag group '01' – GP configuration flag group
P2	Flag number as specified in 5.1.3 and 5.1.4
Lc	'00'
Data	Not present
Le	Not present

The data field of the command message is not present.

7.3.5.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 43. SET FLAG status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
'6B'	00	P1 or P2 are invalid

7.3.6 GET FLAG

This command checks state of the configuration flag specified in P2.

GET FLAG command shall be sent to RAM applet via OTA.

The command message is coded according to the following table.

7.3.6.1 Command message

Code	Value
CLA	'80'
INS	'02'
P1	'00' – UICC configuration flag group '01' – GP configuration flag group
P2	Flag number as specified in 5.1.3 and 5.1.4
Lc	Not present
Data	Not present
Le	Not present

The data field of the command message is not present.

7.3.6.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

SW1	SW2	Meaning
Normal processing		
'90'	'10'	If a specified configuration flag is not set
'90'	'11'	If a specified configuration flag is set
Errors		
'6B'	00	P1 or P2 are invalid

7.3.7 LOCK PIN MANAGER

After this command has been performed, the PIN manager will be locked forever and will no more accept state changing commands, like INITIALIZE PIN.

The command message is coded according to the following table.

7.3.7.1 Command message

Code	Value
CLA	'80'
INS	'F0'
P1	'00'
P2	'00'
Lc	Not present

Data	Not present
Le	Not present

The data field of the command message is not present.

7.3.7.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 44. LOCK PIN MANAGER status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command.
Errors		
		All other values.

7.3.8 PIN MANAGER STATUS

This command provides information about the lock status of the PIN manager as well as the list of all managed PIN IDs.

The command message is coded according to the following table.

7.3.8.1 Command message

Code	Value
CLA	'80'
INS	'F2'
P1	'00'
P2	'00'
Lc	Not present
Data	Not present
Le	Length of data sent from the UICC

The data field of the command message is not present.

7.3.8.2 Response message

The data field of the response message is coded according to the following table.

Table 45. PIN MAGAER STATUS response message data field

Bytes	Description	Length
1	PIN manager lock: '00' – locked; '01' – unlocked.	1 byte
2	PIN manager capacity. Maximum number of PINs that can be managed.	1 byte

3 – 2*(X+1)	List of PINs, 2 bytes per entry: the first byte is PIN ID, the second byte is PIN instance number. 'FFFF' - indicates empty entries (non-instantiated PINs).	2*X bytes

The following status conditions shall be returned by the UICC.

Table 46. PIN MANAGER STATUS status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'67'	'00'	Lc or Le are invalid
'6B'	'00'	P1 or P2 are invalid

7.3.9 INITIALIZE PIN

This command creates and initializes an UICC PIN. Various PIN properties like verification counters, status, 2G mapping, etc. are set.

7.3.9.1 Command message

The command message is coded according to the following table.

Table 47. INITIALIZE PIN command message

Code	Value
CLA	'80'
INS	'F4'
P1	'00'
P2	'00'
Lc	Length of the subsequent data field
Data	Data sent to the UICC
Le	Not present

The data field of the command message is coded according to the following table.

Table 48. INITIALIZE PIN command message data field

Bytes	Description	Status	Length
3G block:			
1	PIN ID	M	1 byte
2	PIN instance	M	1 byte
3	Status flags	M	1 byte
4	Use counter	M	1 byte
5	Verification attempt counter	M	1 byte
6	Verification attempts counter reload	M	1 byte
7-14	Value	M	8 bytes
15	Unblock attempts counter	M	1 byte

16	Unblock attempts counter reload	M	1 byte
17-24	Unblock PIN value	M	1 byte
2G block:			
25	GSM PIN ID	M	1 byte
26	GSM status position	M	1 byte
27	GSM access level	M	1 byte
Access rights token			
28	Access right token length	M	1 byte
29 – (29 + (X-1))	Access rights token	O	X bytes

PIN ID: UICC PIN reference.

PIN instance: UICC PIN instance (meaningful for local PINs only, for global PINs must always be equal '01').

Status flags: Status of this PIN.

'00' - PIN is disabled. If the PIN is disabled, it is considered automatically verified from the security point of view. If the PIN is disabled and blocked for whatever reason, it is still considered implicitly verified.

'02' - PIN is enabled, meaning that it should be successfully verified before access conditions associated with this PIN are granted.

Use counter: The successful verification attempts counter. It depicts how many times this PIN can be successfully verified. Coded as byte value from '00' to 'FE' (If it is 'FF', there's no limit for successful verifications).

Verification attempt counter: The remaining unsuccessful verification attempts counter. Coded as byte value from '00' to '0F'.

Verification attempts counter reload: The reload value for the unsuccessful verification attempts counter. Coded as byte value from '00' to '0F'.

Value: Reference value for PIN checking, right-padded with 'FF'.

Unblock attempts counter: The successful verification attempts counter for unblock PIN checking. It depicts how many times this unblock PIN can be successfully verified. Coded as byte value from '00' to 'FE' (If it is 'FF', there's no limit for successful verifications).

Unblock attempts counter: The remaining unsuccessful verification attempts counter (for unblock PIN). Coded as byte value from '00' to '0F'.

Unblock attempts counter reload: The reload value for the unsuccessful verification attempts counter (for unblock PIN). Coded as byte value from '00' to '0F'.

Unblock PIN value: Reference value for unblock PIN checking, right-padded with 'FF'.

GSM PIN ID: Coded as one byte. 'FF' - if not mapped.

GSM status position: Offset to the start of the status of this PIN inside the information returned in response to a SELECT or a STATUS command.

'13' to '1A' - useful values;

'FF' - if not mapped.

GSM access level: GSM access level associated with this PIN.

'FF' - if not mapped.

Access rights token: Access rights token representing rights granted by successful verification of this PIN. Normally, the PIN will only be responsible for a single bit in this token, directly corresponding to the PIN's ID and instance. But if one needs to implement hierarchical PINs, this token can be set up to grant multiple access rights.

Access rights tokens are of a variable length, defined by the number of local PINs and ADMs. A token is coded according following table:

Table 49. Coding of Access rights token

Byte	Meaning
0	Status of global PINs (APINx): - bit 0: APIN1 - bit 1: APIN2 - bit 2: APIN3 - bit 3: APIN4 - bit 4: APIN5 - bit 5: APIN6 - bit 6: APIN7 - bit 7: APIN8
1	Status of global ADMs and UPIN: - bit 0: ADM1 - bit 1: ADM2 - bit 2: ADM3 - bit 3: ADM4 - bit 4: ADM5 - bit 5: PP - bit 6: UPIN - bit 7: ALWays
2	Status of local PINs (2APINx), instance #1: - bit 0: 2APIN1 - bit 1: 2APIN2 - bit 2: 2APIN3 - bit 3: 2APIN4 - bit 4: 2APIN5 - bit 5: 2APIN6

	- bit 6: 2APIN7 - bit 7: 2APIN8
…	…
2+M-1	Status of local PINs (2APINx), instance #M: - bit 0: 2APIN1 - bit 1: 2APIN2 - bit 2: 2APIN3 - bit 3: 2APIN4 - bit 4: 2APIN5 - bit 5: 2APIN6 - bit 6: 2APIN7 - bit 7: 2APIN8
2+M	Status of local ADMs, instance #1: - bit 0: ADM6 - bit 1: ADM7 - bit 2: ADM8 - bit 3: ADM9 - bit 4: ADM10 - bit 5: RFU - bit 6: RFU - bit 7: RFU
…	…
2-M+N-1	Status of local ADMs, instance #N: - bit 0: ADM6 - bit 1: ADM7 - bit 2: ADM8 - bit 3: ADM9 - bit 4: ADM10 - bit 5: RFU - bit 6: RFU - bit 7: RFU

For example: the basic ART for UPIN is [0, 0x40]. But if we describe ART for UPIN like [0x01, 0x40], it'll mean that after the successful verification of this PIN, the privileges of APIN1 will be granted as well.

7.3.9.2 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 50. INITIALIZE PIN status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'65'	'81'	The PIN manager capacity is exhausted
'67'	'00'	Lc or Le are invalid

'69'	'84'	The reference PIN or unblock value is invalid (e.g. only '00' or 'FF')
'69'	'85'	The PIN manager is locked
'6A'	'89'	The PIN/instance pair is already allocated to another PIN
'6B'	'00'	P1 or P2 are invalid

7.3.10 DESCRIBE PIN

This command provides information about the given PIN. Various PIN properties like verification counters, status, 2G mapping, etc. are returned.

7.3.10.1 Command message

The command message is coded according to the following table.

Table 51. DESCRIBE PIN command message

Code	Value
CLA	'80'
INS	'F6'
P1	'00'
P2	'00'
Lc	Length of the subsequent data field
Data	Data sent to the UICC
Le	Not present

The data field of the command message is coded according to the following table.

Table 52. DESCRIBE PIN command message data field

Bytes	Description	Status	Length
1	PIN ID	M	1 byte
2	PIN instance	O	1 byte

PIN ID: UICC PIN reference (see 7.3.9.1 for more details).

PIN instance: UICC PIN instance number (default value is '01').

Special values of PIN instance:

'FF' - PIN ID will be treated as GSM CHV ID;

'FE' - PIN ID will be treated as GSM access level.

7.3.10.2 Response message

The data field of the response message is coded according to the following table.

Table 53. DESCRIBE PIN response message data field

Bytes	Description	Status	Length
3G block:			
1	PIN ID	M	1 byte

2	PIN instance	M	1 byte
3	Status flags	M	1 byte
4	Use counter	M	1 byte
5	Verification attempt counter	M	1 byte
6	Verification attempts counter reload	M	1 byte
15	Unblock attempts counter	M	1 byte
16	Unblock attempts counter reload	M	1 byte
2G block:			
25	GSM PIN ID	M	1 byte
26	GSM status position	M	1 byte
27	GSM access level	M	1 byte
Access rights token			
28	Access right token length (NOTE 1)	M	1 byte
29 – (29 + (X-1))	Access rights token (NOTE 2)	M	6 bytes

See 7.3.9.1 for more info about content.

NOTE 1: Access right token length equals '06'. It's the default size of an internal buffer with access rights token (coded as 1 byte for global APINs + 1 byte for global ADMs & UPIN + 2 bytes for local APINs + 2 bytes for local ADMs).

NOTE 2: This field is mandatory in response. This command returns data from internal array that can be filled via INITIALIZE PIN command.

The following status conditions shall be returned by the UICC.

Table 54. DESCRIBE PIN status conditions

SW1	SW2	Meaning
Normal processing		
'90'	'00'	Normal ending of the command
Errors		
'67'	'00'	Lc or Le are invalid
'6A'	'88'	The PIN/instance pair is already allocated to another PIN
'6B'	'00'	P1 or P2 are invalid

7.4 Global Platform Commands

7.4.1 PUT KEY Command

Default GP card's state after initialization is OP_READY. In this state it is possible to use PUT KEY command with keys in clear text without key check value and authentication.

At least one GP keyset should be personalized with TDES (16 bytes length) keys to disable default GP keys.

7.4.1.1 Command message

The command message is coded according to the following table.

Table 55. PUT KEY command message

Code	Value
CLA	'80'
INS	'D8'
P1	'00'
P2	'81' Multiple keys in one command. First key ID - 1.
Lc	'22' for DES (8 bytes) keys '3A' for TDES (16 bytes) keys '52' for TDES (24 bytes) keys '40' for AES (16 bytes) keys Variable for TLS keys
Data	See Table 56. PUT KEY command message data field in case of DES keys See Table 57. PUT KEY command message data field in case of AES keys See Table 58. PUT KEY command message data field in case of TLS keys
Le	Not present.

The data field of the command message is coded according to the following tables.

Table 56. PUT KEY command message data field in case of DES keys

Value	Description	Status	Length
'01'	Keyset Version Number	M	1 byte
'80'	Key Type (DES)	M	1 byte
'10'	Key Length ('08' for DES, '10' or '18' for TDES)	M	1 byte
'XX..XX'	KIC Key Value	M	8, 16, or 24 bytes
'00'	Key Check Value Length	M	1 byte
'80'	Key Type (DES)	M	1 byte
'10'	Key Length ('08' for DES, '10' or '18' for TDES)	M	1 byte
'XX..XX'	KID Key Value	M	8, 16, or 24 bytes
'00'	Key Check Value Length	M	1 byte
'80'	Key Type (DES)	M	1 byte
'10'	Key Length ('08' for DES, '10' or '18' for TDES)	M	1 byte
'XX..XX'	KIK Key Value	M	8, 16, or 24 bytes
'00'	Key Check Value Length	M	1 byte

Table 57. PUT KEY command message data field in case of AES keys

Value	Description	Status	Length

‘01’	Keyset Version Number	M	1 byte
‘88’	Key Type (AES)	M	1 byte
‘12’	Length of Key Component Data	M	1 byte
‘10’	Key Length	M	1 byte
‘XX..XX’	KIC Key Value	M	16 byte
‘08’	MAC Length	M	1 byte
‘00’	Key Check Value Length	M	1 byte
‘88’	Key Type (AES)	M	1 byte
‘12’	Length of Key Component Data	M	1 byte
‘10’	Key Length	M	1 byte
‘XX..XX’	KID Key Value	M	16 bytes
‘08’	MAC Length	M	1 byte
‘00’	Key Check Value Length	M	1 byte
‘88’	Key Type (AES)	M	1 byte
‘12’	Length of Key Component Data	M	1 byte
‘10’	Key Length	M	1 byte
‘XX..XX’	KIK Key Value	M	16 bytes
‘08’	MAC Length	M	1 byte
‘00’	Key Check Value Length	M	1 byte

Table 58. PUT KEY command message data field in case of TLS keys

Value	Description	Status	Length
‘41’	Keyset Version Number	M	1 byte
‘85’	Key Type (TLS)	M	1 byte
‘10’	Key Length	M	1 byte
‘XX..XX’	KIC Key Value	M	16 byte
‘00’	Key Check Value Length	M	1 byte
‘85’	Key Type (TLS)	M	1 byte
‘10’	Key Length	M	1 byte
‘XX..XX’	KID Key Value	M	16 bytes
‘00’	Key Check Value Length	M	1 byte

Second Key used as KEK for update TLS key via remote management (SCP81).

7.4.1.2 PUT KEY Command for SSD Keys

In order to use Supplementary Security Domain it have to be personalized. It can be done by GP PUT KEY command (same as for OTA (GP) keys). Security Domain have to be selected. After personalization Life Cycle State of Security Domain changes to PERSONALIZED.

7.4.1.3 PUT KEY Command for DAP Keys

In order to use DAP verification SSD DAP key have to be personalized. It can be done by GP PUT KEY command for DAP key. Security Domain have to be selected.

The command message is coded according to the following table.

Table 59. PUT KEY command message for DAP key

Code	Value
CLA	'80'
INS	'D8'
P1	'00'
P2	'11' One key in one command. Key ID – 11 (DES DAP key).
Lc	'14' for DES DAP key
Data	See
Le	Not present.

The data field of the command message is coded according to the following table.

Table 60. PUT KEY command message data field in case of DAP key

Value	Description	Status	Length
'72'	Keyset Version Number	M	1 byte
'80'	Key Type (DES)	M	1 byte
'10'	Key Length	M	1 byte
'XX..XX'	KIC Key Value	M	16 bytes
'00'	Key Check Value Length	M	1 byte

7.4.1.4 Response message

The data field of the response message is not present.

The following status conditions shall be returned by the UICC.

Table 61. PUT KEY status conditions

SW1	SW2	Meaning
Normal processing		
'61'	'xx'	Normal ending of the command
Errors		
		All other values.

8 Card Production Process

8.1 Card Initialization

Card initialization process consists from following steps.

1. Initialize card. For INITIALIZE CARD command coding and description see 7.3.1INITIALIZE CARD.
2. Install SIM application. For installation parameters see 6.2 SIM Application.
3. Install TLS application. For installation parameters see 6.5 TLS application.
4. Install BIP application. For installation parameters see 6.6 BIP application.
5. Install HTTP Admin application. For installation parameters see 6.7 HTTP Admin application.
6. Select ISD and load third party applications that can be used in MNO profiles. This step is optional.

8.2 Card Personalization

Card personalization process consists from following steps..

1. Select ISD and personalize at least one SCP02 or SCP03 Key Set. For PUT KEY command coding see 7.4.1 PUT KEY Command. It shall be TripleDES Key Set. Optionally personalize OTA keys and TLS keys if SCP80 and SCP81 enabled for ISD.Since card is in GP OP_READY state it is possible to use keys data in clear text and without check value.
2. Select ISD and set card Life Cycle State to INITIALIZED or SECURED.

9 List of Figures

10 List of Tables

11 References

Reference	Title	Revision