TRUSTED CONNECTIVITY ALLIANCE

eUICC Profile Package: Interoperable Format Test Specification

Version 2.0

Published by simalliance now Trusted Connectivity Alliance

July 2016

Intellectual Property Rights (IPR) Disclaimer

Attention is drawn to the possibility that some of the elements of any material available for download from the specification pages on Trusted Connectivity Alliance's website may be the subject of Intellectual Property Rights (IPR) of third parties, some, but not all, of which are identified below.

Trusted Connectivity Alliance shall not be held responsible for identifying any or all such IPR, and has made no inquiry into the possible existence of any such IPR. TRUSTED CONNECTIVITY ALLIANCE SPECIFICATIONS ARE OFFERED WITHOUT ANY WARRANTY WHATSOEVER, AND IN PARTICULAR, ANY WARRANTY OF NON-INFRINGEMENT IS EXPRESSLY DISCLAIMED. ANY IMPLEMENTATION OF ANY TRUSTED CONNECTIVITY ALLIANCE SPECIFICATION SHALL BE MADE ENTIRELY AT THE IMPLEMENTER'S OWN RISK, AND NEITHER TRUSTED CONNECTIVITY ALLIANCE, NOR ANY OF ITS MEMBERS OR SUBMITTERS, SHALL HAVE ANY LIABILITY WHATSOEVER TO ANY IMPLEMENTER OR THIRD PARTY FOR ANY DAMAGES OF ANY NATURE WHATSOEVER DIRECTLY OR INDIRECTLY ARISING FROM THE IMPLEMENTATION OF ANY TRUSTED CONNECTIVITY ALLIANCE SPECIFICATION.

Table of Contents

1. Objective

The objective of this document is to define the test specification of the interoperable eUICC Profile. This specification is based on [SA PP TS].

2. Introduction

This specification has the objective of testing if a profile is correctly interpreted and correctly loaded on an eUICC.

This document is agnostic on the format of the eUICC: both soldered (embedded in a device) and non-soldered (stand-alone) eUICCs can be the subject of testing. The test cases are written so that they can be used to test both soldered and non-soldered eUICC formats.

The elements within the scope of this test specification are described in the following figure:

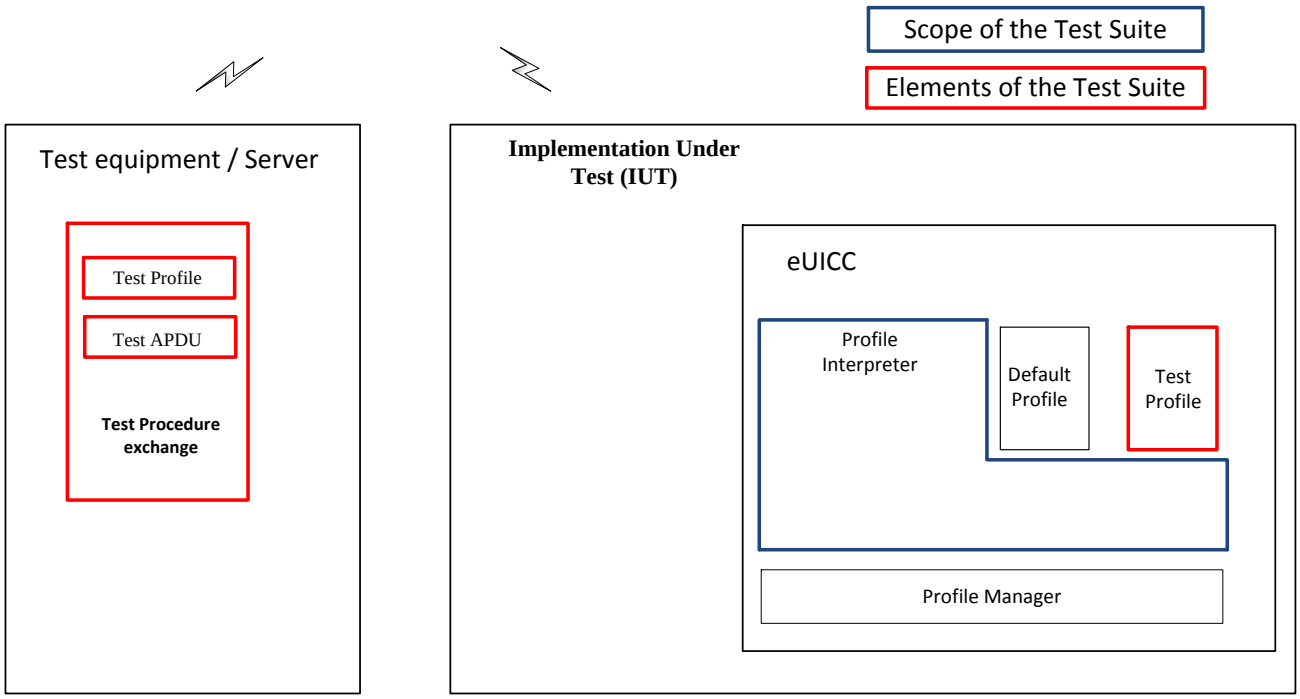

Figure 1: Scope of the testing

3. References

3.1 Normative References

- [SA PP RS]: SIMalliance eUICC Profile Package: Interoperability Functional Requirements V1.1
- [SA PP TS]: SIMalliance eUICC Profile Package: Interoperable Format Technical Specification V2.0
- [101 220]: ETSI TS 101 220 - V12.0.0: Smart Cards; ETSI numbering system for telecommunication application providers (Release 12)
- [102 221]: ETSI TS 102 221 V12.0.0: Smart Cards; UICC-Terminal interface; Physical and logical characteristics (Release 12)
- [102 222]: ETSI TS 102 222 V7.1.0: Integrated Circuit Cards (ICC); Administrative commands for telecommunications applications (Release 7)
- [102 226]: ETSI TS 102 226 V12.0.0: Smart Cards; Remote APDU structure for UICC based applications (Release 12)
- [USIM]: 3GPP TS 31.102 V12.6.0: Characteristics of the Universal Subscriber Identity Module (USIM) application (Release 12)
- [ISIM]: 3GPP TS 31.103 V12.2.0: Characteristics of the IP Multimedia Services Identity Module (ISIM) application (Release 12)
- [CSIM]: 3GPP2 C.S0065-C v1.0: cdma2000 Application on UICC for Spread Spectrum Systems
- [GP CS]: GlobalPlatform Card Specification V2.2.1
- [GP UC]: GlobalPlatform Card Specification UICC Configuration V1.0.1
- [GP AA]: Confidential Card Content Management; GlobalPlatform Card Specification Amendment A v1.0.1
- [GP AB]: GlobalPlatform Card Remote Application Management over HTTP Card Specification v2.2 – Amendment B v1.1.2
- [X.680]: ITU-T X.680 (11/2008): Abstract Syntax Notation One (ASN.1): Specification of basic notation including Corrigendum 1 and 2
- [X690]: ITU-T X.690 (11/2008): ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) including Corrigendum 1 and 2
- [102 230-2]: ETSI TS 102 230-2 V9.0.0: Smart Cards; UICC-Terminal interface; Physical, electrical and logical test specification; Part 2: UICC features;
- [103 484-2]: ETSI TS 103 484-2 V9.0.0: Smart Cards; Test specification for the Secure Channel interface; Part 2: UICC features
- [USIM Test]: 3GPP TS 31.122 V12.0.0: Universal Subscriber Identity Module (USIM) conformance test specification (Release 12)
- [UICC]: 3GPP TS 31.101 V12.2.0: UICC-terminal interface; Physical and logical characteristics (Release 12)

- [GS RPT]: GSMA Remote Provisioning Architecture for Embedded UICC Technical Specification V3.1, 27 May 2016
- [MILENAGE]: 3GPP TS 35.207: 3G Security; Specification of the MILENAGE algorithm set: An example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 3: Implementors' test data
- [TUAK]: 3GPP TS 35.233: Specification of the TUAK algorithm set: A second example algorithm set for the 3GPP authentication and key generation functions f1, f1*, f2, f3, f4, f5 and f5*; Document 3: Design conformance test data
- [GS RPAT]: GSMA Remote Provisioning Architecture for Embedded UICC, Test Specification Version 3.1, 31 May 2016

3.2 Informative References

- [GS RPA]: GSMA Remote Provisioning Architecture for Embedded UICC V1.1
- [102 383]: ETSI TS 103 383 V12.7.0: Smart Cards; Embedded UICC; Requirements Specification (Release 12)

4. Abbreviations

ADF	Application Dedicated File
AID	Application Identifier
AKA	Authentication and Key Agreement
APDU	Application Protocol Data Unit
ASN.1	Abstract Syntax Notation One
CASD	Controlling Authority Security Domain
CD	Configuration Data
CDMA	Code Division Multiple Access
CSIM	cdma2000 Subscriber Identify Identity Module
CIN	Card Image Number / Card Identification Number
DF	Dedicated File
DGI	Data Grouping Identifier
DO	Data Object
EAP	Extensible Authentication Protocol
EF	Elementary File
eUICC	embedded UICC
EUM	eUICC Manufacturer
FCP	File Control Parameters
FFS	For Further Study
GBA	Generic Bootstrapping Architecture
HCI	Host Controller Interface
ICCID	Integrated Circuit Card ID
ID	Identifier
IIN	Issuer Identification Number

IMSI	International Mobile Subscriber Identity
ISD-P	Issuer Security Domain Profile
ISIM	IP Multimedia Services Identity Module
IUT	Implementation Under Test
LCSI	Life Cycle Status Information
M2M	Machine to Machine
MAC	Message Authentication Code
MAC-A	MAC used for authentication and key agreement
MBMS	Multimedia Broadcast/Multicast Service
MNO	Mobile Network Operator
MNO-SD	Mobile Network Operator Security Domain (Root SD of a Profile)
NAA	Network Access Application
NAC	Network Access Control
OID	Object Identifier
OS	Operating System (of the eUICC)
OTA	Over the Air
PE	Profile Element
PIN	Personal Identification Number
POL	Policy Rules within the Profile
PUK	PIN Unblocking Key
RAM	Remote Application Management
RFM	Remote File Management
RQ	Requirement
SCP	Secure Channel Protocol
SD	Security Domain
SP	Service Provider
SQN	Sequence Number
SSD	Supplementary Security Domain
SW	Status Word
SWP	Single Wire Protocol
USIM	Universal Subscriber Identity Module
T	Test Tool

5. Definitions

Default Profile	A profile which can be used to connect to the network.
embedded UICC	An UICC which is not easily accessible or replaceable, is not intended to be removed or replaced in the terminal, and enables the secure changing of subscriptions.
Policy Rules	Defines the atomic action of a policy and the conditions under which it is executed.
Profile	Combination of a file structure, data and applications on an eUICC.
Profile Creator	External entity in charge of creating the Profile Package based on MNO requirements, protecting the Profile Package from modification and/or content access.

Profile Element	A Profile Element is a part of the Profile Package representing one or several features of the Profile encoded using TLV structures based on ASN.1 description.
Profile Interpreter	On card entity which interprets and translates the ASN profile data to objects residing on the eUICC (files, SD-s, applications, keys, etc.).
Profile Manager	On-card entity, which is able to load, install, activate and deactivate a profile as per GSMA [GS RPT].
Profile Package	A Personalised Profile using an interoperable description format transmitted to an eUICC in order to load and install a Profile.
Provisioning	The downloading and installation of a Profile into an eUICC.
Remote Provisioning	Provisioning done by the subscription manager on an eUICC outside of his premises, using a secure data link.

6. Test environment

6.1 Table of optional features

The supplier of the implementation shall state the support of possible options in **Table 1**.

Table 1: Options

Item	Option	Support	Mnemonic
1	Support of USIM		O_USIM
2	Support of ISIM		O_ISIM
3	Support of CSIM		O_CSIM
4	Support of milenage		O_MILENAGE
5	Support of TUAK		O_TUAK
6	Support of CAVE		O_CAVE
7	Support of GBA-USIM		O_GBA_USIM
8	Support of GBA-ISIM		O_GBA_ISIM
9	Support of MBMS		O_MBMS
10	Support of EAP		O_EAP
11	Support Contactless		O_CONTACTLESS
12	Support of Java Card		O_JAVACARD
13	Support of Multos		O_MULTOS
14	Support of ETSI TS 102 613 and TS 102 622 Card-emulation Mode		O_CARDEMULATION
15	Support of ETSI TS 102 613 and TS 102 622 Reader Mode		O_READER_MODE
16	Support of GlobalPlatform UICC Configuration		O_UICC_CONFIGURATION
17	VOID		
18	VOID		
19	For ApplicationLoadPackage, the following parameters are supported: nonVolatileCodeLimitC6 volatileDataLimitC7 nonVolatileDataLimitC8		O_MEMORY_LIMIT
20	For ApplicationLoadPackage hashValue is supported		O_HASHVALUE
21	The eUICC reports error when profile with PE-USIM before PE-MF is loaded		O_ERROR_FOR_PE_USIM_BEFORE PE_MF
22	The eUICC reports error when profile with PE-Application before PE-SecurityDomain is loaded		O_ERROR_FOR_PE_APPLICATION_BEFORE PE_SECURITYDOMAIN
23	The eUICC reports error when profile with PE-RFM before PE-SecurityDomain is loaded		O_ERROR_FOR_PE_RFM_BEFORE PE_SECURITYDOMAIN
24	The eUICC is able to correctly load profiles with PE-USIM before PE-MF		O_SUPPORT_PE_USIM_BEFORE PE_MF
25	The eUICC is able to correctly load profiles with PE-Application before PE-SecurityDomain		O_SUPPORT_PE_APPLICATION_BEFORE PE_SECURITYDOMAIN
26	The eUICC is able to correctly load profiles with PE-RFM before PE-SecurityDomain		O_SUPPORT_PE_RFM_BEFORE PE_SECURITYDOMAIN
27	Support of PE MF (OID: 2.23.143.1.2.1) creation by template		O_SUPPORT_PE_MF_BY_TEMPLATE
28	Support of PE USIM (OID: 2.23.143.1.2.4) creation by template		O_SUPPORT_PE_USIM_BY_TEMPLATE
29	Support of PE OPT USIM (OID: 2.23.143.1.2.5) creation by template		O_SUPPORT_PE_OPT_USIM_BY_TEMPLATE

The following dependencies exist between the options:

- At least one of the NAA options O_USIM and O_CSIM shall be supported.
- If O_USIM is supported, then the algorithm option O_MILENAGE shall be supported.
- When O_GBA_USIM is supported also O_USIM shall be supported.
- When O_GBA_ISIM is supported also O_ISIM shall be supported.
- At least one of the runtime environments O_JAVACARD and O_MULTOS shall be supported.

6.2 Applicability table

Table 2 specifies the applicability of each test case to the IUT.

Table 2: Applicability of tests

Test case	Test case title	Version 2.0	Support
	Profile Package General Structure tests		
	FFS		
	Profile Package Elements Definition tests		
	Check Profile Format		
8.2.1.1	Installing PE-MF, PE-USIM and PE-OPT-USIM when eUICC supports file system creation by generic file manager	C009	
8.2.1.2	Installing PE-MF, PE-USIM and PE-OPT-USIM when eUICC supports file system creation by template	C010	
8.2.1.3	Installing PE-USIM when eUICC does not support USIM	C003	
8.2.1.4	Installing profile without ProfileHeader PE	C009	
8.2.1.5	Installing profile with PE-USIM before PE-MF, eUICC reports error	C006	
8.2.1.6	Installing profile with PE-Application before PE-SecurityDomain, eUICC reports error	C007	
8.2.1.7	Installing profile with PE-RFM before PE-SecurityDomain, eUICC reports error	C008	
8.2.1.8	Installing profile with PE-USIM before PE-MF	C011	
	Check PE Security Domain		
8.2.2.1	Check mandatory elements in PE Security Domain	C009	
8.2.2.2	Check key list in PE Security Domain	C009	
8.2.2.3	Check number of keyComponent objects	C009	
8.2.2.4	Check sdPersoData	C009	
8.2.2.5	Check OTA HTTPs Personalisation	C009	
	Check PE Application		
8.2.3.1	Check Application PE (PE_Applet1) and mandatory elements in ApplicationInstance	C009	
8.2.3.2	Check all elements in ApplicationLoadPackage – taking size into account	C009	
8.2.3.3	Check all elements in ApplicationInstance	C009	
8.2.3.4	Error when load a PE-Applet4 and bad library is provided	C009	
8.2.3.5	Check multiple ApplicationInstance	C009	
8.2.3.6	Check processData	C009	

Table 3: Conditional items referenced by Table 2

Conditional item	Condition
C001	VOID
C002	VOID
C003	IF O_USIM NOT SUPPORTED THEN M ELSE N/A
C004	IF O_MEMORY_LIMIT SUPPORTED THEN M ELSE N/A
C005	IF O_HASHVALUE SUPPORTED THEN M ELSE N/A
C006	IF (O_USIM SUPPORTED AND O_ERROR_FOR_PE_USIM_BEFORE PE_MF SUPPORTED) THEN M ELSE N/A
C007	IF (O_USIM SUPPORTED AND O_ERROR_FOR_PE_APPLICATION_BEFORE PE_SECURITYDOMAIN SUPPORTED) THEN M ELSE N/A
C008	IF (O_USIM SUPPORTED AND O_ERROR_FOR_PE_RFM_BEFORE PE_SECURITYDOMAIN SUPPORTED) THEN M ELSE N/A
C009	IF O_USIM SUPPORTED THEN M ELSE N/A
C010	IF (O_USIM SUPPORTED AND O_SUPPORT_PE_MF_BY_TEMPLATE SUPPORTED AND O_SUPPORT_PE_USIM_BY_TEMPLATE SUPPORTED AND O_SUPPORT_PE_OPT_USIM_BY_TEMPLATE SUPPORTED) THEN M ELSE N/A
C011	IF (O_USIM SUPPORTED AND O_SUPPORT_PE_USIM_BEFORE PE_MF SUPPORTED) THEN M ELSE N/A

6.3 Optional features and applicability tables formatting

6.3.1 Format of the table of optional features

The columns in **Table 1** have the following meaning.

Column	Meaning
Option:	The optional feature supported or not by the implementation.
Support:	The support columns are to be filled in by the supplier of the implementation. The following common notations are used for the support column in table 1. • Y or y supported by the implementation; • N or n not supported by the implementation; • N/A, or n/a - no answer required (allowed only if the status is N/A, directly or after evaluation of a conditional status).
Mnemonic:	The mnemonic column contains mnemonic identifiers for each item.

6.3.2 Format of the applicability table

The applicability of every test in **Table 2** is formally expressed by the use of Boolean expressions defined in the following clause 6.3.3.

The columns in **Table 2** have the following meaning:

Column	Meaning
Test case:	The “Test case” column gives a reference to the test case number(s) detailed in the present document.
Test case title:	The “Test case title” column gives the title of the test case.
Version X:	The “Version X” column indicates which test cases are applicable for the given Technical Specification version. Several different status notifications can be used in this column. They are defined in clause 6.3.3.
Support:	The “Support” column is blank in the proforma, and is to be completed by the manufacturer in respect of each particular requirement to indicate the choices that have been made in the implementation.

6.3.3 Status and Notations

The “Version X” columns show the status of the entries as follows:
The following notations are used for the status column:

M mandatory – the capability is required to be supported.
O optional – the capability may be supported or not.
N/A not applicable – in the given context, it is impossible to use the capability.
X prohibited (excluded) – there is a requirement not to use this capability in the given context.
O.i qualified optional – for mutually exclusive or selectable options from a set. “i” is an integer which identifies an unique group of related optional items and the logic of their selection, which is defined immediately following the table.
Ci conditional – the requirement on the capability (“M”, “O”, “X” or “N/A”) depends on the support of other optional or conditional items. “i” is an integer identifying an unique conditional status expression, which is defined immediately following the table. For nested conditional expressions, the syntax “IF ... THEN (IF ... THEN ... ELSE...) ELSE ...” is to be used to avoid ambiguities.

6.4 Test environment description

The general architecture for the test environment is:

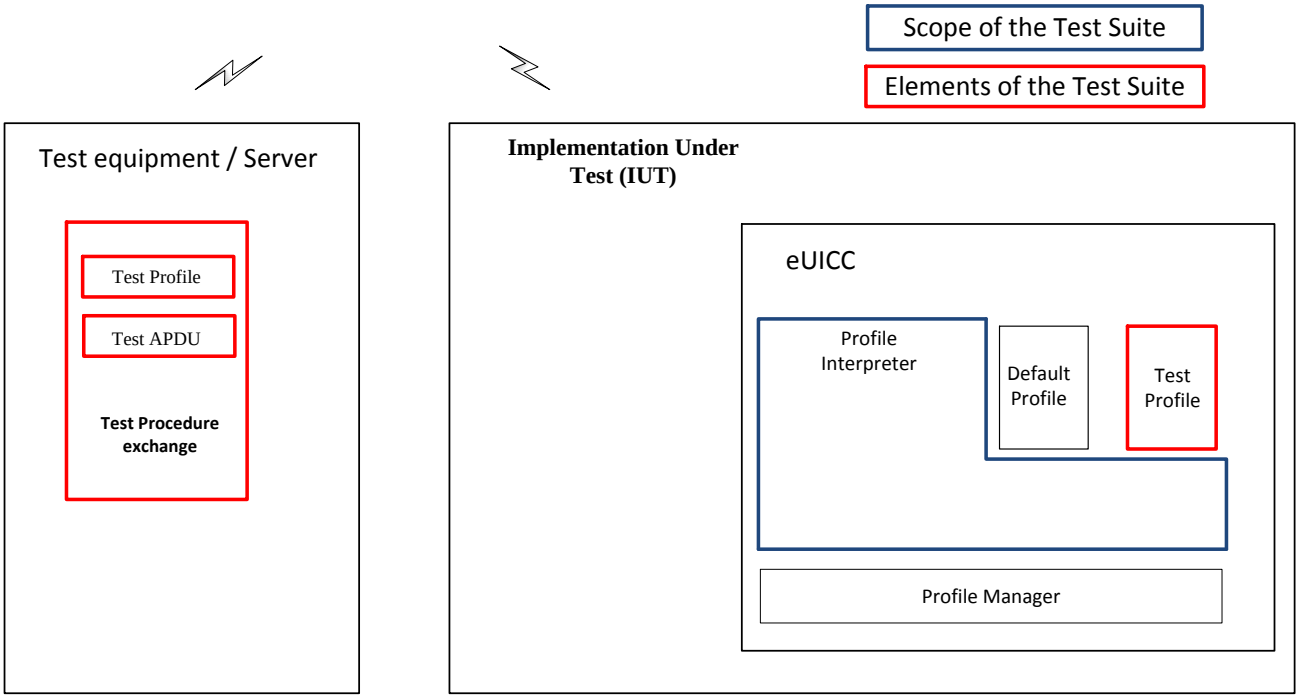

Figure 2: Test environment description

6.5 Test equipment

The test equipment shall meet the following requirements:

- The result of I/O commands shall be presented at the application layer.
- It shall be able to provide results of the tests.
- It shall be able to accept all valid status codes returned.
- It shall send all data specified in the test profile.
- It may be able to send and receive commands remotely to/from the IUT, OR
- It may provide a terminal simulation that is connected to the IUT during test procedure execution, unless otherwise specified. With respect to the eUICC, the terminal simulation shall act according to ETSI TS 102 221 [102 221], 3GPP TS 31.101 [UICC] (if this interface is present at the UICC) and 3GPP TS 31.102 [USIM], unless otherwise specified. The terminal simulation may provide the possibility to monitor the eUICC on the ETSI TS 102 221 [102 221] interface if this interface is accessible.

6.6 Test execution

The order of the PE-s in the Test Profiles shall be kept as it is defined in the “Test Execution” subchapter of each test case.

After each test case execution, the eUICC shall be put back to its initial state.

6.6.1 General Initial Conditions

The General Initial Conditions are a set of general prerequisites for the IUT prior to the execution of testing. For each test procedure described in the present document, the following rules apply to the Initial Conditions:

- Unless otherwise stated, the IUT shall be reset before each test procedure.
- The ISD-P shall be installed and personalised.

6.7 Pass criterion

A test shall be considered as successful, only if the test procedure was carried out successfully with the IUT respecting all conformance requirements referenced in the test procedure.

6.8 Indications concerning support of features

For the following features, if the file system is using the PE template, the eUICC shall support the given related PEs (optional for the profiles):

- When supporting the USIM feature, the following PEs are mandatory to support: PE-USIM, PE-CD, PE-TELECOM, PE-OPT-USIM, PE-GSM-ACCESS, PE-PHONEBOOK, USIM Related Files and Directories PEs.
- When supporting the ISIM feature, the following PEs are mandatory to support: PE-ISIM, PE-CD, PE-OPT-ISIM, ISIM Related Files and Directories PEs.
- When supporting the CSIM feature, the following PEs are mandatory to support: PE-CSIM, PE-CD, PE-OPT-CSIM, CSIM Related Files and Directories PEs.

When supporting the milenage feature, support of the following PE is mandatory: PE-AKAParameters.

When supporting the tuak feature, support of the following PE is mandatory: PE-AKAParameters.

The following PEs are mandatorily supported by the eUICC, regardless of the supported feature: PE-PINCodes, PE-PUKCodes, PE-SecurityDomain, PE-Application, PE-RFM, PE-End, file systems PEs (PE-MF, PE-CD, PE-TELECOM), Generic File management PEs.

The eUICC is required to recognise PE-NonStandard in a profile, but the processing of the content is not mandatory.

File management templates may also be expressed using the appropriate generic file management.

6.9 eUICC Initalisation Procedures

This procedure shall be applied by the test tool only when the eUICC under test is in an unsoldered format. When the eUICC under test is embedded in a device, the initialisation procedure is accomplished by the device.
To initialise the communication between T and the eUICC, these commands shall be executed:

Step	Direction	Description	RQ
1	T → eUICC	RESET	
2	eUICC -> T	ATR	
3	T → eUICC	[TERMINAL_PROFILE]	
4	eUICC -> T	Toolkit initialization SW=’9000’	

The value of the [TERMINAL_PROFILE] is the same as specified by [GS RPAT] in Annex E1.

NOTE: It is assumed that some proactive commands may be sent by the eUICC after sending the TERMINAL PROFILE (i.e. SET UP EVENT LIST, POLL INTERVAL, PROVIDE LOCAL INFORMATION…). In this case, T shall send the corresponding FETCH and TERMINAL RESPONSE (successfully performed) commands.

6.10 Profile loading

Profile packages shall be loaded using the respective standard procedures supported by the eUICC (e.g. [GS RPT]).

6.11 Profile enabling

Profile packages shall be enabled using the respective standard procedures supported by the eUICC (e.g. [GS RPT]).

6.12 Test PE description

6.12.1 Basic Profile Package PE-s

The Basic Profile contains the following components:

- MF and USIM ADF.
- PIN and PUK codes.
- NAA using milenage algorithm.
- MNO-SD supporting SCP80 in 3DES.
- SSD supporting SCP80 in 3DES.
- Applet.
- RFM application.

The parameters below have been chosen to personalise the Profile:

- Profile type: “SIMalliance Profile Package”.
- ICCID: ‘89019990001234567893’.
- IMSI: 234101943787656.
- MNO-SD AID / TAR: ‘A000000151000000’ / ‘B20100’.
- RFM application AID / TAR: ‘A00000055910100001’ / ‘B00000’.
- Executable Load File AID for SD: ‘A0000001515350’.
- Executable Module AID for SD: ‘A000000151535041’.
- SSD AID / TAR: ‘A00000055910100102736456616C7565’ / ‘6C7565’.

If not stated otherwise access rules are taken from section “Access Rules Definition” of [SA PP TS].

Two additional Access Rules are used in this specification:

Table 4: Additional Access Rules

File Access Conditions						Access Rules	Values
Read	Update	Incr.	Act.	Deact.	Delete		
ALWAYS	PIN 1 OR PIN 2	NEVER	ADM 1	ADM 1	ADM 1	15	8001019000800102A010A40683 0101950108A406830102950108 800158A40683010A950108
ALWAYS	PIN 1 AND ADM 1	NEVER	ADM 1	ADM 1	ADM 1	16	8001019000800102AF10A40683 0101950108A40683010A950108 800158A40683010A950108

6.12.1.1. Profile Header

Profile_HEADER

```
headerValue ProfileElement ::= header : {
  major-version 2,
  minor-version 0,
  profileType "SIMalliance Profile Package",
  iccid '89019990001234567893'H,
  eUICC-Mandatory-services {
    usim NULL,
    milenage NULL,
    javacard NULL
  },
  eUICC-Mandatory-GFSTEList {
  }
}
```

```
A0398001 02810100 821B5349 4D616C6C
69616E63 65205072 6F66696C 65205061
636B6167 65830A89 01999000 12345678
93A50681 0084008B 00A600
```

6.12.1.2. PE MF

6.12.1.2.1. PE MF by Template

PE_MF (Template)

```
mfVal ProfileElement ::= mf : {
  mf-header {
    mandated NULL,
    identification 1
  },
  templateID { 2 23 143 1 2 1 },
  mf {
    fileDescriptor : {
      pinStatusTemplateDO '01020A'H
    }
  },
  ef-pl {
    fileDescriptor : {
-- EF_PL modified to use Access Rule 15 within
EF_ARR
      securityAttributesReferenced '0F'H
    }
  },
  ef-iccid {
    fileDescriptor : {
    -- use Access Rule 16 within EF_ARR
      securityAttributesReferenced '10'H
    }
-- swapped ICCID: 98109909002143658739
    fillFileContent : '98109909002143658739'H
  },
  ef-dir {
    fileDescriptor : {
-- Shareable Linear Fixed File
-- 4 records, record length: 38 bytes
      fileDescriptor '42210026'H,
      efFileSize '0098'H
    },
-- USIM AID: A0000000871002FF33FF018900000100
    fillFileContent :
'61184F10A0000000871002FF33FF01890000010050045
553494D'H
  },
  ef-arr {
    fileDescriptor : {
```

```
B0820225 A0058000 81010181 0667810F
010201A2 07A105C6 0301020A A305A103
8B010FA4 11A1038B 0110830A 98109909
00214365 8739A528 A10A8204 42210026
80020098 831A6118 4F10A000 00008710
02FF33FF 01890000 01005004 5553494D
A68201C5 A10A8204 42210025 80020250
831B8001 01900080 0102A406 83010195
01088001 58A40683 010A9501 0882010A
83168001 01A40683 01019501 0880015A
A4068301 0A950108 82010F83 0B80015B
A4068301 0A950108 82011A83 0A800101
90008001 5A970082 011B8316 800103A4
06830101 95010880 0158A406 83010A95
01088201 0F831680 0111A406 83010195
01088001 4AA40683 010A9501 0882010F
83218001 03A40683 01019501 08800158
A4068301 0A950108 840132A4 06830101
95010882 01048321 800101A4 06830101
95010880 0102A406 83018195 01088001
58A40683 010A9501 08820104 831B8001
01900080 011AA406 83010195 01088001
40A40683 010A9501 0882010A 83108001
01900080 015AA406 83010A95 01088201
15831580 01019000 800118A4 0683010A
95010880 01429700 82011083 10800101
A4068301 01950108 80015A97 00820115
83168001 13A40683 01019501 08800148
A4068301 0A950108 82010F83 0B80015E
A4068301 0A950108 82011A83 25800101
90008001 02A010A4 06830101 950108A4
06830102 95010880 0158A406 83010A95
01088325 80010190 00800102 AF10A406
```

```
-- Shareable Linear Fixed File
-- 16 records, record length: 37 bytes
-- ARR created with content recommended in
Annex A (Section 9.9) plus two additional
records for use with EF_PL and EF ICCID
      fileDescriptor '42210025'H,
      efFileSize '0250'H
    },
    fillFileContent :
'8001019000800102A406830101950108800158A406830
10A950108'H,
    fillFileOffset : 10,
    fillFileContent :
'800101A40683010195010880015AA40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'80015BA40683010A950108'H,
    fillFileOffset : 26,
    fillFileContent : '800101900080015A9700'H,
    fillFileOffset : 27,
    fillFileContent :
'800103A40683010195010880015 8A40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'800111A40683010195010880014AA40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'800103A40683010195010880015 8A40683010A9501088
40132A406830101950108'H,
    fillFileOffset : 4,
    fillFileContent :
'800101A40683010195010880010 2A4068301819501088
00158A40683010A950108'H,
    fillFileOffset : 4,
    fillFileContent :
'800101900080011AA40683010195010880014 0A406830
10A950108'H,
    fillFileOffset : 10,
    fillFileContent :
'800101900080015AA40683010A950108'H,
    fillFileOffset : 21,
    fillFileContent :
'8001019000800118A40683010A9501088001429700'H,
    fillFileOffset : 16,
    fillFileContent :
'800101A40683010195010880015A9700'H,
    fillFileOffset : 21,
    fillFileContent :
'800113A40683010195010880014 8A40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'80015EA40683010A950108'H,
    fillFileOffset : 26,
-- Rule 15: [Read: Always][Update/CreateEF:
PIN Appl 1|PIN Appl 2][Deactivate, Activate,
DeleteSelf: ADM1]
    fillFileContent :
'8001019000800102A010A4068301019501 8A40683010
2950108800158A40683010A950108'H,
-- Rule 16: [Read: Always][Update/CreateEF:
PIN Appl 1 & ADM 1][Deactivate, Activate,
83010195 0108A406 83010A95 01088001
58A40683 010A9501 08
```

```
DeleteSelf: ADM1]
    fillFileContent :
'80010190008001O2AF10A406830101950108A40683010
A950108800158A40683010A950108'H
  }
}
```

6.12.1.2.2. PE MF by Generic File Management

PE_MF (Generic File Management)

```
altMFVal ProfileElement ::=
genericFileManagement : {
  gfm-header {
    mandated NULL,
    identification 1
  },
  fileManagementCMD {
    {
-- create MF
      createFCP : {
        fileDescriptor '7821'H,
        fileID '3F00'H,
        securityAttributesReferenced '0E'H,
        pinStatusTemplateDO '01020A'H
      },
-- create PL
      createFCP : {
        fileDescriptor '4121'H,
        fileID '2F05'H,
        securityAttributesReferenced '0F'H,
        efFileSize '03'H,
        shortEFID '28'H
      },
-- create ICCID
      createFCP : {
        fileDescriptor '4121'H,
        fileID '2FE2'H,
        securityAttributesReferenced '10'H,
        efFileSize '0A'H
      },
-- swapped ICCID: 98109909002143658739
      fillFileContent :
'98109909002143658739'H,
-- create DIR
-- Shareable Linear Fixed File
-- 4 records, record length: 38 bytes
      createFCP : {
        fileDescriptor '42210026'H,
        fileID '2F00'H,
        securityAttributesReferenced '0A'H,
        efFileSize '0098'H,
        shortEFID 'F0'H
      },
-- USIM AID: A0000000871002FF33FF018900000100
      fillFileContent :
'61184F10A0000000871002FF33FF01890000010050045
553494D'H,

-- create ARR
      createFCP : {
-- Shareable Linear Fixed File
-- 15 records, record length: 37 bytes
        fileDescriptor '42210025'H,
        fileID '2F06'H,
A182025E A0058000 810101A1 82025330
82024F62 10820278 2183023F 008B010E
C6030102 0A621182 02412183 022F058B
010F8001 03880128 620E8202 41218302
2FE28B01 1080010A 810A9810 99090021
43658739 62148204 42210026 83022F00
8B010A80 02009888 01F0811A 61184F10
A0000000 871002FF 33FF0189 00000100
50045553 494D6211 82044221 00258302
2F068B01 0A800202 50811B80 01019000
800102A4 06830101 95010880 0158A406
83010A95 01080201 0A811680 0101A406
83010195 01088001 5AA40683 010A9501
0802010F 810B8001 5BA40683 010A9501
0802011A 810A8001 01900080 015A9700
02011B81 16800103 A4068301 01950108
800158A4 0683010A 95010802 010F8116
800111A4 06830101 95010880 014AA406
83010A95 01080201 0F812180 0103A406
83010195 01088001 58A40683 010A9501
08840132 A4068301 01950108 02010481
21800101 A4068301 01950108 800102A4
06830181 95010880 0158A406 83010A95
01080201 04811B80 01019000 80011AA4
06830101 95010880 0140A406 83010A95
01080201 0A811080 01019000 80015AA4
0683010A 95010802 01158115 80010190
00800118 A4068301 0A950108 80014297
0002010E 81108001 01A40683 01019501
0880015A 97000201 15811680 0113A406
83010195 01088001 48A40683 010A9501
0802010D 810B8001 5EA40683 010A9501
0802011A 81258001 01900080 0102A010
A4068301 01950108 A4068301 02950108
800158A4 0683010A 95010881 25800101
90008001 02AF10A4 06830101 950108A4
0683010A 95010880 0158A406 83010A95
0108620E 82024121 83022F08 8B010A80
0105
```

```
        securityAttributesReferenced '0A'H,
        efFileSize '0250'H
      },
    fillFileContent :
'8001019000800102A4068301019501088001 58A406830
10A950108'H,
    fillFileOffset : 10,
    fillFileContent :
'800101A40683010195010880015AA40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'80015BA40683010A950108'H,
    fillFileOffset : 26,
    fillFileContent : '800101900080015A9700'H,
    fillFileOffset : 27,
    fillFileContent :
'800103A40683010195010880015 8A40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'800111A40683010195010880014AA40683010A950108'
H,
    fillFileOffset : 15,
    fillFileContent :
'800103A40683010195010880015 8A40683010A9501088
40132A406830101950108'H,
    fillFileOffset : 4,
    fillFileContent :
'800101A40683010195010880010 2A4068301819501088
00158A40683010A950108'H,
    fillFileOffset : 4,
    fillFileContent :
'800101900080011AA40683010195010880014 0A406830
10A950108'H,
    fillFileOffset : 10,
    fillFileContent :
'800101900080015AA40683010A950108'H,
    fillFileOffset : 21,
    fillFileContent :
'8001019000800118A40683010A9501088001429700'H,
    fillFileOffset : 14,
    fillFileContent :
'800101A40683010195010880015A9700'H,
    fillFileOffset : 21,
    fillFileContent :
'800113A40683010195010880014 8A40683010A950108'
H,
    fillFileOffset : 13,
    fillFileContent :
'80015EA40683010A950108'H,
    fillFileOffset : 26,
-- Rule 15: [Read: Always][Update/CreateEF:
PIN Appl 1| PIN Appl 1][Deactivate, Activate,
DeleteSelf: ADM1] – see note below
    fillFileContent :
'8001019000800102A010A406830101950108A40683010
2950108800158A40683010A950108'H,
-- Rule 16: [Read: Always][Update/CreateEF:
PIN Appl 1 & ADM 1][Deactivate, Activate,
DeleteSelf: ADM1]
    fillFileContent :
'8001019000800102AF10A406830101950108A40683010
A950108800158A40683010A950108'H,
-- create UMPC
```

```
      createFCP : {
        fileDescriptor '4121'H,
        fileID '2F08'H,
        securityAttributesReferenced '0A'H,
        efFileSize '05'H
      }
    }
  }
}
```

6.12.1.3. PE PUKCodes

PE_PUKCodes

```
pukVal ProfileElement ::= pukCodes : {
  -- PUK PE needs be right after the MF
  puk-Header {
    mandated NULL,
    identification 2
  },
  pukCodes {
    {
      keyReference pukAppl1,
      pukValue '3132333435363738'H,
      -- maxNumOfAttemps:9, retryNumLeft:9
      maxNumOfAttemps-retryNumLeft 153
    },
    {
      keyReference pukAppl2,
      pukValue '3132333435363738'H
    },
    {
      keyReference secondPUKAppl1,
      pukValue '3132333435363738'H,
      -- maxNumOfAttemps:8, retryNumLeft:8
      maxNumOfAttemps-retryNumLeft 136
    }
  }
}
```

```
A33FA005 80008101 02A13630 11800101
81083132 33343536 37388202 0099300D
80010281 08313233 34353637 38301280
02008181 08313233 34353637 38820200
88
```

6.12.1.4. PE PINCodes

PE_PINCodes

```
pinVal ProfileElement ::= pinCodes : {
-- the PIN codes for global PINs have to be
created under the MF context and right after
the MF context
  pin-Header {
    mandated NULL,
    identification 3
  },
  pinCodes pinconfig : {
    {
      keyReference pinAppl1,
      pinValue '3132333435363738'H,
      unblockingPINReference pukAppl1
```

```
A251A005 80008101 03A148A0 46301080
01018108 31323334 35363738 82010130
0D800102 81083132 33343536 3738300D
80010A81 08313233 34353637 38301480
02008181 08313233 34353637 38830101
840122
```

```
    },
    {
      keyReference pinAppl2,
      pinValue '3132333435363738'H
    },
    {
      keyReference adm1,
      pinValue '3132333435363738'H
    },
    {
      keyReference secondPINAppl1,
      pinValue '3132333435363738'H,
      -- PIN is enabled
      pinAttributes 1,
      -- maxNumOfAttemps:2, retryNumLeft:2
      maxNumOfAttemps-retryNumLeft 34
    }
  }
}
```

6.12.1.5. PE USIM

6.12.1.5.1. PE USIM by Template

PE_USIM (Template)

```
usimValue ProfileElement ::= usim : {
  usim-header {
    mandated NULL,
    identification 4
  },
  templateID { 2 23 143 1 2 4 },
  adf-usim {
     fileDescriptor : {
        fileID '7FF1'H,
        dfName
'A0000000871002FF33FF018900000100'H,
        pinStatusTemplateDO '01810A'H
     }
  },
  ef-imsi {
    -- numerical format: 234101943787656
    fillFileContent : '082943019134876765'H
  },
  ef-arr {
    fileDescriptor : {
         linkPath '2F06'H
    }
   },
  ef-ust {
    fileDescriptor : {
      fileDescriptor '4121'H,
      efFileSize '0E'H  -- plus one byte
    },
    -- Service Dialling Numbers, Short Message
Storage
    fillFileContent :
'0A2E178CE7320400000000000000'H
```

```
B38185A0 05800081 01048106 67810F01
0204A21D A11B8302 7FF18410 A0000000
871002FF 33FF0189 00000100 C6030181
0AA30B83 09082943 01913487 6765A406
A104C702 2F06A819 A1078202 41218001
0E830E0A 2E178CE7 32040000 00000000
00AD0E83 0C025349 4D616C6C 69616E63
65AE0383 0100B204 83020040 B6068304
19F1FF01 B8028000
```

```
  },
  ef-spn {
    -- ASCII format: "SIMalliance"
    fillFileContent :
'0253494D616C6C69616E6365'H
  },
  ef-est {
    -- Services deactivated
    fillFileContent : '00'H
  },
  ef-acc {
    -- Access class 4
    fillFileContent : '0040'H
  },
  ef-ecc {
    -- Emergency Call Code 911
    fillFileContent : '19F1FF01'H
  }
  ef-epsloci {
    -- do not create EF EPSLOCI
    doNotCreate NULL
  }

}
```

6.12.1.5.2. PE USIM by Generic File Management

PE_USIM (Generic File Management)

```
altUsimValue ProfileElement ::=
genericFileManagement : {
  gfm-header {
    mandated NULL,
    identification 4
  },
  fileManagementCMD {
    {
-- ADF_USIM
      createFCP : {
        fileDescriptor '7821'H,
        fileID '7FF1'H,
        dfName
'A0000000871002FF33FF018900000100'H,
        securityAttributesReferenced '0A'H,
        pinStatusTemplateDO '01810A'H
      },

-- EF_IMSI
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F07'H,
        securityAttributesReferenced '02'H,
        efFileSize '09'H,
        shortEFID '38'H
      },
```

```
A182029E A0058000 810104A1 82029330
82028F62 22820278 2183027F F18410A0
00000087 1002FF33 FF018900 0001008B
010AC603 01810A62 11820241 2183026F
078B0102 80010988 01388109 08294301
91348767 65621482 04422100 2583026F
068B010A 8801B8C7 022F0662 1A820241
2183026F 088B0105 80012188 0140A507
C00180C1 0207FF62 1A820241 2183026F
098B0105 80012188 0148A507 C00180C1
0207FF62 16820241 2183026F 318B0102
80010188 0190A503 C1010A62 11820241
2183026F 388B0102 80010E88 0120810D
0A2E178C E7320400 00000000 00621982
04422100 1A83026F 3B8B0108 80020208
8800A504 C10200FF 62198204 422100B0
83026F3C 8B010580 0206E088 00A504C1
0200FF62 12820442 21002683 026F428B
01058001 26880062 15820241 2183026F
438B0105 80010288 00A503C0 01806212
82024121 83026F46 8B036F06 0A800111
8800810C 0253494D 616C6C69 616E6365
62118202 41218302 6F568B01 08800101
88012881 0100621B 82024121 83026F5B
8B010580 01068801 78A508C0 0180C203
```

```
    -- provide content for EF_IMSI
    -- numerical format: 234101943787656
      fillFileContent : '082943019134876765'H,

-- EF_ARR Link
      createFCP : {
        fileDescriptor '42210025'H,
        fileID '6F06'H,
        securityAttributesReferenced '0A'H,
        shortEFID 'B8'H,
        linkPath '2F06'H
      },

-- EF_Keys
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F08'H,
        securityAttributesReferenced '05'H,
        efFileSize '21'H,
        shortEFID '40'H,
        proprietaryEFInfo {
          specialFileInformation '80'H,
          fillPattern '07FF'H
        }
      },

-- EF_KeysPS
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F09'H,
        securityAttributesReferenced '05'H,
        efFileSize '21'H,
        shortEFID '48'H,
        proprietaryEFInfo {
          specialFileInformation '80'H,
          fillPattern '07FF'H
        }
      },

-- EF_HPPLMN
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F31'H,
        securityAttributesReferenced '02'H,
        efFileSize '01'H,
        shortEFID '90'H,
        proprietaryEFInfo {
-- specialFileInformation with Default value
          specialFileInformation '00'H,
          fillPattern '0A'H
        }
      },

-- EF_UST
      createFCP : {
```

```
F0000062 16820241 2183026F 5C8B0102
80010388 0180A503 C0018062 16820241
2183026F 738B0105 80010E88 0160A503
C0018002 01078107 00F11000 00FF0162
11820241 2183026F 788B0102 80010288
01308102 00406211 82024121 83026F7B
8B010580 010C8801 68621682 02412183
026F7E8B 01058001 0B880158 A503C001
80020107 81040000 FF016216 82024121
83026FAD 8B010A80 01048801 18A503C1
01000201 03810102 62138204 42210004
83026FB7 8B010A80 01048801 08810419
F1FF0162 15820241 2183026F C48B0105
80018088 00A503C0 01806216 82024121
83026FE3 8B010580 01128801 F0A503C0
01800201 0F810300 00016216 82024121
83026FE4 8B010580 01508801 C0A503C0
0180
```

```
        fileDescriptor '4121'H,
        fileID '6F38'H,
        securityAttributesReferenced '02'H,
        efFileSize '0E'H,
        shortEFID '20'H
      },
      -- provide UST settings
      -- Service Dialling Numbers, Short
Message Storage
      fillFileContent :
'0A2E178CE73204000000000000'H,

-- EF_FDN
      createFCP : {
        fileDescriptor '4221001A'H,
        fileID '6F3B'H,
        securityAttributesReferenced '08'H,
        efFileSize '0208'H,
        shortEFID ''H,
        proprietaryEFInfo {
          fillPattern '00FF'H
        }
      },

-- EF_SMS
      createFCP : {
        fileDescriptor '422100B0'H,
        fileID '6F3C'H,
        securityAttributesReferenced '05'H,
        efFileSize '06E0'H,
        shortEFID ''H,
        proprietaryEFInfo {
          fillPattern '00FF'H
        }
      },

-- EF_SMSP
      createFCP : {
        fileDescriptor '42210026'H,
        fileID '6F42'H,
        securityAttributesReferenced '05'H,
        efFileSize '26'H,
        shortEFID ''H

      },

-- EF_SMSS
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F43'H,
        securityAttributesReferenced '05'H,
        efFileSize '02'H,
        shortEFID ''H,
        proprietaryEFInfo {
          specialFileInformation '80'H
```

```
        }
      },

-- EF_SPN
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F46'H,
-- provide the full access rule including
EF_ARR File ID
        securityAttributesReferenced
'6F060A'H,
        efFileSize '11'H,
        shortEFID ''H
      },
      -- ASCII format: "SIMalliance"
      fillFileContent :
'0253494D616C6C69616E6365'H,

-- EF_EST
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F56'H,
        securityAttributesReferenced '08'H,
        efFileSize '01'H,
        shortEFID '28'H
      },
      -- EST Services deactivated
      fillFileContent : '00'H,

-- EF_START-HFN
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F5B'H,
        securityAttributesReferenced '05'H,
        efFileSize '06'H,
        shortEFID '78'H,
        proprietaryEFInfo {
          specialFileInformation '80'H,
-- ues of repeat pattern to initialize the
content
          repeatPattern 'F00000'H
        }
      },

-- EF_THRESHOLD
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F5C'H,
        securityAttributesReferenced '02'H,
        efFileSize '03'H,
        shortEFID '80'H,
        proprietaryEFInfo {
          specialFileInformation '80'H
        }
      },
```

```
-- EF_PSLOCI
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F73'H,
        securityAttributesReferenced '05'H,
        efFileSize '0E'H,
        shortEFID '60'H,
        proprietaryEFInfo  {
          specialFileInformation '80'H
        }
      },
      -- Initialize PSLOCI
      fillFileOffset : 7,
      fillFileContent : '00F1100000FF01'H,

-- EF_ACC
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F78'H,
        securityAttributesReferenced '02'H,
        efFileSize '02'H,
        shortEFID '30'H
      },
      -- Provide Content for ACC
      -- Access class 4
      fillFileContent : '0040'H,

-- EF_FPLMN
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F7B'H,
        securityAttributesReferenced '05'H,
        efFileSize '0C'H,
        shortEFID '68'H
      },

-- EF_LOCI
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6F7E'H,
        securityAttributesReferenced '05'H,
        efFileSize '0B'H,
        shortEFID '58'H,
        proprietaryEFInfo {
          specialFileInformation '80'H
        }
      },
      -- Initialize LOCI
      fillFileOffset : 7,
      fillFileContent : '0000FF01'H,

-- EF_AD
      createFCP : {
        fileDescriptor '4121'H,
```

```
        fileID '6FAD'H,
        securityAttributesReferenced '0A'H,
        efFileSize '04'H,
        shortEFID '18'H,
        proprietaryEFInfo {
-- use of fillPattern in Combination with
fillFileContent (not efficient in this
example)
          fillPattern '00'H
        }
      },
      -- Initialize AD
      fillFileOffset : 3,
      fillFileContent : '02'H,

-- EF_ECC
      createFCP : {
        fileDescriptor '42210004'H,
        fileID '6FB7'H,
        securityAttributesReferenced '0A'H,
        efFileSize '04'H,
        shortEFID '08'H
      },
      -- Initialize ECC
      -- Emergency Call Code 911
      fillFileContent : '19F1FF01'H,

-- EF_NETPAR
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6FC4'H,
        securityAttributesReferenced '05'H,
        efFileSize '80'H,
        shortEFID ''H,
        proprietaryEFInfo {
          specialFileInformation '80'H
        }
      },

-- EF_EPSLOCI
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6FE3'H,
        securityAttributesReferenced '05'H,
        efFileSize '12'H,
        shortEFID 'F0'H,
        proprietaryEFInfo {
          specialFileInformation '80'H
        }
      },

      -- Initialize EF_EPSLOCI
      fillFileOffset : 15,
      fillFileContent : '000001'H,
```

```
-- EF_EPSNSC
      createFCP : {
        fileDescriptor '4121'H,
        fileID '6FE4'H,
        securityAttributesReferenced '05'H,
        efFileSize '50'H,
        shortEFID 'C0'H,
        proprietaryEFInfo {
          specialFileInformation '80'H
        }
      }
    }
  }
}
```

6.12.1.5.3. PE OPT-USIM (Template)

PE_OPT-USIM (Template)

```
usimValue ProfileElement ::= opt-usim : {
  optusim-header {
    mandated NULL,
    identification 6
  },
  templateID { 2 23 143 1 2 5 },
    ef-li {
  },
  ef-ext5 {
   },
  ef-ext4 {
   },
  ef-ipd {
    fileDescriptor : {
      fileDescriptor '42210004'H,
      efFileSize '10'H,
      }
   }
}
```

```
B423A005 80008101 06810667 810F0102 05A200B6
00B800BF 4B0BA109 82044221 00048001 10
```

6.12.1.5.4. PE OPT-USIM by Generic File Management

PE_OPT-USIM (Generic File Management)

```
gmOptUsimValue ProfileElement ::=
genericFileManagement : {
  gfm-header {
    mandated NULL,
    identification 6
  },

  fileManagementCMD {
  {
-- ef-li
    createFCP : {
```

```
A15AA005 80008101 06A15130 4F621182 02412183
026F058B 01018001 06880110 62128204 4221000D
83026F4E 8B010580 01828800 62128204 4221000D
83026F55 8B010880 01688800 62128204 42210004
83026FF2 8B010380 01108800
```

```
      fileDescriptor '4121'H,
      fileID '6F05'H,
      securityAttributesReferenced '01'H,
      efFileSize '06'H,
      shortEFID '10'H
    },
--   ef-ext5
    createFCP : {
      fileDescriptor '4221000D'H,
      fileID '6F4E'H,
      securityAttributesReferenced '05'H,
      efFileSize '82'H,
      shortEFID ''H
     },
--   ef-ext4
    createFCP : {
      fileDescriptor '4221000D'H,
      fileID '6F55'H,
      securityAttributesReferenced '08'H,
      efFileSize '68'H,
      shortEFID ''H
     },
--   ef-ipd
    createFCP : {
      fileDescriptor '42210004'H,
      fileID '6FF2'H,
      securityAttributesReferenced '03'H,
      efFileSize '10'H,
      shortEFID ''H
     }
  }
 }
}
```

6.12.1.6. PE PINCodes (Local PIN)

PE_Local_PIN_Value	

localPinValue ProfileElement ::= pinCodes : {
  pin-Header {
  mandated NULL,
  identification 20
  },
  pinCodes pinconfig : {
  {
  keyReference secondPINAppl1,
  pinValue '0101010101010101'H,
  pinAttributes 1,
  maxNumOfAttemps-retryNumLeft 34
  }
  }
}
	A221A005 80008101 14A118A0 16301480
	02008181 08010101 01010101 01830101
	840122

6.12.1.7. PE AKA Parameter

PE_AKA_Parameters

```
akaMilenage ProfileElement ::= akaParameter :
{
   aka-header {
      mandated NULL,
      identification 30
      },
   algoConfiguration algoParameter : {
      algorithmID milenage,
      algorithmOptions '01'H,      -- RES and
MAC 64 bits, CK and IK 128 bits

      key '000102030405060708090A0B0C0D0E0F'H,
      opc '0102030405060708090A0B0C0D0E0F00'H,
--    rotationConstants  uses default:
'4000204060'H,
--    xoringConstants    uses default:
'00000000000000000000000000000000100000000000000
0020000000000000004000000000000008'H,
      authCounterMax '010203'H
   }
-- sqnOptions uses default: '02'H, i.e.
Anonymity key used, SQN wrap around not
allowed
-- sqnDelta   uses default: '000010000000'H
-- sqnAgeLimit  uses default: '000010000000'H
-- sqnInit:     uses default: all bytes zero
}
```

```
A43AA005 80008101 1EA131A1 2F800101
81010182 10000102 03040506 0708090A
0B0C0D0E 0F831001 02030405 06070809
0A0B0C0D 0E0F0086 03010203
```

6.12.1.8. PE SecurityDomain (MNO SD)

PE_SecurityDomain_MNO_SD

```
mnoSdValue ProfileElement ::= securityDomain :
{
  sd-Header {
    mandated NULL,
    identification 40
  },
  instance {
    applicationLoadPackageAID
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID 'A000000151000000'H,
    applicationPrivileges '82FC80'H,
    -- Secured
    lifeCycleState '0F'H,
    -- SCP80 supported acc. UICC Config.
    applicationSpecificParametersC9
'810280008201F08701F0'H,
    -- other parameters may be necessary
    applicationParameters {
      -- TAR: B20100, MSL: 12

uiccToolkitApplicationSpecificParametersField
         '01000001000000002011203B2010000'H
    }
  },
  keyList {
    {
```

```
A681BBA0 05800081 0128A144 4F07A000
00015153 504F08A0 00000151 5350414F
08A00000 01510000 00820382 FC808301
0FC90A81 02800082 01F08701 F0EA1180
0F010000 01000000 02011203 B2010000
A26C3022 95013882 01018301 01301730
15800180 86101122 33445566 77889910
11121314 15163022 95013482 01028301
01301730 15800180 86101122 33445566
77889910 11121314 15163022 9501C882
01038301 01301730 15800180 86101122
33445566 77889910 11121314 1516
```

```
      -- C-ENC + R-ENC
      keyUsageQualifier '38'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- ENC key
      keyIdentifier '01'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-MAC + R-MAC
      keyUsageQualifier '34'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- MAC key
      keyIdentifier '02'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-DEK + R-DEK
      keyUsageQualifier 'C8'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- data ENC key
      keyIdentifier '03'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    }
  }
}
```

6.12.1.9. PE Security Domain (SSD)

PE_SecurityDomain_SSD	

ssdValue ProfileElement ::= securityDomain : {
  sd-Header {
  mandated NULL,
  identification 41
  },
  instance {
  applicationLoadPackageAID
	A681C0A0 05800081 0129A149 4F07A000
	00015153 504F08A0 00000151 5350414F
	10A00000 05591010 01027364 56616C75
	65820380 80008301 0FC90781 02800082
	01F0EA11 800F0100 00010000 00020112

```
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID
'A00000055910100102736456616C7565'H,
    applicationPrivileges '808000'H,
    lifeCycleState '0F'H,
    applicationSpecificParametersC9
'810280008201F0'H,
    applicationParameters {

uiccToolkitApplicationSpecificParametersField
'0100000100000020112036C756500'H
    }
  },
  keyList {
    {
      keyUsageQualifier '38'H,
      keyAccess '00'H,
      keyIdentifier '01'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          keyType '80'H,
          keyData
'11223344556677881122334455667788'H
        }
      }
    },
    {
      keyUsageQualifier '34'H,
      -- keyAccess '00'H,
      keyIdentifier '02'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          keyType '80'H,
          keyData
'11223344556677881122334455667788'H
        }
      }
    },
    {
      keyUsageQualifier 'C8'H,
      keyAccess '00'H,
      keyIdentifier '03'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          keyType '80'H,
          keyData
'11223344556677881122334455667788'H
        }
      }
    }
  }
}
```

```
036C7565 00A26C30 22950138 82010183
01013017 30158001 80861011 22334455
66778811 22334455 66778830 22950134
82010283 01013017 30158001 80861011
22334455 66778811 22334455 66778830
229501C8 82010383 01013017 30158001
80861011 22334455 66778811 22334455
667788
```

6.12.1.10. PE Application

PE_Application

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 50
  },
  loadBlock {
    loadPackageAID 'A000000559101001'H,
      -- Java file for the applet1 in [GS RPAT
Annex A1]
    loadBlockObject
'01002EDECAFFED020204000108A0000005591010011B6
36F6D2F67736D612F65756963632F746573742F6170706
C657431020021002E0021000F003B002A00210066000A0
00E0000008A040F000000000000004010004003B0403010
7A000000062010100011 0A0000000090005FFFFFFFF891
2000000010110A0000000871005FFFFFFFF89132000000
00107A000000062000103000F010BA0000005591010011
122330008060021000044800300FF000504000000033FFF
F00300040810700820000800200810801080700660001 1
0188C00007A04328F00013D8C00022E181D25290416046
1081B8B0003700C1B181D044116048B00041B8C00057A0
0207A02301E046B0719670418770177022 11D750016000
1000200098D00062D1A048E0200071770027A02108D000
8058E020009007A08000A0000000000000000000005002
A000A0680030001000200060000010380030103800302 0
600005A06810F000181040006811000018109000 9000E0
000000A0506040E0C0420070905'H
  },
  instanceList {
    {
      applicationLoadPackageAID
'A000000559101001'H,
      classAID 'A0000005591010011 12233'H,
      instanceAID 'A0000005591010011 1223301'H,
      applicationPrivileges '000000'H,
      -- Selectable
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H,
      applicationParameters {

uiccToolkitApplicationSpecificParametersField
          -- TAR: 112233
          '0100000000000311223300'H
      }
    }
  }
}
```

```
A88201D9 A0058000 810132A1 82018B4F
08A00000 05591010 01C48201 7D01002E
DECAFFED 02020400 0108A000 00055910
10011B63 6F6D2F67 736D612F 65756963
632F7465 73742F61 70706C65 74310200
21002E00 21000F00 3B002A00 21006600
0A000E00 00008A04 0F000000 00000004
01000400 3B040301 07A00000 00620101
000110A0 00000009 0005FFFF FFFF8912
00000001 0110A000 00008710 05FFFFFF
FF891320 00000001 07A00000 00620001
03000F01 0BA00000 05591010 01112233
00080600 21000044 800300FF 00050400
000033FF FF003000 40810700 82000080
02008108 01080700 66000110 188C0000
7A04328F 00013D8C 00022E18 1D252904
16046108 1B8B0003 700C1B18 1D044116
048B0004 1B8C0005 7A00207A 02301E04
6B071967 04187701 7702211D 75001600
01000200 098D0006 2D1A048E 02000717
70027A02 108D0008 058E0200 09007A08
000A0000 00000000 00000000 05002A00
0A068003 00010002 00060000 01038003
01038003 02060000 5A06810F 00018104
00068110 00018109 0009000E 0000000A
0506040E 0C042007 0905A241 303F4F08
A0000005 59101001 4F0BA000 00055910
10011122 334F0CA0 00000559 10100111
22330182 03000000 830107C9 0100EA0D
800B0100 00000000 03112233 00
```

6.12.1.11. PE RFM

PE_RFM

```
rfmValue ProfileElement ::= rfm : {
  rfm-header {
    mandated NULL,
    identification 60
  },
  instanceAID 'A00000055910100001'H,
  tarList {
    'B00000'H
  },
  minimumSecurityLevel '12'H,
  uiccAccessDomain '00'H,
  uiccAdminAccessDomain '00'H
  adfRFMAccess {
    adfAID
'A0000000871002FF33FF018900000100'H,
    adfAccessDomain '00'H,
    adfAdminAccessDomain '00'H
  }
}
```

```
A73CA005 80008101 3C4F09A0 00000559
10100001 A0050403 B0000081 01120401
00040100 30188010 A0000000 871002FF
33FF0189 00000100 81010082 0100
```

6.12.1.12. PE End

PE_END

```
endValue ProfileElement ::= end : {
  end-header {
    mandated NULL,
    identification 99
  }
}
```

```
AA07A005 80008101 63
```

6.12.2 Customised PEs

The content of the Customised PEs is based on the content of the Basic Profile Package PEs and is modified according to the testing needs.

6.12.2.1. PE Security Domain

6.12.2.1.1. PE SecurityDomain (MNO_SD1)

This PE provides only the mandatory objects a PE Security Domain has to contain.

Compared to the Basic Profile Package PE Security Domain definition defined in 6.12.1.8, all optional definitions are removed.

PE_SecurityDomain_MNO_SD1

```
mnoSdValue ProfileElement ::= securityDomain :
{
  sd-Header {
    mandated NULL,
    identification 3
  },
  instance {
    applicationLoadPackageAID
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID 'A000000151000000'H,
    applicationPrivileges '82FC80'H,
    -- Secured
    lifeCycleState '0F'H,
    -- no SCP defined
    applicationSpecificParametersC9 '00'H,
  }
}
```

```
A631A005 80008101 03A1284F 07A00000
01515350 4F08A000 00015153 50414F08
A0000001 51000000 820382FC 8083010F
C90100
```

6.12.2.1.2. PE SecurityDomain (MNO SD2)

Compared to the Basic Profile Package PE Security Domain definition defined in 6.12.1.8, the first key of the keylist contains two key components definitions.

PE_ SecurityDomain_MNO_SD2

```
mnoSdValue ProfileElement ::= securityDomain :
{
  sd-Header {
    mandated NULL,
    identification 40
  },
  instance {
    applicationLoadPackageAID
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID 'A000000151000000'H,
    applicationPrivileges '82FC80'H,
    -- Secured
    lifeCycleState '0F'H,
    -- SCP80 supported acc. UICC Config.
    applicationSpecificParametersC9
'810280008201F08701F0'H,
    -- other parameters may be necessary
    applicationParameters {
      -- TAR: B20100, MSL: 12

uiccToolkitApplicationSpecificParametersField
        '01000001000000002011203B2010000'H
    }
  },
  keyList {
    {
      -- C-ENC + R-ENC
      keyUsageQualifier '38'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- ENC key
      keyIdentifier '01'H,
      keyVersionNumber '01'H,
```

```
A681D3A0 05800081 0128A144 4F07A000
00015153 504F08A0 00000151 5350414F
08A00000 01510000 00820382 FC808301
0FC90A81 02800082 01F08701 F0EA1180
0F010000 01000000 02011203 B2010000
A2818330 39950138 82010183 0101302E
30158001 80861011 22334455 66778899
10111213 14151630 15800180 86101122
33445566 77889910 11121314 15163022
95013482 01028301 01301730 15800180
86101122 33445566 77889910 11121314
15163022 9501C882 01038301 01301730
15800180 86101122 33445566 77889910
11121314 1516
```

```
      keyCompontents {
        {


          -- DES mode implicitly known
          keyType '80'H,
          keyData
'11223344556677889910111213141516'H
        },
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'11223344556677889910111213141516'H
        }
      }
    },
    {
      -- C-MAC + R-MAC
      keyUsageQualifier '34'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- MAC key
      keyIdentifier '02'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'11223344556677889910111213141516'H
        }
      }
    },
    {
      -- C-DEK + R-DEK
      keyUsageQualifier 'C8'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- data ENC key
      keyIdentifier '03'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'11223344556677889910111213141516'H
        }
      }
    }
  }
}
```

6.12.2.1.3. PE SecurityDomain (MNO SD3)

Compared to the Basic Profile Package PE Security Domain definition defined in 6.12.1.8, the PE SD contains the sdPerso Data definition.

PE_SecurityDomain_MNO_SD3	
`mnoSdValue ProfileElement ::= securityDomain : {`	`A681D7A0 05800081 0128A144 4F07A000 00015153 504F08A0 00000151 5350414F`

```
  sd-Header {
    mandated NULL,
    identification 40
  },
  instance {
    applicationLoadPackageAID
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID 'A000000151000000'H,
    applicationPrivileges '82FC80'H,
    -- Secured
    lifeCycleState '0F'H,
    -- SCP80 supported acc. UICC Config.
    applicationSpecificParametersC9
'810280008201F08701F0'H,
    -- other parameters may be necessary
    applicationParameters {
      -- TAR: B20100, MSL: 12

uiccToolkitApplicationSpecificParametersField
         '01000001000000002011203B2010000'H
    }
  },
  keyList {
    {
      -- C-ENC + R-ENC
      keyUsageQualifier '38'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- ENC key
      keyIdentifier '01'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-MAC + R-MAC
      keyUsageQualifier '34'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- MAC key
      keyIdentifier '02'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-DEK + R-DEK
      keyUsageQualifier 'C8'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- data ENC key
      keyIdentifier '03'H,
```

```
08A00000 01510000 00820382 FC808301
0FC90A81 02800082 01F08701 F0EA1180
0F010000 01000000 02011203 B2010000
A26C3022 95013882 01018301 01301730
15800180 86101122 33445566 77889910
11121314 15163022 95013482 01028301
01301730 15800180 86101122 33445566
77889910 11121314 15163022 9501C882
01038301 01301730 15800180 86101122
33445566 77889910 11121314 1516A31A
040B0070 08420601 02030405 06040B00
70084506 06050403 0201
```

```
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'11223344556677889910111213141516'H
        }
      }
    }
  },
      -- IIN and CIN
  sdPersoData {
    '0070084206010203040506'H,
    '0070084506060504030201'H
  }
}
```

6.12.2.1.4. PE SecurityDomain (MNO SD4)

Compared to the Basic Profile Package PE Security Domain definition defined in 6.12.1.8, the instance definition is extended by the processData definition containing HTTPs configuration data.

PE_SecurityDomain_MNO_SD4

```
mnoSdValue ProfileElement ::= securityDomain :
{
  sd-Header {
    mandated NULL,
    identification 40
  },
  instance {
    applicationLoadPackageAID
'A0000001515350'H,
    classAID 'A000000151535041'H,
    instanceAID 'A000000151000000'H,
    applicationPrivileges '82FC80'H,
    -- Secured
    lifeCycleState '0F'H,
    -- SCP80 supported acc. UICC Config.
    applicationSpecificParametersC9
'8102800081028101'H,
    -- other parameters may be necessary
    applicationParameters {
      -- TAR: B20100, MSL: 12

uiccToolkitApplicationSpecificParametersField
         '0100000100000002011203B2010000'H
    },
      -- HTTP Configuration according Amend.B
     processData{

'80E21000428581AB84243507020000030000023902057
8470947534D4165554943433C03021F413E05217F00000
1850A0650534B49443102400189778A096C6F63616C686
F7374'H,

'80E290016C8B582F2F73652D69642F6569642F3030363
3363835363030303030303030303030303030303030303
030303737373B2F2F61612D69642F6169642F413030303
030303031382F343334443038303930413042304330303
03030308C102F67736D612F61646D696E6167656E74'H
    }
```

```
A68201E8 A0058000 810128A1 8201014F
07A00000 01515350 4F08A000 00015153
50414F08 A0000001 51000000 820382FC
8083010F C9088102 80008102 8101EA11
800F0100 00010000 00020112 03B20100
003081BC 044780E2 10004285 81AB8424
35070200 00030000 02390205 78470947
534D4165 55494343 3C03021F 413E0521
7F000001 850A0650 534B4944 31024001
89778A09 6C6F6361 6C686F73 74047180
E290016C 8B582F2F 73652D69 642F6569
642F3030 36333638 35363030 30303030
30303030 30303030 30303030 30303037
37373B2F 2F61612D 69642F61 69642F41
30303030 30303031 382F3433 34443038
30393041 30423043 30303030 30308C10
2F67736D 612F6164 6D696E61 67656E74
A281D930 29950138 82010183 01018505
00000000 00301730 15800180 86101122
33445566 77889910 11121314 15163029
95013482 01028301 01850500 00000000
30173015 80018086 10112233 44556677
88991011 12131415 16302995 01C88201
03830101 85050000 00000030 17301580
01808610 11223344 55667788 99101112
13141516 30329501 3C820101 83014030
27302580 01858620 F0C0FAAC 0EF1364A
3E5EB422 9CF797A3 752CD0C8 27784457
6B3E05D5 05A03F21 30229501 C8820102
83014030 17301580 01808610 11223344
55667788 99101112 13141516
```

```
  },
  keyList {
    {
      -- C-ENC + R-ENC
      keyUsageQualifier '38'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- ENC key
      keyIdentifier '01'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-MAC + R-MAC
      keyUsageQualifier '34'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- MAC key
      keyIdentifier '02'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
    {
      -- C-DEK + R-DEK
      keyUsageQualifier 'C8'H,
      -- may be used by SD and application
      keyAccess '00'H,
      -- data ENC key
      keyIdentifier '03'H,
      keyVersionNumber '01'H,
      keyCompontents {
        {
          -- DES mode implicitly known
          keyType '80'H,
          keyData
'112233445566778899101112131415 16'H
        }
      }
    },
{
      -- PSK
      keyUsageQualifier '3C'H,
      -- may be used by SD and application
      keyAccess '00'H,
      keyIdentifier '01'H,
      keyVersionNumber '40'H,
      keyCompontents {
        {
          -- PSK
```

```
          keyType '85'H,
          keyData
'F0C0FAAC0EF1364A3E5EB4229CF797A3752CD0C827784
4576B3E05D505A03F21'H
        }
      }
    },
    {
      keyUsageQualifier 'C8'H,
      keyAccess '00'H,
      keyIdentifier '02'H,
      keyVersionNumber '40'H,
      keyCompontents {
        {
          keyType '80'H,
          keyData
'1122334455667788991011213141516'H,
        }
      }
    }
  }
}
```

6.12.2.2. PE Application

6.12.2.2.1. void

6.12.2.2.2. PE Application 2

PE_APPLICATION_2

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 21
  },
  loadBlock {
    loadPackageAID 'A000000559101001'H,
    securityDomainAID 'A000000151000000'H,
    nonVolatileCodeLimitC6 '0000'H,
    volatileDataLimitC7 'FFFF'H,
    nonVolatileDataLimitC8 '0000'H,
      -- Java file for the applet1 in [GS RPAT
Annex A1]
    loadBlockObject
'01002EDECAFFED020204000108A0000005591010011B63
6F6D2F67736D612F65756963632F746573742F6170706C6
57431020021002E0021000F003B002A00210066000A000E
0000008A040F000000000000004010004003B04030107A00
000006201010001l0A0000000090005FFFFFFFF89120000
00010110A0000000871005FFFFFFFF8913200000000107A
0000000620001030O0F010BA0000005591010011122330O
0806002100004480O300FF00050400000033FFFF0030004
0810700820000800200810801080700660001101l8C0000
7A04328F00013D8C00022E181D252904160461081B8B000
3700C1B181D044116048B00041B8C00057A00207A02301E
046B0719670418770177022llD75001600010002000980D0
0062D1A048E0200071770027A02108D0008058E02000900
7A08000A000000000000000000000005002A000A068003000
1000200060000010380030103800302060000SA06810F00
0181040006811000018109000900OE0000000A0506040E0
C0420070905'H
  },
  instanceList {
    {
      applicationLoadPackageAID
'A000000559101001'H,
      classAID 'A000000559101002112233'H,
      instanceAID 'A00000055910100211223301'H,
      applicationPrivileges '000000'H,
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H,
      applicationParameters {

uiccToolkitApplicationSpecificParametersField
          '0100000000000311223300'H
      }
    }
  }
}
```

```
A88201EF A0058000 810115A1 8201A14F
08A00000 05591010 014F08A0 00000151
000000C6 020000C7 02FFFFC8 020000C4
82017D01 002EDECA FFED0202 04000108
A0000005 59101001 1B636F6D 2F67736D
612F6575 6963632F 74657374 2F617070
6C657431 02002100 2E002100 0F003B00
2A002100 66000A00 0E000000 8A040F00
00000000 00040100 04003B04 030107A0
00000062 01010001 10A00000 00090005
FFFFFFFF 89120000 00010110 A0000000
871005FF FFFFFF89 13200000 000107A0
00000062 00010300 0F010BA0 00000559
10100111 22330008 06002100 00448003
00FF0005 04000000 33FFFF00 30004081
07008200 00800200 81080108 07006600
0110188C 00007A04 328F0001 3D8C0002
2E181D25 29041604 61081B8B 0003700C
1B181D04 4116048B 00041B8C 00057A00
207A0230 1E046B07 19670418 77017702
211D7500 16000100 0200098D 00062D1A
048E0200 07177002 7A02108D 0008058E
02000900 7A08000A 00000000 00000000
00000500 2A000A06 80030001 00020006
00000103 80030103 80030206 00005A06
810F0001 81040006 81100001 81090009
000E0000 000A0506 040E0C04 20070905
A241303F 4F08A000 00055910 10014F0B
A0000005 59101002 1122334F 0CA00000
05591010 02112233 01820300 00008301
07C90100 EA0D800B 01000000 00000311
223300
```

6.12.2.2.3. PE Application 3

PE_APPLICATION_3

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 22
  },
  loadBlock {
    loadPackageAID 'A000000559101001'H,
      -- Java file for the applet1 in [GS RPAT
Annex A1]
    loadBlockObject
'01002EDECAFFED020204000108A0000005591010011B63
6F6D2F67736D612F65756963632F746573742F6170706C6
57431020021002E0021000F003B002A00210066000A000E
0000008A040F000000000000004010004003B04030107A00
000006201010001l0A0000000090005FFFFFFFF89120000
00010110A0000000871005FFFFFFFF8913200000000107A
0000000620001030O0F010BA0000005591010011122330O
0806002100004480030OFF00050400000033FFFF0030004
0810700820000800200810801080700660001101l8C0000
7A04328F00013D8C00022E181D252904160461081B8B000
3700C1B181D044116048B00041B8C00057A00207A02301E
046B0719670418770177022llD75001600010002000980D0
0062D1A048E0200071770027A02108D0008058E02000900
7A08000A0000000000000000000005002A000A068003000
1000200060000010380030103800302060000SA06810F00
0181040006811000018109000900OE0000000A0506040E0
C0420070905'H
  },
  instanceList {
    {
      applicationLoadPackageAID
'A000000559101001'H,
      classAID 'A0000005591010031122233'H,
      instanceAID 'A0000005591010011322330l'H,
      extraditeSecurityDomainAID
'A000000151000000'H,
      applicationPrivileges '000000'H,
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H,
      systemSpecificParameters {
         volatileMemoryQuotaC7 '0000'H,
         nonVolatileMemoryQuotaC8 '0000'H,
         implicitSelectionParameter 'CF0180'H,
         volatileReservedMemory 'D7020000'H,
         nonVolatileReservedMemory 'D8020000'H
      },
      applicationParameters {

uiccToolkitApplicationSpecificParametersField
'0100000000000311223300'H,
      uiccAccessParams '810400010000'H,
uiccAdministrativeAccessApplicationSpecificPara
metersField '820400010000'H
      }
    }
  }
}
```

```
A882020E A0058000 810116A1 82018B4F
08A00000 05591010 01C48201 7D01002E
DECAFFED 02020400 0108A000 00055910
10011B63 6F6D2F67 736D612F 65756963
632F7465 73742F61 70706C65 74310200
21002E00 21000F00 3B002A00 21006600
0A000E00 00008A04 0F000000 00000004
01000400 3B040301 07A00000 00620101
000110A0 00000009 0005FFFF FFFF8912
00000001 0110A000 00008710 05FFFFFF
FF891320 00000001 07A00000 00620001
03000F01 0BA00000 05591010 01112233
00080600 21000044 800300FF 00050400
000033FF FF003000 40810700 82000080
02008108 01080700 66000110 188C0000
7A04328F 00013D8C 00022E18 1D252904
16046108 1B8B0003 700C1B18 1D044116
048B0004 1B8C0005 7A00207A 02301E04
6B071967 04187701 7702211D 75001600
01000200 098D0006 2D1A048E 02000717
70027A02 108D0008 058E0200 09007A08
000A0000 00000000 00000000 05002A00
0A068003 00010002 00060000 01038003
01038003 02060000 5A06810F 00018104
00068110 00018109 0009000E 0000000A
0506040E 0C042007 0905A276 30744F08
A0000005 59101001 4F0BA000 00055910
10031122 334F0CA0 00000559 10100113
2233014F 08A00000 01510000 00820300
00008301 07C90100 EF19C702 0000C802
0000CF03 CF0180D7 04D70200 00D804D8
020000EA 1D800B01 00000000 00031122
33008106 81040001 00008206 82040001
0000
```

6.12.2.2.4. PE Application 4

PE_APPLICATION_4

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 23
  },
  loadBlock {
    loadPackageAID 'A000000559101002'H,
      -- Java file based on the applet1 in [GS
RPAT Annex A1 with AID modified]
    loadBlockObject
'01002EDECAFFED020204000108A0000005591010011B63
6F6D2F67736D612F65756963632F746573742F6170706C6
57431020021002E0021000F003B002A00210066000A000E
0000008A040F000000000000004010004003B04030107A01
000006201010001l0A0000000090005FFFFFFFF89120000
00010110A0000000871005FFFFFFFF8913200000000107A
000000062000103000F010BA0000005591010011122330O
0806002100004480030OFF00050400000033FFFF0030004
08107008200008002008108010807006600011018BC0000
7A04328F00013D8C00022E181D252904160461081B8B000
3700C1B181D044116048B00041B8C00057A00207A02301E
046B071967041877017702211D75001600010002000980D0
0062D1A048E0200071770027A02108D0008058E02000900
7A08000A000000000000000000000005002A000A068003000
1000200060000010380030103800302060000SA06810F00
018104000681100001810900090O0E0000000A0506040E0
C0420070905'H
  }
}
```

```
A8820196 A0058000 810117A1 82018B4F
08A00000 05591010 01C48201 7D01002E
DECAFFED 02020400 0108A000 00055910
10011B63 6F6D2F67 736D612F 65756963
632F7465 73742F61 70706C65 74310200
21002E00 21000F00 3B002A00 21006600
0A000E00 00008A04 0F000000 00000004
01000400 3B040301 07A01000 00620101
000110A0 00000009 0005FFFF FFFF8912
00000001 0110A000 00008710 05FFFFFF
FF891320 00000001 07A00000 00620001
03000F01 0BA00000 05591010 01112233
00080600 21000044 800300FF 00050400
000033FF FF003000 40810700 82000080
02008108 01080700 66000110 188C0000
7A04328F 00013D8C 00022E18 1D252904
16046108 1B8B0003 700C1B18 1D044116
048B0004 1B8C0005 7A00207A 02301E04
6B071967 04187701 7702211D 75001600
01000200 098D0006 2D1A048E 02000717
70027A02 108D0008 058E0200 09007A08
000A0000 00000000 00000000 05002A00
0A068003 00010002 00060000 01038003
01038003 02060000 5A06810F 00018104
00068110 00018109 0009000E 0000000A
0506040E 0C042007 0905
```

6.12.2.2.5. PE Application 5

PE_APPLICATION_5

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 24
  },
  loadBlock {
    loadPackageAID 'A000000559101001'H,
      -- Java file for the applet1 in [GS RPAT
Annex A1]
    loadBlockObject
'01002EDECAFFED020204000108A0000005591010011B63
6F6D2F67736D612F65756963632F746573742F6170706C6
57431020021002E0021000F003B002A00210066000A000E
0000008A040F000000000000004010004003B04030107A00
000006201010001l0A0000000090005FFFFFFFF89120000
00010110A0000000871005FFFFFFFF8913200000000107A
000000062000103000F010BA0000005591010011122330O
0806002100004480030OFF00050400000033FFFF0030004
0810700820000800200810801080700660001101l8C0000
7A04328F00013D8C00022E181D252904160461081B8B000
3700C1B181D044116048B00041B8C00057A00207A02301E
046B071967041877017702211D75001600010002000980D0
0062D1A048E0200071770027A02108D0008058E02000900
7A08000A000000000000000000000005002A000A068003000
1000200060000010380030103800302060000SA06810F00
018104000681100001810900090000E0000000A0506040E0
C0420070905'H
  },
  instanceList {
    {
      applicationLoadPackageAID
'A000000559101001'H,
      classAID 'A0000005591010005445566'H,
      instanceAID 'A0000005591010054455660l'H,
      applicationPrivileges '000000'H,
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H
    }
    -- Second Instance
    {
      applicationLoadPackageAID
'A000000559101001'H,
      classAID 'A0000005591010005445566'H,
      instanceAID 'A000000559101005112233O2'H,
      applicationPrivileges '000000'H,
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H
    }
  }
}
```

```
A88201FC A0058000 810118A1 82018B4F
08A00000 05591010 01C48201 7D01002E
DECAFFED 02020400 0108A000 00055910
10011B63 6F6D2F67 736D612F 65756963
632F7465 73742F61 70706C65 74310200
21002E00 21000F00 3B002A00 21006600
0A000E00 00008A04 0F000000 00000004
01000400 3B040301 07A00000 00620101
000110A0 00000009 0005FFFF FFFF8912
00000001 0110A000 00008710 05FFFFFF
FF891320 00000001 07A00000 00620001
03000F01 0BA00000 05591010 01112233
00080600 21000044 800300FF 00050400
000033FF FF003000 40810700 82000080
02008108 01080700 66000110 188C0000
7A04328F 00013D8C 00022E18 1D252904
16046108 1B8B0003 700C1B18 1D044116
048B0004 1B8C0005 7A00207A 02301E04
6B071967 04187701 7702211D 75001600
01000200 098D0006 2D1A048E 02000717
70027A02 108D0008 058E0200 09007A08
000A0000 00000000 00000000 05002A00
0A068003 00010002 00060000 01038003
01038003 02060000 5A06810F 00018104
00068110 00018109 0009000E 0000000A
0506040E 0C042007 0905A264 30304F08
A0000005 59101001 4F0BA000 00055910
10054455 664F0CA0 00000559 10100544
55660182 03000000 830107C9 01003030
4F08A000 00055910 10014F0B A0000005
59101005 4455664F 0CA00000 05591010
05112233 02820300 00008301 07C90100
```

6.12.2.2.6. PE Application 6

PE_APPLICATION_6

```
appletValue ProfileElement ::= application : {
  app-Header {
    mandated NULL,
    identification 25
  },
  loadBlock {
    loadPackageAID 'A000000559101001'H,
    loadBlockObject
'010012DECAFFED01020400010800000055910100102001F0012001F000F0057006E00250161001A002C0000010300040001000B060100040057060401070A000000062010105011 0A0000000090005FFFFFFFF8912000000060110A0000000871005FFFFFFFF8913200000050106A00000015100060210A0000000090003FFFFFFFF891071000200010700000006200010300 0F010BA0000055591010011122330018060025458003020002050500000080FFFF0040008D00D581070082000080020081080108830801090701610002101 88C000218110100900B87D84C0085EB7DB9060018110100900B87017A04328F00033D8C00042E181D252904160461081B8B0005700C1B181D044116048B00061B8C00077A04220331188B000860037A198B00092E1B04257500120001FFCA0009181B038C000A317008116D008D000B198B000C3B191E08438B000D19081E08438B000E7A02301E046B07196704187701770523033 11D056B358D000F2E1B1B8B0010AD01031B8B00118B00123B18AD01038C000A318D001328041504AD01081E08438B00141504109F8B00157A02108D0016058E020017007A07621007290619088D0018290716077D00196B3919160625022FE4B03D557273B26B09160604415B2906160604415B29061916062510926B1C191606AD000319160604412505418D001A3B15041605033804781 16A808D000B037806310332191E05418D00181100926B23AD0003256015AD0003191E0841AD000425054 18D001A327010116A888D000B7008116A888D000B1F7808001A00040001000103000B48454C4C4F20574F524C44000000020070050 06E001B020000000200000106800300010000000600000103800301038003020600 00C90380030303800A01060001280680070103800A0703800A0903800A040684080003840 80D0384080E03840842CC07EF33E438D702068406000384061003840612068110000181090068010040500000206801001090 02C00090E08890C0E512F0607001F0516040E0C040907130904080819060703080 40D0706050D07300F0D1B09080B01030100010000050004000881078200800281088308000900000000038FF0A0000028004000200000000380102000001003800810001003A00150000000001090018003C00260000000007010040006400 3E0000000005010080006800 0B0000000080100 8D0044003A00000000FF0200C9003A000A000000009010 0D5006F005100000000FF020128004100370079AEC93548B8F1D3000000001B00380038003AFFFF003A003A003C003A003F00380041004400460044004800 4B004600460004F00530057005A005CFFFF00410046006001B0011004B431012003B4400241014003441005684080054B44400568406004B44102310568109006B4B444066800A10B68006368002006B44B44A2A5EC474DE8B2DE'H
  },
  instanceList {
    {
      applicationLoadPackageAID
A8820485 A0058000 810119A1 8204284F
08A00000 05591010 01C48204 1A010012
DECAFFED 01020400 0108A000 00055910
10010200 1F001200 1F000F00 57006E00
25016100 1A002C00 00010300 04000100
0B060100 04005706 040107A0 00000062
01010501 10A00000 00090005 FFFFFFFF
89120000 00060110 A0000000 871005FF
FFFFFF89 13200000 050106A0 00000151
00060210 A0000000 090003FF FFFFFF89
10710002 000107A0 00000062 00010300
0F010BA0 00000559 10100111 22330018
06002545 80030200 02050500 000080FF
FF004000 8D00D581 07008200 00800200
81080108 83080109 07016100 0210188C
00021811 0100900B 87D84C00 85EB7DB9
06001811 0100900B 87017A04 328F0003
3D8C0004 2E181D25 29041604 61081B8B
0005700C 1B181D04 4116048B 00061B8C
00077A04 22033118 8B000860 037A198B
00092E1B 04257500 120001FF CA000918
1B038C00 0A317008 116D008D 000B198B
000C3B19 1E08438B 000D1908 1E08438B
000E7A02 301E046B 07196704 18770177
05230331 1D056B35 8D000F2E 1B1B8B00
10AD0103 1B8B0011 8B00123B 18AD0103
8C000A31 8D001328 041504AD 01081E08
438B0014 1504109F 8B00157A 02108D00
16058E02 0017007A 07621007 29061908
8D001829 0716077D 00196B39 19160625
022FE4B0 3D557273 B26B0916 0604415B
29061606 04415B29 06191606 2510926B
1C191606 AD000319 16060441 2505418D
001A3B15 04160503 38047811 6A808D00
0B037806 31033219 1E05418D 00181100
926B23AD 00032560 15AD0003 191E0841
AD000425 05418D00 1A327010 116A888D
000B7008 116A888D 000B1F78 08001A00
04000100 0103000B 48454C4C 4F20574F
524C4400 00000200 7005006E 001B0200
00000200 00010680 03000100 00000600
00010380 03010380 03020600 00C90380
03030380 0A010600 01280680 07010380
0A070380 0A090380 0A040684 08000384
080D0384 080E0384 0842CC07 EF33E438
D7020684 06000384 06100384 06120681
10000181 09000680 10040500 00020680
10010900 2C00090E 08890C0E 512F0607
001F0516 040E0C04 09071309 04080819
06070308 040D0706 050D0730 0F0D1B09
080B0103 01000100 00050004 00088107
```

```
'A000000559101001'H,
      classAID 'A000000559101001112233'H,
      instanceAID 'A00000055910100111223306'H,
      applicationPrivileges '000000'H,
      lifeCycleState '07'H,
      applicationSpecificParametersC9 '00'H,
         applicationParameters {

uiccToolkitApplicationSpecificParametersField
          '0100010000000311223300'H
      },
      processData {
        '80E2880009007006920411223344'H
      }
    }
  }
}
```

```
82008002 81088308 00090000 000038FF
0A000002 80040002 00000000 38010200
00010038 00810001 003A0015 00000000
01090018 003C0026 00000000 07010040
0064003E 00000000 05010080 0068000B
00000000 0801008D 0044003A 00000000
FF0200C9 003A000A 00000000 090100D5
006F0051 00000000 FF020128 00410037
0079AEC9 3548B8F1 D3000000 001B0038
0038003A FFFF003A 003A003C 003A003F
00380041 00440046 00440048 004B0046
0046004F 00530057 005A005C FFFF0041
00460060 01B00110 04B43101 2003B440
02410140 03441005 68408005 4B444005
68406004 B4410231 05681090 06B4B444
066800A1 0B680063 68002006 B44B44A2
A5EC474D E8B2DEA2 50304E4F 08A00000
05591010 014F0BA0 00000559 10100111
22334F0C A0000005 59101001 11223306
82030000 00C90100 EA0D800B 01000100
00000311 22330030 10040E80 E2880009
00700692 04112233 44
```

Note: this PE Application implies that data is returned using RP-ACK protocol. In order that data would be returned by MO SMS this PE Application is expected to be modified in the next version of this specification.

6.12.2.3. Profile Header

6.12.2.3.1. Profile Header 2

Profile_HEADER 2

```
headerValue ProfileElement ::= header : {
  major-version 2,
  minor-version 0,
  profileType "SIMalliance Profile Package",
  iccid '89019990001234567893'H,
  eUICC-Mandatory-services {
    usim NULL,
    milenage NULL,
    javacard NULL
  },
  eUICC-Mandatory-GFSTEList {
    -- MF-ID
    {2 23 143 1 2 1},
    -- USIM-ID
    {2 23 143 1 2 4}
  }
}
```

```
A0498001 02810100 821B5349 4D616C6C
69616E63 65205072 6F66696C 65205061
636B6167 65830A89 01999000 12345678
93A50681 0084008B 00A61006 0667810F
01020106 0667810F 010204
```

7. Profile Package General Structure

7.1 Test requirements

The test requirements are extracted from sections 7.2 and 7.3 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ7.1.1.1	Each PE is described and can be processed by the eUICC independently from the others.
RQ7.1.1.2	An identification number shall be associated to every PE.
RQ7.1.1.3	A PE starts with a header containing the following information: • PE identification number. • Optional flag indicating that the support of this PE is mandatory. • PE type. • PE length.
RQ7.1.1.4	If a feature in a PE flagged as mandatory is not supported by the eUICC: • an error is reported to the Profile Creator. • the processing of the Profile Package is cancelled. • and all of the PE already processed shall be discarded.
RQ7.1.1.5	If a PE is not flagged as mandatory, and if the eUICC does not support the associated feature, the error is reported but the processing of the Profile Package continues.
RQ7.1.1.6	In order to avoid errors and warnings during the processing of a Profile Package, the Profile Creator may audit the targeted eUICC before building a Profile Package. In that case, all the features described in the Profile Package will be entirely supported by the eUICC.
RQ7.1.1.7	The features that shall be supported by the Profile are also described in the Profile header. In case the eUICC does not support one of the features listed in this Profile header, the eUICC shall immediately return an error code and abort the processing of the Profile.
NOTE: RQ7.1.1.1, RQ7.1.1.2 and RQ7.1.1.3 are implicitely tested in test cases loading profiles NOTE 2: Testing of RQ7.1.1.4 and RQ7.1.1.5 is FFS. NOTE 3: RQ7.1.1.6 is out of the scope of this specification.	

7.2 Test cases / scenarios

FFS.

8. Profile Package Elements Definition

8.1 Test requirements

8.1.1 Common types

The test requirements are extracted from section 8.1 of “eUICC Profile Package: Interoperable Format Technical Specification” [SA PP TS].

RQ8.1.1.1	The Profile Package shall respect the size constaints 0 to 255 for the basic integer type Uint8.
RQ8.1.1.2	The Profile Package shall respect the size constaints 0 to 32267 for the basic integer type Uint15.
RQ8.1.1.3	The Profile Package shall respect the size constaints 0 to 65535 for the basic integer type Uint16.
RQ8.1.1.4	VOID
RQ8.1.1.5	The Application Identifier (AID) shall be an OCTET STRING with the size of 5 to 16 bytes.
RQ8.1.1.6	The PE Header shall be present at the beginning of all PE-s described in this specification.
RQ8.1.1.7	The PE Header may consist of an optional “mandated” field. The type of the mandated field shall be NULL.
RQ8.1.1.8	If the mandated field is set the support of this PE is mandatory for the installation of this Profile. If the eUICC does not support the following PE, it shall abort the processing of the profile and return an error to the sender of the profile.
RQ8.1.1.9	The PE Header shall consist of an “identification” field. The type of the identification field shall be Uint15.
RQ8.1.1.10	The identification field is used to uniquely identify the PE within a profile. It will be used for error reporting to the sender of the profile.
RQ8.1.1.11	Void
RQ8.1.1.12	The ProfileHeader shall be the first element and provided once within a profile download only.
RQ8.1.1.13	VOID
RQ8.1.1.13a	The PE MF may be provided once as the first element of the file system creation after the ProfileHeader PE.
RQ8.1.1.13b	If PE MF is not used, the MF shall be created as the first element of the file system using the PE Generic File Management.
RQ8.1.1.14	The PE-CD is optional and shall come after the creation of the MF.
RQ8.1.1.15	The PE-TELEKOM is optional and shall come after the creation of the MF.
RQ8.1.1.16	The PE-USIM is optional and shall come after the creation of the MF.
RQ8.1.1.17	The PE-ISIM is optional and shall come after the creation of the MF.
RQ8.1.1.18	The PE-CSIM is optional and shall come after the creation of the MF.
RQ8.1.1.19	The PE-OPT-USIM is optional and shall come after the PE-USIM.
RQ8.1.1.20	The PE-GSM-ACCESS is optional and shall come after the PE-USIM.
RQ8.1.1.21	The PE-PHONENOOK is optional and shall come after the PE-USIM.
RQ8.1.1.22	The PE-OPT-ISIM is optional and shall come after the PE-ISIM.
RQ8.1.1.23	The PE-OPT-CSIM is optional and shall come after the PE-CSIM.
RQ8.1.1.24	Dependencies within the file system creation need to be considered.
RQ8.1.1.25	PE-PINCodes shall be created in the context according to their scope.
RQ8.1.1.26	Global PINs (Application PINs according to ETSI TS 102 221) shall be provided once in the context of the creation of the MF of the UICC.
RQ8.1.1.26b	Local PINs may be provided once in the context of the creation of a DF or ADF
RQ8.1.1.27	PE-PINCodes shall only be provided once within each DF (ADF).
RQ8.1.1.28	VOID
RQ8.1.1.28a	If PE-AKAParameters is provided, it shall be present in the context of the creation of a NAA filesystem.
RQ8.1.1.29	VOID
RQ8.1.1.29a	PE-AKAParameters may be provided once or several times per NAA. If several set of parameters are provided for one NAA, the set of parameters used by this NAA is not defined.
RQ8.1.1.30	PE-AKAParameters is not allowed in the context of MF.
RQ8.1.1.31	PE-AKAParameters is not allowed in the context of SDs.
RQ8.1.1.32	PE-AKAParameters is not allowed in the context of applications.
RQ8.1.1.33	VOID
RQ8.1.1.33a	PE-PUKCodes may only be provided once within the context of the UICC file system (MF). If PE-PUKCodes is not present in the Profile Package then no PUK codes are defined.
RQ8.1.1.34	PE-PUKCodes shall include all PUK codes for the complete profile.
RQ8.1.1.35	PE-SecurityDomain is optional and shall be provided after the creation of the file system, NAA parameters and PIN/PUK configuration.
RQ8.1.1.36	PE-Application is optional and should be provided after the creation of the SDs.
RQ8.1.1.37	PE-RFM is optional. It shall be provided after the creation of the SDs the RFM parameters shall be assigned to.
RQ8.1.1.38	PE-NonStandard is optional and in general may be provided in any position after the profile header. Further restrictions depend on the respective application.
RQ8.1.1.39	PE-End shall be provided once at the end of the Profile Package.
NOTE: RQ8.1.1.10 and RQ8.1.1.38 are not testable. NOTE 2: VOID NOTE 3: Requirements RQ8.1.1.1, RQ8.1.1.2, RQ8.1.1.3 and RQ8.1.1.5 are implicitly tested in test cases loading profiles. NOTE 4: VOID NOTE 5: testing of RQ8.1.1.17, RQ8.1.1.18, RQ8.1.1.22 and RQ8.1.1.23 is FFS (ISIM and CSIM related)	

NOTE 6: testing of RQ8.1.1.30, RQ8.1.1.31 and RQ8.1.1.32 is FFS (not nominal tests) NOTE 7: RQ8.1.1.35 is implicitly tested every time a PE-SecurityDomain is used in test cases NOTE 8: Testing of RQ8.1.1.14, RQ8.1.1.15, RQ8.1.1.20, RQ8.1.1.21 and RQ8.1.1.24 is FFS.

8.1.2 Profile header

The test requirements are extracted from section 8.2 of “eUICC Profile Package: Interoperable Format Technical Specification” [SA PP TS].

<table>
<tr><td>RQ8.1.2.1</td><td>The ProfileHeader PE shall be used once and shall be the first PE of the Profile Package.</td></tr>
<tr><td>RQ8.1.2.2</td><td>The ServiceList type is used to indicate the services that shall be supported by the eUICC. The type of the fields in the ServiceList shall be NULL.</td></tr>
<tr><td>RQ8.1.2.3</td><td>The following list gives the features that the eUICC shall support in order to provide the associated service. When a service is present in the sequence, it indicates that this service is mandatory.
<table>
<tr><th>Service</th><th>Feature provided by te eUICC</th></tr>
<tr><td>contactless</td><td>support the SWP and HCI interfaces as well as the associated APIs.</td></tr>
<tr><td>usim</td><td>the USIM application as defined by 3GPP [USIM].</td></tr>
<tr><td>isim</td><td>the ISIM application as defined by 3GPP [ISIM].</td></tr>
<tr><td>csim</td><td>the CSIM application as defined by 3GPP2 [CSIM].</td></tr>
<tr><td>milenage</td><td>the milenage AKA authentication algorithm.</td></tr>
<tr><td>tuak128</td><td>the TUAK AKA authentication algorithm with 128 bit key length.</td></tr>
<tr><td>tuak256</td><td>the TUAK AKA authentication algorithm with 256 bit key length</td></tr>
<tr><td>cave</td><td>the CAVE authentication algorithm.</td></tr>
<tr><td>gba-usim</td><td>support of GBA authentication context in the USIM application.</td></tr>
<tr><td>gba-isim</td><td>support of GBA authentication context in the ISIM application.</td></tr>
<tr><td>mbms</td><td>support of the MBMS authentication context in the USIM application.</td></tr>
<tr><td>eap</td><td>support of the UICC EAP client.</td></tr>
<tr><td>javacard</td><td>support of the Java Card TM runtime environment.</td></tr>
<tr><td>multos</td><td>support of the Multos TM runtime environment.</td></tr>
<tr><td>multiple-usim</td><td>support of multiple USIM instances – requires “usim” to be present in the list</td></tr>
<tr><td>multiple-isim</td><td>support of multiple ISIM instances – requires “isim” to be present in the list</td></tr>
<tr><td>multiple-csim</td><td>support of multiple CSIM instances – requires “csim” to be present in the list</td></tr>
</table></td></tr>
<tr><td>RQ8.1.2.4</td><td>The ProfileHeader shall contain the “major-version”. The type of the major-version shall be Uint8.</td></tr>
<tr><td>RQ8.1.2.5</td><td>When receiving the ProfileHeader, the eUICC shall check the major-version. If the version indicated by the Profile is not supported by the eUICC (e.g. if it is an earlier or an older version), the eUICC shall return an error “unsupported-profile-version” and stop the processing of the Profile.</td></tr>
<tr><td>RQ8.1.2.6</td><td>The ProfileHeader shall contain the “minor-version”. The type of the minor-version shall be Uint8.</td></tr>
<tr><td>RQ8.1.2.7</td><td>The minor-version is only informative. It may indicate that the profile contains elements that the eUICC will not be able to process if it supports an older version of the specification. In that case, these elements shall be ignored by the eUICC unless they are marked as mandatory in the PE header.</td></tr>
<tr><td>RQ8.1.2.8</td><td>The ProfileHeader may contain the “profileType”. The type of the profileType shall be UTF8String. The “profileType” is a free optional text indicating for example, the name of the Profile issuer and the type of Profile.</td></tr>
<tr><td>RQ8.1.2.9</td><td>The ProfileHeader shall contain the “iccid”. The type of iccid shall be OCTET STRING (SIZE (10)).</td></tr>
<tr><td>RQ8.1.2.9a</td><td>The “iccid” shall be be encoded non-swapped as per ITU E.118 representation and padded with 'F' if less digits are used (Example:8947010000123456784F) (see NOTE 4)</td></tr>
<tr><td>RQ8.1.2.10</td><td>The ProfileHeader may contain the “pol”. The type of the pol shall be OCTET STRING. The pol contains the policy rules within a Profile.</td></tr>
<tr><td>RQ8.1.2.11</td><td>If pol is not supplied in the Profile Package, its value shall be set to all 0 in the eUICC.</td></tr>
<tr><td>RQ8.1.2.12</td><td>The ProfileHeader shall contain the “eUICC-Mandatory-services”. The type of the eUICC-Mandatory-services shall be ServiceList.</td></tr>
<tr><td>RQ8.1.2.13</td><td>The ProfileHeader shall contain the “eUICC-Mandatory-GFSTEList”.</td></tr>
<tr><td>RQ8.1.2.14</td><td>The “eUICC-Mandatory-GFSTEList” contains a list of OIDs identifying file system templates used in the Profile Package that shall be supported by the eUICC in order for the Profile to be correctly installed on the eUICC.</td></tr>
<tr><td>RQ8.1.2.15</td><td>This list may contain the OIDs associated to the file system template defined in “ANNEX A (Normative): File Structure Templates Definition” of this specification.</td></tr>
<tr><td>RQ8.1.2.16</td><td>The ProfileHeader may contain the “connectivityParameters”. The “connectivityParameters” contains the connectivity parameters as defined in GSMA in [GS RPT], in table 52, not including '3A07' DGI.</td></tr>
<tr><td colspan="2">NOTE 1: RQ8.1.2.5 and RQ8.1.2.7 are FFS.
NOTE 2: RQ8.1.2.10 and RQ8.1.2.11 are not testable (there is no interoperable command to read the value).
NOTE 3: RQ8.1.2.13 is implicitely tested everytime ProfileHeader is used.</td></tr>
</table>

NOTE 4: REQ8.1.2.9a is out of scope of this specification.

8.1.3 File system

The test requirements are extracted from section 8.3 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.3.1	Templates need to be created according to the specified settings.
RQ8.1.3.2	Templates can be sent in any order considering the dependencies (e.g. some templates require that a NAA has already been created).
RQ8.1.3.3	Parameters which alter the default given in a template needs to result in the desired configuration; e.g. change of file size, access rule reference.
RQ8.1.3.4	In case a file within a template is specified as 'do not create' it must not be available within the created file system.
RQ8.1.3.5	It shall be possible to mix templates with Generic FileSystem Commands.
RQ8.1.3.6	It shall be possible to create a complete profile by Generic FileSystem Commands without use of any templates.
RQ8.1.3.7	Using a template marked as mandated but which is not supported by the eUICC shall lead to an error.
RQ8.1.3.8	The eUICC shall support any template it claims to support; e.g the profile header is passed which requires the need for specific templates creation of the template shall work provided it is correctly used.
RQ8.1.3.9	It shall be possible to create multiple instances of the following templates: - USIM - ISIM - CSIM - EAP-AKA
RQ8.1.3.10	Templates shall always be created within the current context. E.g. the optional USIM EFs template shall be created in the currently selected application.
RQ8.1.3.11	The eUICC shall be able to create multiple instances of a file from a template by following the process described in figure 2 of [SA PP TS].
RQ8.1.3.12	It shall not be possible to create two files with the same file path irrespective of whether templates or a generic file system is used.
RQ8.1.3.13	void
RQ8.1.3.14	The eUICC shall be able to handle the "template modification rules" described within the specification.
RQ8.1.3.15	File content provided within the profile package shall be applied to the created file.
RQ8.1.3.16	Within an optional template, files shall only be created if the respective TLV is explicitly included in the profile package.
RQ8.1.3.17	For mandatory file templates all files shall be created unless they are explicitly marked as "do not create".
RQ8.1.3.18	For all files which are not fully defined in the template specification (open parameters like size) the respective parameters shall be included in the profile package.
RQ8.1.3.19	FCP of files which have been created may include proprietary information. These parameters shall be ignored when checking the settings of files which have been created.
RQ8.1.3.20	The access conditions which have been configured shall apply for the respective files; e.g.: Never shall always be Never and not readable even if other PINs are verified; in case PIN1 is specified for read it shall only be possible to read the file if PIN1 has been verified; The eUICC shall apply all provided FCP parameters according to ETSI TS 102 221 [102 221].
RQ8.1.3.21	The eUICC shall support access rule conditions according to the UICC specification ETSI TS 102 221 [102 221]; also supporting AND/OR conditions like PIN1 || ADM1.
NOTE: RQ8.1.3.1 and RQ8.1.3.19 are out of scope of this specification. NOTE 2: Testing of RQ8.1.3.2, RQ8.1.3.7, RQ8.1.3.9, RQ8.1.3.11, RQ8.1.3.12 and RQ8.1.3.18 is FFS. NOTE 3: RQ8.1.3.8 and RQ8.1.3.15 are implicitly tested in all test cases.	

8.1.4 NAA(s)

The test requirements are extracted from section 8.4 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.4.1	The PE-AKAParameters shall be tested with the USIM and ISIM NAA.
RQ8.1.4.2	PE-AKAParameters shall be tested using both options: milenage and TUAK.
RQ8.1.4.3	For milenage PE-AKAParameters shall be tested with the following parameters: key: 16 byte length opc: 16 byte length RES Length Options: 32bits, 64bits, 128bits MAC-A, MAC-C Size: does not apply. To be set to 0 (64 bit) CK and IK size: 128 bits Rotation constants shall have a length of 5 bytes xoringConstants shall have a length of 80 Bytes.
RQ8.1.4.4	For testing milenage the test vectors from 3GPP [MILENAGE] shall be used: PE-AKAParameters shall be initialised with the respective settings.
RQ8.1.4.5	For testing TUAK the test vectors from 3GPP [TUAK] shall be used: PE-AKAParameters shall be initialised with the respective settings.
RQ8.1.4.6	Using Authenticate within USIM NAA in 2G Compatibility mode shall only work if service 38 within the UST is enabled.
RQ8.1.4.7	Authenticate command shall only work if respective Application PIN for the NAA has been verified (e.g. PIN1).
RQ8.1.4.8	Sharing network credentials via the mapping function shall be tested between USIM NAAs, ISIM NAAs and USIM/ISIM. Same `algorithmID`, `algorithmOptions`, `key`, `(T)opc`, `rotationConstants`, `xoringConstants` and `authCounterMax`for both NAAs is to be anticipated. The following mapping permutations shall be tested: - Share `sqnInit`, `sqnOptions`, `sqnDelta`, `sqnAgeLimit`. - Share `sqnOptions`, `sqnDelta`, `sqnAgeLimit`.
RQ8.1.4.9	DEFAULT values shall be verified by the relevant test to ensure that they are set correctly.
RQ8.1.4.10	It shall be tested if the DEFAULT values can be overwritten by the profile package; it shall also be checked that the DEFAULT values can be provided as well.
RQ8.1.4.11	Values for rotationConstants and xoringConstants shall only be provided in case milenage is used.
RQ8.1.4.12	SQN handling shall be tested with the available options: - Authentication shall not work for blocked SQN when the wrap around option deactivated. - In case SQN value has reached the maximum value 07FFFFFFFFFF authentication shall still work (by disabled SQN verification) if the wrap around option is activated. - If incoming SQN is out of range (depends on delta and age limit) the eUICC shall indicate the need for resynchronization – provided the authentication vector passes authentication. - Authentication shall work if SQN is within the desired range (considering Delta and Age limit).
RQ8.1.4.13	In case a value is provided for authCounterMax it shall be tested. It defines the accumulated number of Authenticate Commands for all the NAA-s which share the counter over the complete life time of the profile (independent from resets, profile de-/activation). It shall be provided once in a Profile Package. Once the actual number of Authenticate commands reaches the defined value the command should fail and return '6F00' as the respective error code. This behaviour shall apply to the NAA-s sharing the same parameters where the counter max has been reached.
RQ8.1.4.14	The PE-AKAParameters shall be send once per NAA.
NOTE : Testing of all RQs in this sections is FFS (priority for phase 2).	

8.1.5 PIN and PUK codes

The test requirements are extracted from section 8.5 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.5.1	Global PINs created by the PE-PINCodes shall be valid within the complete FileSystem.
RQ8.1.5.2	Local PINs shall only be valid within the context (DF/ADF and sub DFs) where they are defined.
RQ8.1.5.3	VOID
RQ8.1.5.4	Local PINs shared shall share remaining attempts in all contexts where they are valid.
RQ8.1.5.5	It shall be possible to create Global PINs in the context of the MF. E.g. after creation of the MF or also after selection of the MF using Generic File System.
RQ8.1.5.6	VOID
RQ8.1.5.7	It shall be possible to share one PUK for multiple PIN values.
RQ8.1.5.8	Blocked PINs cannot be verified via I/O, but applets with the respective access rights may execute the authorised commands (update, read, create, delete, …)
RQ8.1.5.9	Within the FCP of the ADF and the MF the eUICC has to indicate the status of the PINs/PUKs as specified within the template (e.g. remaining attempts, PINs initialised, PINs available, PIN activated/deactivated) provided that the settings have not been altered after profile installation.
RQ8.1.5.10	The eUICC needs to support the PIN attributes specified: - PINs enabled: in this case the PIN shall be enabled. - PIN may be changed: PIN change allowed; otherwise not. - PIN can be disabled: Means that status of the PIN may not be altered. - disabled PIN may not be enabled. - enabled PIN may not be disabled.
RQ8.1.5.11	It shall be possible to create all possible global PINs within the global PE-PINCodes.
RQ8.1.5.12	It shall be possible to create all second Application PINs within one or more DFs.
RQ8.1.5.13	Two local PINs which have been created separately in two DFs with the same second application PIN ID shall have separate status; own remaining attempts; own verified status; own enabled/disable status; also different attributes may be applied for the two PINs.
RQ8.1.5.14	PIN Values shall have a length of 8 Bytes. Unused Bytes are to be padded with FF..FF.
RQ8.1.5.15	It shall be possible to define any value for any PIN: Random Hex Values and also coded as string for user PINs (e.g. PIN 1234 > 31 32 33 34 FF FF FF FF).
RQ8.1.5.16	It shall be possible to assign a PUK value for any PIN.
RQ8.1.5.17	maxNumOfAttemps-retryNumLeft: It shall be possible to assign any value from 0…F for maxNumberOfAttempts and retryNumLeft independent from each other.
RQ8.1.5.18	It shall be possible to create any PIN in enabled or disabled mode.
RQ8.1.5.19	It shall be possible to create any PIN with “PIN can be disabled” stated to define that a PIN status cannot be changed from enabled to disabled and vice versa.
RQ8.1.5.20	It shall be possible to define any PIN with “PIN can be changed” set to allow changing the PIN value; if “PIN can be changed” is not set it shall not be possible to change the PIN.
NOTE: Testing of these RQs is FFS.	

8.1.6 Security domains

The test requirements are extracted from section 8.6 of “eUICC Profile Package: Interoperable Format Technical Specification” [SA PP TS].

RQ8.1.6.1	The PE Security Domain shall consist of a PE header and an Application Instance object.
RQ8.1.6.2	The values standardised for Supplementary SDs shall be used for the Application Instance object.
RQ8.1.6.3	The PE Security Domain may consist of a keylist and sdPersoData objects.
RQ8.1.6.4	The PE-SecurityDomain shall be used for every SD creation, starting from MNO-SD.
RQ8.1.6.5	The MNO-SD shall be defined and created explicitly using "PE-SecurityDomain" within the Profile Package.
RQ8.1.6.6	The MNO-SD shall be created first before any other SD, before any RFM Parameters are set, or before any applets are created.
RQ8.1.6.7	Since no package AID nor classAID is standardised for the MNO-SD, it shall use the values defined for supplementary SD creation.
RQ8.1.6.8	The first SD within the sequence of the Profile Package shall be categorised as the MNO-SD by definition.
RQ8.1.6.9	The MNO-SD shall be installed with the special MNO-SD rights defined by the GSMA.
RQ8.1.6.10	All subsequent following instances of SDs shall be installed like regular supplementary SDs as known from GlobalPlatform Card Specification [GP CS].
RQ8.1.6.11	The keylist optional present in the Security Domain PE shall be a sequence of key objects.
RQ8.1.6.12	A key object shall contain a keyUsageQualifier, tag number [21] which shall be an OCTET STRING with SIZE of 1.
RQ8.1.6.13	A key object shall contain a keyAccess, tag number [22] which shall be an OCTET STRING with SIZE of 1.
RQ8.1.6.14	A key object shall contain a keyIdentifier, tag number [2] which shall be an OCTET STRING with SIZE of 1.
RQ8.1.6.15	A key object shall contain a keyVersionNumber, tag number [3] which shall be an OCTET STRING with SIZE of 1.
RQ8.1.6.16	A key object shall contain a list of keyComponents.
RQ8.1.6.17	A keyComponent shall contain a keyType, tag number [0], which shall be an OCTET STRING.
RQ8.1.6.18	A keyComponent shall contain a keyData which shall be an OCTET STRING.
RQ8.1.6.19	VOID
RQ8.1.6.20	A key object may contain a keyCounterValue, tag number [5] which shall be an OCTET STRING.
RQ8.1.6.20a	If the keyCounterValue is present, it indicates the initial counter associated for that keyset.
RQ8.1.6.20b	If the keyCounterValue is absent, the initial counter value shall be set according to the default value of the related protocol (e.g. for SCP02 keyset the default value is ‘0000’h, for SCP03 it is ‘000000’h, for SCP80 it is ‘0000000000’h).
RQ8.1.6.21	VOID
RQ8.1.6.22	Each key to be personalised shall be listed only once.
RQ8.1.6.23	VOID
RQ8.1.6.24	VOID
RQ8.1.6.25	VOID
RQ8.1.6.26	Only keyTypes defined in GlobalPlatform Card Specification [GP CS], Table 11-16, may be part of the list of keyComponents.
RQ8.1.6.27	Each keyComponent shall be specified only once per key (e.g. including two times the same keyType within one KeyObject will lead to an error).
RQ8.1.6.28	In case the sdPersoData is present it shall contain the data field of a STORE DATA command used to personalise the SD.
RQ8.1.6.29	The content of the data field of the STORE DATA command shall not be encrypted and shall use DGI format.
RQ8.1.6.30	The complete DGI structure for the SD personalisation shall be sent in one complete byte array.
RQ8.1.6.31	Each DGI shall be provided in its own sdPersoData record.
RQ8.1.6.32	Only standardised DGIs, according to GlobalPlatform Card Specification [GP CS], shall be sent when addressing a SD.
RQ8.1.6.33	Installation of the CASD, if required inside a Profile, shall use the same personalisation procedure as defined for SDs.
RQ8.1.6.34	In case RAM and OTA HTTPs is added to a SD the settings can be configured according to GlobalPlatform Card Specification [GP CS] and ETSI specifications.
RQ8.1.6.35	In case RAM is be added to a SD the TAR values for RAM can be configured as follows: - Bytes 13-15 of the SD instance AID. - TAR List within SD install parameters.
RQ8.1.6.36	VOID
RQ8.1.6.36a	In case OTA HTTPS is added to a SD OTA HTTPs may be provided within the sdPersoData included in DGI '0070' using in tag ‘85’ according to GlobalPlatform Amd B [GP AB] (Section 3.7.1 TLV: Security Domain Administration Session Parameters) in the PE-SecurityDomain structure of the respective security domain.
RQ8.1.6.37	In the case where RAM is added to a SD the security level for RAM shall be defined by the MSL parameter of the SD installation parameters.

RQ8.1.6.38	VOID
RQ8.1.6.38a	In the case where RAM is added to a SD, TAR values to the Security Domains as specified in TS 101 220 [101 220] should be assigned.
RQ8.1.6.39	The configuration of the PoR (Proof of Receipt) handling shall not be part of the Profile definition.
RQ8.1.6.40	The eUICC shall follow the latest ETSI and 3GPP release to provide the necessary level of security.
RQ8.1.6.41	There may be SSDs which belong to independent SD hierarchies with a self-extradited SSD as root SD.
RQ8.1.6.42	A keyComponent shall contain a macLength which shall be an UInt8 DEFAULT 8.
RQ8.1.6.43	If macLength is for AES KID keys, indicates the length of the MAC in bytes as defined in TS 102 226 [102 226].
RQ8.1.6.44	macLength shall be ignored for other key types than AES KID
RQ8.1.6.45	If keyType or any other KeyObject parameters are not supported by the eUICC, the error code feature-not-supported shall be returned and the installation of the Profile Package shall be aborted.
RQ8.1.6.46	Parameters using TLV format may be included in DGI '0070' as defined by GlobalPlatform Card Specification [GP CS].
NOTE: RQ8.1.6.9 is not tested in this specification. Its verification is under the scope of the GSMA. NOTE 2: testing of RQ8.1.6.20, RQ8.1.6.20a, RQ8.1.6.20b, RQ8.1.6.33, RQ8.1.6.41, RQ8.1.6.42, RQ8.1.6.43, RQ8.1.6.44 and RQ8.1.6.45 is FFS. NOTE 3: RQ8.1.6.39 is not testable. NOTE 4: RQ8.1.6.32 is not tested in this specification. Its verification is under the scope of GlobalPlatform. NOTE 5: RQ8.1.6.22 and RQ8.1.6.40are out of scope of this specification.	

8.1.7 Application loading and installation

The test requirements are extracted from section 8.7 of “eUICC Profile Package: Interoperable Format Technical Specification” [SA PP TS].

RQ8.1.7.1	A library shall be loaded when only a ApplicationLoadPackage object is provided within one Application PE.
RQ8.1.7.2	A preloaded application shall be installed only when an ApplicationInstance object is provided within one Application PE.
RQ8.1.7.3	Multiple instances of the same application shall be installed when multiple ApplicationInstance objects are provided within one Application PE.
RQ8.1.7.4	An application shall be loaded providing an ApplicationLoadPackage object and installed via an ApplicationInstance.
RQ8.1.7.5	An application shall be installed when an ApplicationInstance object is provided within one Application PE.
RQ8.1.7.6	If PEHeader object is set to mandatory, profile installation shall fail if one of the subsequent elements cannot be executed (e.g. load fails because of API incompatibility, install fails because of duplicate TAR values …).
RQ8.1.7.7	If PEHeader object is not set to mandatory, profile installation shall continue with the next PE if one of the subsequent elements cannot be executed (e.g. load fails because of API incompatibility, install fails because of duplicate TAR values …).
RQ8.1.7.8	The loadPackageAID object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS].
RQ8.1.7.9	The loadPackageAID object is mandatory and shall be an ApplicationIdentifier.
RQ8.1.7.10	The securityDomainAID object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS].
RQ8.1.7.11	The securityDomainAID object is optional and shall be an ApplicationIdentifier.
RQ8.1.7.12	The nonVolatileCodeLimitC6 object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS].
RQ8.1.7.13	The nonVolatileCodeLimitC6 object is optional and it shall be an OCTET STRING.
RQ8.1.7.14	The volatileDataLimitC7 object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS].
RQ8.1.7.15	The volatileDataLimitC7 object is optional and it shall be an OCTET STRING.
RQ8.1.7.16	The nonVolatileDataLimitC8 object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS]
RQ8.1.7.17	The nonVolatileDataLimitC8 object is optional and it shall be an OCTET STRING.
RQ8.1.7.18	The hashValue object shall be based on the GP2.2 Load Command according to GlobalPlatform Card Specification [GP CS]
RQ8.1.7.19	The hashValue object is optional and it shall be an OCTET STRING.
RQ8.1.7.20	The loadBlockObject object shall contain the complete load block.
RQ8.1.7.21	The loadBlockObject object is mandatory and it shall be an OCTET STRING.
RQ8.1.7.22	The coding of applicationLoadPackageAID object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.23	The applicationLoadPackageAID object is mandatory and shall be an ApplicationIdentifier.
RQ8.1.7.24	The coding of classAID object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.25	The classAID object is mandatory and shall be an ApplicationIdentifier.
RQ8.1.7.26	The coding of instanceAID object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.27	The instanceAID object is mandatory and shall be an ApplicationIdentifier.
RQ8.1.7.28	The extraditeSecurityDomainAID object shall have the same effect like the Install for Extradition command defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.29	The extraditeSecurityDomainAID object is optional and shall be an ApplicationIdentifier.
RQ8.1.7.30	If the extraditeSecurityDomainAID object value is not provided, the instance shall be associated to the MNO-SD by default.
RQ8.1.7.31	The coding of applicationPrivileges object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.32	The applicationPrivileges object is mandatory and it shall be an OCTET STRING.
RQ8.1.7.33	The coding of lifeCycleState object shall follow the coding Life Cycle State defined within GlobalPlatform Card Specification [GP CS] (section 11.1.1 Life Cycle Coding).
RQ8.1.7.34	VOID
RQ8.1.7.34a	The lifeCycleState object is optional for the profile package and it shall be an OCTET STRING. If not provided the default value '07'H shall be taken into account as if provided.
RQ8.1.7.35	The coding of applicationSpecificParametersC9 object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.36	The applicationSpecificParametersC9 object is mandatory and it shall be an OCTET STRING.

RQ8.1.7.37	The coding of systemSpecificParameters object shall follow the coding defined for Install for Install defined by GlobalPlatform Card Specification [GP CS].
RQ8.1.7.38	The systemSpecificParameters object is optional and it shall be an ApplicationSystemParameters.
RQ8.1.7.39	The coding of applicationParameters object shall follow the coding defined in ETSI TS 102 226 [102 226].
RQ8.1.7.40	The applicationParameters object is optional and it shall be an UICCApplicationParameters.
RQ8.1.7.41	The applicationParameters can be used to define the access domain for an applet.
RQ8.1.7.42	The applicationParameters can be used to define the MSL (Minimum Security Level) for an applet or an RFM instance.
RQ8.1.7.43	The processData object is optional and it shall be a SEQUENCE OF OCTET STRING.
RQ8.1.7.44	The processData object octet string shall be directly sent to the respective application instance for processing through the "processData" method of the "Application" or "Personalization" interface of the application.
RQ8.1.7.45	The processData object may contain all the bytes contained in a STORE DATA command (Including CLA,INS, P1, P2, L) if required by the application but encryption shall not be used. Note: This test specification will consider this as mandatory otherwise it is not predictable.
RQ8.1.7.46	The processData object shall contain data for the application and no decryption shall be performed by the respective SD.
RQ8.1.7.47	The volatileMemoryQuotaC7 is optional and it shall be an OCTET STRING.
RQ8.1.7.48	The nonvolatileMemoryQuotaC8 is optional and it shall be an OCTET STRING.
RQ8.1.7.49	The globalServiceParameters is optional and it shall be an OCTET STRING.
RQ8.1.7.50	The implicitSelectionParameter is optional and it shall be an OCTET STRING.
RQ8.1.7.51	The volatileReservedMemory is optional and it shall be an OCTET STRING.
RQ8.1.7.52	The nonVolatileReservedMemory is optional and it shall be an OCTET STRING.
RQ8.1.7.53	The ts102226SIMFileAccessToolkitParameter is optional and it shall be an OCTET STRING.
RQ8.1.7.54	The ts102226AdditionalContactlessParameters is optional and it shall be a TS102226AdditionalContactlessParameters.
RQ8.1.7.55	The uiccToolkitApplicationSpecificParametersField is optional and it shall be an OCTET STRING.
RQ8.1.7.56	The uiccAccessApplicationSpecificParametersField is optional and it shall be an OCTET STRING.
RQ8.1.7.57	VOID
RQ8.1.7.58	The uiccAdministrativeAccessApplicationSpecificParametersField is optional and it shall be an OCTET STRING.
RQ8.1.7.59	The protocolParameterData is mandatory and it shall be OCTET STRING.
RQ8.1.7.60	The processData object shall be provided to the respective applet instance, with the supported processData method according to GlobalPlatform Card Specification [GP CS].
RQ8.1.7.61	The Application PE shall be used after the security domain to which the application instance is associated to is created by using PE-SecurityDomain.
RQ8.1.7.62	In case no value for the optional parameter `securityDomainAID` is provided, the package will be associated to the MNO-SD by default.
RQ8.1.7.63	The contactlessProtocolParameters is optional and it shall be OCTET STRING.
RQ8.1.7.64	The contactlessProtocolParameters shall be coded according to Contactless Protocol Parameters Structure as defined in GlobalPlatform Amd. C [XXX]
RQ8.1.7.65	The userInteractionContactlessParameters is optional and it shall be OCTET STRING
RQ8.1.7.66	The userInteractionContactlessParameters shall be coded according to User Interaction Parameters Structure as defined in GlobalPlatform Amd. C [XXX]
RQ8.1.7.67	The protocolParameterData shall be encoded according to ETSI TS 102 226 [XXX]
RQ8.1.7.68	The whole PE should be discarded, if the processData object is provided in the PE, but the application does not implement the "processData" method.
NOTE: Testing of RQ8.1.7.2, RQ8.1.7.28, RQ8.1.7.29, RQ8.1.7.30, RQ8.1.7.37, RQ8.1.7.38, RQ8.1.7.40, RQ8.1.7.42, RQ8.1.7.47, RQ8.1.7.48, RQ8.1.7.49, RQ8.1.7.50, RQ8.1.7.51, RQ8.1.7.52, RQ8.1.7.53, RQ8.1.7.54, RQ8.1.7.56, RQ8.1.7.58, RQ8.1.7.59 , RQ8.1.7.61, RQ8.1.7.62, RQ8.1.7.63, RQ8.1.7.64, RQ8.1.7.65, RQ8.1.7.66, RQ8.1.7.67 and RQ8.1.7.68 is FFS.	

8.1.8 RFM Parameters

The test requirements are extracted from section 8.8 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.8.1	RFM Parameters PE shall appear after PE containing the related SD.
RQ8.1.8.1b	RFM Parameters PE shall appear after PE containing the related ADF.
RQ8.1.8.2	RFM Parameters PE is optional and may be used several times.
RQ8.1.8.3	The securityDomainAID object is optional. If present an RFM instance shall be associated with the referenced SD. If not present, the RFM instance shall be associated with the MNO-SD.
RQ8.1.8.4	A RFM instance shall be addressable with a given TAR values.
RQ8.1.8.5	A RFM instance shall be associated with at most one ADF.
RQ8.1.8.6	RFM Parameters PE shall contain PEHeader object.
RQ8.1.8.7	RFM Parameters PE may contain securityDomainAID of ApplicationIdentifier type, tag 15.
RQ8.1.8.8	RFM Parameters may contain tarList as a sequence of OCTET STRING of size 3, tag 0.
RQ8.1.8.8a	tarList shall include at least one TAR if available.
RQ8.1.8.8b	In case tarList is not available the TAR value defined within bytes 13-15 of the instanceAID is used.
RQ8.1.8.9	RFM Parameters shall contain minimumSecurityLevel of OCTET STRING of size 1, tag 1.
RQ8.1.8.10	The Minimum Security Level (MSL) for the RFM instance shall be interpreted according to ETSI TS 102 226.
RQ8.1.8.11	RFM Parameters shall contain uiccAccessDomain of OCTET STRING of variable size.
RQ8.1.8.12	RFM Parameters shall contain uiccAdminAccessDomain field of OCTET STRING of variable size.
RQ8.1.8.13	RFM Parameters may contain adfRFMAccess of ADFRFMAccess type.
RQ8.1.8.14	ADFRFMAccess object shall contain adfAID of ApplicationIdentifier type.
RQ8.1.8.15	ADFRFMAccess object shall contain adfAccessDomain of OCTET STRING of variable size.
RQ8.1.8.16	ADFRFMAccess object shall contain adfAdminAccessDomain of OCTET STRING of variable size.
RQ8.1.8.17	If `adfRFMAccess` is not provided, the RFM instance shall be linked only to the MF.
RQ8.1.8.18	If `adfRFMAccess` is provided, corresponding ADF shall be selected by default in the context of an RFM script.
RQ8.1.8.19	If `adfRFMAccess` is not provided, the MF shall be selected by default in the context of an RFM script.
RQ8.1.8.20	RFM Parameters PE shall contain instanceAID of ApplicationIdentifier type, tag 15
NOTE: Testing of these RQs is FFS.	

8.1.9 Non standardised content

The test requirements are extracted from section 8.9 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.9.1	The Profile Package can use as many PE-NonStandard profile elements as required.
RQ8.1.9.2	PE-NonStandard shall contain a "nonstandard-header" object. The type of the "nonstandard-header" object is PEHeader.
RQ8.1.9.3	PE-NonStandard shall contain an "issuerID" object. The type of the issuerID shall be OBJECT IDENTIFIER.
RQ8.1.9.4	PE-NonStandard shall contain "content" object. The type of the content shall be OCTET STRING.
Note; RQ8.1.9.1, RQ8.1.9.2, RQ8.1.9.3 and RQ8.1.9.4 are out of scope of this specification.	

8.1.10 Profile Package end

The test requirements are extracted from section 8.10 of "eUICC Profile Package: Interoperable Format Technical Specification" [SA PP TS].

RQ8.1.10.1	The PE-End shall contain an "end-header" object. The type of the "end-header" object is PE Header.
RQ8.1.10.2	The support of PE-End is mandatory for eUICC.
RQ8.1.10.3	The PE shall be used as the last element of the Profile Package.

8.1.11 eUICC Response type

The test requirements are extracted from section 8.11 and 9.5.2 of “eUICC Profile Package: Interoperable Format Technical Specification” [SA PP TS].

RQ8.1.11.1	EUICCResponse object shall contain peStatus field of SEQUENCE OF PEStatus type.
RQ8.1.11.2	EUICCResponse object may contain profileInstallationAborted field of NULL type.
RQ8.1.11.2a	When profileInstallationAborted is used, it shall be present in the last EUICCResponse sent by the eUICC.
RQ8.1.11.3	EUICCResponse object may contain statusMessage field of UTF8String type.
RQ8.1.11.4	PEStatus object shall contain status field of INTEGER type.
RQ8.1.11.5	PEStatus object may contain identification field of Uint15 type.
RQ8.1.11.6	The identification field, if present, shall indicate the identification number of the PE triggering the error.
RQ8.1.11.7	The identification field shall be present if any of following statuses are reported: - PE-not-supported. - bad-values.
RQ8.1.11.8	PEStatus object may contain additional-information field of Uint8 type.
RQ8.1.11.9	EUICCResponse with ok status shall be sent at the end of the profile installation when the profile has been processed successfully, and only if there is nothing to report.
RQ8.1.11.10	EUICCResponse with ok status shall not indicate any PE identification.
RQ8.1.11.11	EUICCResponse with PE-not-supported status shall be sent if a specific PE is not supported by the eUICC.
RQ8.1.11.12	EUICCResponse with PE-not-supported status shall include profileInstallationAborted tag if an unsupported PE is indicated as “mandated”.
RQ8.1.11.13	In case of profile installation failure due to internal memory issue, EUICCResponse with memory-failure status shall be sent.
RQ8.1.11.13a	If memory-failure is reported, the eUICC shall abort profile installation.
RQ8.1.11.14	bad-values status shall be sent if any of values is out of its acceptable value range.
RQ8.1.11.14a	In the case where bad-values is reported, eUICC may abort profile installation if it is not able to recover the error
RQ8.1.11.15	If eUICC has not enough free memory to install the Profile, eUICCResponse with not-enough-memory status shall be sent.
RQ8.1.11.15a	If the eUICC runs out of memory during processing PE-MF, it shall abort profile installation.
RQ8.1.11.15b	If the eUICC runs out of memory during processing a PE with “mandated” flag set, it shall abort profile installation.
RQ8.1.11.16	If eUICC finds a structure of a PE unkown or badly formatted, eUICCResponse with invalid-request-format status shall be sent.
RQ8.1.11.16a	invalid-request-format status code shall be used to indicate the incorrect order of the PEs. Note: It is not required that the eUICC is able to detect and reject all the incorrect order of the PEs or all invalid formats.
RQ8.1.11.16b	The eUICC shall abort profile installation if invalid-request-format error is triggered by any of following PEs: - PE-AKA-Parameters - PE-CSIM-Parameters - PE-PIN-Code - PE-PUK-Code - PE-Security-Domain - PE-RFM-Parameters For other PE-s the eUICC may abort profile installation in case invalid-request-format error is triggered and the eUICC is not able to recover the error.
RQ8.1.11.17	If eUICC does not support a parameter in a particular PE, eUICCResponse with invalid-parameter shall be sent.
RQ8.1.11.17a	The eUICC shall abort profile installation if invalid-parameter error is triggered by any of following PEs: - PE-AKA-Parameters - PE-CSIM-Parameters - PE-PIN-Code - PE-PUK-Code - PE-Security-Domain - PE-RFM-Parameters For other PE-s the eUICC may abort profile installation if an invalid parameter is detected, and the eUICC is not able to recover the error
RQ8.1.11.18	If any PE-Application in the Profile requires a runtime environment that is not supported by the eUICC, EUICCResponse with runtime-not-supported status shall be sent.
RQ8.1.11.18a	If an unsupported runtime environment is requested by a PE with “mandated” flag set the eUICC shall abort profile installation.
RQ8.1.11.19	If any PE-Application in the Profile depends on a library that is not available in the eUICC, EUICCResponse with lib-not-supported status shall be sent.

RQ8.1.11.19a	If a missing library is requested by a PE with “mandated” flag set the eUICC shall abort profile installation.
RQ8.1.11.20	If a generic file system template indicated by OID is not supported by the eUICC, EUICCResponse with template-not-supported status shall be sent.
RQ8.1.11.20a	template-not-supported status shall be sent if a file system template PE contained in the profile package is not supported.
RQ8.1.11.20b	If the eUICC does not support any of the file system templates identified in the Profile Header the eUICC shall abort profile installation.
RQ8.1.11.20c	In case a file system template PE triggering the template-not-supported error has “mandated” flag set the eUICC shall abort profile installation.
RQ8.1.11.21	feature-not-supported status shall be sent if the profile header mentions a feature the eUICC does not support.
RQ8.1.11.22	feature-not-supported status shall be sent if Optional USIM EFs PE contains any of EF GBABP, EF MSK, EF MUK, EF GBANL and EF NAFKCA and respective services are not supported at the eUICC operating system level. In this case, PEStatus object shall contain additional-information field set to ‘1’ if GBA is not supported, to ‘2’ if MBMS if not supported and ‘3’ if both are not supported.
RQ8.1.11.22a	The eUICC may send feature-not-supported status, if a PE requests a feature that the eUICC does not support (e.g. PE-Security-Domain contains a key of unsupported key type).
RQ8.1.11.22b	The eUICC shall abort profile installation, if the feature-not-supported error is triggered by Profile Header
RQ8.1.11.23	If a major version indicated in the Profile header is not supported by the eUICC, EUICCResponse with unsupported-profile-version status shall be sent (in the respect of specified versions) and the eUICC shall abort the profile installation.
RQ8.1.11.24	VOID
RQ8.1.11.25	If the installation of the Profile is aborted EUICCResponse shall contain profileInstallationAborted tag.
NOTE 1: RQ8.1.11.1 is implicitely tested everytime UICC response with PEStatus is sent. NOTE 2: Testing of RQ8.1.11.2, RQ8.1.11.2a, RQ8.1.11.3, RQ8.1.11.4, RQ8.1.11.5, RQ8.1.11.6, RQ8.1.11.7, RQ8.1.11.8, RQ8.1.11.10, RQ8.1.11.11, RQ8.1.11.12, RQ8.1.11.13, RQ8.1.11.13a, RQ8.1.11.14, RQ8.1.11.15a, RQ8.1.11.18, RQ8.1.11.18a, RQ8.1.11.20, RQ8.1.11.20a, RQ8.1.11.20b, RQ8.1.11.20c, RQ8.1.11.22, RQ8.1.11.22a and RQ8.1.11.23 is FFS.	

8.2 Test cases / scenarios

8.2.1 Check Profile Format

8.2.1.1. Installing PE-MF, PE-USIM and PE-OPT-USIM when eUICC supports file system creation by generic file manager

8.2.1.1.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_OPT-USIM (Generic File Managment)	6.12.1.5.4
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.1.2. Initial Conditions

None.

8.2.1.1.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10 .	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.8 RQ8.1.1.9 RQ8.1.1.12 RQ8.1.1.13b RQ8.1.1.16 RQ8.1.1.25 RQ8.1.1.26 RQ8.1.1.26b RQ8.1.1.27 RQ8.1.1.28a RQ8.1.1.29a RQ8.1.1.33a RQ8.1.1.34 RQ8.1.1.39 RQ8.1.2.1 RQ8.1.2.2 RQ8.1.2.3 RQ8.1.2.4 RQ8.1.2.6 RQ8.1.2.8

			RQ8.1.2.9 RQ8.1.2.12 RQ8.1.2.13 RQ8.1.2.14 RQ8.1.2.15 RQ8.1.3.6 RQ8.1.3.15 RQ8.1.3.20 RQ8.1.3.21 RQ8.1.10.1 RQ8.1.10.2 RQ8.1.10.3
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T ↔ eUICC	Select all files in PE MF and verify their FCPs	
5	T ↔ eUICC	Read all files in PE MF and verify that the content is the same as defined in the PE MF (Generic File Management) 6.12.1.2.2	
6	T ↔ eUICC	Select all files in PE USIM and verify their FCPs	
7	T ↔ eUICC	Read all files in PE USIM and verify that the content is the same as defined in the PE USIM (Generic File Management) 6.12.1.5.2	

8.2.1.2. Installing PE-MF, PE-USIM and PE-OPT-USIM when eUICC supports file system creation by template

8.2.1.2.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader 2	6.12.2.3.1
PE-MF (Template)	6.12.1.2.1
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Template)	6.12.1.5.1
PE_OPT-USIM (Template)	6.12.1.5.3
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.2.2. Initial Conditions

None.

8.2.1.2.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10 .	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.8 RQ8.1.1.9 RQ8.1.1.12 RQ8.1.1.13a RQ8.1.1.16

			RQ8.1.1.25 RQ8.1.1.26 RQ8.1.1.26b RQ8.1.1.27 RQ8.1.1.28a RQ8.1.1.29a RQ8.1.1.33a RQ8.1.1.34 RQ8.1.1.39 RQ8.1.2.1 RQ8.1.2.2 RQ8.1.2.3 RQ8.1.2.4 RQ8.1.2.6 RQ8.1.2.8 RQ8.1.2.9 RQ8.1.2.12 RQ8.1.2.13 RQ8.1.2.14 RQ8.1.2.15 RQ8.1.3.3 RQ8.1.3.4 RQ8.1.3.5 RQ8.1.3.10 RQ8.1.3.14 RQ8.1.3.16 RQ8.1.3.17 RQ8.1.10.1 RQ8.1.10.2 RQ8.1.10.3
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T ↔ eUICC	Select all files in PE MF and verify their FCPs	
5	T ↔ eUICC	Read all files in PE MF and verify that the content is the same as defined in the PE MF (Template) 6.12.1.2.1	
6	T ↔ eUICC	Select all files in PE USIM and verify their FCPs	
7	T ↔ eUICC	Read all files in PE USIM and verify that the content is the same as defined in the PE USIM (Template) 6.12.1.5.1	

8.2.1.3. Installing PE-USIM when eUICC does not support USIM

8.2.1.3.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.3.2. Initial Conditions

None.

8.2.1.3.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile according to 6.10.	RQ7.1.1.7 RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.9 RQ8.1.1.12 RQ8.1.2.1 RQ8.1.2.2
2	eUICC → T	eUICC responses with PEStatus (10) feature-not-supported eUICC response contains profileInstallationAborted tag.	RQ8.1.11.21 RQ8.1.11.22b RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.11 fail.	

8.2.1.4. Installing profile without ProfileHeader PE

8.2.1.4.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.4.2. Initial Conditions

None.

8.2.1.4.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10.	RQ8.1.1.12 RQ8.1.2.1
2	eUICC → T	eUICC responses with PEStatus (5) invalid-request-format. eUICCResponse contains profileInstallationAborted tag.	RQ8.1.11.16 RQ8.1.11.16b RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.11 fail	

8.2.1.5. Installing profile with PE-USIM before PE-MF, eUICC reports error.

8.2.1.5.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE_USIM (Generic File Management)	6.12.1.5.2
PE_MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.5.2. Initial Conditions

None.

8.2.1.5.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10 .	RQ8.1.1.16
2	eUICC → T	eUICC responses with PEStatus (5) invalid-request-format. eUICC response contains profileInstallationAborted tag.	RQ8.1.11.16a RQ8.1.11.16b RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.11 fail	

8.2.1.6. Installing profile with PE-Application before PE-SecurityDomain, eUICC reports error.

8.2.1.6.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE_MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_Application	6.12.1.10
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.6.2. Initial Conditions

None.

8.2.1.6.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10.	RQ8.1.1.36
2	eUICC → T	eUICC responses with PEStatus (5) invalid-request-format or PEStatus (6) invalid-parameter. eUICC response contains profileInstallationAborted tag.	RQ8.1.6.6 RQ8.1.11.16a RQ8.1.11.16b RQ8.1.11.17 RQ8.1.11.17a RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.11 fail	

8.2.1.7. Installing profile with PE-RFM before PE-SecurityDomain, eUICC reports error.

8.2.1.7.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE_MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_RFM	6.12.1.11
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_END	6.12.1.12

8.2.1.7.2. Initial Conditions

None.

8.2.1.7.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.1.37
2	eUICC → T	eUICC responses with PEStatus (5) invalid-request-format or PEStatus (6) invalid-parameter. eUICC response contains profileInstallationAborted tag.	RQ8.1.6.6 RQ8.1.11.16a RQ8.1.11.16b RQ8.1.11.17 RQ8.1.11.17a RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.116.11 fail	

8.2.1.8. Installing profile with PE-USIM before PE-MF.

8.2.1.8.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE_USIM (Generic File Management)	6.12.1.5.2
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.1.8.2. Initial Conditions

None.

8.2.1.8.3. Test Procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10 .	RQ8.1.1.16
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T ↔ eUICC	Select all files in PE MF and verify their FCPs	
5	T ↔ eUICC	Read all files in PE MF and verify that the content is the same as defined in the PE MF (Generic File Management) 6.12.1.2.2	
6	T ↔ eUICC	Select all files in PE USIM and verify their FCPs	
7	T ↔ eUICC	Read all files in PE USIM and verify that the content is the same as defined in the PE USIM (Generic File Management) 6.12.1.5.2	

8.2.1.9. Installing profile with PE-Application before PE-SecurityDomain, eUICC supports the installation.

FFS

8.2.1.10. Installing profile with PE-RFM before PE-SecurityDomain, eUICC supports the installation.

FFS

8.2.2 Check PE Security Domain

8.2.2.1. Check mandatory elements in PE Security Domain

This test shall check all the mandatory objects.

8.2.2.1.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.2.1.2. Initial conditions

None.

8.2.2.1.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.6.1 RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = '80' P2 = '02' Data = '5C 05 4F 9F 70 C5 C4 '	RQ8.1.6.4 RQ8.1.6.5 RQ8.1.6.7
5	eUICC → T	GET STATUS command responses with • AID of MNO-SD (#instanceAID) • Life cycle state (#lifeCyleState) • Privileges (#applicationPrivileges) • Application Executable Load file AID (#applicationLoadPackageAID) • • SW='9000'	

8.2.2.2. Check key list in PE Security Domain

This test shall check if the optional key list object is correctly processed.

8.2.2.2.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.2.2.2. Initial conditions

None

8.2.2.2.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.6.1
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T → eUICC	Send GET DATA command to MNO-SD with P1 = '00' P2 = 'E0'	RQ8.1.6.1 RQ8.1.6.2 RQ8.1.6.3 RQ8.1.6.4 RQ8.1.6.5 RQ8.1.6.8 RQ8.1.6.10 RQ8.1.6.11 RQ8.1.6.12 RQ8.1.6.13 RQ8.1.6.14 RQ8.1.6.15 RQ8.1.6.16 RQ8.1.6.17 RQ8.1.6.18 RQ8.1.6.26
5	eUICC → T	GET DATA command responses with • key information data containing `#keyIdentifier`, `#keyVersionNumber` and `#keyType`. • SW='9000'	

6	T → eUICC	Send GET_STATUS command using SCP80 to MNO-SD with P1 = '80' P2= '02' Data = '5C 05 4F 9F 70 C5 C4'	RQ8.1.6.4 RQ8.1.6.5 RQ8.1.6.7 RQ8.1.6.18 RQ8.1.6.34 RQ8.1.6.35 RQ8.1.6.37 RQ8.1.6.38a
7	eUICC → T	GET STATUS command responses with • AID of MNO-SD (`#instanceAID`) • Life cycle state (`#lifeCyleState`) • Privileges (`#applicationPrivileges`) • Application Executable Load file AID (`#applicationLoadPackageAID`) • SW='9000'	
8	T	1) Decrypt the response packet with the #SCP80_ENC_KEY 2) Verify the cryptographic checksum using #SCP80_AUTH_KEY	

8.2.2.3. Check number of keyComponent objects

This test shall check if keyComponent is assigned just once per key.

8.2.2.3.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.2.1.2
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.2.3.2. Initial conditions

None

8.2.2.3.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according. To 6.10.	RQ8.1.6.27
2	eUICC → T	eUICC response shall contain at least one PEStatus different from ok (0)	

8.2.2.4. Check sdPersoData

This test shall check if sdPersoData is processed.

8.2.2.4.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.2.1.3
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.2.4.2. Initial conditions

None

8.2.2.4.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.6.1
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T → eUICC	Send GET DATA command to MNO-SD with P1 = '00' P2 = '42' (Issuer Identification Number)	RQ8.1.6.28 RQ8.1.6.29 RQ8.1.6.30 RQ8.1.6.31 RQ8.1.6.46
5	eUICC → T	GET DATA command responses with • IIN out of `#sdPersoData`. • SW='9000'	
6	T → eUICC	Send GET DATA command to MNO-SD with P1 = '00' P2 = '45' (Card Image Number)	RQ8.1.6.28 RQ8.1.6.29 RQ8.1.6.30 RQ8.1.6.31
7	eUICC → T	GET DATA command responses with • CIN out of #sdPersoData. • SW='9000'	

8.2.2.5. Check OTA HTTPs Personalisation

This test shall check if MNO_SD is personalised with OTA HTTPs Data.

8.2.2.5.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_PINCodes	6.12.1.6
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.2.1.4
PE_SecurityDomain	6.12.1.9
PE_Application	6.12.1.10
PE_RFM	6.12.1.11
PE_END	6.12.1.12

8.2.2.5.2. Initial conditions

None

8.2.2.5.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.6.1
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile according to 6.11	
4	T → eUICC	Send GET DATA command to MNO-SD with P1 = '00' P2 = '85'	RQ8.1.6.36a RQ8.1.7.45
5	eUICC → T	GET DATA command responses with • Security Domain Administration Session Parameters contained in `#processData`. • SW='9000'	

8.2.3 Check PE Application

8.2.3.1. Check Application PE (PE_Applet) and mandatory elements in ApplicationInstance

8.2.3.1.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.1.10
PE_END	6.12.1.12

8.2.3.1.2. Initial conditions

None

8.2.3.1.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.9 RQ8.1.7.4 RQ8.1.7.5 RQ8.1.7.7 RQ8.1.7.8 RQ8.1.7.9 RQ8.1.7.20 RQ8.1.7.21 RQ8.1.7.22 RQ8.1.7.23 RQ8.1.7.24 RQ8.1.7.25 RQ8.1.7.26 RQ8.1.7.27 RQ8.1.7.31 RQ8.1.7.32 RQ8.1.7.33 RQ8.1.7.34a RQ8.1.7.35 RQ8.1.7.36 RQ8.1.7.39 RQ8.1.7.55
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile (see description in 6.11)	
4	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = ‘40’ P2 = ‘02’ Data =’4F LL #instanceAID 5C 05 4F 9F 70 C5 C4’	RQ8.1.6.7

5	eUICC → T	GET STATUS command responses with • AID of application (`#instanceAID`) • Life cycle state (`#lifeCyleState`) • Privileges (`#applicationPrivileges`) • Application Executable Load file AID (`#applicationLoadPackageAID`) • • SW='9000'	

8.2.3.2. Check all elements in ApplicationLoadPackage – taking size into account.

8.2.3.2.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.2.2.2
PE_END	6.12.1.12

8.2.3.2.2. Initial conditions

None

8.2.3.2.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.7.8 RQ8.1.7.9 RQ8.1.7.10 RQ8.1.7.11 RQ8.1.7.12 RQ8.1.7.13 RQ8.1.7.14 RQ8.1.7.15 RQ8.1.7.16 RQ8.1.7.17 RQ8.1.7.18 RQ8.1.7.19 RQ8.1.7.20 RQ8.1.7.21
2	eUICC → T	If O_MEMORY_LIMIT , the eUICC responses with PEStatus (4) not-enough-memory and the eUICC response contains profileInstallationAborted tag If not O_MEMORY_LIMIT, the eUICC responses with PEStatus (0) ok or with PEStatus (6) invalid-parameter	RQ8.1.11.15 RQ8.1.11.15b RQ8.1.11.25 RQ8.1.11.9 RQ8.1.11.17
3	T ↔ eUICC	If O_MEMORY_LIMIT enabling the Test Profile according to 6.11 fail. If not O_MEMORY_LIMIT the result of enabling the Test Profile according to 6.11 is unspecified.	

8.2.3.3. Check all elements in ApplicationInstance

8.2.3.3.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.2.2.3
PE_END	6.12.1.12

8.2.3.3.2. Initial conditions

None

8.2.3.3.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.7.4 RQ8.1.7.5 RQ8.1.7.22 RQ8.1.7.23 RQ8.1.7.24 RQ8.1.7.25 RQ8.1.7.26 RQ8.1.7.27 RQ8.1.7.28 RQ8.1.7.29 RQ8.1.7.30 RQ8.1.7.31 RQ8.1.7.32 RQ8.1.7.33 RQ8.1.7.34a RQ8.1.7.35 RQ8.1.7.36 RQ8.1.7.37 RQ8.1.7.38 RQ8.1.7.39 RQ8.1.7.40 RQ8.1.7.41 RQ8.1.7.42 RQ8.1.7.47 RQ8.1.7.48 RQ8.1.7.50 RQ8.1.7.51 RQ8.1.7.52 RQ8.1.7.55 RQ8.1.7.58
2	eUICC → T	eUICC responses with PEStatus (0) ok	RQ8.1.11.9
3	T ↔ eUICC	Enable Test Profile (see description in 6.11)	
4	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = ‘40’ P2 = ‘02’ Data =’4F LL #instanceAID 5C 05 4F 9F 70 C5 C4’	RQ8.1.6.7

5	eUICC → T	GET STATUS command responses with • AID of application (#instanceAID) • Life cycle state (#lifeCyleState) • Privileges (#applicationPrivileges) • Application Executable Load file AID (#applicationLoadPackageAID) SW='9000'	

8.2.3.4. Error when load a PE-Applet4 and bad library is provided.

8.2.3.4.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.2.2.4
PE_END	6.12.1.12

8.2.3.4.2. Initial conditions

None

8.2.3.4.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.9 RQ8.1.7.1 RQ8.1.7.7 RQ8.1.7.8 RQ8.1.7.9
2	eUICC → T	eUICC response with PEStatus (8) lib-not-supported and the eUICC response contains profileInstallationAborted tag	RQ8.1.11.19 RQ8.1.11.19a RQ8.1.11.25
3	T ↔ eUICC	Enabling Test Profile according to 6.11 fail	

8.2.3.5. Check multiple ApplicationInstance.

8.2.3.5.1. Test execution

This test is executed only if multiple instances are supported

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.2.2.5
PE_END	6.12.1.12

8.2.3.5.2. Initial conditions

None

8.2.3.5.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.9 RQ8.1.7.3 RQ8.1.7.7 RQ8.1.7.8 RQ8.1.7.9 RQ8.1.7.20 RQ8.1.7.21 RQ8.1.7.22 RQ8.1.7.23 RQ8.1.7.24 RQ8.1.7.25 RQ8.1.7.26 RQ8.1.7.27 RQ8.1.7.31 RQ8.1.7.32 RQ8.1.7.33 RQ8.1.7.35 RQ8.1.7.36
2	eUICC → T	eUICC response with PEStatus (0) ok	RQ8.1.11.9
3	T → eUICC	Enable Test Profile (see description in 6.11)	
4	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = '40' P2 = '02' Data ='4F LL #instanceAID 5C 05 4F 9F 70 C5 C4' (first application)	RQ8.1.6.7

5	eUICC → T	GET STATUS command responses with • AID of application1 (#instanceAID) • Life cycle state (#lifeCyleState) • Privileges (#applicationPrivileges) • Application Executable Load file AID (#applicationLoadPackageAID) • SW='9000'	
6	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = '40' P2 = '02' Data ='4F LL #instanceAID (second application) 5C 06 4F 9F 70 C5 C4 84'	RQ8.1.6.7 RQ8.1.7.3
7	eUICC → T	GET STATUS command responses with • AID of application2 (#instanceAID) • Life cycle state (#lifeCyleState) • Privileges (#applicationPrivileges) • Application Executable Load file AID (#applicationLoadPackageAID) • SW='9000'	

8.2.3.6. Check processData.

8.2.3.6.1. Test execution

The Test Profile is defined as follows:

TYPE	VALUE or REFERENCE
ProfileHeader	6.12.1.1
PE-MF (Generic File Management)	6.12.1.2.2
PE_PUKCodes	6.12.1.3
PE_PINCodes	6.12.1.4
PE_USIM (Generic File Management)	6.12.1.5.2
PE_AKAParameters	6.12.1.7
PE_SecurityDomain	6.12.1.8
PE-Application	6.12.2.2.5
PE_END	6.12.1.12

8.2.3.6.2. Initial conditions

None

8.2.3.6.3. Test procedure

Step	Direction	Description	RQ
1	T → eUICC	Load Test Profile to the eUICC according to 6.10	RQ8.1.1.6 RQ8.1.1.7 RQ8.1.1.9 RQ8.1.7.6 RQ8.1.7.8 RQ8.1.7.9 RQ8.1.7.20 RQ8.1.7.21 RQ8.1.7.22 RQ8.1.7.23 RQ8.1.7.24 RQ8.1.7.25

			RQ8.1.7.26 RQ8.1.7.27 RQ8.1.7.31 RQ8.1.7.32 RQ8.1.7.33 RQ8.1.7.34a RQ8.1.7.35 RQ8.1.7.36 RQ8.1.7.39 RQ8.1.7.40 RQ8.1.7.41 RQ8.1.7.42 RQ8.1.7.43 RQ8.1.7.44 RQ8.1.7.45 RQ8.1.7.46 RQ8.1.7.55 RQ8.1.7.60
2	eUICC → T	eUICC response with PEStatus (0) ok	RQ8.1.11.9
3	T → eUICC	Enable Test Profile (see description in 6.11)	
4	T → eUICC	Send GET STATUS command to MNO-SD using SCP80 with P1 = ‘40’ P2 = ‘02’ Data =’4F LL #instanceAID 5C 05 4F 9F 70 C5 C4’	RQ8.1.6.7
5	eUICC → T	GET STATUS command responses with • AID of application1 (#instanceAID) • Life cycle state (#lifeCyleState) • Privileges (#applicationPrivileges) • Application Executable Load file AID (#applicationLoadPackageAID) • SW=’9000’	
6	T → eUICC	Send GET DATA command to TAR Application 6 using SCP80 with P1 = ‘00’ P2 = ’92’ Lc = ‘00’ Le = ‘00’	
7	eUICC → T	GET DATA command responses with • #processData information	

9. ANNEX A (Informative) : Document history

The table below indicates changes that have been incorporated into the present document since it was created by SIMalliance.

Version	Date	Brief Description of Change
V1.0.	14/04/2016	1st Release of Document
V2.0.	06/07/2016	-Test PE-s are updated in Ch 6.12 to align to eUICC Profile Package: Interoperable Format Technical Specification v2.0; also new Test PE-s are added -RQs are updated in Ch 7.1 and 8.1 to align to eUICC Profile Package: Interoperable Format Technical Specification v2.0; also new RQs are added -Test cases are updated, especially new Test PE-s are referenced -New test cases are added: 8.2.1.1; 8.2.1.8; 8.2.3.6 -References are updated, applicability table and related chapters are updated, Ch 6.7 is updated