

**Remote Provisioning Architecture for Embedded UICC Test**

**Specification**

**Version 3.4**

**16 February 2023**

#### This Industry Specification is a Non-binding Permanent Reference Document of the GSMA

###### Security Classification: Non-confidential

Access to and distribution of this document is restricted to the persons permitted by the security classification. This document is confidential to the Association and is subject to copyright protection. This document is to be used only for the purposes for which it has been supplied and information contained in it must not be disclosed or in any other way made available, in whole or in part, to persons other than those permitted under the security classification without the prior written approval of the Association.

**Copyright Notice**

Copyright © 2023 GSM Association

**Disclaimer**

The GSM Association (“Association”) makes no representation, warranty or undertaking (express or implied) with respect to and does not accept any responsibility for, and hereby disclaims liability for the accuracy or completeness or timeliness of the information contained in this document. The information contained in this document may be subject to change without prior notice.

**Compliance Notice**

The information contain herein is in full compliance with the GSM Association’s antitrust compliance policy.

This Permanent Reference Document is classified by GSMA as an Industry Specification, as such it has been developed and is maintained by GSMA in accordance with the provisions set out in GSMA AA.35 - Procedures for Industry Specifications.

V3.4 Page 1 of 590

**Table of Contents**

1. [Introduction 6](#_bookmark0)
   1. [Overview 6](#_bookmark1)
   2. [Scope 6](#_bookmark2)
   3. [Definition of Terms 6](#_bookmark3)
   4. [Abbreviations 9](#_bookmark4)
   5. [Document Cross-references 12](#_bookmark5)
   6. [Conventions 13](#_bookmark22)
2. [Testing Rules 14](#_bookmark23)
   1. [Applicability 14](#_bookmark24)
      1. [Format of the Optional Features Table 14](#_bookmark25)
      2. [Format of the Applicability Table 14](#_bookmark26)
      3. [Applicability and Notations 14](#_bookmark27)
      4. [Optional Features Table 15](#_bookmark28)
      5. [Applicability Table 15](#_bookmark30)
   2. [General Consideration 20](#_bookmark32)
      1. [Test Cases Definition 20](#_bookmark33)
      2. [Test Cases Format 20](#_bookmark34)
      3. [Using of Methods, Constants and Dynamic Content 23](#_bookmark35)
      4. [Commands and Responses 23](#_bookmark36)
      5. [Referenced Requirements 23](#_bookmark37)
      6. [Pass Criterion 23](#_bookmark38)
      7. [Future Study 24](#_bookmark39)
3. [Testing Architecture 25](#_bookmark40)
   1. [Testing Scope 25](#_bookmark41)
   2. [Testing Execution 26](#_bookmark42)
      1. [Interfaces Compliancy 26](#_bookmark43)
      2. [System Behaviour 28](#_bookmark44)
   3. [Void 31](#_bookmark46)
4. [Interface Compliancy Testing 32](#_bookmark47)
   1. [General Overview 32](#_bookmark48)
   2. [eUICC Interfaces 32](#_bookmark49)
      1. [Generic Sub-sequences 32](#_bookmark50)
      2. [OTA Transport Protocols 38](#_bookmark58)
      3. [ES5 (SM-SR – eUICC): CreateISDP 43](#_bookmark62)
      4. [ES5 (SM-SR – eUICC): EnableProfile 49](#_bookmark68)
      5. [ES5 (SM-SR – eUICC): DisableProfile 56](#_bookmark75)
      6. [ES5 (SM-SR – eUICC): SetFallbackAttribute 65](#_bookmark80)
      7. [ES5 (SM-SR – eUICC): DeleteProfile 70](#_bookmark84)
      8. [ES5 (SM-SR – eUICC): eUICCCapabilityAudit 79](#_bookmark88)
      9. [ES5 (SM-SR – eUICC): MasterDelete 92](#_bookmark92)
      10. [ES5 (SM-SR – eUICC): EstablishISDRKeySet 109](#_bookmark96)
      11. [ES5 (SM-SR – eUICC): FinaliseISDRhandover 121](#_bookmark100)
      12. [ES5 (SM-SR – eUICC): UpdateSMSRAddressingParameters 129](#_bookmark104)
      13. [ES5 (SM-SR – eUICC): Notification on Profile Enabling 137](#_bookmark107)
      14. [ES5 (SM-SR – eUICC): Notification on Profile Disabling 153](#_bookmark112)
      15. [ES6 (MNO – eUICC): UpdatePOL1byMNO 164](#_bookmark115)
      16. [ES6 (MNO – eUICC): UpdateConnectivityParametersByMNO 173](#_bookmark119)
      17. [ES8 (SM-DP – eUICC): EstablishISDPKeySet 180](#_bookmark121)
      18. [ES8 (SM-DP – eUICC): DownloadAndInstallation 196](#_bookmark127)
      19. [ES8 (SM-DP – eUICC): UpdateConnectivityParameters 208](#_bookmark132)
   3. [Off-card Interfaces 214](#_bookmark139)
      1. [ES1 (EUM – SM-SR): RegisterEIS 214](#_bookmark140)
      2. [ES2 (MNO – SM-DP): GetEIS 217](#_bookmark142)
      3. [ES2 (MNO – SM-DP): DownloadProfile 220](#_bookmark144)
      4. [ES2 (MNO – SM-DP): UpdatePolicyRules 224](#_bookmark146)
      5. [ES2 (MNO – SM-DP): UpdateSubscriptionAddress 227](#_bookmark148)
      6. [ES2 (MNO – SM-DP): EnableProfile 229](#_bookmark150)
      7. [ES2 (MNO – SM-DP): DisableProfile 234](#_bookmark153)
      8. [ES2 (MNO – SM-DP): DeleteProfile 238](#_bookmark155)
      9. [ES3 (SM-DP – SM-SR): GetEIS 242](#_bookmark157)
      10. [ES3 (SM-DP – SM-SR): AuditEIS 244](#_bookmark159)
      11. [ES3 (SM-DP – SM-SR): CreateISDP 245](#_bookmark161)
      12. [ES3 (SM-DP – SM-SR): SendData 247](#_bookmark163)
      13. [ES3 (SM-DP – SM-SR): UpdatePolicyRules 249](#_bookmark165)
      14. [ES3 (SM-DP – SM-SR): UpdateSubscriptionAddress 251](#_bookmark167)
      15. [ES3 (SM-DP – SM-SR): UpdateConnectivityParameters 253](#_bookmark169)
      16. [ES3 (SM-DP – SM-SR): EnableProfile 255](#_bookmark171)
      17. [ES3 (SM-DP – SM-SR): DisableProfile 258](#_bookmark173)
      18. [ES3 (SM-DP – SM-SR): DeleteISDP 261](#_bookmark175)
      19. [ES4 (MNO – SM-SR): GetEIS 264](#_bookmark177)
      20. [ES4 (MNO – SM-SR): UpdatePolicyRules 265](#_bookmark179)
      21. [ES4 (MNO – SM-SR): UpdateSubscriptionAddress 267](#_bookmark181)
      22. [ES4 (MNO – SM-SR): AuditEIS 269](#_bookmark183)
      23. [ES4 (MNO – SM-SR): EnableProfile 271](#_bookmark185)
      24. [ES4 (MNO – SM-SR): DisableProfile 274](#_bookmark187)
      25. [ES4 (MNO – SM-SR): DeleteProfile 277](#_bookmark189)
      26. [ES4 (MNO – SM-SR): PrepareSMSRChange 281](#_bookmark191)
      27. [ES4 (MNO – SM-SR): SMSRchange 283](#_bookmark193)
      28. [ES7 (SM-SR – SM-SR): HandoverEUICC 287](#_bookmark195)
      29. [ES7 (SM-SR – SM-SR): AuthenticateSMSR 289](#_bookmark197)
      30. [ES7 (SM-SR – SM-SR): CreateAdditionalKeySet 293](#_bookmark199)
5. [System Behaviour Testing 297](#_bookmark200)
   1. [General Overview 297](#_bookmark201)
   2. [eUICC Behaviour 297](#_bookmark202)
      1. [Device – eUICC 297](#_bookmark203)
      2. [LOCKED State Unsupported by ISD-R and ISD-P 298](#_bookmark205)
      3. [Components and Visibility 301](#_bookmark208)
      4. [Security and Responsibility 316](#_bookmark215)
      5. [Confidential Setup of MNO Secure Channel Keys 329](#_bookmark220)
      6. [Full Profile Installation Process 332](#_bookmark223)
   3. [Platform Behaviour 336](#_bookmark225)
      1. [eUICC Identity Check 336](#_bookmark226)
      2. [Profile Download and Installation Process 339](#_bookmark229)
      3. [Profile Enabling Process 351](#_bookmark233)
      4. [Profile Disabling Process 371](#_bookmark237)
      5. [Profile Deletion Process 390](#_bookmark240)
      6. [Master Delete Process 398](#_bookmark243)
      7. [SM-SR Change Process 398](#_bookmark244)
      8. [Update Connectivity Parameters Process 416](#_bookmark249)
6. [Test Specifications 419](#_bookmark251)
   1. [SIMAlliance eUICC Profile Package Test Specification 419](#_bookmark252)

[Annex A Reference Applications 420](#_bookmark253)

* 1. [Applet1 420](#_bookmark254)
     1. [Description 420](#_bookmark255)
     2. [AID 420](#_bookmark256)
     3. [Source Code (Java Card) 420](#_bookmark257)
  2. [Applet2 422](#_bookmark258)
     1. [Description 422](#_bookmark259)
     2. [AID 422](#_bookmark260)
     3. [Source Code (Java Card) 422](#_bookmark261)
  3. [Applet3 422](#_bookmark262)
     1. [Description 422](#_bookmark263)
     2. [AID 422](#_bookmark264)
     3. [Source Code (Java Card) 422](#_bookmark265)

[Annex B Constants 424](#_bookmark266)

* 1. [Hexadecimal Constants 424](#_bookmark267)
  2. [ASCII Constants 426](#_bookmark268)
  3. [eUICC Settings 428](#_bookmark269)
  4. [Platforms Settings 429](#_bookmark270)
  5. [RPS Elements 433](#_bookmark271)
  6. [Profiles Information 447](#_bookmark272)
  7. [Profile Package Description 449](#_bookmark273)
     1. [Profile Package Content 450](#_bookmark274)
     2. [Access Rules 463](#_bookmark277)
     3. [Additional Profile Elements 464](#_bookmark279)

[Annex C Dynamic Content 468](#_bookmark280)

[Annex D Methods 471](#_bookmark281)

[Annex E Commands and Responses 486](#_bookmark282)

* 1. [Commands 486](#_bookmark283)
  2. [Responses 505](#_bookmark284)

[Annex F Bearer Independent Protocol 518](#_bookmark285)

[Annex G CAT\_TP PDUs 520](#_bookmark286)

[Annex H TLS Records 523](#_bookmark288)

[Annex I Initial States 526](#_bookmark289)

[Annex J Requirements 530](#_bookmark290)

* 1. [Format of the Requirements Table 530](#_bookmark291)
  2. [Requirements in Scope 530](#_bookmark292)
  3. [Out of Scope Requirements 584](#_bookmark293)

1. [Document History 587](#_bookmark294)
   1. [Document Owner 590](#_bookmark295)

# Introduction

## Overview

The main aim of the GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) & [[2]](#_bookmark7) is to provide a technical description of the ‘over the air’ remote provisioning mechanism for machine-to-machine Devices.

This Test Plan provides a set of test cases to be used for testing the implementations of the GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) & [[2]](#_bookmark7). This document offers stakeholders a unified test strategy and ensures interoperability between different implementations.

## Scope

This document is intended for:

* Test tools and platforms’ suppliers
* Vendors (Device & eUICC Manufacturers)
* Operators

The Test Plan consists of a set of test cases relevant for testing all entities defined in the eUICC remote provisioning ecosystem. The testing scopes developed in this document are:

* Interface compliancy testing
* System behaviour testing

For each test case specified within this Test Plan, there is a reference to one or more requirements.

## Definition of Terms

|  |  |
| --- | --- |
| **Term** | **Description** |
| Actor | Physical entity (person, company or organization) that can assume a Role in the functional architecture. It is possible for an Actor to assume multiple Roles in the same functional architecture. |
| Connectivity Parameters | A set of data (e.g. SMS-C address) required by the eUICC to open a communication channel (e.g. SMS, HTTPS) on a dedicated network. |
| Device | Equipment into which an Embedded UICC and a communication module are inserted during assembly. Examples include Utility meter, car and camera. |
| Disabled (Profile) | The state of a Profile where all files and applications (e.g. NAA) present in the Profile are not selectable over the eUICC - Terminal interface. |
| Domain Name System | A internet protocol for translating domain names (or hostnames) into IP addresses. |
| Embedded UICC | A UICC which is not easily accessible or replaceable, is not intended to be removed or replaced in the Device, and enables the secure changing of Profiles. |
| Enabled (Profile) | The state of a Profile when its files and/or applications (e.g. NAA) are selectable over the UICC-Terminal interface. |

|  |  |
| --- | --- |
| **Term** | **Description** |
| eUICC Certificate | A certificate issued by the EUM for a specific eUICC. This certificate can be verified using the EUM Certificate. |
| eUICC Manufacturer | Supplier of the eUICCs and resident software (e.g. firmware and operating system). |
| EUM Certificate | A certificate issued to a GSMA accredited EUM which can be used to verify eUICC Certificates. |
| Executable Load File | An on-card container of one or more application's executable code as defined in GlobalPlatform Card Specification [[3].](#_bookmark8) |
| Executable Module | The on-card executable code of a single application present within an Executable Load File as defined in GlobalPlatform Card Specification [[3].](#_bookmark8) |
| Fall-back Attribute | This is an attribute of a Profile which, when set, identifies the Profile to be enabled by the Fall-back Mechanism. Only one Profile on the eUICC can have the Fall-back Attribute set at a time. |
| Fall-back Mechanism | eUICC based mechanism which enables the Profile with Fall-back Attribute set. |
| Integrated Circuit Card ID | Unique number to identify a Profile in an eUICC.  Note: the ICCID throughout this specification is used to identify the Profile. |
| International Mobile  Subscriber Identity | Unique identifier owned and issued by Mobile operators to (U)SIM applications to enable Devices to attach to a network and use services. |
| Issuer Security Domain | A security domain on the UICC as defined by GlobalPlatform Card Specification [[3].](#_bookmark8) |
| MNO-SD | Security domain part of the Profile, owned by the MNO, providing the Secured Channel to the MNO’s OTA Platform. It is used to manage the content of a Profile once the Profile is Enabled. |
| Mobile Network Operator | An entity providing access capability and communication services to its Customers through a mobile network infrastructure. |
| Network Access Application | An application residing on a UICC which provides authorization to access a network e.g. a USIM application. |
| OTA Keys | The credentials included in the Profile, used in conjunction with OTA Platforms. |
| OTA Platform | An MNO platform for remote management of UICCs and the content of Enabled MNO Profiles on eUICCs. |
| PIX | Proprietary application Identifier eXtension, the value of which is part of the AID. |
| Platform Management | A set of functions related to the enabling, disabling and deletion of a Profile and the transport of Profile Management functions to an eUICC. Platform Management actions are protected by Platform Management Credentials shared between the SM-SR and the ISD-R. Platform Management does not affect the content of a Profile. |
| Platform Management Credentials | Data required within an eUICC so that a secured communication can be set up between an external entity and the eUICC in order to enable, disable and delete Profiles on the eUICC and to transport Profile Management functions. |

|  |  |
| --- | --- |
| **Term** | **Description** |
| Policy | Principles reflected in a set of rules that governs the behaviour of eUICC and/or entities involved in the remote management of the eUICC. |
| Policy Rule | Defines the atomic action of a Policy and the conditions under which it is executed. |
| Profile | Combination of a file structure, data and applications to be provisioned onto, or present on, an eUICC and which allows, when Enabled, the access to a specific mobile network infrastructure. |
| Profile Component | A Profile Component is an element of the Profile and MAY be one of the following: |
| 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 (as defined in SIMAlliance eUICC Profile Package specification [[16]](#_bookmark21)). |
| Profile Management | A set of functions related to the downloading, installation and content update of a Profile in a dedicated ISD-P on the eUICC. Download and installation are protected by Profile Management Credentials shared between the SM-DP and the ISD-P. |
| Profile Management Credentials | Data required within an eUICC so that a Profile downloaded from an external entity can be decrypted and installed on the eUICC. |
| Profile Package | A personalised Profile using an interoperable description format transmitted to an eUICC in order to load and install a Profile (as defined in SIMAlliance eUICC Profile Package specification [[16]](#_bookmark21)). |
| RID | Registered Application Provider Identifier, the value of which is part of the AID. |
| Roles | Roles are representing a logical grouping of functions. |
| Root Certificate | Self-signed certificate of the CI, used to authenticate certificates issued to other entities. |
| Subscriber | An entity (associated with one or more users) that is engaged in a Subscription with a Telecommunication Service Provider. The Subscriber is allowed to subscribe and unsubscribe to services, to register a user or a list of users authorized to use those services, and also to set the limits relative to the use that associated users make of those services. |
| Subscription | Describes the commercial relationship between the Subscriber and the Telecommunication Service Provider. |
| Subscription Address | A unique network address, such as MSISDN, IMSI or SIP-URI, of a mobile Subscription within a mobile network. It is used to route messages, e.g. SMS, to the eUICC. |

|  |  |
| --- | --- |
| **Term** | **Description** |
| Subscription Manager Secure Routing | Role that securely performs functions of Platform Management commands and the transport of Profile Management commands. |
| Subscription Manager Data Preparation | Role that prepares the Profiles and manages the secure download and installation of these Profiles onto the eUICC. |
| Telecommunication Service Provider | The organization through which the Subscriber obtains PLMN telecommunication services. This is usually the network operator or possibly a separate body. |
| Test Plan | Current document describing the test cases that allow testing the eUICC Remote Provisioning Architecture. |

## Abbreviations

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| ADF | Application Dedicated File |
| AES | Advanced Encryption Standard |
| AID | Application Identifier |
| AKA | Authentication and Key Agreement |
| APDU | Application Protocol Data Unit |
| ASN.1 | Abstract Syntax Notation One |
| ATR | Answer To Reset |
| ATS | Answer To Select |
| BIP | Bearer Independent Protocol |
| C-APDU | Command APDU |
| CASD | Controlling Authority Security Domain |
| CAT\_TP | Card Application Toolkit Transport Protocol |
| CERT.DP.ECDSA | Certificate of the SM-DP for its ECDSA key |
| CERT.ECASD.ECKA | Certificate of the ECASD for its ECKA key |
| CERT.SR.ECDSA | Certificate of the SM-SR for its ECDSA key |
| CI | Certificate Issuer |
| CLA | Class byte of the command message |
| DER | Distinguished Encoding Rule |
| DF | Dedicated File |
| DGI | Data Grouping Identifier |
| DNS | Domain Name System |
| DR | Derivation Random |
| DS | Device Simulator |
| ECASD | eUICC Controlling Authority Security Domain |
| ECDSA | Elliptic Curve cryptography Digital Signature Algorithm |

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| ECKA | Elliptic Curve cryptography Key Agreement algorithm |
| EF | Elementary File |
| EID | eUICC-ID |
| EIS | eUICC Information Set |
| ePK.DP.ECKA | ephemeral Public Key of the SM-DP used for ECKA |
| ePK.SR.ECKA | ephemeral Public Key of the SM-SR used for ECKA |
| eSK.DP.ECKA | ephemeral Private Key of the SM-DP used for ECKA |
| eSK.SR.ECKA | ephemeral Private Key of the SM-SR used for ECKA |
| ETSI | European Telecommunications Standards Institute |
| eUICC | Embedded UICC |
| eUICC-UT | eUICC Under Test |
| EUM | eUICC Manufacturer |
| EUM-S | eUICC Manufacturer Simulator |
| EVT | Event |
| FFS | For Future Study |
| GSMA | GSM Association |
| HTTPS | HyperText Transfer Protocol Secure |
| ICCID | Integrated Circuit Card ID |
| IMEI | International Mobile Equipment Identity |
| IMSI | International Mobile Subscriber Identity |
| INS | Instruction byte of the command message |
| ISD | Issuer Security Domain |
| ISD-P | Issuer Security Domain Profile |
| ISD-R | Issuer Security Domain Root |
| ISO | International Organization for Standardization |
| MAC | Message Authentication Code |
| MEID | Mobile Equipment IDentifier |
| MF | Master File |
| MNO | Mobile Network Operator |
| MNO-S | MNO Simulator |
| MSL | Minimum Security Level |
| NAA | Network Access Application |
| NAN | Network Access Name |
| NPI | Numbering Plan Identifier |
| OID | Object IDentifier |
| OTA | Over The Air |
| P1 | Reference control parameter 1 |
| P2 | Reference control parameter 2 |

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| PDU | Protocol Data Unit |
| PE | Profile Element |
| PIN | Personal Identification Number |
| PIX | Proprietary application Identifier eXtension |
| PK.CI.ECDSA | Public Key of the CI in the ECASD for verifying certificate signatures |
| PK.DP.ECDSA | Public Key of the SM-DP, part of the CERT.DP.ECDSA, for verifying his signatures |
| PK.ECASD.ECKA | Public Key of the ECASD used for ECKA |
| PK.SR.ECDSA | Public Key of the SM-SR part of the CERT.SR.ECDSA, for verifying his signatures |
| PLMN | Public Land Mobile Network |
| POL1 | Policy Rules within the Profile |
| POL2 | Policy Rules associated to a Profile and stored in the relevant EIS at the SM-SR |
| POR | Proof Of Receipt |
| PSK | Pre-Shared Key |
| PUK | PIN Unblocking Key |
| R-APDU | Response APDU |
| REQ | Requirement |
| RFM | Remote File Management |
| R-MAC | Response MAC |
| RPS | GSMA Embedded UICC Remote Provisioning messages |
| SCP | Secure Channel Protocol |
| SD | Security Domain |
| SDIN | Security Domain Image Number |
| SDU | Service Data Unit |
| ShS | Shared Secret |
| SIM | Subscriber Identity Module |
| SIN | Security Domain Provider Identification Number |
| SK.CI.ECDSA | Private key of the CI for signing certificates |
| SK.DP.ECDSA | Private Key of the of SM-DP for creating signatures |
| SK.ECASD.ECKA | Private Key of the ECASD used for ECKA |
| SK.SR.ECDSA | Private Key of the SM-SR for creating signatures |
| SM | Subscription Manager |
| SM-DP | Subscription Manager Data Preparation |
| SM-DP-S | Subscription Manager Data Preparation Simulator |
| SM-DP-UT | Subscription Manager Data Preparation Under Test |
| SMS-C | Short Message Service Centre |

|  |  |
| --- | --- |
| **Abbreviation** | **Description** |
| SM-SR | Subscription Manager Secure Routing |
| SM-SR-S | Subscription Manager Secure Routing Simulator |
| SM-SR-TP | Third Party Subscription Manager Secure Routing |
| SM-SR-UT | Subscription Manager Secure Routing Under Test |
| SSD | Supplementary Security Domain |
| SW | Status Word |
| TAR | Toolkit Application Reference |
| TLS | Transport Layer Security |
| TLV | Tag, Length, Value |
| TON | Type Of Number |
| URI | Uniform Resource Identifier |
| USIM | Universal Subscriber Identity Module |
| W3C | World Wide Web Consortium |
| XML | Extensible Markup Language |

## Document Cross-references

|  |  |
| --- | --- |
| **Ref** | **Title** |
| [1] | GSMA SGP.01 - Embedded SIM Remote Provisioning Architecture v1.1 |
| [2] | GSMA SGP.02 - Remote Provisioning Architecture for Embedded UICC - Technical Specification v3.2 |
| [3] | GlobalPlatform Card Specification v.2.2.1 |
| [4] | ETSI TS 102 225 - Secured packet structure for UICC based applications; Release 12 |
| [5] | 3GPP TS 23.040 - Technical Specification Group Core Network and Terminals;  Technical realization of the Short Message Service (SMS) |
| [6] | ETSI TS 102 226 - Remote APDU structure for UICC based applications; Release 9 |
| [7] | ETSI TS 102 127 - Transport protocol for CAT applications; Release 6 |
| [8] | RFC 5246 - The TLS Protocol – Version 1.2 |
| [9] | RFC 5487 - Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode |
| [10] | ISO/IEC 7816-4 - Identification cards – Integrated circuit cards - Part 4: Organization, security and commands for interchange |
| [11] | GlobalPlatform Card Specification v.2.2 - Amendment D: Secure Channel Protocol 03 v1.1.1 |
| [12] | GlobalPlatform Card Specification v.2.2 - Amendment E: Security Upgrade for Card Content Management v1.0.1 |
| [13] | GlobalPlatform Card Specification v.2.2.1 - UICC Configuration v1.0.1 |
| [14] | GlobalPlatform Card Specification v.2.2 - Amendment C: Contactless Services v1.1.1 |
| [15] | RFC 4346 - The TLS Protocol – Version 1.1 |
| [16] | SIMAlliance eUICC Profile Package: Interoperable Format Technical Specification Version 2.1 |
| [17] | Trusted Connectivity Alliance (TCA) eUICC Profile Package: Interoperable Format Test Specification Version 3.2.1 |

|  |  |
| --- | --- |
| [18] | GlobalPlatform Card Specification v.2.2 Amendment B: Remote Application Management over HTTP v1.1.3 |
| [19] | RFC 2119 - Key words for use in RFCs to Indicate Requirement Levels, S. Bradner <http://www.ietf.org/rfc/rfc2119.txt> |

## Conventions

Throughout this document, normative requirements are highlighted by use of key words as described below.

The key words "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", and "MAY" in this

document SHALL be interpreted as described in RFC 2119 [19].

# Testing Rules

## Applicability

## Format of the Optional Features Table

The columns in [Table 4](#_bookmark29) 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:  Y supported by the implementation.  N not supported by the implementation. |
| Mnemonic | The mnemonic column contains mnemonic identifiers for each item. |

###### Table 1: Format of the Optional Features Table

## Format of the Applicability Table

The applicability of every test in [Table 5](#_bookmark31) is formally expressed by the use of Boolean expression defined in the following clause.

The columns in [Table 5](#_bookmark31) have the following meaning:

|  |  |
| --- | --- |
| **Column** | **Meaning** |
| Test case | The "Test case" column gives a reference to the test case number detailed in the present document and is required to validate the implementation of the corresponding item in the "Name" column. |
| Name | In the "Name" column, a short non-exhaustive description of the test is found. |
| Roles | SM-SR, SM-DP or eUICC  Entities under test that take in charge the functions used in the test case. |
| Applicability | See clause [2.1.3](#_bookmark27) 'Applicability and Notations'. |

###### Table 2: Format of the Applicability Table

## Applicability and Notations

The following notations are used for the Applicability column:

|  |  |
| --- | --- |
| **Applicability code** | **Meaning** |
| M | mandatory - the capability is required to be supported. |
| N/A | not applicable - in the given context, it is impossible to use the capability. |
| Ci | conditional - the requirement on the capability depends on the support of other 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. |

###### Table 3: Applicability and Notations

## Optional Features Table

The supplier of the implementation SHALL state the support of possible options in [Table 4](#_bookmark29). Items indicated as O\_XYZ (for example, O\_HTTPS) refer to features supported by a Role.

|  |  |  |  |
| --- | --- | --- | --- |
| **Item** | **Option** | **Support** | **Mnemonic** |
| 1 | Support of HTTPS |  | O\_HTTPS |
| 2 | Support of CAT\_TP |  | O\_CAT\_TP |
| 3 | HTTPS enabled on the default MNO-SD |  | O\_MNO\_HTTPS |
| 4 | Confidential setup of default Profile keys using scenario #2.B supported |  | O\_MNO\_SC2B |
| 5 | Confidential setup of default Profile keys using scenario #3 supported |  | O\_MNO\_SC3 |

###### Table 4: Options

All these features are related to the eUICC. As consequence, only the EUM is responsible for stating the support of these features.

Note that O\_HTTPS and O\_CAT\_TP are linked. At least, one of these options SHALL be supported. The support of the optional feature O\_MNO\_HTTPS supposes that the O\_HTTPS is also supported.

## Applicability Table

[Table 5](#_bookmark31) specifies the applicability of each test case. See clause [2.1.2](#_bookmark26) for the format of this table.

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Name** | **Roles** | **Applicability** |
| Interfaces Compliancy Test Cases | | | |
| [4.2.2.2.1](#_bookmark59) | TC.TP.SMS.1:Transport\_SMS | eUICC | M |
| [4.2.2.2.2](#_bookmark60) | TC.TP.CAT\_TP.2:Transport\_CAT\_TP | eUICC | C2 |
| [4.2.2.2.3](#_bookmark61) | TC.TP.HTTPS.3:Transport\_HTTPS | eUICC | C1 |
| [4.2.3.2.1](#_bookmark63) | TC.ES5.CISDP.1:CreateISDP\_SMS | eUICC | M |
| [4.2.3.2.2](#_bookmark64) | TC.ES5.CISDP.2:CreateISDP\_CAT\_TP | eUICC | C2 |
| [4.2.3.2.3](#_bookmark66) | TC.ES5.CISDP.3:CreateISDP\_HTTPS | eUICC | C1 |
| [4.2.4.2.1](#_bookmark69) | TC.ES5.EP.1:EnableProfile\_SMS | eUICC | M |
| [4.2.4.2.2](#_bookmark71) | TC.ES5.EP.2:EnableProfile\_CAT\_TP | eUICC | C2 |
| [4.2.4.2.3](#_bookmark73) | TC.ES5.EP.3:EnableProfile\_HTTPS | eUICC | C1 |
| [4.2.5.2.1](#_bookmark76) | TC.ES5.DISP.1:DisableProfile\_SMS | eUICC | M |
| [4.2.5.2.2](#_bookmark78) | TC.ES5.DISP.2:DisableProfile\_CAT\_TP | eUICC | C2 |
| [4.2.5.2.3](#_bookmark79) | TC.ES5.DISP.3:DisableProfile\_HTTPS | eUICC | C1 |
| [4.2.6.2.1](#_bookmark81) | TC.ES5.FB.1:SetFallbackAttribute\_SMS | eUICC | M |
| [4.2.6.2.2](#_bookmark82) | TC.ES5.FB.2:SetFallbackAttribute\_CAT\_TP | eUICC | C2 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Name** | **Roles** | **Applicability** |
| [4.2.6.2.3](#_bookmark83) | TC.ES5.FB.3:SetFallbackAttribute\_HTTPS | eUICC | C1 |
| [4.2.7.2.1](#_bookmark85) | TC.ES5.DP.1:DeleteProfile\_SMS | eUICC | M |
| [4.2.7.2.2](#_bookmark86) | TC.ES5.DP.2:DeleteProfile\_CAT\_TP | eUICC | C2 |
| [4.2.7.2.3](#_bookmark87) | TC.ES5.DP.3:DeleteProfile\_HTTPS | eUICC | C1 |
| [4.2.8.2.1](#_bookmark89) | TC.ES5.ECA.1:eUICCCapabilityAudit\_SMS | eUICC | M |
| [4.2.8.2.2](#_bookmark90) | TC.ES5.ECA.2:eUICCCapabilityAudit\_CAT\_TP | eUICC | C2 |
| [4.2.8.2.3](#_bookmark91) | TC.ES5.ECA.3:eUICCCapabilityAudit\_HTTPS | eUICC | C1 |
| [4.2.9.2.1](#_bookmark93) | TC.ES5.MD.1:MasterDelete\_SMS | eUICC | M |
| [4.2.9.2.1.7](#_bookmark94) | TC.ES5.MD.2:MasterDelete\_CAT\_TP | eUICC | C2 |
| [4.2.9.2.3](#_bookmark95) | TC.ES5.MD.3:MasterDelete\_HTTPS | eUICC | C1 |
| [4.2.10.2.1](#_bookmark97) | TC.ES5.EISDRK.1:EstablishISDRKeyset\_SMS | eUICC | M |
| [4.2.10.2.2](#_bookmark98) | TC.ES5.EISDRK.2:EstablishISDRKeyset\_CAT\_TP | eUICC | C2 |
| [4.2.10.2.3](#_bookmark99) | TC.ES5.EISDRK.3:EstablishISDRKeyset\_HTTPS | eUICC | C1 |
| [4.2.11.2.1](#_bookmark101) | TC.ES5.FIH.1:FinaliseISDRHandover\_SMS Test Sequence N°1 | eUICC | C1 |
| [4.2.11.2.1](#_bookmark101) | TC.ES5.FIH.1:FinaliseISDRHandover\_SMS Test Sequence N°2, Test Sequence N°3 | eUICC | M |
| [4.2.11.2.2](#_bookmark102) | TC.ES5.FIH.2:FinaliseISDRHandover\_CAT\_TP Test Sequence N°1 | eUICC | C9 |
| [4.2.11.2.2](#_bookmark102) | TC.ES5.FIH.2:FinaliseISDRHandover\_CAT\_TP Test Sequence N°2 | eUICC | C8 |
| [4.2.11.2.3](#_bookmark103) | TC.ES5.FIH.3:FinaliseISDRHandover\_HTTPS | eUICC | C1 |
| [4.2.12.2.1](#_bookmark105) | TC.ES5.USAP.1:UpdateSMSRAddrParam\_SMS | eUICC | M |
| [4.2.12.2.2](#_bookmark106) | TC.ES5.USAP.2:UpdateSMSRAddrParam\_CAT\_TP | eUICC | C2 |
| 4.2.12.2.3 | TC.ES5.USAP.3:UpdateSMSRAddrParam\_HTTPS | eUICC | C1 |
| [4.2.13.2.1](#_bookmark108) | TC.ES5.NOTIFPE.1:Notification\_SMS | eUICC | M |
| [4.2.13.2.2](#_bookmark110) | TC.ES5.NOTIFPE.2:Notification\_CAT\_TP | eUICC | C2 |
| 4.2.13.2.3 | TC.ES5.NOTIFPE.3:Notification\_HTTPS | eUICC | C1 |
| [4.2.14.2.1](#_bookmark113) | TC.ES5.NOTIFPD.1:Notification\_SMS | eUICC | M |
| [4.2.14.2.2](#_bookmark114) | TC.ES5.NOTIFPD.2:Notification\_CAT\_TP | eUICC | C2 |
| 4.2.14.2.3 | TC.ES5.NOTIFPD.3:Notification\_HTTPS | eUICC | C1 |
| [4.2.15.2.1](#_bookmark116) | TC.ES6.UPOL1MNO.1:UpdatePOL1byMNO\_SMS | eUICC | M |
| [4.2.15.2.2](#_bookmark117) | TC.ES6.UPOL1MNO.2:UpdatePOL1byMNO\_CAT\_TP | eUICC | C2 |
| [4.2.15.2.3](#_bookmark118) | TC.ES6.UPOL1MNO.3:UpdatePOL1byMNO\_HTTPS | eUICC | C5 |
| [4.2.16.2.1](#_bookmark120) | TC.ES6.UCPMNO.1:UpdateConnectParamByMNO\_SMS Test Sequence N°1 | eUICC | M |
| [4.2.16.2.1](#_bookmark120) | TC.ES6.UCPMNO.1:UpdateConnectParamByMNO\_SMS Test Sequence N°2 | eUICC | C3 |
| [4.2.16.2.1](#_bookmark120) | TC.ES6.UCPMNO.1:UpdateConnectParamByMNO\_SMS Test Sequence N°3 | eUICC | C4 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Name** | **Roles** | **Applicability** |
| [4.2.17.2.1](#_bookmark122) | TC.ES8.EISDPK.1:EstablishISDPKeyset\_SMS | eUICC | M |
| [4.2.17.2.2](#_bookmark123) | TC.ES8.EISDPK.2:EstablishISDPKeyset\_CAT\_TP | eUICC | C2 |
| [4.2.17.2.3](#_bookmark125) | TC.ES8.EISDPK.3:EstablishISDPKeyset\_HTTPS | eUICC | C1 |
| [4.2.18.2.1](#_bookmark128) | TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP | eUICC | C2 |
| [4.2.18.2.2](#_bookmark130) | TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS | eUICC | C1 |
| [4.2.19.2.1](#_bookmark133) | TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS Test Sequence N°1 | eUICC | M |
| [4.2.19.2.1](#_bookmark133) | TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS Test Sequence N°2, Test Sequence N°4 | eUICC | C3 |
| [4.2.19.2.1](#_bookmark133) | TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS Test Sequence N°3, Test Sequence N°5 | eUICC | C4 |
| [4.2.19.2.2](#_bookmark135) | TC.ES8.UCP.2:UpdateConnectivityParameters\_CAT\_TP | eUICC | C2 |
| [4.2.19.2.3](#_bookmark137) | TC.ES8.UCP.3:UpdateConnectivityParameters\_HTTPS | eUICC | C1 |
| [4.3.1.2.1](#_bookmark141) | TC.ES1.REIS.1:RegisterEIS | SM-SR | M |
| [4.3.2.2.1](#_bookmark143) | TC.ES2.GEIS.1:GetEIS | SM-DP | M |
| [4.3.3.2.1](#_bookmark145) | TC.ES2.DP.1:DownloadProfile | SM-DP | M |
| [4.3.4.2.1](#_bookmark147) | TC.ES2.UPR.1:UpdatePolicyRules | SM-DP | M |
| [4.3.5.2.1](#_bookmark149) | TC.ES2.USA.1:UpdateSubscriptionAddress | SM-DP | M |
| [4.3.6.2.1](#_bookmark151) | TC.ES2.EP.1:EnableProfile | SM-DP | M |
| [4.3.6.2.2](#_bookmark152) | TC.ES2.EP.2:EnableProfileWithDeletion | SM-DP | M |
| [4.3.7.2.1](#_bookmark154) | TC.ES2.DISP.1:DisableProfile | SM-DP | M |
| [4.3.8.2.1](#_bookmark156) | TC.ES2.DP.1:DeleteProfile | SM-DP | M |
| [4.3.9.2.1](#_bookmark158) | TC.ES3.GEIS.1:GetEIS | SM-SR | M |
| [4.3.10.2.1](#_bookmark160) | TC.ES3.AEIS.1:AuditEIS | SM-SR | M |
| [4.3.11.2.1](#_bookmark162) | TC.ES3.CISDP.1:CreateISDP | SM-SR | M |
| [4.3.12.2.1](#_bookmark164) | TC.ES3.SDATA.1:SendData | SM-SR | M |
| [4.3.13.2.1](#_bookmark166) | TC.ES3.UPR.1:UpdatePolicyRules | SM-SR | M |
| [4.3.14.2.1](#_bookmark168) | TC.ES3.USA.1:UpdateSubscriptionAddress | SM-SR | M |
| [4.3.15.2.1](#_bookmark170) | TC.ES3.UCP.1:UpdateConnectivtyParameters | SM-SR | M |
| [4.3.16.2.1](#_bookmark172) | TC.ES3.EP.1:EnableProfile | SM-SR | M |
| [4.3.17.2.1](#_bookmark174) | TC.ES3.DISP.1:DisableProfile | SM-SR | M |
| [4.3.18.2.1](#_bookmark176) | TC.ES3.DISDP.1:DeleteISDP | SM-SR | M |
| [4.3.19.2.1](#_bookmark178) | TC.ES4.GEIS.1:GetEIS  Test Sequence N°1 | SM-SR | M |
| [4.3.19.2.1](#_bookmark178) | TC.ES4.GEIS.1:GetEIS  Test Sequence N°2 | SM-SR | N/A |
| [4.3.20.2.1](#_bookmark180) | TC.ES4.UPR.1:UpdatePolicyRules | SM-SR | M |
| [4.3.21.2.1](#_bookmark182) | TC.ES4.USA.1:UpdateSubscriptionAddress | SM-SR | M |
| [4.3.22.2.1](#_bookmark184) | TC.ES4.AEIS.1:AuditEIS | SM-SR | M |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Name** | **Roles** | **Applicability** |
| [4.3.23.2.1](#_bookmark186) | TC.ES4.EP.1:EnableProfile | SM-SR | M |
| [4.3.24.2.1](#_bookmark188) | TC.ES4.DISP.1:DisableProfile | SM-SR | M |
| [4.3.25.2.1](#_bookmark190) | TC.ES4.DP.1:DeleteProfile | SM-SR | M |
| [4.3.26.2.1](#_bookmark192) | TC.ES4.PSMSRC.1:PrepareSMSRChange | SM-SR | M |
| [4.3.27.2.1](#_bookmark194) | TC.ES4.SMSRC.1:SMSRChange | SM-SR | M |
| [4.3.28.2.1](#_bookmark196) | TC.ES7.HEUICC.1:HandoverEUICC | SM-SR | M |
| [4.3.29.2.1](#_bookmark198) | TC.ES7.ASMSR.1:AuthenticateSMSR | SM-SR | M |
| [4.3.29.2.1](#_bookmark198) | TC.ES7.CAK.1:CreateAdditionalKeyset | SM-SR | M |
| System Behaviour Test Cases | | | |
| [5.2.1.2.1](#_bookmark204) | TC.ECASD.1:EIDRetrieval | eUICC | M |
| [5.2.2.2.1](#_bookmark206) | TC.LOCKISDR.1:LockISDR | eUICC | M |
| [5.2.2.2.2](#_bookmark207) | TC.LOCKISDP.1:LockISDP | eUICC | M |
| [5.2.3.2.1](#_bookmark209) | TC.CV.1:ComponentVisibility | eUICC | M |
| [5.2.3.2.2](#_bookmark210) | TC.CV.2:ISDRVisibility | eUICC | M |
| [5.2.3.2.3](#_bookmark211) | TC.CV.3:ISDPNotEnabled  Test Sequence N°1, Test Sequence N°3 | eUICC | C2 |
| [5.2.3.2.3](#_bookmark211) | TC.CV.3:ISDPNotEnabled  Test Sequence N°2, Test Sequence N°4 | eUICC | C1 |
| [5.2.3.2.3.4](#_bookmark212) | TC.CV.4:TarAllocation Test Sequence N°1 | eUICC | C2 |
| [5.2.3.2.3.4](#_bookmark212) | TC.CV.4:TarAllocation Test Sequence N°2 | eUICC | C1 |
| [5.2.3.2.3.4](#_bookmark212) | TC.CV.4:TarAllocation Test Sequence N°3 | eUICC | M |
| [5.2.3.2.5](#_bookmark213) | TC.CV.5:AIDAllocation Test Sequence N°1 | eUICC | C2 |
| [5.2.3.2.5](#_bookmark213) | TC.CV.5:AIDAllocation Test Sequence N°2 | eUICC | C1 |
| [5.2.3.2.5](#_bookmark213) | TC.CV.5:AIDAllocation Test Sequence N°3 | eUICC | M |
| [5.2.3.2.6](#_bookmark214) | TC.CV.6:MNOSDDefinition | eUICC | M |
| [5.2.4.2.1](#_bookmark216) | TC.SAR.1:SecurityError\_SMS | eUICC | M |
| [5.2.4.2.1.2](#_bookmark217) | TC.SAR.2:ISDRResponsibility | eUICC | M |
| 5.2.4.2.3 | TC.SAR.3:ReplayAttack | eUICC | M |
| [5.2.4.2.4](#_bookmark218) | TC.SAR.4:HTTPSRestrictions | eUICC | C1 |
| [5.2.4.2.5](#_bookmark219) | TC.SAR.5:SCP03t\_ErrorManagement | eUICC | M |
| [5.2.5.2.1](#_bookmark221) | TC.CSMNOSCK.1:Scenario#2.B | eUICC | C6 |
| [5.2.5.2.2](#_bookmark222) | TC.CSMNOSCK.2:Scenario#3 | eUICC | C7 |
| [5.2.6.2.1](#_bookmark224) | TC.FPIP.1:ProfileDownloadAndEnabling  Test Sequence N°1 | eUICC | C2 |
| [5.2.6.2.1](#_bookmark224) | TC.FPIP.1:ProfileDownloadAndEnabling | eUICC | C1 |

|  |  |  |  |
| --- | --- | --- | --- |
| **Test case** | **Name** | **Roles** | **Applicability** |
|  | Test Sequence N°2 |  |  |
| [5.3.1.2.1](#_bookmark227) | TC.EUICCIC.1:eUICCEligibilitySMDP | SM-DP | M |
| [5.3.1.2.2](#_bookmark228) | TC.EUICCIC.2:eUICCEligibilitySMSR | SM-SR | M |
| [5.3.2.2.1](#_bookmark230) | TC.PROC.DIP.1:DownloadAndInstallProfile Test Sequence N°1 | SM-DP,  SM-SR | C3 |
| [5.3.2.2.1](#_bookmark230) | TC.PROC.DIP.1:DownloadAndInstallProfile Test Sequence N°2 | SM-DP, SM-SR | C4 |
| [5.3.2.2.2](#_bookmark232) | TC.PROC.DIP.2:DownloadAndInstallProfileAndEnable | SM-DP, SM-SR | M |
| [5.3.3.2.1](#_bookmark234) | TC.PROC.PE.1.ProfileEnablingByMNO | SM-SR | M |
| [5.3.3.2.2](#_bookmark236) | TC.PROC.PE.2.ProfileEnablingBySMDP | SM-DP, SM-SR | M |
| [5.3.4.2.1](#_bookmark238) | TC.PROC.DIS.1:ProfileDisablingByMNO | SM-SR | M |
| [5.3.4.2.2](#_bookmark239) | TC.PROC.DIS.2:ProfileDisablingBySMDP | SM-DP, SM-SR | M |
| [5.3.5.2.1](#_bookmark241) | TC.PROC.DEL.1:ProfileDeletionByMNO | SM-SR | M |
| [5.3.5.2.1.3](#_bookmark242) | TC.PROC.DEL.2:ProfileDeletionBySMDP | SM-DP, SM-SR | M |
| [5.3.7.2.1](#_bookmark245) | TC.PROC.SMSRCH.1:SMSRChange | SM-DP, SM-SR | M |
| [5.3.7.2.2](#_bookmark246) | TC.PROC.SMSRCH.2:SMSRChange | SM-SR | M |
| [5.3.7.2.3](#_bookmark247) | TC.PROC.SMSRCH.3:SMSRChange | SM-SR | M |
| [5.3.7.2.4](#_bookmark248) | TC.PROC.SMSRCH.4:SMSRChange | SM-SR | M |
| [5.3.8.2.1](#_bookmark250) | TC.PROC.UCP.1:UpdateConnectivityParameters  Test Sequence N°1 | SM-SR | M |
| [5.3.8.2.1](#_bookmark250) | TC.PROC.UCP.1:UpdateConnectivityParameters  Test Sequence N°2 | SM-SR | C3 |
| 5.3.8.2.1 | TC.PROC.UCP.1:UpdateConnectivityParameters  Test Sequence N°3 | SM-SR | C4 |
| Test Specifications | | | |
| 6.1 | SIMAlliance eUICC Profile Package Test Specification | eUICC | M |

###### Table 5: Applicability of Tests

|  |  |
| --- | --- |
| **Conditional item** | **Condition** |
| C1 | IF (NOT O\_CAT\_TP OR O\_HTTPS) THEN M ELSE N/A |
| C2 | IF (NOT O\_HTTPS OR O\_CAT\_TP) THEN M ELSE N/A |
| C3 | IF (O\_CAT\_TP) THEN M ELSE N/A |
| C4 | IF (O\_HTTPS) THEN M ELSE N/A |
| C5 | IF (O\_HTTPS AND O\_MNO\_HTTPS) THEN M ELSE N/A |

|  |  |
| --- | --- |
| **Conditional item** | **Condition** |
| C6 | IF (O\_MNO\_SC2B) THEN M ELSE N/A |
| C7 | IF (O\_MNO\_SC3) THEN M ELSE N/A |
| C8 | IF (O\_HTTPS AND O\_CAT\_TP) THEN M ELSE N/A |
| C9 | IF (NOT O\_HTTPS) THEN M ELSE N/A |
| C10 | VOID |

###### Table 6: Conditional Items Referenced by Table 5

## General Consideration

This section contains some general considerations about the test cases defined in this document. Note that some external test specifications are referred to in chapter 6. Consequently, the following sub sections SHALL only apply for test cases defined in sections 4 and 5.

## Test Cases Definition

Test descriptions are independent.

For each test described in this document, a chapter provides a general description of the initial conditions applicable for the whole test. This description is completed by specific configurations to each individual sub-case.

It is implicitly assumed that all entities under test SHALL be compliant with the initial states described in [Annex I](#_bookmark289). An initial state SHALL be considered as a pre-requisite to execute all the test cases described in this Test Plan.

After completing the test, the configuration is reset before the execution of the following test.

## Test Cases Format

Here is an explanation of the way to define the test cases in chapters [4](#_bookmark47) and [5](#_bookmark200).

**4.X.Y.Z Test Cases General Initial Conditions**

* Test cases - general condition 1
* Test cases - general condition 2

**Test Environment**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Entity1 Entity2**  Standard Message Optional Message  Informative Message **Entity3**  Request Message  Backend Message  Response Message   * + - * 1. **TC.TEST\_NAME.1: TEST\_TITLE**   **Test Purpose**  *Description of the aim of the test case TC.TEST\_NAME.1*  **Referenced Requirements**   * REQ1, REQ2   **Initial Conditions**   * Test case TC.TEST\_NAME.1 - initial condition 1 * Test case TC.TEST\_NAME.1 - initial condition 2   **Test Sequence N°1 Initial Conditions**  Test sequence N°1 - initial condition 1  Test sequence N°1 - initial condition 2 | | | | | |
|  | **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Entity1 → Entity2 | Command or Message to send from Entity1 to Entity2 | 1. expected result N°1.1 2. expected result N°1.2 | REQ1 |
| 2 | Entity2 → Entity3 | Command or Message to send from Entity2 to Entity3 |  |  |
| *Note: Global note for the test sequence N°1* | | | | |
| **4.X.Y.Z.1.2 Test Sequence N°2 Initial Conditions**   * None | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |  |
| 1 | Entity1 → Entity2 | Command or Message to send from Entity1 to Entity2 |  |  |  |
| 2 | Entity2 → Entity3 | Command or Message to send from Entity2 to Entity3 | 1. expected result N°2.1 2. expected result N°2.2 (see Note 1) | REQ2 |  |
| *Note 1: Note about the expected result N°2.2* | | | | |  |
| **4.X.Y.Z.2 TC.TEST\_NAME.2: TEST\_TITLE**  … | | | | | |  |

The test cases TC.TEST\_NAME.1:TEST\_TITLE and TC.TEST\_NAME.2:TEST\_TITLE are referenced in [Table 5](#_bookmark31) that allows indicating the applicability of the tests.

The test environment allows describing the different entities involved in the test sequences of the test case. Different types of messages are used:

* standard message: message exchanged between two entities (e.g. an APDU, a RPS Message) composed of a request and a response
* optional message: standard message that MAY be sent or not depending of the aim of the test
* informative message: message used to facilitate the understanding of the test case. It is not exchanged by any entities (e.g. messages between simulators)
* request message: message sent to an entity that MAY trigger messages to other entities to generate the corresponding response
* backend message: message exchanged between two entities that cannot be checked by the current test case
* response message: a response related to a request message

In the test case TC.TEST\_NAME.1:TEST\_TITLE, the requirements REQ1 and REQ2 are respectively covered by the test sequences N°1 and N°2.

The test sequence N°1 SHALL be executed if and only if these conditions are met:

* Test cases - general condition 1
* Test cases - general condition 2
* Test case TC.TEST\_NAME.1 - initial condition 1
* Test case TC.TEST\_NAME.1 - initial condition 2
* Test sequence N°1 - initial condition 1
* Test sequence N°1 - initial condition 2

The test sequence N°2 SHALL be executed if and only if these conditions are met:

* Test cases - general condition 1
* Test cases - general condition 2
* Test case TC.TEST\_NAME.1 - initial condition 1
* Test case TC.TEST\_NAME.1 - initial condition 2

In the test sequence N°1, in the step N°1, if the expected results N°1 and N°2 are validated, the requirement REQ1 (or a part of the REQ1) SHALL be considered as implemented.

Note that all initial states (described in [Annex I](#_bookmark289)) SHALL be implemented by the entity under test whatever the test cases to execute.

## Using of Methods, Constants and Dynamic Content

In several test sequences described in this document, some methods, constants and dynamic values are used.

A constant is used as follow:

#NAME\_OF\_THE\_CONSTANT: SHALL be replaced by the value of the corresponding constant defined in [Annex B](#_bookmark266).

A dynamic content is described in [Annex C](#_bookmark280) and used as follow:

{NAME\_OF\_THE\_VARIABLE}

A dynamic content is either generated by an entity under test or by a test tool provider.

A method is used as follow:

NAME\_OF\_THE\_METHOD(PARAM1, PARAM2…): the method and the parameters are described in [Annex D](#_bookmark281).

The implementation of these methods is under the responsibility of the test tool providers.

## Commands and Responses

In several test sequences described in this document, some commands and responses are used. These elements are explained in [Annex E](#_bookmark282).

A reference to a command or a response is used as follow: [NAME\_OF\_THE\_COMMAND\_OR\_RESPONSE]: SHALL be replaced by the value defined in [Annex E](#_bookmark282).

## Referenced Requirements

All requirements referenced in this document by their identifiers are present and described in [Annex J](#_bookmark290). These requirements have been extracted from the specifications:

* GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

## Pass Criterion

A test execution is considered as successful only if the test procedure was fully carried out successfully.

A test execution is considered as failed if the tested feature provides an unexpected behaviour during the steps indicated with a white background in the tables.

A test execution is considered as inconclusive when the pass criteria cannot be evaluated due to issues during the setup of the initial conditions or during the steps indicated with a pink background in the tables.

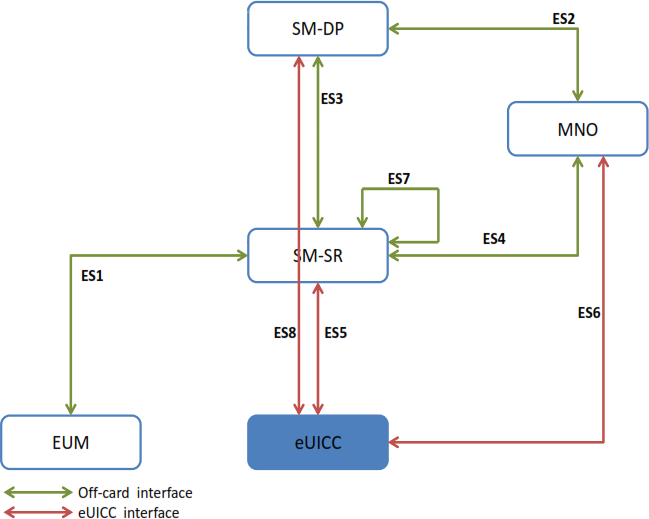
## Future Study

Some of the test cases or test sequences described in this Test Plan are FFS (For Future Study). This means that some clarifications are expected at the requirement level to conclude on a test method. As consequence, the corresponding test SHALL NOT be executed.

# Testing Architecture

## Testing Scope

Here are all the interfaces that are tested in this document.



###### Figure 1: Scope of the Tests

|  |  |
| --- | --- |
| **Interface** | **Description** |
| ES1 | Interface between the EUM and the SM-SR that allows the registration of an eUICC within the SM-SR. |
| ES2 | Interface between the MNO and the SM-DP that allows managing a Profile and to trigger Profile loading. |
| ES3 | Interface between the SM-DP and the SM-SR that allows managing a Profile and to trigger Profile loading. |
| ES4 | Interface between the MNO and the SM-SR that allows enabling, disabling and deleting Profiles. |
| ES5 | Interface between the SM-SR and the eUICC that allows the OTA communication. |
| ES6 | Interface between the MNO and the eUICC that allows managing the content of the MNO’s Profile. |
| ES7 | Interface between two SM-SR that allows managing the SM-SR change process. |
| ES8 | Interface between the SM-DP and the eUICC that allows downloading of a Profile within the eUICC. |

###### Table 7: Interfaces Descriptions

The DNS resolution defined in SGP.02 [2], section 2.4.5, is an optional feature and is defined as FFS in this version of the specification. All eUICC test cases defined in this document remain applicable even if this feature is supported considering that the ISD-R has always an IP address either configured in the Connection Parameters of the Security Domain Administration Session Parameters or supplied in the Administration Session Triggering Parameters (as defined by GlobalPlatform Amendment B [18]). As a consequence, the eUICC SHALL NOT perform any DNS resolution during the execution of the HTTPs test cases defined in sections 4.2 and 5.2.

The support of Java Card is considered as mandatory in the scope of this specification.

## Testing Execution

This chapter aims to describe the different testing environments and equipment to allow executing the test cases.

To allow the execution of the different test cases described in this Test Plan, some simulators SHALL be used. Here are the different simulators that have been defined:

* DS: the Device simulator used to simulate the Device and to send some commands to the eUICC-UT using ISO/IEC 7816-4 [[10]](#_bookmark15) on the contact interface. The provisioning commands sent by the DS refer to commands sent by the system Actors (i.e. SM-SR, SM-DP and MNO)
* SM-DP-S: the SM-DP simulator used to simulate the SM-DP and to test a SM-SR
* SM-SR-S: the SM-SR simulator used to simulate the SM-SR and to test a SM-DP or a SM-SR
* MNO-S: the MNO simulator used to simulate the MNO and to test a SM-DP or a SM- SR
* EUM-S: the EUM simulator used to simulate the EUM and to test a SM-SR

Implementation of these simulators remains the responsibility of the test tool providers.

## Interfaces Compliancy

The aim of all the test cases related to the interfaces compliancy (see section [4](#_bookmark47)) is to verify the compliancy of an Actor (i.e. eUICC, SM-DP, SM-SR).

###### eUICC Interfaces

Figure 2 shows the different entities used during the execution of the test cases related to the eUICC interfaces (see section [4.2](#_bookmark49)).

ES8

ES5, ES8

ES6

MNO-S

* *SMS*
* *CAT\_TP*
* *HTTPS*

SM-DP-S

SM-SR-S

* *SMS*
* *CAT\_TP*
* *HTTPS*

eUICC-UT

DS

Legend:

Simulator

Back-end simulator

Entity under test

###### Figure 2: eUICC Interfaces Test Environment

The aim of the interface compliancy test cases, related to the interfaces ES5, ES6 and ES8, is to test the eUICC. The Device Simulator (DS) allows simulating the SM-SR, the SM-DP or the MNO. As consequence, the DS SHALL include SMS, HTTPS and CAT\_TP entities to simulate the OTA communication with the eUICC (i.e. the SM-SR-S, SM-DP-S and MNO-S SHALL be considered as parts of the DS).

The CAT\_TP entity generates CAT\_TP PDUs according the [Annex G](#_bookmark286). The HTTPS entity generates TLS records according the [0](#_bookmark287).

The Device Simulator SHALL honor any POLL INTERVAL proactive commands issued by the eUICC, and accordingly send STATUS commands at the interval requested.

The Device Simulator SHALL honor any TIMER MANAGEMENT proactive commands issued by the eUICC, and accordingly sed an ENVELOPE (TIMER EXPIRATION) command after the specified time, if a timer has been activated.

###### Off-card Interfaces

The off-card test cases assume that all simulated platforms (i.e. EUM-S, MNO1-S, MNO2-S, SM-DP-S, SM-SR-S) identified by EUM\_S\_ID, MNO1\_S\_ID, MNO2\_S\_ID, SM\_DP\_S\_ID, SM\_SR\_S\_ID SHALL be well known to the platforms under test (i.e. SM-DP-UT, SM-SR-UT) as specified in the initial conditions of each test. All simulated platforms SHALL be compliant with the security level mandated by the platforms under test.

Figure 3 shows the different entities used during the execution of the test cases related to the off-card interfaces (see section 4.3).

ES1

ES3

ES4

SM-DP-S

SM-SR-UT

EUM-S



ES5\*

SM-DP-UT

MNO1-S MNO2-S

SM-SR-S

ES7

Device

eUICC

ES2

ES3

ES2

Legend:

Simulator

Entity under test

Unused equipment

###### Figure 3: Off-card Interfaces Test Environment

*\* All OTA interfaces between the SM-SR-UT and an eUICC (ES5 or ES8 over ES5) are out of the scope defined for the off-card interfaces testing. The test cases involving the SM-SR- UT and an eUICC are defined in the section “5 -* [System Behaviour Testing](#_bookmark200)*”.*

## System Behaviour

The aim of all the test cases related to the system behaviour (see section [5](#_bookmark200)) is to verify the functional behaviour of the eUICC ecosystem composed of the following Actors:

* MNO
* eUICC
* SM-DP
* SM-SR

###### eUICC Behaviour

Figure 4 shows the different entities used during the execution of the test cases related to the eUICC behaviour (see section [5.2](#_bookmark202)).

DS

eUICC-UT

Legend:

Simulator

Entity under test

###### Figure 4: eUICC Behaviour Test Environment

###### Platform Behaviour

Figure 5 shows the different entities used during the execution of the test cases related to the platforms behaviour (see section 5.3).

ES5

ES3

ES7

ES4

ES3

ES7

ES2

ES3

MNO1-S

MNO2-S

SM-DP-S

SM-SR-S

SM-DP-UT

SM-SR-TP

SM-SR-UT

|  |  |  |
| --- | --- | --- |
| Device | | |
|  | eUICC |  |

Legend:

Simulator

Entity under test

Equipment used for testing

Black-box

###### Figure 5: Platform Behaviour Test Environment

A black box testing method is used in order to ensure that the system functional scenarios are properly implemented. In this context, it is assumed that:

* The OTA communication between the SM-SR-UT and the Device equipment (i.e. ES5) SHALL be based on real wireless network provided by MNO (see [Figure 7](#_bookmark45)). OTA operations performed by the SM-SR-UT are not checked by test tool providers: the verification of the correctness of commands coming from the SM-SR-UT is performed by the eUICC/Device.
* The SM-DP-UT and the SM-SR-UT are well known to each other and the functions of the ES3 interface are individually tested in accordance with the test cases described in section 4.3.
* The Device used for testing SHALL support all mandatory requirements described in the GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification / Annex G [[2]](#_bookmark7).
* The functions of the eUICC interface (i.e. ES5 and ES8 over ES5) SHALL be supported by the eUICC.
* The entity SM-SR-TP SHALL be considered as a third party platform used to test the SM-SR-UT. As consequence, the functions of the ES7 interface SHALL be supported by this platform.

Figure 6 shows the eUICC configuration that SHALL be used to execute the test cases:



**eUICC**

Personalized

#SM\_SR\_DEST\_ADDR *#SM\_SR\_UDP\_IP #SM\_SR\_UDP\_PORT #SM\_SR\_TCP\_IP #SM\_SR\_TCP\_PORT #SM\_SR\_HTTP\_URI #SM\_SR\_HTTP\_HOST*

Pre-installed Enabled

MSISDN: #MSISDN

ICCID: #ICCID

Owned by MNO2-S

Pre-installed Disabled

**…**

Not Created by default MSISDN: #NEW\_MSISDN ICCID: #NEW\_ICCID

Owned by MNO1-S

#MNO2\_CON\_NAN #MNO2\_CON\_LOGIN #MNO2\_CON\_PWD #MNO2\_CON\_TON\_NPI #MNO2\_CON\_DIAL\_NUM

#MNO1\_CON\_NAN

#MNO1\_CON\_LOGIN #MNO1\_CON\_PWD

Connectivity Parameters

Connectivity Parameters

NEW ISD-P

Installed during test execution

ISD-P n

(optional)

ISD-P

ECASD

Addressing Parameters

ISD-R

###### Figure 6: eUICC Configuration

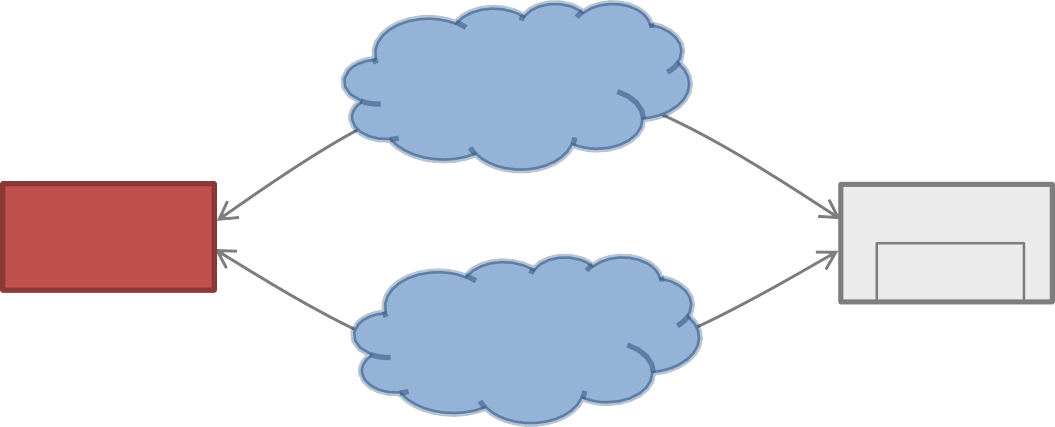
The eUICC, used to execute the test cases defined in the section 5.3, SHALL be compliant with the figure above. A Profile, identified by #ICCID, SHALL be Enabled. Other pre-installed Profiles MAY be present (i.e. if present, they SHALL be Disabled). The Profile, identified by #NEW\_ICCID, is dynamically downloaded during the test cases execution: as consequence, it SHALL NOT be pre-installed. It is implicitly assumed that all mandatory Profile Components SHALL be present in the Profiles identified by #ICCID and #NEW\_ICCID to allow connectivity network (i.e. file system, NAA…).

Regarding the addressing parameters, except the #SM\_SR\_DEST\_ADDR which is mandatory, the HTTPS and the CAT\_TP settings are conditional depending on the eUICC implementation.

Note that the Subscription Addresses of the Profile dynamically downloaded during the tests (i.e. #NEW\_MSISDN / #NEW\_ICCID) and the pre-installed Profile (i.e. #MSISDN / #ICCID) SHALL be provided by real MNOs (named MNO1 and MNO2 in the [Figure 7](#_bookmark45)). It means that the SM-SR-UT is able to communicate with these MNOs’ networks (as mentioned in the initial conditions of the test cases defined in section 5.3).

In the sections dealing with the platform behaviour testing, MNO1-S and MNO2-S stand for MNO platforms simulators which only allow sending requests to the SM-DP-UT and SM-SR- UT.

Figure 7 shows how the SM-SR-UT SHALL communicate OTA with the eUICC.



MNO1

Network

ES5

SM-SR-UT

Device

eUICC

MNO2 ES5

Network

###### Figure 7: Required Network Access for SM-SR-UT

## Void

# Interface Compliancy Testing

## General Overview

This section focuses on the implementation of the different interfaces according to the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7). The aim is to verify the compliancy of all interfaces within the system.

## eUICC Interfaces

## Generic Sub-sequences

This section describes some generic sub-sequences used in the eUICC interfaces compliancy test cases. These test sequences are part of test cases and SHALL NOT be executed in standalone mode.

###### Initialization Sequence

To initialize the communication between the DS and the eUICC, these commands SHALL be executed:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization SW=’9000’ |  |
| *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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.* | | | | |

###### Open CAT\_TP Session on ISD-R

To open a CAT\_TP session on the ISD-R, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54 |
| 1 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [OPEN\_CHANNEL\_FOR\_BIP]; |  |
|  |  | [OPEN\_CHANNEL\_FOR\_CATTP]) |  |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The buffer size is equal to   #BUFFER\_SIZE   1. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #UDP\_PORT   1. The IP is equal to   #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ53 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The CAT\_TP PDU used here after SHALL be compliant with the* [*Annex G.*](#_bookmark286) | | | | |
| 6 | eUICC-UT → DS | SYN | The identification data MAY contain the #EID | EUICC\_REQ18 |
| 7 | DS → eUICC-UT | SYN\_ACK |  |  |
| 8 | eUICC-UT → DS | ACK\_NO\_DATA | The CAT\_TP session is open. | EUICC\_REQ18 |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The SCP80 status code is equal to ‘00’ – POR OK | EUICC\_REQ21 |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |

This sub-sequence allows testing these requirements:

* + - * + EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Open CAT\_TP Session on MNO-SD

To open a CAT\_TP session on the #MNO\_SD\_AID, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_SD\_TAR, [OPEN\_CHANNEL\_FOR\_BIP];  [OPEN\_CHANNEL\_FOR\_CATTP])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The buffer size is equal to   #BUFFER\_SIZE   1. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #UDP\_PORT   1. The IP is equal to   #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The CAT\_TP PDU used here after SHALL be compliant with the* [*Annex G.*](#_bookmark286) | | | | |
| 6 | eUICC-UT → DS | SYN |  | EUICC\_REQ18 |
| 7 | DS → eUICC-UT | SYN\_ACK |  |  |
| 8 | eUICC-UT → DS | ACK\_NO\_DATA | The CAT\_TP session is open. | EUICC\_REQ18 |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. The SCP80 status code is equal to ‘00’ – POR OK |  |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |

This sub-sequence allows testing these requirements:

* + - * + EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ22

###### Close CAT\_TP Session

To close a CAT\_TP session, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RST |  | EUICC\_REQ18 |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND:*  CLOSE CHANNEL | The CAT\_TP session is closed. | EUICC\_REQ18 |
| 3 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |

This sub-sequence allows testing this requirement:

* + - * + EUICC\_REQ18

###### Open HTTPS Session on ISD-R

To open an HTTPS session on the ISD-R, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [OPEN\_SCP81\_SESSION]) |  | EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ54 |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The SCP80 status code is equal to ‘00’ – POR OK | EUICC\_REQ21 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* OPEN CHANNEL |  |  |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to   #BEARER\_DESCRIPTION   1. The buffer size is equal to   #BUFFER\_SIZE   1. The NAN is equal to #NAN\_VALUE 2. The port is equal to #TCP\_PORT 3. The IP is equal to #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ42 |
| 9 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The TLS records used here after SHALL be compliant with the Annex H.* | | | | |
| 10 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The CLIENT\_HELLO SHALL contain at least one of the cipher-suites accepted by the HTTPS server. | EUICC\_REQ14, EUICC\_REQ43 |
| 11 | DS → eUICC-UT | TLS\_SERVER\_HELLO  and TLS\_SERVER\_HELLO\_DONE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 12 | eUICC-UT → DS | TLS\_CLIENT\_KEY\_EXCHANGE  and TLS\_CHANGE\_CIPHER\_SPEC  and  TLS\_FINISHED | The CLIENT\_KEY\_EXCHANGE  SHALL contain the #PSK\_ID | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ45 |
| 13 | DS → eUICC-UT | TLS\_CHANGE\_CIPHER\_SPEC  and TLS\_FINISHED |  |  |
| 14 | eUICC-UT → DS | TLS\_APPLICATION with the first POST message | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher- suite negotiated during the TLS handshake 2. The HTTP content is empty 3. The POST URI is equal to   #POST\_URI   1. The headers are equal to #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47 |

This sub-sequence allows testing these requirements:

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ54

###### Open HTTPS Session on MNO-SD

To open an HTTPS session on the #MNO\_SD\_AID, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_SD\_TAR,  [OPEN\_SCP81\_MNO\_SESSION])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22 |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | *PROACTIVE COMMAND:* SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. The SCP80 status code is equal to ‘00’ – POR OK |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The buffer size is equal to   #BUFFER\_SIZE   1. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #TCP\_PORT   1. The IP is equal to   #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ14 |
| 9 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The TLS records used here after SHALL be compliant with the Annex H.* | | | | |
| 10 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The CLIENT\_HELLO SHALL  contain at least one of the cipher-suites accepted by the HTTPS server. | EUICC\_REQ14, EUICC\_REQ43 |
| 11 | DS → eUICC-UT | TLS\_SERVER\_HELLO  and TLS\_SERVER\_HELLO\_DONE |  |  |
| 12 | eUICC-UT → DS | TLS\_CLIENT\_KEY\_EXCHANGE  and TLS\_CHANGE\_CIPHER\_SPEC  and  TLS\_FINISHED | The CLIENT\_KEY\_EXCHANGE  SHALL contain the  #MNO\_PSK\_ID | EUICC\_REQ14, EUICC\_REQ43 |
| 13 | DS → eUICC-UT | TLS\_CHANGE\_CIPHER\_SPEC  and TLS\_FINISHED |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 14 | eUICC-UT → DS | TLS\_APPLICATION with the first POST message | 1. Decrypt the TLS record with the #MNO\_SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The HTTP content is empty 3. The POST URI is equal to   #POST\_URI   1. The headers are equal to #HOST #X\_ADMIN\_PROTOCOL   #X\_ADMIN\_FROM\_MNO | EUICC\_REQ14, EUICC\_REQ43 |

This sub-sequence allows testing these requirements:

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ22, EUICC\_REQ43

###### Close HTTPS Session

To close an HTTPS session, here are the different steps to execute:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | TLS\_APPLICATION with the HTTP code equal to #HTTP\_CODE\_204.  The header X-Admin-Protocol SHALL be present and equal to #X\_ADMIN\_PROTOCOL. |  |  |
| 2 | eUICC-UT → DS | TLS\_ALERT\_CLOSE\_NOTIFY |  | EUICC\_REQ14, EUICC\_REQ43 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND:*  CLOSE CHANNEL | The HTTP session is closed. | EUICC\_REQ14 |
| 4 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |

This sub-sequence allows testing these requirements:

* + - * + EUICC\_REQ14, EUICC\_REQ43

## OTA Transport Protocols

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ21\_1, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

SMS / CAT\_TP / HTTPS

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.TP.SMS.1: Transport\_SMS

###### Test Purpose

*To ensure remote application management is possible using SMS. The aim is to send an APDU (GET STATUS) over SMS. The compliance of the GET STATUS response is not verified during these tests.*

###### Referenced Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is in expanded format with definite length | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.TP.CAT\_TP.2: Transport\_CAT\_TP

###### Test Purpose

*To ensure remote application management is possible using CAT\_TP. The aim is to send an APDU (GET STATUS) over CAT\_TP. The compliance of the GET STATUS response is not verified during these tests.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is in expanded format with definite length | EUICC\_REQ13, EUICC\_REQ18 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.TP.HTTPS.3: Transport\_HTTPS

###### Test Purpose

*To ensure remote application management is possible using HTTPS. The aim is to send an APDU (GET STATUS) command over HTTPS. The compliance of the GET STATUS response is not verified during these tests.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ21\_1, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher- suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data in expanded format with indefinite length | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ48 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

###### Test Sequence N°2 – Nominal Case: No POR required in the SMS for HTTPS session triggering

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE\_NO\_POR, #ISD\_R\_TAR,  [OPEN\_SCP81\_SESSION]) | No POR sent by the eUICC | EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ54,  EUICC\_REQ21  \_1 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* OPEN CHANNEL |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The buffer size is equal to   #BUFFER\_SIZE   1. The NAN is equal to   #NAN\_VALUE   1. The port is equal to #TCP\_PORT 2. The IP is equal to #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ42 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 7 | Execute the generic sub-sequence “Open HTTPS Session on ISD-R” from step 10 to step 14 (as described in section 4.2.1.5) | | | |
| 8 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 9 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher- suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data in expanded format with indefinite length | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ48 |
| 10 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): CreateISDP

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ3, PF\_REQ7
        + EUICC\_REQ4, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + ISD-P #ISD\_P\_AID1 not present on the eUICC

###### Test Environment

ES5-CreateISDP

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.CISDP.1: CreateISDP\_SMS

###### Test Purpose

*To ensure the ISD-P creation process is well implemented on the eUICC using SMS. Several INSTALL commands with different parameters are sent. After ISD-P creation, the lifecycle state of the security domain is checked (SHALL be SELECTABLE).*

###### Referenced Requirements

* + - * + PF\_REQ3, PF\_REQ7
        + EUICC\_REQ4, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [INSTALL\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_009000] | PF\_REQ3, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum   using #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E3\_ISDP1\_07] | PF\_REQ3, PF\_REQ7, EUICC\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 - Nominal Case: Memory Quota Set

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [INSTALL\_ISDP\_MEM]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using   #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_009000] | PF\_REQ3, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E3\_ISDP1\_07] | PF\_REQ3, PF\_REQ7, EUICC\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.CISDP.2: CreateISDP\_CAT\_TP

###### Test Purpose

*To ensure the ISD-P creation process is well implemented on the eUICC using CAT\_TP. After ISD-P creation, the lifecycle state of the security domain is checked (SHALL be SELECTABLE).*

###### Referenced Requirements

PF\_REQ3, PF\_REQ7

EUICC\_REQ4, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [INSTALL\_ISDP]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_009000] | PF\_REQ3, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ23 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP1\_07] | PF\_REQ3, PF\_REQ7, EUICC\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.CISDP.3: CreateISDP\_HTTPS

###### Test Purpose

*To ensure the ISD-P creation process is well implemented on the eUICC using HTTPS. After ISD-P creation, the lifecycle state of the security domain is checked (SHALL be SELECTABLE).*

###### Referenced Requirements

PF\_REQ3, PF\_REQ7

EUICC\_REQ4, EUICC\_REQ12, EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [INSTALL\_ISDP]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to #POST\_URI 3. The different headers are   equal to  #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_009000] | PF\_REQ3, EUICC\_REQ12, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to #POST\_URI 3. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP1\_07] | PF\_REQ3, PF\_REQ7, EUICC\_REQ4, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): EnableProfile

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ4, PF\_REQ7
        + SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC
        + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### Test Environment

ES5-EnableProfile

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.EP.1: EnableProfile\_SMS

###### Test Purpose

*To ensure the Profile enabling process is well implemented on the eUICC using SMS. Some error cases due to incompatible initial conditions are also defined. In these error cases, the lifecycle state of the corresponding ISD-P is checked to make sure that it remains unchanged.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

* + - * + PF\_REQ4, PF\_REQ7
        + SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [ENABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_9000] | PF\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* REFRESH | see Note 1 |  |
| 8 | DS → eUICC-UT | FETCH |  |  |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ4 |
| 10 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Note 1: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

###### Test Sequence N°2 - Error Case: ISD-P Not Disabled

###### Initial Conditions

#ISD\_P\_AID1 in SELECTABLE state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [ENABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_6985] | PF\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_07] | PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 - Error Case: ISD-P with Incompatible POL1

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID contains the POL1 “Disabling of the Profile not allowed”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [ENABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_69E1] | PF\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ14 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.EP.2: EnableProfile\_CAT\_TP

###### Test Purpose

*To ensure the Profile enabling process is well implemented on the eUICC using CAT\_TP.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

PF\_REQ4

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [ENABLE\_ISDP1]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_9000] | PF\_REQ4, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) see Note 1 | | | |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* REFRESH | see Note 2 |  |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND: REFRESH* |  | PF\_REQ4 |
| 9 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| *Note 1: The closing of the CAT\_TP session MAY be performed automatically by the eUICC by sending the RST.*  *Note 2: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

###### TC.ES5.EP.3: EnableProfile\_HTTPS

###### Test Purpose

*To ensure the Profile enabling process is well implemented on the eUICC using HTTPS.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

PF\_REQ4

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [ENABLE\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_9000] | PF\_REQ4, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) see Note 1 | | | |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  REFRESH | see Note 2 |  |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ4 |
| 9 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Note 1: The closing of the HTTPS session MAY be performed automatically by the eUICC by sending the TLS\_ALERT\_CLOSE\_NOTIFY*  *Note 2: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

## ES5 (SM-SR – eUICC): DisableProfile

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ5, PF\_REQ7
        + SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

ES5-DisableProfile

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.DISP.1: DisableProfile\_SMS

###### Test Purpose

*To ensure the Profile disabling process is well implemented on the eUICC using SMS. Some error cases due to incompatible initial conditions are also defined. In these error cases, the lifecycle state of the corresponding ISD-P is checked to make sure that it remains unchanged.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

* + - * + PF\_REQ5, PF\_REQ7
        + SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID in Disabled state

No POL1 is defined on the #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DISABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_9000] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* REFRESH | see Note 1 |  |
| 8 | DS → eUICC-UT | FETCH |  |  |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ5 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Note 1: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

###### Test Sequence N°2 – Error Case: ISD-P Not Enabled

###### Initial Conditions

#ISD\_P\_AID1 in SELECTABLE state

#DEFAULT\_ISD\_P\_AID in Enabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [DISABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_6985] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_07] | PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Error Case: ISD-P with the Fall-back Attribute Set

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID in Disabled state

No POL1 is defined on the #ISD\_P\_AID1

#ISD\_P\_AID1 is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [DISABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_6985] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_3F] | PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Error Case: ISD-P with Incompatible POL1

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID in Disabled state

#ISD\_P\_AID1 contains the POL1 “Disabling of the Profile not allowed”

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DISABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_69E1] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ14 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_3F] | PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.DISP.2: DisableProfile\_CAT\_TP

###### Test Purpose

*To ensure the Profile disabling process is well implemented on the eUICC using CAT\_TP.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

PF\_REQ5

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID in Disabled state

No POL1 is defined on the #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [DISABLE\_ISDP1]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_9000] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) see Note 1 | | | |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* REFRESH | see Note 2 |  |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ5 |
| 9 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Note 1: The closing of the CAT\_TP session MAY be performed automatically by the eUICC by sending the RST.*  *Note 2: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

###### TC.ES5.DISP.3: DisableProfile\_HTTPS

###### Test Purpose

*To ensure the Profile disabling process is well implemented on the eUICC using HTTPS.*

*Note: As the update of the lifecycle states of the Profiles MAY become effective after the REFRESH command, the check of the lifecycle states cannot be performed in this test case.*

###### Referenced Requirements

PF\_REQ5

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID in Disabled state

No POL1 is defined on the #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [DISABLE\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_9000] | PF\_REQ5, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) see Note 1 | | | |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  REFRESH | see Note 2 |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | DS → eUICC-UT | FETCH |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ5 |
| 9 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Note 1: The closing of the HTTPS session MAY be performed automatically by the eUICC by sending the TLS\_ALERT\_CLOSE\_NOTIFY.*  *Note 2: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.* | | | | |

## ES5 (SM-SR – eUICC): SetFallbackAttribute

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7, PF\_REQ9
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC
        + #ISD\_P\_AID1 in Disabled state
        + #DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

###### Test Environment

ES5-SetFallbackAttribute

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.FB.1: SetFallbackAttribute\_SMS

###### Test Purpose

*To ensure it is possible to set the Fall-back Attribute on the eUICC using SMS. After changing the security domain with the Fall-back Attribute, a GET STATUS command is sent to make sure that the attribute is set on the targeted ISD-P.*

###### Referenced Requirements

* + - * + PF\_REQ7, PF\_REQ9
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [SET\_FALLBACK]) |  | EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_9000] | PF\_REQ9, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_FALLBACK]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E3\_ISDP1\_E1] | PF\_REQ7, PF\_REQ9, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.FB.2: SetFallbackAttribute\_CAT\_TP

###### Test Purpose

*To ensure it is possible to set the Fall-back Attribute on the eUICC using CAT\_TP. After changing the security domain with the Fall-back Attribute, a GET STATUS command is sent to make sure that the attribute is set on the targeted ISD-P.*

###### Referenced Requirements

PF\_REQ7, PF\_REQ9

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [SET\_FALLBACK]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_9000] | PF\_REQ9, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_FALLBACK]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP1\_E1] | PF\_REQ7, PF\_REQ9, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.FB.3: SetFallbackAttribute\_HTTPS

###### Test Purpose

*To ensure it is possible to set the Fall-back Attribute on the eUICC using HTTPS. After changing the security domain with the Fall-back Attribute, a GET STATUS command is sent to make sure that the attribute is set on the targeted ISD-P.*

###### Referenced Requirements

PF\_REQ7, PF\_REQ9

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [SET\_FALLBACK]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_9000] | PF\_REQ9, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_FALLBACK]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP1\_E1] | PF\_REQ7, PF\_REQ9, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): DeleteProfile

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ6, PF\_REQ7
        + SEC\_REQ12, SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

**DS**

ES5-DeleteProfile

ES5-eUICCCapabilityAudit

**eUICC-UT**

**SM-SR-S**

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC

###### Test Environment

###### TC.ES5.DP.1: DeleteProfile\_SMS

###### Test Purpose

*To ensure the Profile deletion process is well implemented on the eUICC using SMS. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC. Some error cases due to incompatible initial conditions are also defined.*

###### Referenced Requirements

* + - * + PF\_REQ6, PF\_REQ7
        + SEC\_REQ12, SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 defined on #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_009000] | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88] | PF\_REQ6, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ12 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Error Case: ISD-P Not Disabled

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

No POL1 defined on #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [DELETE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6985] | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_3F] | PF\_REQ6, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Error Case: ISD-P with the Fall-back Attribute Set

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 defined on #ISD\_P\_AID1

#ISD\_P\_AID1 is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6985] | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ6, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Error Case: ISD-P with Incompatible POL1

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#ISD\_P\_AID1 contains the POL1 “Deletion of the Profile not allowed”

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [DELETE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_69E1] | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ14 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ6, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°5 – Error Case: ISD-P not present on the eUICC

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 defined on #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

The Profile identified by the ISD-P AID #ISD\_P\_AID\_UNKNOWN is not present on the eUICC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE\_ISDP\_UNKNOWN]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88]   Note: Status code 6A82 MAY also be returned. | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.DP.2: DeleteProfile\_CAT\_TP

###### Test Purpose

*To ensure the Profile deletion process is well implemented on the eUICC using CAT\_TP. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC.*

###### Referenced Requirements

PF\_REQ6, PF\_REQ7

SEC\_REQ12

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 defined on #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [DELETE\_ISDP1]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_009000] | PF\_REQ6, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_6A88] | PF\_REQ6, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, SEC\_REQ12 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.DP.3: DeleteProfile\_HTTPS

###### Test Purpose

*To ensure the Profile deletion process is well implemented on the eUICC using HTTPS. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC.*

###### Referenced Requirements

PF\_REQ6, PF\_REQ7

SEC\_REQ12

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

No POL1 is defined on the #ISD\_P\_AID1

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [DELETE\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_009000] | PF\_REQ6, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_6A88] | PF\_REQ6, PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, SEC\_REQ12 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): eUICCCapabilityAudit

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.ECA.1: eUICCCapabilityAudit\_SMS

###### Test Purpose

*To ensure it is possible to audit the eUICC using SMS. GET STATUS and GET DATA commands are sent to retrieve the ISD-P list, the ECASD certificate, the eUICC recognition data and the card resources information.*

###### Referenced Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: Retrieve all ISD-P

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_LIST]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST3]   (see Note 1) | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: If more than one Profile is pre-installed on the eUICC, this response SHALL be adapted in consequence (in addition of the Enabled ISD-P identified by the AID #DEFAULT\_ISD\_P\_AID and the ISD-P identified by the AID #ISD\_P\_AID1, other Profiles MAY be present).* | | | | |

###### Test Sequence N°2 – Nominal Case: Retrieve Default Enabled ISD-P

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: Retrieve Disabled ISD-P

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_DISABLED]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] (see Note 1) | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: If more than one Profile is pre-installed on the eUICC (i.e. several Disabled Profiles exist), this response SHALL be adapted in consequence (in addition of the ISD-P identified by the AID #ISD\_P\_AID1, other Profiles MAY be present).* | | | | |

###### Test Sequence N°4 – Nominal Case: Retrieve Card Resources Information

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_FF21]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_FF21] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°5 – Nominal Case: Retrieve ECASD Recognition Data

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_BF30\_REC]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_BF30\_REC] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°6 – Nominal Case: Retrieve ECASD Certificate Store

###### Initial Conditions

None

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_BF30\_CERT]) |  | | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  | |  |
| 4 | DS → eUICC-UT | FETCH |  | |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_BF30\_ECASD] 2. The #PK\_ECASD\_ECKA is   equal to the content of the TAG ‘7F49’   1. The signature (i.e. TAG ‘5F37’) SHALL be verified using the #EUM\_PK\_ECDSA 2. TAG ‘42’ is equal to   #EUM\_OID   1. TAG ‘95’ is equal to   #KEY\_USAGE   1. TAG ’73’ contains the TLV ‘C0’, ‘C1’, ‘C2’ and ‘C9’ 2. TAG ‘C9’ is equal to   #EUM\_SUBJECT\_KEY\_ID   1. TAG ‘5F20’ contains the   #EID | | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ | |  |

###### Test Sequence N°7 – Nominal Case: Retrieve ISD-P with Memory Information

###### Initial Conditions

#ISD\_P\_AID1 in SELECTABLE state and created using the command

[INSTALL\_ISDP\_MEM]

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1\_MEM]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_MEM] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Void

###### TC.ES5.ECA.2: eUICCCapabilityAudit\_CAT\_TP

###### Test Purpose

*To ensure it is possible to audit the eUICC using CAT\_TP. GET STATUS and GET DATA commands are sent to retrieve the ISD-P list, the ECASD certificate, the eUICC recognition data and the card resources information.*

###### Referenced Requirements

PF\_REQ7

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: Retrieve all Information

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_LIST]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP\_LIST3]  (see Note 1) | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 7 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP\_DISABLED]) |  | EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP1\_1F]  (see Note 2) | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 9 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_DATA\_FF21]) |  | EUICC\_REQ54 |
| 10 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_FF21] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 11 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_BF30\_REC]) |  | EUICC\_REQ54 |
| 12 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_BF30\_REC] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 13 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_BF30\_CERT]) |  | EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 14 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_BF30\_ECASD] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 15 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| *Note 1: If more than one Profile is pre-installed on the eUICC, this response SHALL be adapted in consequence (in addition of the Enabled ISD-P identified by the AID #DEFAULT\_ISD\_P\_AID and the ISD-P identified by the AID #ISD\_P\_AID1, other Profiles MAY be present).*  *Note 2: If more than one Profile is pre-installed on the eUICC (i.e. several Disabled Profiles exist), this response SHALL be adapted in consequence (in addition of the ISD-P identified by the AID #ISD\_P\_AID1).* | | | | |

###### TC.ES5.ECA.3: eUICCCapabilityAudit\_HTTPS

###### Test Purpose

*To ensure it is possible to audit the eUICC using HTTPS. GET STATUS and GET DATA commands are sent to retrieve the ISD-P list, the ECASD certificate, the eUICC recognition data and the card resources information.*

###### Referenced Requirements

PF\_REQ7

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: Retrieve all Information

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP\_LIST]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP\_LIST3]   (see Note 1) | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP\_DISABLED]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 8 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP1\_1F]   (see Note 2) | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 9 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DATA\_FF21]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 10 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_FF21] | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 11 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DATA\_BF30\_REC]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 12 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_BF30\_REC] | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 13 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DATA\_BF30\_CERT]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 14 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_BF30\_CERT] | PF\_REQ7, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 15 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |
| *Note 1: If more than one Profile is pre-installed on the eUICC, this response SHALL be adapted in consequence (in addition of the Enabled ISD-P identified by the AID #DEFAULT\_ISD\_P\_AID and the ISD-P identified by the AID #ISD\_P\_AID1, other Profiles MAY be present).*  *Note 2: If more than one Profile is pre-installed on the eUICC (i.e. several Disabled Profiles exist), this response SHALL be adapted in consequence (in addition of the ISD-P identified by the AID #ISD\_P\_AID1).* | | | | |

## ES5 (SM-SR – eUICC): MasterDelete

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7, PF\_REQ8, PF\_REQ8\_1, PF\_REQ8\_2
        + SEC\_REQ12, SEC\_REQ14
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process *ES8-EstablishISDPKeySet* has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

* + - * + #ISD\_P\_AID1 contains a keyset ‘70’ with an AES key (16 bytes long)

A PUT KEY command as defined in the GlobalPlatform Card Specification [[3]](#_bookmark8) SHOULD be used to initialize the {TOKEN\_KEY}

The value of the {TOKEN\_KEY} can be freely chosen by the test tool

* + - * + #ISD\_P\_AID1 contains the SDIN value #ISD\_P\_SDIN\*
        + #ISD\_P\_AID1 contains the SIN value #ISD\_P\_SIN\*
        + #ISD\_P\_AID1 contains the Application Provider Identifier value

#ISD\_P\_PROV\_ID\*

* *To set the SDIN, SIN and the Application Provider Identifier, the sequence below SHALL be executed just after the establishment of the ISD-P keysets:*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]; [STORE\_SIN]; [STORE\_PROV\_ID]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | PF\_REQ8\_1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:    1. SW=’9000’ or ‘6108’ | PF\_REQ8\_1 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Environment

ES5-MasterDelete

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.MD.1: MasterDelete\_SMS

###### Test Purpose

*To ensure the master deletion process is well implemented on the eUICC using SMS. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC. Some error cases due to incompatible initial conditions or incorrect values in commands are also defined.*

###### Referenced Requirements

* PF\_REQ7, PF\_REQ8, PF\_REQ8\_1, PF\_REQ8\_2
* SEC\_REQ12, SEC\_REQ14
* EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_009000] | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ12 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: With default Application Provider identifier (5F20)

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

#ISD\_P\_AID1 contains the SDIN value #ISD\_P\_SDIN\*

#ISD\_P\_AID1 contains the SDN value #ISD\_P\_SIN\*

#ISD\_P\_AID1 does not contain any Application Provider Identifier value \*

* *To set the SDIN and the SIN, the sequence below SHALL be executed just after the establishment of the ISD-P keysets (this overrides the related general initial condition defined in this section):*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  | ENVELOPE\_SMS\_PP( |  |  |
|  |  | #SPI\_VALUE, |  |
|  |  | #ISD\_P\_TAR1, |  |
| 1 | DS → eUICC-UT | SCP03\_SCRIPT(  #SCP03\_KVN, | PF\_REQ8\_1 PF\_REQ8\_2 |
|  |  | [STORE\_SDIN]; |  |
|  |  | [STORE\_SIN])) |  |
|  |  | Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | PROACTIVE COMMAND: SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:    1. SW=’9000’ or ‘6108’ | PF\_REQ8\_1 PF\_REQ8\_2 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [MASTER\_DEL\_ISDP1\_RID]) |  | EUICC\_REQ22, EUICC\_REQ54, PF\_REQ8\_2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_009000] | PF\_REQ8, PF\_REQ8\_2,  EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_6A88] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ12 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: ISD-P with POL1 “Deletion not allowed”

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

#ISD\_P\_AID1 contains the POL1 “Deletion of the Profile not allowed”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_009000] | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ14 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, SEC\_REQ12 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Error Case: ISD-P Not Disabled

###### Initial Conditions

#ISD\_P\_AID1 in Enabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_6985] | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_3F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°5 – Error Case: ISD-P with the Fall-back Attribute Set

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#ISD\_P\_AID1 is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_6985] | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°6 – Error Case: Wrong Token Value

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [BAD\_MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_6985]  (see Note 1) | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6982’* | | | | |

###### Test Sequence N°7 – Error Case: With empty Application Provider identifier (5F20)

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

#ISD\_P\_AID1 contains the SDIN value #ISD\_P\_SDIN\*

#ISD\_P\_AID1 contains the SIN value #ISD\_P\_SIN\*

#ISD\_P\_AID1 does not contain any Application Provider Identifier value \*

*\* To set the SDIN and the SIN, the sequence below SHALL be executed just after the establishment of the ISD-P keysets (this overrides the related general initial condition defined in this section):*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  | ENVELOPE\_SMS\_PP( |  |  |
|  |  | #SPI\_VALUE, |  |
|  |  | #ISD\_P\_TAR1, |  |
| 1 | DS → eUICC-UT | SCP03\_SCRIPT(  #SCP03\_KVN, | PF\_REQ8\_1 PF\_REQ8\_2 |
|  |  | [STORE\_SDIN]; |  |
|  |  | [STORE\_SIN])) |  |
|  |  | Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |
| 2 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 3 | DS → eUICC-UT | FETCH |  |  |
| 4 | eUICC-UT → DS | PROACTIVE COMMAND: SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:    1. SW=’9000’ or ‘6108’ | PF\_REQ8\_1 PF\_REQ8\_2 |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS →  eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [MASTER\_DEL\_ISDP1\_NO\_PROV\_ID]) |  | EUICC\_REQ22, EUICC\_REQ54, PF\_REQ8\_2 |
| 3 | eUICC-UT  → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS →  eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT  → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the   cryptographic checksum using #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_6985]   (see Note 1) | PF\_REQ8, PF\_REQ8\_2,  EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS →  eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS →  eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT  → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS →  eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT  → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is   equal to [R\_AB\_E3\_ISDP1\_1 F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS →  eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6982’* | | | | |

###### Test Sequence N°8 – Error Case: With incorrect SDIN

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [MASTER\_DEL\_ISDP1\_INV\_SDIN]) |  | EUICC\_REQ22, EUICC\_REQ54, |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC- UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6985]   (see Note 1) | PF\_REQ8,  EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC- UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6982’* | | | | |

###### Test Sequence N°9 – Error Case: With incorrect SIN

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [MASTER\_DEL\_ISDP1\_INV\_SIN]) |  | EUICC\_REQ22, EUICC\_REQ54, |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC- UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6985]   (see Note 1) | PF\_REQ8,  EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC- UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6982’* | | | | |

###### Test Sequence N°10 – Error Case: With incorrect Application Provider ID

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

No POL1 defined on #ISD\_P\_AID1

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [MASTER\_DEL\_ISDP1\_RID]) |  | EUICC\_REQ22, EUICC\_REQ54, |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC- UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6985]   (see Note 1) | PF\_REQ8,  EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC- UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6982’* | | | | |

###### TC.ES5.MD.2: MasterDelete\_CAT\_TP

###### Test Purpose

*To ensure the master deletion process is well implemented on the eUICC using CAT\_TP. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC.*

###### Referenced Requirements

PF\_REQ7, PF\_REQ8, PF\_REQ8\_1

SEC\_REQ12

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_009000] | PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_6A88] | PF\_REQ7, PF\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, SEC\_REQ12 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.MD.3: MasterDelete\_HTTPS

###### Test Purpose

*To ensure the master deletion process is well implemented on the eUICC using HTTPS. After ISD-P deletion, a GET STATUS command is sent to make sure that the security domain is no longer present on the eUICC.*

###### Referenced Requirements

PF\_REQ7, PF\_REQ8, PF\_REQ8\_1

SEC\_REQ12

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

#ISD\_P\_AID1 in Disabled state

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [MASTER\_DEL\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_009000] | PF\_REQ8, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_6A88] | PF\_REQ7, PF\_REQ8, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, SEC\_REQ12 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): EstablishISDRKeySet

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54
        + PROC\_REQ13\_1

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

ES5-EstablishISDRKeyset

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.EISDRK.1: EstablishISDRKeyset\_SMS

###### Test Purpose

*To ensure the ISD-R keyset establishment process is well implemented on the eUICC using SMS. After SCP80 keys initialization on ISD-R, a new secure channel session is opened to make sure that the new keys have been set. During the key establishment, different parameters are used (DR, HostID) to make sure that all configurations are supported on the eUICC. An error case is defined to test that an incorrect SM-SR certificate is rejected.*

###### Referenced Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [STORE\_SR\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RC] 4. Retrieve the {RC} 5. The {RC} length is either 16 or 32 bytes | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24, PROC\_REQ13  \_1 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
|  |  | #ISD\_R\_TAR, |  |
| 7 | DS → eUICC-UT | STORE\_ISDR\_KEYS( |  |
|  |  | #SC3\_NO\_DR; |  |
|  |  | {RC}), |  |
|  |  | #LAST\_SCRIPT) |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
|  |  |  | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RECEIPT] 4. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 5. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:* |  |
|  |  | SEND SHORT MESSAGE |  |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 12 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [GET\_ISDP\_ENABLED]) |  |
|  |  | Use {SCP\_KENC}, {SCP\_KMAC} and  {SCP\_KDEK} |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the {SCP\_KENC} 2. Verify the cryptographic checksum using   {SCP\_KMAC}   1. The response data is equal to [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal case: DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [STORE\_SR\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RC] 4. Retrieve the {RC} 5. The {RC} length is either 16 or 32 bytes | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24, PROC\_REQ13  \_1 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
|  |  | #ISD\_R\_TAR, |  |
| 7 | DS → eUICC-UT | STORE\_ISDR\_KEYS( |  |
|  |  | #SC3\_DR; |  |
|  |  | {RC}), |  |
|  |  | #LAST\_SCRIPT) |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
|  |  |  | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_02RECEIPT\_DR] 2. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 3. Derive keyset from ShS and   {DR} and retrieve the  {SCP\_KENC}, {SCP\_KMAC}  and {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:* |  |
|  |  | SEND SHORT MESSAGE |  |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 12 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [GET\_ISDP\_ENABLED]) |  |
|  |  | Use {SCP\_KENC}, {SCP\_KMAC} and  {SCP\_KDEK} |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the {SCP\_KENC} 2. Verify the cryptographic checksum using   {SCP\_KMAC}   1. The response data is equal to [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: DR, Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [STORE\_SR\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RC] 4. Retrieve the {RC} 5. The {RC} length is either 16 or 32 bytes | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24, PROC\_REQ13  \_1 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
|  |  | #ISD\_R\_TAR, |  |
| 7 | DS → eUICC-UT | STORE\_ISDR\_KEYS( |  |
|  |  | #SC3\_DR\_HOST; |  |
|  |  | {RC}), |  |
|  |  | #LAST\_SCRIPT) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
|  |  |  | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RECEIPT\_DR] 4. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 5. Derive keyset from ShS (using {DR}, #HOST\_ID, #ISD\_R\_SIN and #ISD\_R\_SDIN) and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:* |  |
|  |  | SEND SHORT MESSAGE |  |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 12 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [GET\_ISDP\_ENABLED]) |  |
|  |  | Use {SCP\_KENC}, {SCP\_KMAC} and  {SCP\_KDEK} |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the {SCP\_KENC} 2. Verify the cryptographic checksum using   {SCP\_KMAC}   1. The response data is equal to [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Error Case: Invalid SM-SR Certificate

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR, |  |
|  |  | [STORE\_INVALID\_SR\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
|  |  |  | 1- Decrypt the response | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
|  |  |  | packet with the |
|  |  |  | #SCP80\_ENC\_KEY |
|  |  |  | 2- Verify the cryptographic |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | checksum using  #SCP80\_AUTH\_KEY  3- The response data is equal |
|  |  |  | to [R\_AB\_026982] |
|  |  |  | (see Note) |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: The SW MAY be also ‘6A80’* | | | | |

###### TC.ES5.EISDRK.2: EstablishISDRKeyset\_CAT\_TP

###### Test Purpose

*To ensure the ISD-R keyset establishment process is well implemented on the eUICC using CAT\_TP. After ISD-R keys initialization, a new secure channel is opened to make sure that the new keys have been set.*

###### Referenced Requirements

PF\_REQ7

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ24, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [STORE\_SR\_CERTIF], #FIRST\_SCRIPT) |  | EUICC\_REQ54 |
|  |  |  | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_02RC]   1. Retrieve the {RC} | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ24 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR |  |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_ISDR\_KEYS(  #SC3\_NO\_DR;  {RC}), #LAST\_SCRIPT) |  | EUICC\_REQ54 |
|  |  |  | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_02RECEIPT]   1. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 2. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ24 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_ENABLED])  Use {SCP\_KENC}, {SCP\_KMAC}  and {SCP\_KDEK} |  | EUICC\_REQ54 |
| 8 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the {SCP\_KENC} 3. Verify the cryptographic   checksum using {SCP\_KMAC}   1. The response data is equal to   [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ24 |
| 9 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.EISDRK.3: EstablishISDRKeyset\_HTTPS

###### Test Purpose

*To ensure the ISD-R keyset establishment process is well implemented on the eUICC using HTTPS. After ISD-R keys initialization, a new secure channel is opened to make sure that the new keys have been set.*

###### Referenced Requirements

PF\_REQ7

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ24, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [STORE\_SR\_CERTIF]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_RC] 2. Retrieve the {RC} | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ24, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( STORE\_ISDR\_KEYS(  #SC3\_NO\_DR;  {RC})) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_RECEIPT] 2. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 3. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ24, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |
| 8 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_ENABLED])  Use {SCP\_KENC}, {SCP\_KMAC}  and {SCP\_KDEK} |  | EUICC\_REQ54 |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 10 | DS → eUICC-UT | FETCH |  |  |
| 11 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the {SCP\_KENC} 2. Verify the cryptographic checksum using {SCP\_KMAC} 3. The response data is equal to   [R\_AB\_E3\_ISDP\_3F] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ24 |
| 12 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

## ES5 (SM-SR – eUICC): FinaliseISDRhandover

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + An additional keyset with the key version number #SCP80\_NEW\_KVN is initialized on the ISD-R

###### Test Environment

ES5-FinaliseISDRhandover

Get keys information

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.FIH.1: FinaliseISDRhandover\_SMS

###### Test Purpose

*To ensure it is possible to delete ISD-R keys on the eUICC using SMS. After keysets deletion, a GET DATA (TAG ‘E0’ – key information template) is sent to retrieve all the keysets present on the ISD-R to make sure that the range of keyset has been deleted correctly. Some error cases due to inconsistent values in commands are also defined.*

###### Referenced Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: Delete All Keys except SCP80 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [DELETE1\_KEYSETS]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_009000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_E0]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E0\_SCP80]   (i.e. no #SCP80\_NEW\_KVN  returned) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [PUTKEY\_SCP81]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_PUTKEY] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |

###### Test Sequence N°2 – Nominal Case: Delete All Keys except SCP80 and SCP81 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE2\_KEYSETS]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_009000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DATA\_E0]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E0\_SCP80\_SCP81]   (i.e. no #SCP80\_NEW\_KVN  returned) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Error Case: Delete All SCP80 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE\_SCP80\_KEYSETS]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_6985] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ25 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES5.FIH.2: FinaliseISDRhandover\_CAT\_TP

###### Test Purpose

*To ensure it is possible to delete ISD-R keys on the eUICC using CAT\_TP. After keysets deletion, a GET DATA (TAG ‘E0’ – key information template) is sent to retrieve all the keysets present on the ISD-R to make sure that the range of keyset has been deleted correctly.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ25, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: Delete All Keys except SCP80 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [DELETE1\_KEYSETS]) |  | EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_009000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ25 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_DATA\_E0]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E0\_SCP80]  (i.e. no #SCP80\_NEW\_KVN  returned) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ25 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### Test Sequence N°2 – Nominal Case: Delete All Keys except SCP80 and SCP81 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [DELETE2\_KEYSETS]) |  | EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_009000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ25 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_DATA\_E0]) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_E0\_SCP80\_SCP81] (i.e. no #SCP80\_NEW\_KVN  returned) | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ25 |
| 7 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.FIH.3: FinaliseISDRhandover\_HTTPS

###### Test Purpose

*To ensure it is possible to delete ISD-R keys on the eUICC using HTTPS. After keysets deletion, a GET DATA (TAG ‘E0’ – key information template) is sent to retrieve all the keysets present on the ISD-R to make sure that the range of keyset has been deleted correctly.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ25, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: Delete All Keys except SCP80 and SCP81 Keys

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [DELETE2\_KEYSETS]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_009000] | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ25, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_DATA\_E0]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E0\_SCP80\_SCP81   (i.e. no #SCP80\_NEW\_KVN  returned) | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ25, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 7 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): UpdateSMSRAddressingParameters

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26, EUICC\_REQ26\_1, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

ES5-UpdateSMSRAddressingParameters

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.USAP.1: UpdateSMSRAddrParam\_SMS

###### Test Purpose

*To ensure it is possible to update SM-SR addressing parameters on the eUICC using SMS, and that the eUICC deletes all previously stored information related to each concerned protocol subtag and just store the new set of parameters.*

*N.B.: Each of the subtags ‘A3’, ‘A4’, ‘A5’ is related to a different protocol, and can be updated without altering the configuration for the other protocols.*

*Some error cases due to inconsistent values in commands are also defined.*

###### Referenced Requirements

* EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26, EUICC\_REQ26\_1, EUICC\_REQ54

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case: Update SMS Parameters

###### Initial Conditions

* #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)
* #DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute
* #ISD\_P\_AID1 in Disabled state
* #ISD\_P\_AID1 has been personalized with the following SCP03 keys: o {SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK}
* No POL1 is defined on the #DEFAULT\_ISD\_P\_AID and on the #ISD\_P\_AID1
* The SMS mode is the default way (priority order1) to send the notification
* TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR2
* SMS-C parameters have been set on #DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1

with #TON\_NPI and #DIALING\_NUMBER

* For both #DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1, TP-PID and TP-DCS are set to default values (no specific values have been set)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 3 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now enabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR2 (see step 5 of the test sequence defined in section 4.2.13.2.1.1)  Check that TP-PID and TP-DCS are set to default value:   * TP-PID = ‘00’ * TP-DCS = ‘04’ |  |
| 4 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [STORE\_SMS\_PARAM]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 5 | eUICC-UT → DS | *PROACTIVE COMMANDPENDING:*  SEND SHORT MESSAGE |  |  |
| 6 | DS → eUICC- UT | FETCH |  |  |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_9000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26 |
| 8 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 9 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN,  [STORE\_SMS\_PARAM\_MNO1])) |  | EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ54 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 11 | DS → eUICC- UT | FETCH |  |  |
| 12 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:    1. SW=’9000’ or ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 13 | DS → eUICC- UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 14 | Execute the test sequence defined in section 4.2.5.2.1.1 (TC.ES5.DISP.1:DisableProfile\_SMS) from step 2 to step 10 in order to disable the #ISD\_P\_AID1 | | All steps executed successfully |  |
| 15 | Execute the test sequence defined in section 4.2.14.2.1.1 (TC.ES5.NOTIFPD.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Disabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR (see step 5 of the test sequence defined in section 4.2.14.2.1.1)  Check that TP-PID and TP-DCS are set to default value:   * TP-PID = ‘00’ * TP-DCS = ‘04’ | EUICC\_REQ26 |
| 16 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 17 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now enabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR (see step 5 of the test sequence defined in section 4.2.13.2.1.1)  Check that TP-PID and TP-DCS are the values set in Step 9 :   * TP-PID is set to #PID * TP-DCS is set to #DCS |  |

###### Test Sequence N°2 – Nominal Case: Update SMS Parameters with Profiles-Specific SM-SR Destination Addresses

###### Initial Conditions

* #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)
* #DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute
* #ISD\_P\_AID1 in Disabled state
* No POL1 is defined on the #DEFAULT\_ISD\_P\_AID and on the #ISD\_P\_AID1
* The SMS mode is the default way (priority order1) to send the notification
* TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR
* SMS-C parameters have been set on #DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1

with #TON\_NPI and #DIALING\_NUMBER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| *Set a specific SM-SR destination address on both Profiles (#DEFAULT\_ISD\_P\_AID and on the #ISD\_P\_AID1)* | | | | |
| 2 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [STORE\_SMS\_PARAM\_ISDPS]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC- UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_9000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26, EUICC\_REQ26\_1 |
| 6 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 7 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now enabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR3 (see step 5 of the test sequence defined in section 4.2.13.2.1.1) | EUICC\_REQ26, EUICC\_REQ26\_1 |
| 8 | Execute the test sequence defined in section 4.2.5.2.1.1 (TC.ES5.DISP.1:DisableProfile\_SMS) from step 2 to step 10 in order to disable the #ISD\_P\_AID1 | | All steps executed successfully |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | Execute the test sequence defined in section 4.2.14.2.1.1 (TC.ES5.NOTIFPD.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Disabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR2 (see step 5 of the test sequence defined in section 4.2.14.2.1.1) | EUICC\_REQ26, EUICC\_REQ26\_1 |
| *Set a specific SM-SR destination address only on the Default Profile (#DEFAULT\_ISD\_P\_AID)* | | | | |
| 10 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [STORE\_SMS\_PARAM\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 11 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 12 | DS → eUICC- UT | FETCH |  |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_9000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26,  EUICC\_REQ26\_1 |
| 14 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 15 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now enabled | | All steps successfully executed  The TP-Destination-Address present in the eUICC Notification (SMS) is equal to #DEST\_ADDR (see step 5 of the test sequence defined in  section 4.2.13.2.1.1) | EUICC\_REQ26, EUICC\_REQ26\_1 |

###### VOID

###### VOID

###### TC.ES5.USAP.2: UpdateSMSRAddrParam\_CAT\_TP

###### Test Sequence N°1 – Nominal Case: Update CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [STORE\_CATTP\_PARAM]) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to [R\_AB\_9000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ26 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES5.USAP.3: UpdateSMSRAddrParam\_HTTPS

###### Test Purpose

*To ensure it is possible to update SM-SR addressing parameters on the eUICC using HTTPS.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ26, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: Update HTTPS Parameters

###### Initial Conditions

#DEFAUT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

#ISD\_P\_AID1 in Disabled state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID and on the #ISD\_P\_AID1

The HTTP mode is the default way (priority order 1) to send the notification in both

#DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1

HTTPS Connectivity Parameters have been set on #ISD\_R\_AID with #TCP\_PORT, #IP\_VALUE2, #ADMIN\_HOST, #AGENT\_ID, #PSK\_ID, #SCP81\_KVN, #SCP81\_KEY\_ID and #ADMIN\_URI

HTTPS Connectivity Parameters have been set on #ISD\_P\_AID1 and on the

#DEFAULT\_ISD\_P\_AID with #BEARER\_DESCRIPTION, #NAN\_VALUE, #LOGIN

and #PWD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Execute the test sequence defined in section 4.2.4.2.3 (TC.ES5.EP.3:EnableProfile\_HTTPS) from step 2 to step 9 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 3 | Execute the test sequence defined in section 4.2.13.2.3.1 (TC.ES5.NOTIFPE.3:Notification\_HTTPS) from step 2 to step 14 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed  The Data Destination-Address present in the OPEN CHANNEL is equal to #IP\_VALUE2 (see step 5 of the test sequence defined in section 4.2.13.2.3.1) |  |
| 4 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_HTTPS\_PARAM) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 5 | DS → eUICC- UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_9000] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ26 |
| 7 | Execute the test sequence defined in section 4.2.5.2.1.1 (TC.ES5.DISP.1:DisableProfile\_SMS) from step 2 to step 10 in order to disable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 8 | Execute the test sequence defined in section 4.2.13.2.3.1 (TC.ES5.NOTIFPE.3:Notification\_HTTPS) from step 2 to step 14 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Disabled | | All steps successfully executed  The Data Destination-Address present in the OPEN CHANNEL is equal to #IP\_VALUE (see step 5 of the test sequence defined in section 4.2.13.2.3.1) | EUICC\_REQ26 |
| 9 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES5 (SM-SR – eUICC): Notification on Profile Enabling

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ4, PF\_REQ7
        + PM\_REQ3, PM\_REQ4
        + PROC\_REQ6, PROC\_REQ8, PROC\_REQ20, PROC\_REQ2, PROC\_REQ5\_1
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + The #ISD\_P\_AID1 has just been Enabled

REFRESH proactive command has been sent by the eUICC

To Enable this Profile, the Profile enabling process SHALL be used (i.e. the test sequence defined in section [4.2.4.2.1.1](#_bookmark70) MAY be executed)

###### Test Environment

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.NOTIFPE.1: Notification\_SMS

###### Test Purpose

*To ensure SMS notification procedure is well implemented when a Profile is Enabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case.*

###### Referenced Requirements

* + - * + PF\_REQ4, PF\_REQ7
        + PM\_REQ3, PM\_REQ4
        + PROC\_REQ6, PROC\_REQ8, PROC\_REQ20, PROC\_REQ5\_1
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ54

###### Initial Conditions

* + - * + The SMS mode is the default way (priority order 1) to send the notification
        + TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR
        + SMS-C parameters have been set on #ISD\_P\_AID1 with #TON\_NPI and

#DIALING\_NUMBER

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #DEFAULT\_ISD\_P\_AID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the   {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20, |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]; [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST1] | PM\_REQ3, PM\_REQ4, PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### Test Sequence N°2 – Nominal Case: Follow-up Activity

###### Initial Conditions

The previous Enabled ISD-P’s (i.e. #DEFAULT\_ISD\_P\_AID) POL1 contains the rule “Profile deletion is mandatory when it is disabled”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the   {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_NOTIF1] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using   . #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_6A88] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### Test Sequence N°3 – Nominal Case: No Follow-up Activities when the Profile is set with the Fall-Back Attribute and POL1 “Profile deletion is mandatory when its state is changed to disabled”

###### Initial Conditions

POL1 “Profile deletion is mandatory when its state is changed to disabled” is defined in the previous Enabled ISD-P (i.e. #DEFAULT\_ISD\_P\_AID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination- Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the   {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20, |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20,  PROC\_REQ5\_ 1 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1];  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST1] | PM\_REQ3, PM\_REQ4, PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22,  PROC\_REQ5\_ 1 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### Test Sequence N°4 – Error Case: SM-SR Unreachable

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #DEFAULT\_ISD\_P\_AID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *Start loop while maximum retries number is not reached*  *(The maximum number of retries to wait for a Notification SHALL be given by the EUM to the Test Tool Provider)* | | | | |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE  See Note 4 | This proactive command MAY be triggered by either an ENVELOPE(TIMER MANAGEMENT) or a STATUS  command (maximum number of STATUS commands SHALL be given by the EUM to the Test Tool Provider) | EUICC\_REQ27, PROC\_REQ6, PROC\_REQ8, PROC\_REQ20 |
| 8 | DS → eUICC-UT | FETCH |  |  |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the {NOTIF\_NUMBER}: it SHALL be the same as the previous one | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ6, PROC\_REQ8, PROC\_REQ20 |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *End loop* | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 11 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* REFRESH  See note 4 | This proactive command MAY be triggered by either an ENVELOPE(TIMER MANAGEMENT) or a STATUS  command (maximum number of STATUS commands SHALL be given by the EUM to the Test Tool Provider) | EUICC\_REQ27, PROC\_REQ6, PROC\_REQ8, PROC\_REQ20 |
| 12 | DS → eUICC-UT | FETCH |  |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PM\_REQ3, PROC\_REQ6, PROC\_REQ8 |
| 14 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 15 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 16 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 17 | DS → eUICC-UT | FETCH |  |  |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_ROLL\_BACK   (see Note 1)   1. Extract the {NOTIF\_NUMBER}: it SHALL NOT be the same as the previous one | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ6, PROC\_REQ8 |
| 19 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 20 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [NOTIF\_CONFIRMATION]) |  | PROC\_REQ6, PROC\_REQ8, EUICC\_REQ54 |
| 21 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 22 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 23 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using   2. #SCP80\_AUTH\_KEY  3- The response data is equal to  [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ6, PROC\_REQ8 |
| 24 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 25 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ54 |
| 26 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 27 | DS → eUICC-UT | FETCH |  |  |
| 28 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP\_3F] | PM\_REQ3, PM\_REQ4, PF\_REQ7, PROC\_REQ6, PROC\_REQ8, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29 |
| 29 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).*  *Note 4: It is assumed that some proactive commands TIMER MANAGEMENT or POLL INTERVALL MAY be sent by the eUICC between iterations of the loop. The Device Simulator SHALL honor these commands as per section 3.2.1.1* | | | | |

###### TC.ES5.NOTIFPE.2: Notification\_CAT\_TP

###### Test Purpose

*To ensure CAT\_TP notification procedure is well implemented when a Profile is Enabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case.*

###### Referenced Requirements

* PF\_REQ4, PF\_REQ7
* PM\_REQ3, PM\_REQ4
* EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ54

###### Initial Conditions

* The CAT\_TP mode is the default way (priority order 1) to send the notification

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #DEFAULT\_ISD\_P\_AID)

CAT\_TP Connectivity Parameters have been set on #ISD\_R\_AID with #UDP\_PORT, #CAT\_TP\_PORT and #IP\_VALUE

CAT\_TP Connectivity Parameters have been set on #ISD\_P\_AID1 with

#BEARER\_DESCRIPTION, #NAN\_VALUE, #LOGIN and #PWD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL | 3. |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #UDP\_PORT   1. The IP is equal to   #IP\_VALUE   1. The login/password are equal to #LOGIN/#PWD | EUICC\_REQ18, EUICC\_REQ27 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The CAT\_TP PDU used here after SHALL be compliant with the* [*Annex G.*](#_bookmark286) | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | eUICC-UT → DS | SYN | The identification data MAY contain the #EID | EUICC\_REQ18 |
| 8 | DS → eUICC-UT | SYN\_ACK |  |  |
| 9 | eUICC-UT → DS | ACK\_NO\_DATA | The CAT\_TP session is open. | EUICC\_REQ18 |
| 10 | eUICC-UT → DS | ACK\_DATA containing the notification | 1. The ACK\_DATA contains a command packet 2. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE   (see Note 1)   1. Extract the   {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ27, EUICC\_REQ54 |
| 11 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [NOTIF\_CONFIRMATION]) |  | EUICC\_REQ54 |
| 12 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ29 |
| 13 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 14 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1];  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 16 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 17 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST1] | PM\_REQ3, PM\_REQ4, PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 18 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### TC.ES5.NOTIFPE.3: Notification\_HTTPS

###### Test Purpose

*To ensure HTTPS notification procedure is well implemented when a Profile is Enabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case.*

###### Referenced Requirements

PF\_REQ4, PF\_REQ7

PM\_REQ3, PM\_REQ4

PROC\_REQ21

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS mode is the default way (priority order 1) to send the notification

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #DEFAULT\_ISD\_P\_AID)

HTTPS Connectivity Parameters have been set on #ISD\_R\_AID with #TCP\_PORT, #IP\_VALUE, #ADMIN\_HOST, #AGENT\_ID, #PSK\_ID, #SCP81\_KVN, #SCP81\_KEY\_ID and #ADMIN\_URI

HTTPS Connectivity Parameters have been set on #ISD\_P\_AID1 with

#BEARER\_DESCRIPTION, #NAN\_VALUE, #LOGIN and #PWD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #TCP\_PORT   1. The IP is equal to   #IP\_VALUE   1. The login/password are equal to #LOGIN/#PWD | EUICC\_REQ13, EUICC\_REQ14, PROC\_REQ21 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The TLS records used here after SHALL be compliant with the Annex H.* | | | | |
| 7 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The CLIENT\_HELLO SHALL  contain at least one of the cipher-suites accepted by the HTTPS server. | EUICC\_REQ14, EUICC\_REQ43, PROC\_REQ21 |
| 8 | DS → eUICC-UT | TLS\_SERVER\_HELLO  and TLS\_SERVER\_HELLO\_DONE |  | PROC\_REQ21 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | eUICC-UT → DS | TLS\_CLIENT\_KEY\_EXCHANGE  and TLS\_CHANGE\_CIPHER\_SPEC  and  TLS\_FINISHED | The CLIENT\_KEY\_EXCHANGE  SHALL contain the #PSK\_ID | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ45, PROC\_REQ21 |
| 10 | DS → eUICC-UT | TLS\_CHANGE\_CIPHER\_SPEC  and TLS\_FINISHED |  | PROC\_REQ21 |
| 11 | eUICC-UT → DS | TLS\_APPLICATION with the first POST message | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The HTTP content is empty 3. The POST URI is equal to   #POST\_URI\_NOTIF (see  Note 1)   1. The headers are equal to #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R | EUICC\_REQ14, EUICC\_REQ27, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, PROC\_REQ21 |
| 12 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [NOTIF\_CONFIRMATION]) |  | EUICC\_REQ29, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, PROC\_REQ21 |
| 13 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_NOTIF] | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ29, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, PROC\_REQ21 |
| 14 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 15 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]; [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 16 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 17 | DS → eUICC-UT | FETCH |  |  |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST1] | PM\_REQ3, PM\_REQ4, PF\_REQ4, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 19 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

## ES5 (SM-SR – eUICC): Notification on Profile Disabling

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ5, PF\_REQ7
        + PM\_REQ3, PM\_REQ4
        + PROC\_REQ20, PROC\_REQ21
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ43,

EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + The #ISD\_P\_AID1 has just been Disabled

REFRESH proactive command has been sent by the eUICC

To Disable this Profile, the Profile disabling process SHALL be used (i.e. the test sequence defined in section [4.2.5.2.1.1](#_bookmark77) MAY be executed)

* + - * + #DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

###### Test Environment

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES5-eUICCCapabilityAudit

**eUICC-UT**

**DS**

**SM-SR-S**

###### TC.ES5.NOTIFPD.1: Notification\_SMS

###### Test Purpose

*To ensure SMS notification procedure is well implemented when a Profile is Disabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case (the ISD-P with the Fall-back Attribute SHALL be Enabled).*

###### Referenced Requirements

* + - * + PF\_REQ5, PF\_REQ7
        + PM\_REQ3, PM\_REQ4
        + PROC\_REQ20
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ54

###### Initial Conditions

* + - * + The SMS mode is the default way (priority order 1) to send the notification
        + TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR
        + SMS-C parameters have been set on #DEFAULT\_ISD\_P\_AID with #TON\_NPI and

#DIALING\_NUMBER

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #ISD\_P\_AID1)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE2   (see Note 1)   1. Extract the {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1];  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP\_LIST2] | PM\_REQ3, PM\_REQ4, PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### Test Sequence N°2 – Nominal Case: Follow-up Activity

###### Initial Conditions

The previous Enabled ISD-P’s (i.e. #ISD\_P\_AID1) POL1 contains the rule “Profile deletion is mandatory when it is disabled”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination-Address is equal to #DEST\_ADDR 2. The SMS-C address is equal   to #TON\_NPI + #DIALING\_NUMBER   1. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE2   (see Note 1)   1. Extract the {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_NOTIF2] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20, |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to   [R\_AB\_6A88] | PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### TC.ES5.NOTIFPD.2: Notification\_CAT\_TP

###### Test Purpose

*To ensure CAT\_TP notification procedure is well implemented when a Profile is Disabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case (the ISD-P with the Fall-back Attribute SHALL be Enabled).*

###### Referenced Requirements

PF\_REQ5, PF\_REQ7

PM\_REQ3, PM\_REQ4

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ54

###### Initial Conditions

The CAT\_TP mode is the default way (priority order 1) to send the notification

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #ISD\_P\_AID1)

CAT\_TP Connectivity Parameters have been set on #ISD\_R\_AID with #UDP\_PORT, #CAT\_TP\_PORT and #IP\_VALUE

CAT\_TP Connectivity Parameters have been set on #DEFAULT\_ISD\_P\_AID with

#BEARER\_DESCRIPTION, #NAN\_VALUE, #LOGIN and #PWD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* OPEN CHANNEL |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #UDP\_PORT   1. The IP is equal to #IP\_VALUE 2. The login/password are equal to #LOGIN/#PWD | EUICC\_REQ18, EUICC\_REQ27 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The CAT\_TP PDU used here after SHALL be compliant with the* [*Annex G.*](#_bookmark286) | | | | |
| 7 | eUICC-UT → DS | SYN | The identification data MAY contain the #EID | EUICC\_REQ18 |
| 8 | DS → eUICC-UT | SYN\_ACK |  |  |
| 9 | eUICC-UT → DS | ACK\_NO\_DATA | The CAT\_TP session is open. | EUICC\_REQ18 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | ACK\_DATA containing the notification | 1. The ACK\_DATA contains a command packet 2. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHANGE2   (see Note 1)   1. Extract the {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ27, EUICC\_REQ54 |
| 11 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [NOTIF\_CONFIRMATION]) |  | EUICC\_REQ54 |
| 12 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ29 |
| 13 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 14 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1];  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 16 | DS → eUICC-UT | FETCH |  |  |
| 17 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_E3\_ISDP\_LIST2] | PM\_REQ3, PM\_REQ4, PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 18 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

###### TC.ES5.NOTIFPD.3: Notification\_HTTPS

###### Test Purpose

*To ensure HTTPS notification procedure is well implemented when a Profile is Disabled.*

*Note: As the update of the lifecycle states MAY become effective after the REFRESH command, the check of the lifecycle states of the Profiles is performed in this test case (the ISD-P with the Fall-back Attribute SHALL be Enabled).*

###### Referenced Requirements

PF\_REQ5, PF\_REQ7

PM\_REQ3, PM\_REQ4

PROC\_REQ21

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS mode is the default way (priority order 1) to send the notification

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: No Follow-up Activities

###### Initial Conditions

No POL1 defined in the previous Enabled ISD-P (i.e. #ISD\_P\_AID1)

HTTPS Connectivity Parameters have been set on #ISD\_R\_AID with #TCP\_PORT, #IP\_VALUE, #ADMIN\_HOST, #AGENT\_ID, #PSK\_ID, #SCP81\_KVN, #SCP81\_KEY\_ID and #ADMIN\_URI

HTTPS Connectivity Parameters have been set on #DEFAULT\_ISD\_P\_AID with

#BEARER\_DESCRIPTION, #NAN\_VALUE, #LOGIN and #PWD

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| 2 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #TCP\_PORT   1. The IP is equal to   #IP\_VALUE   1. The login/password are equal to #LOGIN/#PWD | EUICC\_REQ13, EUICC\_REQ14, PROC\_REQ21 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The TLS records used here after SHALL be compliant with the Annex H.* | | | | |
| 7 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The CLIENT\_HELLO SHALL  contain at least one of the cipher-suites accepted by the HTTPS server. | EUICC\_REQ14, EUICC\_REQ43, PROC\_REQ21 |
| 8 | DS → eUICC-UT | TLS\_SERVER\_HELLO  and TLS\_SERVER\_HELLO\_DONE |  | PROC\_REQ21 |
| 9 | eUICC-UT → DS | TLS\_CLIENT\_KEY\_EXCHANGE  and TLS\_CHANGE\_CIPHER\_SPEC  and  TLS\_FINISHED | The CLIENT\_KEY\_EXCHANGE  SHALL contain the #PSK\_ID | EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ45, PROC\_REQ21 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | DS → eUICC-UT | TLS\_CHANGE\_CIPHER\_SPEC  and TLS\_FINISHED |  | PROC\_REQ21 |
| 11 | eUICC-UT → DS | TLS\_APPLICATION with the first POST message | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The HTTP content is empty 3. The POST URI is equal to   #POST\_URI\_NOTIF2 (see  Note 1)   1. The headers are equal to #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R | EUICC\_REQ14, EUICC\_REQ27, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, PROC\_REQ21 |
| 12 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [NOTIF\_CONFIRMATION]) |  | EUICC\_REQ29, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52, PROC\_REQ21 |
| 13 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_NOTIF] | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ29, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, PROC\_REQ21 |
| 14 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |
| 15 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1];  [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 16 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 17 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP\_LIST2] | PM\_REQ3, PM\_REQ4, PF\_REQ5, PF\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 19 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The tag ‘14’ (or ‘94’) with the IMEI value and the tag ‘6D’ (or ‘ED’) with the MEID provided in the TERMINAL*  *RESPONSE(PROVIDE LOCAL INFORMATION) sent during the toolkit initialization process MAY be also present in the notification.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events before issuing the notification (within a maximum time interval of 10 STATUS events).* | | | | |

## ES6 (MNO – eUICC): UpdatePOL1byMNO

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PM\_REQ6
        + PROC\_REQ17
        + EUICC\_REQ7, EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ43, EUICC\_REQ48, EUICC\_REQ52

###### Test Cases

###### General Initial Conditions

* + - * + None

###### Test Environment

**DS**

ES6-UpdatePOL1byMNO

**eUICC-UT**

**MNO-S**

###### TC.ES6.UPOL1MNO.1: UpdatePOL1byMNO\_SMS

###### Test Purpose

*To ensure MNO can update POL1 on the eUICC using SMS. Some error cases due to inconsistent values in commands are also defined.*

###### Referenced Requirements

* + - * + PM\_REQ6
        + PROC\_REQ17
        + EUICC\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: No Rule

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ17 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_POL1\_NO\_RULE]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: Disabling Not Allowed

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ17 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_POL1\_DIS]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: Deletion and Disabling Not Allowed

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ17 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_POL1\_DEL\_DIS]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Nominal Case: Delete when Disabled

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  [INSTALL\_PERSO\_RES\_ISDP]; [STORE\_POL1\_DEL\_AUTO])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22, PROC\_REQ17 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°5 – Error Case: Bad POL1 Value

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  [INSTALL\_PERSO\_RES\_ISDP]; [BAD\_STORE\_POL1])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22, PROC\_REQ17 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_026A80] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°6 – Error Case: Associated ISD-P Not Enabled

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID is in Enabled state (SHALL be the initial state of the eUICC)

#ISD\_P\_AID1 in Disabled state

For this test sequence, #MNO\_TAR (MNO-SD TAR of the Profile linked to

#DEFAULT\_ISD\_P\_AID) is set to ‘010203’ and SHALL not be equal to ‘B20100’

MNO-SD TAR of the Profile linked to the #ISD\_P\_AID1 is set to ‘B20100’ (as defined in section B.7.1)

#DEFAULT\_ISD\_P\_AID contains the POL1 “Disabling of the Profile not allowed”

MNO-SD SCP80 keys of the Profile linked to the #ISD\_P\_AID1 are the same as the ones configured in the Profile #DEFAULT\_ISD\_P\_AID (i.e. #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY and #MNO\_SCP80\_DATA\_ENC\_KEY)

The SMS mode is the default way (priority order 1) to send the notification

TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR

SMS-C parameters have been set on #DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1

with #TON\_NPI and #DIALING\_NUMBER

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  [INSTALL\_PERSO\_RES\_ISDP]; [STORE\_POL1\_NO\_RULE])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | DS → eUICC-UT | FETCH | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] |  |
| 5 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 6 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 7 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to  the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| 8 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  [INSTALL\_PERSO\_RES\_ISDP]; [STORE\_POL1\_NO\_RULE])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22, PROC\_REQ17 |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE | See Note 1 |  |
| 10 | DS → eUICC-UT | FETCH | The SCP80 status code is ‘09’ – TAR unknown | PM\_REQ6, PROC\_REQ17, EUICC\_REQ7, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| *Note 1: Depending on the implementation, the eUICC MAY decide to not send back a POR (i.e. SW ‘9000’ on the ENVELOPE command). Therefore, the steps 9, 10 and 11 SHALL be considered as optional.* | | | | |

###### TC.ES6.UPOL1MNO.2: UpdatePOL1byMNO\_CAT\_TP

###### Test Purpose

*To ensure MNO can update POL1 on the eUICC using CAT\_TP.*

###### Referenced Requirements

PM\_REQ6

PROC\_REQ17

EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18, EUICC\_REQ22

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: No Rule

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on MNO-SD as described in section [4.2.1.3](#_bookmark53) | | | |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #MNO\_TAR,  [INSTALL\_PERSO\_RES\_ISDP]; [STORE\_POL1\_NO\_RULE])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  | PROC\_REQ17 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ18 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES6.UPOL1MNO.3: UpdatePOL1byMNO\_HTTPS

###### Test Purpose

*To ensure MNO can update POL1 on the eUICC using HTTPS.*

###### Referenced Requirements

PM\_REQ6

PROC\_REQ17

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ22, EUICC\_REQ43, EUICC\_REQ48, EUICC\_REQ52

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #MNO\_PSK\_ID

PSK value: #MNO\_SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: No Rule

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on MNO-SD as described in section [4.2.1.6](#_bookmark56) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_POL1\_NO\_RULE]) |  | PROC\_REQ17 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #MNO\_SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_MNO #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_029000] | PM\_REQ6, PROC\_REQ17, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ43, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES6 (MNO – eUICC): UpdateConnectivityParametersByMNO

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PM\_REQ7
        + PROC\_REQ18
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Test Cases

###### General Initial Conditions

* + - * + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### Test Environment

ES6-UpdateConnectivityParametersByMNO

**eUICC-UT**

**DS**

**MNO-S**

###### TC.ES6.UCPMNO.1: UpdateConnectParamByMNO\_SMS

###### Test Purpose

*To ensure MNO can update the Connectivity Parameters on the eUICC using SMS, and configure the order of protocols used for the notitications.*

###### Referenced Requirements

* + - * + PM\_REQ7
        + PROC\_REQ18
        + EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ28

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: Update SMS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ18 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_SMS\_PARAM\_MNO]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ7, PROC\_REQ18, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: Update CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ18 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_CATTP\_PARAM\_MNO]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ7, PROC\_REQ18, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: Update HTTPS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ18 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_HTTPS\_PARAM\_MNO]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ7, PROC\_REQ18, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Nominal Case: Update HTTPS + SMS Parameters

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

#DEFAULT\_ISD\_P\_AID is the Profile with the Fall-back Attribute

#ISD\_P\_AID1 present on the eUICC, in Disabled state

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID and on the #ISD\_P\_AID1

The SMS mode is the only way (priority order n°1, and no other protocol set) to send the notification on both ISD-P

SMS-C parameters has been set on #ISD\_P\_AID1 with #TON\_NPI and

#DIALING\_NUMBER

SMS-C parameters has been set on #DEFAULT\_ISD\_P\_AID with #TON\_NPI and

#DIALING\_NUMBER\_INITIAL

TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR

HTTPS Connectivity Parameters have been set on #ISD\_R\_AID with #TCP\_PORT, #IP\_VALUE, #ADMIN\_HOST, #AGENT\_ID, #PSK\_ID, #SCP81\_KVN, #SCP81\_KEY\_ID and #ADMIN\_URI

###### Specific conditions during execution of the test

The test sequence changes the Connectivity Parameters in the #DEFAULT\_ISD\_P\_AID, and also verifies that the following notification sequence obeys the new Connectivity Parameters.

In order to trigger usage of both notification protocols, the DS SHALL be configured to reject HTTPS session opening, but allow SMS notification to succeed.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| *Update Connectivity Parameters via ES6* | | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, PROC\_REQ18 |
|  |  | #MNO\_TAR, |  |
| 2 | DS → eUICC-UT | [INSTALL\_PERSO\_RES\_ISDP];  [STORE\_HTTPSSMS\_PARAM]) |  |
|  |  | Use #MNO\_SCP80\_ENC\_KEY, |  |
|  |  | #MNO\_SCP80\_AUTH\_KEY, |  |
|  |  | #MNO\_SCP80\_DATA\_ENC\_KEY |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KE Y 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_029000] | PM\_REQ7, PROC\_REQ18, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Enable #ISD\_P\_AID1* | | | | |
| 7 | Execute the test sequence defined in section 4.2.4.2.1.1 (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 10 in order to enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 8 | Execute the test sequence defined in section 4.2.13.2.1.1 (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 2 to step 11 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| *Disable #ISD\_P\_AID1* | | | | |
| 9 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [DISABLE\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 11 | DS → eUICC-UT | FETCH |  |  |
| 12 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using   #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_9000] | PF\_REQ5, EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, |
| 13 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 14 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  REFRESH | see Note 1 |  |
| 15 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 16 | eUICC-UT → DS | *PROACTIVE COMMAND:*  REFRESH |  | PF\_REQ5 |
| 17 | DS → eUICC-UT | RESET | ATR returned by eUICC |  |
| *Handle notification sequence such that HTTP notification fails* | | | | |
| 18 | DS → eUICC-UT | [TERMINAL\_PROFILE] | Toolkit initialization see Note 2 and Note 3 |  |
| 19 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 20 | DS → eUICC-UT | FETCH |  |  |
| 21 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTI ON 2. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #TCP\_PORT   1. The IP is equal to   #IP\_VALUE   1. The login/password are equal to #LOGIN/#PWD | EUICC\_REQ13, EUICC\_REQ14, PROC\_REQ21, EUICC\_REQ28 |
| 22 | DS → eUICC-UT | TERMINAL RESPONSE with Result field  = ‘21’ (Network currently unable to process command) | See Note 4 |  |
| *Loop on steps 19 to 22 (see Note 4) while maximum retries number is not reached*  *(The maximum number of retries for HTTP session establishment SHALL be given by the EUM to the Test Tool Provider)* | | | | |
| *Handle notification in SMS sequence such that SMS notification succeeds* | | | | |
| 23 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 24 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 25 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. The TP-Destination- Address is equal to #DEST\_ADDR 2. The SMS-C address is equal to #TON\_NPI + #DIALING\_NUMBER 3. The SPI is equal to   #SPI\_NOTIF   1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The secured data SHALL only contain the TLV #NOTIF\_PROFILE\_CHAN   GE2 (see Note 1) 6- Extract the  {NOTIF\_NUMBER} | EUICC\_REQ16, EUICC\_REQ27, EUICC\_REQ54, PROC\_REQ20, EUICC\_REQ28 |
| 26 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 27 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [NOTIF\_CONFIRMATION]) |  | PROC\_REQ20, EUICC\_REQ54 |
| 28 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 29 | DS → eUICC-UT | FETCH |  |  |
| 30 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using   #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_NOTIF] | EUICC\_REQ13, EUICC\_REQ16, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ29, PROC\_REQ20 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| *Note 1: Before sending the REFRESH command, the eUICC MAY wait for several STATUS events. In this case, the eUICC SHALL issue the REFRESH command within a maximum time interval of 10 STATUS events.*  *Note 2: 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, the DS*  *SHALL send the corresponding FETCH and TERMINAL RESPONSE(successfully performed) commands.*  *Note 3: Depending on the implementation, it MAY be necessary to send an ENVELOPE (EVENT DOWNLOAD - Location status) indicating “normal service” (i.e. ‘00’) in order to trigger the sending of the eUICC notification. This envelope SHALL be sent only if this event (i.e. encoded with the value ‘03’) is present in the SET UP EVENT LIST sent by the eUICC. Moreover, the eUICC MAY also wait for several STATUS events between the notifications (within a maximum time interval of 10 STATUS events).*  *Note 4: It is assumed that some proactive commands TIMER MANAGEMENT or POLL INTERVALL MAY be sent by the eUICC between iterations of the loop. The Device Simulator SHALL honor these commands as per section 3.2.1.1* | | | | |

## ES8 (SM-DP – eUICC): EstablishISDPKeySet

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7
        + PM\_REQ8
        + EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ15, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC
        + #ISD\_P\_AID1 in SELECTABLE state

###### Test Environment

ES8-EstablishISDPKeyset

ES5-eUICCCapabilityAudit

Store SDIN over SCP03

**eUICC-UT**

**DS**

**SM-SR-S**

**SM-DP-S**

###### TC.ES8.EISDPK.1: EstablishISDPKeyset\_SMS

###### Test Purpose

*To ensure the ISD-P keyset establishment process is well implemented on the eUICC using SMS. After ISD-P SCP03 keys initialization, the lifecycle state of the ISD-P is checked (SHALL be PERSONALIZED) and a new secure channel session is opened to make sure that the new keys have been set. During the key establishment, different parameters are used (DR, HostID) to make sure that all configurations are supported on the eUICC. An error case is defined to test that an incorrect SM-DP certificate is rejected.*

###### Referenced Requirements

* + - * + PF\_REQ7
        + PM\_REQ8
        + EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ17, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [INSTALL\_PERSO\_ISDP1]; [STORE\_DP\_CERTIF], #FIRST\_SCRIPT) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_03RC] 4. Retrieve the {RC} 5. The {RC} length is either 16 or 32 bytes | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_ISDP\_KEYS(  #SC3\_NO\_DR;  {RC}), #LAST\_SCRIPT) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RECEIPT] 4. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 5. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 17 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 19 | DS → eUICC-UT | FETCH |  |  |
| 20 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. The response data is equal to [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ8, EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 21 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 22 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ54 |
| 23 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 24 | DS → eUICC-UT | FETCH |  |  |
| 25 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. No SCP03 security error is   raised in the response data (i.e. INITIALIZE UPDATE and EXTERNAL  AUTHENTICATE commands are successfully executed) | EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ23 |
| 26 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR,  [INSTALL\_PERSO\_ISDP1]; |  |
|  |  | [STORE\_DP\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:* | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_03RC] 4. Retrieve the {RC} | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
|  |  | SEND SHORT MESSAGE |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_ISDP\_KEYS(  #SC3\_DR;  {RC}), #LAST\_SCRIPT) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RECEIPT\_DR] 4. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 5. Derive keyset from ShS and {DR} and retrieve the   {SCP\_KENC},  {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ54 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ8, EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 17 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ54 |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 19 | DS → eUICC-UT | FETCH |  |  |
| 20 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. No SCP03 security error is raised in the response data (i.e. INITIALIZE UPDATE and EXTERNAL AUTHENTICATE   commands are successfully executed) | EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ23 |
| 21 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: DR, Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [INSTALL\_PERSO\_ISDP1]; [STORE\_DP\_CERTIF], #FIRST\_SCRIPT) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_03RC] 4. Retrieve the {RC} | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_ISDP\_KEYS(  #SC3\_DR\_HOST;  {RC}), #LAST\_SCRIPT) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_02RECEIPT\_DR] 4. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 5. Derive keyset from ShS (using {DR}, #HOST\_ID, #ISD\_R\_SIN and #ISD\_R\_SDIN) and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 12 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 14 | DS → eUICC-UT | FETCH |  |  |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ8, EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 17 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ54 |
| 18 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 19 | DS → eUICC-UT | FETCH |  |  |
| 20 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. No SCP03 security error is raised in the response data (i.e. INITIALIZE UPDATE and EXTERNAL   AUTHENTICATE commands are successfully executed) | EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ23 |
| 21 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Error Case: Invalid SM-DP Certificate

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ22, EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #ISD\_R\_TAR,  [INSTALL\_PERSO\_ISDP1]; |  |
|  |  | [STORE\_INVALID\_DP\_CERTIF], |  |
|  |  | #FIRST\_SCRIPT) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  |  | 1- Decrypt the response | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
|  |  |  | packet with the |
|  |  |  | #SCP80\_ENC\_KEY |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 2. The response data is equal |
|  |  |  | to [R\_AB\_036982] |
|  |  |  | (see Note) |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: The SW MAY be also ‘6A80’* | | | | |

###### TC.ES8.EISDPK.2: EstablishISDPKeyset\_CAT\_TP

###### Test Purpose

*To ensure the ISD-P keyset establishment process is well implemented on the eUICC using CAT\_TP. After ISD-P SCP03 keys initialization, the lifecycle state of the ISD-P is checked (SHALL be PERSONALIZED) and a new secure channel session is opened to make sure that the new keys have been set.*

###### Referenced Requirements

PF\_REQ7

PM\_REQ8

EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ53, EUICC\_REQ54

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR,  [INSTALL\_PERSO\_ISDP1]; [STORE\_DP\_CERTIF],  #FIRST\_SCRIPT) |  | EUICC\_REQ54 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_03RC]   1. Retrieve the {RC} | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ18 |
| 5 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, STORE\_ISDP\_KEYS(  #SC3\_NO\_DR;  {RC}), #LAST\_SCRIPT) |  | EUICC\_REQ54 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_02RECEIPT]   1. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 2. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | PM\_REQ8, EUICC\_REQ13, EUICC\_REQ18 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ54 |
| 8 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is equal to   [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ15, EUICC\_REQ18 |
| 9 | DS → eUICC-UT | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ54 |
| 10 | eUICC-UT → DS | ACK\_DATA with POR | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. No SCP03 security error is raised in the response data (i.e. INITIALIZE UPDATE and EXTERNAL AUTHENTICATE   commands are successfully executed) | EUICC\_REQ18, EUICC\_REQ23 |
| 11 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES8.EISDPK.3: EstablishISDPKeyset\_HTTPS

###### Test Purpose

*To ensure the ISD-P keyset establishment process is well implemented on the eUICC using HTTPS. After ISD-P SCP03 keys initialization, the lifecycle state of the ISD-P is checked (SHALL be PERSONALIZED) and a new secure channel session is opened to make sure that the new keys have been set.*

###### Referenced Requirements

PF\_REQ7

PM\_REQ8

EUICC\_REQ5, EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ15, EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT(  [INSTALL\_PERSO\_ISDP1]; [STORE\_DP\_CERTIF]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_02RC] 2. Retrieve the {RC} | PM\_REQ8, EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( STORE\_ISDP\_KEYS(  #SC3\_NO\_DR;  {RC})) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_RECEIPT] 2. Calculate ShS from #SM\_ESK\_ECKA and #PK\_ECASD\_ECKA 3. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | PM\_REQ8, EUICC\_REQ14, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 7 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT( [GET\_ISDP1]) |  | EUICC\_REQ49, EUICC\_REQ50, EUICC\_REQ52 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data equal to [R\_AF\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ8, EUICC\_REQ5, EUICC\_REQ14, EUICC\_REQ15, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 9 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1 SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_SDIN]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK}) |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52 |
| 10 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. No SCP03 security error is raised in the response data (i.e. INITIALIZE UPDATE and EXTERNAL AUTHENTICATE   commands are successfully executed) | EUICC\_REQ14, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 11 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## ES8 (SM-DP – eUICC): DownloadAndInstallation

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7
        + PM\_REQ3, PM\_REQ9
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ53, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61
        + SEC\_REQ23

###### Test Cases

###### General Initial Conditions

* + - * + #ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process *ES8-EstablishISDPKeySet* has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

###### Test Environment

**DS**

**SM-SR-S**

ES8-DownloadAnInstallation

ES5-eUICCCapabilityAudit

**eUICC-UT**

**SM-DP-S**

###### TC.ES8.DAI.1: DownloadAndInstallation\_CAT\_TP

###### Test Purpose

*To ensure Profile download is possible on the eUICC using CAT\_TP. A generic Profile is downloaded and script chaining, as defined in ETSI TS 102 226* [*[6]*](#_bookmark11)*, is used in this sequence.*

*After the execution of the download process, an audit is sent to make sure that the new Profile is Disabled. An error case is also defined to check that the ISD-P lifecycle state remains unchanged when the Profile is not fully downloaded.*

###### Referenced Requirements

* + - * + PF\_REQ7
        + PM\_REQ3, PM\_REQ9
        + EUICC\_REQ13, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ53, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61
        + SEC\_REQ23

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The #PROFILE\_PACKAGE SHALL be split in several parts named from

{PROFILE\_PART1} to {PROFILE\_PARTn} in this sequence (n = the last index of the sub part). Each Profile part contains a list of PEs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
|  |  | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, |  | EUICC\_REQ17, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58 |
|  |  | #ISD\_P\_TAR1, |  |
| 3 | DS → eUICC-UT | SCP03T\_SCRIPT( |  |
|  |  | #SCP03\_KVN, |  |
|  |  | {PROFILE\_PART1}), |  |
|  |  | #FIRST\_SCRIPT) |  |
|  |  | Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  |  | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is formatted in an expanded remote command structure with definite length coding 5. The response to the INITIALIZE UPDATE TLV   command (i.e. TAG ‘84’) SHALL be equal to [R\_SCP03T\_INITUP\_OK]   1. The response to the   EXTERNAL AUTHENTICATE TLV  command (i.e. TAG ‘85’) SHALL be equal to [R\_SCP03T\_EXTAUTH\_OK]   1. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61 |
| 4 | eUICC-UT → DS | ACK\_DATA with POR |  |
| *Loop until the Profile part index (named i) is equal to n-1* | | | | |
|  |  | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, |  | EUICC\_REQ17, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58 |
| 5 | DS → eUICC-UT | #ISD\_P\_TAR1, |  |
|  |  | SCP03T\_SUB\_SCRIPT( |  |
|  |  | {PROFILE\_PARTi}), |  |
|  |  | #SUB\_SCRIPT) |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  |  | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is formatted in an expanded remote command structure with definite length coding 5. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ61 |
| 6 | eUICC-UT → DS | ACK\_DATA with POR |  |
| *End loop* | | | | |
|  |  | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, |  | EUICC\_REQ17, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58 |
| 7 | DS → eUICC-UT | #ISD\_P\_TAR1, |  |
|  |  | SCP03T\_SUB\_SCRIPT( |  |
|  |  | {PROFILE\_PARTn}), |  |
|  |  | #LAST\_SCRIPT) |  |
|  |  |  | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is formatted in an expanded remote command structure with definite length coding 5. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned (except for the last one)   1. Decrypt the last SCP03t response using the SCP03 session key and check the R- MAC 2. The content of the last SCP03t response data is equal to #R\_PROF\_PKG\_OK | PM\_REQ9, EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ61, SEC\_REQ23 |
| 8 | eUICC-UT → DS | ACK\_DATA with POR |  |
| 9 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 11 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 12 | DS → eUICC-UT | FETCH |  |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal   to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PM\_REQ3, EUICC\_REQ13, EUICC\_REQ22 |
| 14 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Error Case: Profile Downloading Interrupted

###### Initial Conditions

The #PROFILE\_PACKAGE SHALL be split in several parts named from

{PROFILE\_PART1} to {PROFILE\_PARTn} in this sequence (n = the last index of the sub part). Each Profile part contains a list of PEs. Note that only the

{PROFILE\_PART1} needs to be sent in the following test.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
|  |  | ACK\_DATA containing the result of  SCP80\_PACKET( #SPI\_VALUE, |  | EUICC\_REQ17, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58 |
|  |  | #ISD\_P\_TAR1, |  |
| 3 | DS → eUICC-UT | SCP03T\_SCRIPT( |  |
|  |  | #SCP03\_KVN, |  |
|  |  | {PROFILE\_PART1}), |  |
|  |  | #FIRST\_SCRIPT) |  |
|  |  | Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA contains a response packet 2. Decrypt the response packet with the #SCP80\_ENC\_KEY 3. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 4. The response data is formatted in an expanded remote command structure with definite length coding 5. The response to the INITIALIZE UPDATE TLV   command (i.e. TAG ‘84’) SHALL be equal to [R\_SCP03T\_INITUP\_OK]   1. The response to the   EXTERNAL AUTHENTICATE TLV  command (i.e. TAG ‘85’) SHALL be equal to [R\_SCP03T\_EXTAUTH\_OK]   1. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) (the other Profile Elements SHALL NOT be sent) | | | |
| 6 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 8 | DS → eUICC-UT | FETCH |  |  |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ3, EUICC\_REQ13, EUICC\_REQ22 |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES8.DAI.2: DownloadAndInstallation\_HTTPS

###### Test Purpose

*To ensure Profile download is possible on the eUICC using HTTP. A generic Profile is downloaded. Contrary to the test case that uses CAT\_TP (section 4.2.18.2.1), no script chaining has to be used over HTTP. After the execution of the download process, an audit is sent to make sure that the new Profile is Disabled. An error case is also defined to check that the ISD-P lifecycle state remains unchanged when the Profile is not fully downloaded.*

###### Referenced Requirements

PF\_REQ7

PM\_REQ3, PM\_REQ9

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ54, EUICC\_REQ57, EUICC\_REQ58, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61

SEC\_REQ23

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The #PROFILE\_PACKAGE SHALL be split in several parts named from

{PROFILE\_PART1} to {PROFILE\_PARTn} in this sequence (n = the last index of the sub part). Each Profile part contains a list of PEs.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1, SCP03T\_SCRIPT(  #SCP03\_KVN,  {PROFILE\_PART1}))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ57, EUICC\_REQ58 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data formatted in an expanded remote command structure with indefinite length coding 2. The response to the INITIALIZE UPDATE TLV   command (i.e. TAG ‘84’) SHALL be equal to [R\_SCP03T\_INITUP\_OK]   1. The response to the EXTERNAL AUTHENTICATE TLV   command (i.e. TAG ‘85’) SHALL be equal to [R\_SCP03T\_EXTAUTH\_OK]   1. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ14, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61 |
| *Loop until the Profile part index (named i) is equal to n-1* | | | | |
| 5 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1, SCP03T\_SUB\_SCRIPT(  {PROFILE\_PARTi})) |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ57, EUICC\_REQ58 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data formatted in an expanded remote command structure with indefinite length coding 2. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ14, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, EUICC\_REQ61, SEC\_REQ23 |
| *End loop* | | | | |
| 7 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1, SCP03T\_SUB\_SCRIPT(  {PROFILE\_PARTn})) |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ57, EUICC\_REQ58 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data formatted in an expanded remote command structure with indefinite length coding 2. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned (except for the last one)   1. Decrypt the last SCP03t response using the SCP03 session key and check the R- MAC 2. The content of the last SCP03t response data is equal to #R\_PROF\_PKG\_OK | PM\_REQ9, EUICC\_REQ14, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, EUICC\_REQ61 |
| 9 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |
| 10 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 11 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 12 | DS → eUICC-UT | FETCH |  |  |
| 13 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_1F] | PF\_REQ7, PM\_REQ3, EUICC\_REQ13, EUICC\_REQ22 |
| 14 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Error Case: Profile Downloading Interrupted

###### Initial Conditions

The #PROFILE\_PACKAGE SHALL be split in several parts named from

{PROFILE\_PART1} to {PROFILE\_PARTn} in this sequence (n = the last index of the sub part). Each Profile part contains a list of PEs. Note that only the

{PROFILE\_PART1} needs to be sent in the following test.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | TLS\_APPLICATION containing the result of  HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1, SCP03T\_SCRIPT(  #SCP03\_KVN,  {PROFILE\_PART1}))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ57, EUICC\_REQ58 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. The HTTP content contains a response data formatted in an expanded remote command structure with indefinite length coding 2. The response to the INITIALIZE UPDATE TLV   command (i.e. TAG ‘84’) SHALL be equal to [R\_SCP03T\_INITUP\_OK]   1. The response to the EXTERNAL AUTHENTICATE TLV   command (i.e. TAG ‘85’) SHALL be equal to [R\_SCP03T\_EXTAUTH\_OK]   1. For each SCP03t TLV command sent (i.e. TAG ‘86’), a response [R\_SCP03T\_EMPTY] is   returned | PM\_REQ9, EUICC\_REQ14, EUICC\_REQ23, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) (the other Profile Elements SHALL NOT be sent) | | | |
| 6 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP1]) |  | EUICC\_REQ22, EUICC\_REQ54 |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 8 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_E3\_ISDP1\_0F] | PF\_REQ7, PM\_REQ3, EUICC\_REQ13, EUICC\_REQ22 |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

## ES8 (SM-DP – eUICC): UpdateConnectivityParameters

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ54

###### Test Cases

###### General Initial Conditions

* + - * + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### Test Environment

**DS**

**SM-SR-S**

ES8-UpdateConnectivityParameters

**eUICC-UT**

**SM-DP-S**

###### TC.ES8.UCP.1: UpdateConnectivityParameters\_SMS

###### Test Purpose

*To ensure ISD-P can update the Connectivity Parameters on an Enabled Profile using SMS.*

###### Referenced Requirements

* + - * + EUICC\_REQ13, EUICC\_REQ17, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31, EUICC\_REQ54

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: Update SMS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
|  |  | ENVELOPE\_SMS\_PP( #SPI\_VALUE, |  | EUICC\_REQ17, EUICC\_REQ22,  EUICC\_REQ54 |
| 2 | DS → eUICC-UT | #DEFAULT\_ISD\_P\_TAR,  SCP03\_SCRIPT( |  |
|  |  | #DEFAULT\_ISD\_P\_SCP03\_KVN, |  |
|  |  | [STORE\_SMS\_PARAM\_MNO])) |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: Update CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [STORE\_CATTP\_PARAM\_MNO])) |  | EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or  ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Nominal Case: Update HTTPS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [STORE\_HTTPS\_PARAM\_MNO])) |  | EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or  ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°4 – Nominal Case: Update SMS and CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [STORE\_SMSCATTP\_PARAM])) |  | EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or  ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°5 – Nominal Case: Update HTTPS and SMS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [STORE\_HTTPSSMS\_PARAM])) |  | EUICC\_REQ17, EUICC\_REQ22, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or  ‘6108’ | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ23, EUICC\_REQ31 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.ES8.UCP.2: UpdateConnectivityParameters\_CAT\_TP

###### Test Purpose

*To ensure ISD-P can update the Connectivity Parameters on a Disabled Profile using CAT\_TP.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ17, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ31, EUICC\_REQ54

###### Initial Conditions

#ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process ES8-EstablishISDPKeySet has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

#ISD\_P\_AID1 in Disabled state

###### Test Sequence N°1 – Nominal Case: Update CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
|  |  | ACK\_DATA containing the result of  SCP80\_PACKET( |  | EUICC\_REQ17, EUICC\_REQ54 |
|  |  | #SPI\_VALUE, |  |
|  |  | #ISD\_P\_TAR1, |  |
| 3 | DS → eUICC-UT | SCP03\_SCRIPT( |  |
|  |  | #SCP03\_KVN, |  |
|  |  | [STORE\_CATTP\_PARAM\_MNO])) |  |
|  |  | Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | eUICC-UT → DS | ACK\_DATA with POR | 1. The ACK\_DATA   contains a response packet   1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. For each R-APDU received:   a. SW=’9000’ or ‘6108’ | EUICC\_REQ13, EUICC\_REQ18, EUICC\_REQ23, EUICC\_REQ31 |
| 5 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### TC.ES8.UCP.3: UpdateConnectivityParameters\_HTTPS

###### Test Purpose

*To ensure ISD-P can update the Connectivity Parameters on a Disabled Profile using HTTPS.*

###### Referenced Requirements

EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ17, EUICC\_REQ23, EUICC\_REQ31, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ54

###### Initial Conditions

#ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process ES8-EstablishISDPKeySet has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

#ISD\_P\_AID1 in Disabled state

###### Test Sequence N°1 – Nominal Case: Update HTTPS Parameters

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | DS → eUICC-UT | HTTPS\_CONTENT\_ISDP( #ISD\_P\_AID1, SCP03\_SCRIPT(  #SCP03\_KVN, [STORE\_HTTPS\_PARAM\_MNO]))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  | EUICC\_REQ17, EUICC\_REQ49, EUICC\_REQ51, EUICC\_REQ52, EUICC\_REQ54 |
| 4 | eUICC-UT → DS | TLS\_APPLICATION with POR | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to   #POST\_URI   1. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL #X\_ADMIN\_FROM\_ISD\_R #CONTENT\_TYPE #TRANSFER\_ENCODING #X\_ADMIN\_STATUS\_OK   1. For each R-APDU received:   a. SW=’9000’ or ‘6108’ | EUICC\_REQ14, EUICC\_REQ16, EUICC\_REQ23, EUICC\_REQ31, EUICC\_REQ43, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ48, EUICC\_REQ52 |
| 5 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

## Off-card Interfaces

## ES1 (EUM – SM-SR): RegisterEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ14
* EUICC\_REQ32
* PM\_REQ14

###### Test Cases

###### General Initial Conditions

* #EUM\_S\_ID and #EUM\_S\_ACCESSPOINT well known to the SM-SR-UT
* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #EUM\_S\_PK\_ECDSA well known to the SM-SR-UT

###### Test Environment

ES1-RegisterEIS

ES3-GetEIS

**SM-DP-S**

**SM-SR-UT**

**EUM-S**

###### TC.ES1.REIS.1: RegisterEIS

###### Test Purpose

*To ensure EIS registration is well implemented on SM-SR. The aim is to ask the SM-SR to add a new EIS in its database and check that the new eUICC information set can be returned at any moment by the SM-SR. Some error cases are also described:*

* *the EIS is already registered within the EIS database of the SM-SR*
* *the EIS signature is invalid*
* *the EIS data is invalid because the free memory is bigger than full memory*

###### Referenced Requirements

* PROC\_REQ14
* EUICC\_REQ32
* PM\_REQ14

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS
* The variable {SM\_DP\_ID\_RPS} in the ProfileInfo:
* SHALL be set to #SM\_DP\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | EUM-S → SM-SR-UT | SEND\_REQ(  ES1-RegisterEIS, #EIS\_ES1\_RPS) |  |  |
| 2 | SM-SR-UT → EUM-S | Send the  ES1-RegisterEIS  response | The Status is equal to  #SUCCESS | PROC\_REQ14, EUICC\_REQ32 |
| 3 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3- GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 4 | SM-SR-UT → SM-DP-S | Send the  ES3- GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned is equal to   #EIS\_ES3\_RPS | EUICC\_REQ32, PM\_REQ14 |

###### Test Sequence N°2 – Error Case: Already Registered

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is already provisioned on the SM-SR- UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | EUM-S → SM-SR-UT | SEND\_REQ(  ES1-RegisterEIS, #EIS\_ES1\_RPS) |  |  |
| 2 | SM-SR-UT → EUM-S | Send the  ES1-RegisterEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_OBJ\_EXIST | PROC\_REQ14, EUICC\_REQ32 |

###### Test Sequence N°3 – Error Case: Invalid Signature

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | EUM-S → SM-SR-UT | SEND\_REQ(  ES1-RegisterEIS, #EIS\_BADEUMSIGN\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → EUM-S | Send the  ES1-RegisterEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EIS 2. The Reason code is equal to #RC\_VERIFICATION\_FAIL ED | PROC\_REQ14, EUICC\_REQ32 |

###### Test Sequence N°4 – Error Case: Invalid Data

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | EUM-S → SM-SR-UT | SEND\_REQ(  ES1-RegisterEIS, #INVALID\_EIS\_RPS) |  |  |
| 2 | SM-SR-UT → EUM-S | Send the  ES1-RegisterEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EIS 2. The Reason code is equal to #RC\_INVALID | PROC\_REQ14, EUICC\_REQ32 |

## ES2 (MNO – SM-DP): GetEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ10, PM\_REQ14

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### Test Environment

ES2-GetEIS

ES2-GetEIS

**SM-DP-UT**

**MNO1-S**

ES3-GetEIS

###### TC.ES2.GEIS.1: GetEIS

**SM-SR-S**

###### Test Purpose

*To ensure EIS can be retrieved by the SM-DP through the SM-SR when a MNO requests it. Some error cases are also defined:*

* *the SM-SR is unknown*
* *the EID is unknown to the SM-SR*

###### Referenced Requirements

* PM\_REQ10, PM\_RE14

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
* The variable {SM\_DP\_ID\_RPS} SHALL be set to #SM\_DP\_UT\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-GetEIS, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-GetEIS  request | The EID parameter is equal to  #VIRTUAL\_EID\_RPS | PM\_REQ10, PM\_REQ14 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-GetEIS, #EIS\_ES3\_RPS) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned is equal to   #EIS\_ES2\_RPS | PM\_REQ10 |

###### Test Sequence N°2 – Error Case: Unknown SM-SR

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-GetEIS, #VIRTUAL\_EID\_RPS, #UNKNOWN\_SM\_SR\_ID) |  |  |
| 2 | SM-DP-UT → MNO1-S | Send the  ES2- GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR 2. The Reason code is equal to   #RC\_UNKNOWN | PM\_REQ10 |

###### Test Sequence N°3 – Error Case: Unknown eUICC

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-GetEIS, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the ES3-GetEIS  request | The EID parameter is equal to  #VIRTUAL\_EID\_RPS | PM\_REQ10, PM\_REQ14 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP( ES3-GetEIS, #FAILED, #SC\_EID,  #RC\_ID\_UNKNOWN) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → MNO1-S | Send the ES2-GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to   #RC\_ID\_UNKNOWN | PM\_REQ10 |

## ES2 (MNO – SM-DP): DownloadProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ1, PROC\_REQ2, PROC\_REQ4
* PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17
* PF\_REQ20

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT
* #EUM\_S\_PK\_ECDSA well known to the SM-DP-UT

###### Test Environment

ES2-DownloadProfile

ES3-GetEIS

ES3-CreateISDP

ES3-SendData

ES3-DeleteISDP

ES2-DownloadProfile

**SM-SR-S**

**SM-DP-UT**

**MNO1-S**

###### TC.ES2.DP.1: DownloadProfile

###### Test Purpose

*To ensure Profile download process is well implemented on SM-DP. The aim of the test cases defined below is to make sure that all ES3 methods are correctly sent. Only error cases are defined:*

* *the keys establishment fails*
* *the ISD-P creation fails*
* *a conditional parameter is missing (neither ProfileType nor ICCID are present in the request)*

###### Referenced Requirements

* PROC\_REQ1, PROC\_REQ2, PROC\_REQ4
* PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17
* PF\_REQ20

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
* The variable {SM\_DP\_ID\_RPS} SHALL be set to #SM\_DP\_UT\_ID\_RPS

###### Test Sequence N°1 – Error Case: Keys Establishment Fails

###### Initial Conditions

The Profile Type #PF\_PROFILE\_TYPE\_TO\_DOWNLOAD\_RPS is well known to the SM-DP- UT

An associated Profile, as the #PROFILE\_PACKAGE, is set on the SM-DP-UT

The Profile to download SHALL be compatible with the #EIS\_ES3\_RPS (i.e. enough memory, the Profile to download is compatible with the eUICC…)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS}, #PF\_PROFILE\_TYPE\_TO\_DOWNLOAD\_RPS,  #EP\_FALSE\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-GetEIS  request | The EID parameter is equal to  #VIRTUAL\_EID\_RPS | PROC\_REQ1, PM\_REQ11, PM\_REQ14 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-GetEIS, #EIS\_ES3\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → SM-SR-S | Send the  ES3-CreateISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID parameter is equal to #PF\_ICCID\_TO\_DOWNLOAD\_RPS 3. The MNO-ID parameter is equal to #MNO1\_S\_ID 4. The REQUIRED-MEMORY   parameter is present and lower than 750000   1. The MORE-TO-DO   parameter MAY be present. If present, it SHALL be equal to #MORE\_TODO\_RPS or #NO\_MORE\_TODO\_RPS | PROC\_REQ1, PM\_REQ11, PM\_REQ16 |
| 5 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-CreateISDP, #ISD\_P\_AID1) |  |  |
| 6 | SM-DP-UT → SM-SR-S | Send the  ES3-SendData  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The SD- AID parameter is equal to #ISD\_R\_AID 3. The DATA parameter is present. It SHALL contain APDUs related to the ES8.EstablishISDPKeyset function (i.e. STORE DATA) 4. The MORE-TO-DO   parameter MAY be present. If present, it SHALL be equal to #MORE\_TODO\_RPS or #NO\_MORE\_TODO\_RPS | PROC\_REQ2, PM\_REQ11, PM\_REQ17 |
| 7 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-SendData, #FAILED, #SC\_ISDP,  #RC\_EXECUTION\_ERROR,  #EUICC\_RESP1\_RPS) |  |  |
| 8 | SM-DP-UT → SM-SR-S | Send the  ES3-DeleteISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID parameter is equal to #PF\_ICCID\_TO\_DOWNLOAD\_RPS | PROC\_REQ4, PM\_REQ11, PF\_REQ20 |
| 9 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-DeleteISDP) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #FAILED   1. The euiccResponseData is equal to #EUICC\_RESP1\_RPS | PROC\_REQ4, PM\_REQ11 |

###### Test Sequence N°2 – Error Case: ISDP Creation Fails

###### Initial Conditions

The Profile #PF\_ICCID\_TO\_DOWNLOAD is well known to the SM-DP-UT

An associated Profile, as the #PROFILE\_PACKAGE is set on the SM-DP-UT

The Profile to download SHALL be compatible with the #EIS\_ES3\_RPS (i.e. enough memory, the Profile to download is compatible with the eUICC…)

The SM-SR-S is configured to send an error when receiving the ES3 call

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS}, #PF\_ICCID\_TO\_DOWNLOAD\_RPS,  #EP\_FALSE\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-GetEIS  request | The EID parameter is equal to  #VIRTUAL\_EID\_RPS | PROC\_REQ1, PM\_REQ11, PM\_REQ14 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-GetEIS, #EIS\_ES3\_RPS) |  |  |
| 4 | SM-DP-UT → SM-SR-S | Send the  ES3-CreateISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID parameter is equal to #PF\_ICCID\_TO\_DOWNLOAD\_RPS 3. The MNO-ID parameter is equal to #MNO1\_S\_ID 4. The MORE-TO-DO   parameter MAY be present. If present, it SHALL be equal to #MORE\_TODO\_RPS or #NO\_MORE\_TODO\_RPS | PROC\_REQ1, PM\_REQ11, PM\_REQ16 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-CreateISDP, #FAILED, #SC\_EUICC, #RC\_MEMORY) |  |  |
| 6 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1- The Status is equal to  #FAILED | PM\_REQ11 |

###### Test Sequence N°3 – Error Case: Conditional Parameters Missing

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #EP\_FALSE\_RPS) |  |  |
| 2 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUNCTION 2. The Reason code is equal to #RC\_COND\_PARAM | PM\_REQ11 |

## ES2 (MNO – SM-DP): UpdatePolicyRules

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ16
* PM\_REQ12, PM\_REQ19

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### Test Environment

ES3-UpdatePolicyRules

ES2-UpdatePolicyRules

ES2-UpdatePolicyRules

**SM-DP-UT**

**MNO1-S**

**SM-SR-S**

###### TC.ES2.UPR.1: UpdatePolicyRules

###### Test Purpose

*To ensure POL2 can be updated by the SM-DP through the SM-SR when a MNO requests it. An error case is also defined:*

* *the Profile identified by the ICCID is unknown*

###### Referenced Requirements

* PROC\_REQ16
* PM\_REQ12, PM\_REQ19

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case: No Rule

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-UpdatePolicyRules, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  {SM\_SR\_ID\_RPS},  #POL2\_EMPTY\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-UpdatePolicyRules  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS   1. Check that POL2 parameter is equal to #POL2\_EMPTY\_RPS | PM\_REQ12, PM\_REQ19, PROC\_REQ16 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-UpdatePolicyRules) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-UpdatePolicyRules  response | The Status is equal to  #SUCCESS | PM\_REQ12, PROC\_REQ16 |

###### Test Sequence N°2 – Nominal Case: Rule “Disabling not allowed”

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-UpdatePolicyRules, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  {SM\_SR\_ID\_RPS},  #POL2\_DIS\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-UpdatePolicyRules  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS   1. The POL2 is equal to   #POL2\_DIS\_RPS | PM\_REQ12, PM\_REQ19, PROC\_REQ16 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-UpdatePolicyRules) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-UpdatePolicyRules  response | The Status is equal to  #SUCCESS | PM\_REQ12, PROC\_REQ16 |

###### Test Sequence N°3 – Error Case: Unknown Profile ICCID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-UpdatePolicyRules, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  {SM\_SR\_ID\_RPS},  #POL2\_DEL\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-UpdatePolicyRules  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS   1. The POL2 is equal to   #POL2\_DEL\_RPS | PM\_REQ12, PM\_REQ19, PROC\_REQ16 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-UpdatePolicyRules, #FAILED, #SC\_PROFILE\_ICCID,  #RC\_UNKNOWN) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-UpdatePolicyRules  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_UNKNOWN | PM\_REQ12, PROC\_REQ16 |

## ES2 (MNO – SM-DP): UpdateSubscriptionAddress

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ13, PM\_REQ20

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### Test Environment

ES3-UpdateSubscriptionAddress

ES2-UpdateSubscriptionAddress

ES2-UpdateSubscriptionAddress

**SM-DP-UT**

**MNO1-S**

**SM-SR-S**

###### TC.ES2.USA.1: UpdateSubscriptionAddress

###### Test Purpose

*To ensure Subscription Address can be updated by the SM-DP through the SM-SR when a MNO requests it.*

###### Referenced Requirements

* PM\_REQ13, PM\_REQ20

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
|  |  | SEND\_REQ( |  |  |
|  |  | ES2-UpdateSubscriptionAddress, |
| 1 | MNO1-S  → SM-DP-UT | #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS, |
|  |  | #NEW\_ADDR\_RPS, |
|  |  | {SM\_SR\_ID\_RPS}) |
|  |  |  | 1- The EID parameter is equal to | PM\_REQ13, PM\_REQ20 |
| 2 | SM-DP-UT  → SM-SR-S | Send the  ES3-UpdateSubscriptionAddress  request | #VIRTUAL\_EID\_RPS   1. The ICCID is equal to   #ICCID1\_RPS   1. The Subscription |  |
|  |  |  | Address is equal to |  |
|  |  |  | #NEW\_ADDR\_RPS |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S  → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-UpdateSubscriptionAddress) |  |  |
| 4 | SM-DP-UT  → MNO1-S | Send the  ES2-UpdateSubscriptionAddress  response | The Status is equal to  #SUCCESS | PM\_REQ13 |

## ES2 (MNO – SM-DP): EnableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ7
* PF\_REQ12, PF\_REQ15, PF\_REQ17, PF\_REQ18, PF\_REQ21, PF\_REQ23

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID, #MNO1\_S\_ACCESSPOINT, #MNO2\_S\_ID and

#MNO2\_S\_ACCESSPOINT well known to the SM-DP-UT

* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### TC.ES2.EP.1: EnableProfile

###### Test Purpose

*To ensure a Profile can be Enabled by the SM-DP through the SM-SR when a MNO requests it. After enabling the Profile, the SM-SR sends the notification HandleProfileDisabledNotification to the SM-DP: this notification SHALL be forwarded to the corresponding MNO.*

*Some error cases are also defined:*

* *the Profile identified by the ICCID is known to the SM-SR but installed on another eUICC than the one identified by the SM-DP*
* *the SM-DP is not allowed to perform this function on the target Profile*

###### Test Environment

**MNO1-S**

**SM-DP-UT**

ES2-EnableProfile

**SM-SR-S**

ES3-EnableProfile

ES2-EnableProfile

*ES3-HandleProfileDisabledNotification*

**MNO2-S**

*ES2-HandleProfileDisabledNotification*

###### Referenced Requirements

* PROC\_REQ7
* PF\_REQ12, PF\_REQ15, PF\_REQ18, PF\_REQ21

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-EnableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ7, PF\_REQ12, PF\_REQ18 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-EnableProfile) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | The Status is equal to  #SUCCESS | PROC\_REQ7, PF\_REQ12 |
| 5 | SM-SR-S → SM-DP-UT | SEND\_NOTIF(  ES3-HandleProfile DisabledNotification,  #VIRTUAL\_EID\_RPS, #ICCID2\_RPS #MNO2\_ID\_RPS,  #TIMESTAMP\_RPS) |  |  |
| 6 | SM-DP-UT → MNO2-S | Send the  ES2-HandleProfile DisabledNotification notification | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID2\_RPS   1. The completion timestamp is equal to #TIMESTAMP\_RPS | PROC\_REQ7, PF\_REQ15, PF\_REQ21 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-EnableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ7, PF\_REQ12, PF\_REQ18 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-EnableProfile, #FAILED, #SC\_PROFILE\_ICCID, #RC\_INVALID\_DEST) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_INVALID\_DEST | PROC\_REQ7, PF\_REQ12 |

###### Test Sequence N°3 – Error Case: Not Allowed

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-EnableProfile  request | 1. The EID parameter is equal to   #VIRTUAL\_EID\_RPS   1. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ7, PF\_REQ12, PF\_REQ18 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-EnableProfile, #FAILED, #SC\_PROFILE\_ICCID, #RC\_NOT\_ALLOWED) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_NOT\_ALLOWED | PROC\_REQ7, PF\_REQ12 |

###### TC.ES2.EP.2: EnableProfileWithDeletion

###### Test Purpose

*To ensure MNO can ask the SM-DP to enable a Profile. The notification HandleProfileDeletedNotification is tested considering that the deletion has been triggered by the evaluation of POL1 on SM-SR side.*

###### Test Environment

**MNO1-S**

**SM-DP-UT**

ES2-EnableProfile

**SM-SR-S**

ES3-EnableProfile

ES2-EnableProfile

*ES3-HandleProfileDeletedNotification*

**MNO2-S**

*ES2-HandleProfileDeletedNotification*

###### Referenced Requirements

PROC\_REQ7

PF\_REQ12, PF\_REQ17, PF\_REQ18, PF\_REQ23

###### Initial Conditions

The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-EnableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ7, PF\_REQ12, PF\_REQ18 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-EnableProfile) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | The Status is equal to  #SUCCESS | PROC\_REQ7, PF\_REQ12 |
| 5 | SM-SR-S → SM-DP-UT | SEND\_NOTIF(  ES3-HandleProfile DeletedNotification, #VIRTUAL\_EID\_RPS,  #ICCID2\_RPS  #MNO2\_ID\_RPS, #TIMESTAMP\_RPS) |  |  |
| 6 | SM-DP-UT → MNO2-S | Send the  ES2-HandleProfile DeletedNotification notification | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID2\_RPS   1. The completion timestamp is equal to #TIMESTAMP\_RPS | PROC\_REQ7, PF\_REQ17, PF\_REQ23 |

## ES2 (MNO – SM-DP): DisableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ10
* PF\_REQ13, PF\_REQ16, PF\_REQ19, PF\_REQ22

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID, #MNO1\_S\_ACCESSPOINT, #MNO2\_S\_ID and

#MNO2\_S\_ACCESSPOINT well known to the SM-DP-UT

* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### Test Environment

**MNO1-S**

**SM-DP-UT**

ES2-DisableProfile

**SM-SR-S**

ES3-DisableProfile

ES2-DisableProfile

*ES3-HandleProfileEnabledNotification*

**MNO2-S**

*ES2-HandleProfileEnabledNotification*

###### TC.ES2.DISP.1: DisableProfile

###### Test Purpose

*To ensure Profile can be Disabled by the SM-DP through the SM-SR when a MNO requests it. After disabling the Profile, the SM-SR sends the notification HandleProfileEnabledNotification which SHALL be forwarded to the corresponding MNO. Some error cases are also defined:*

* *error during execution of the enabling command on the eUICC*
* *the POL1 of the impacted Profiles does not allow this operation*

###### Referenced Requirements

* PROC\_REQ10
* PF\_REQ13, PF\_REQ16, PF\_REQ19, PF\_REQ22

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DisableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ10, PF\_REQ13, PF\_REQ19 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-DisableProfile) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | The Status is equal to  #SUCCESS | PROC\_REQ10, PF\_REQ13 |
| 5 | SM-SR-S → SM-DP-UT | SEND\_NOTIF(  ES3-HandleProfile EnabledNotification, #VIRTUAL\_EID\_RPS,  #ICCID2\_RPS #MNO2\_ID\_RPS, #TIMESTAMP\_RPS) |  |  |
| 6 | SM-DP-UT → MNO2-S | Send the  ES2-HandleProfile EnabledNotification notification | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID2\_RPS   1. The completion timestamp is equal to #TIMESTAMP\_RPS | PROC\_REQ10, PF\_REQ16, PF\_REQ22 |

###### Test Sequence N°2 – Error Case: Execution Error

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DisableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ10, PF\_REQ13, PF\_REQ19 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DisableProfile, #FAILED,  #SC\_ISDR,  #RC\_EXECUTION\_ERROR, #EUICC\_RESP1\_RPS) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_ISDR 2. The Reason code is equal to #RC\_EXECUTION\_ERROR | PROC\_REQ10, PF\_REQ13 |

###### Test Sequence N°3 – Error Case: Incompatible POL1

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DisableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ10, PF\_REQ13, PF\_REQ19 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DisableProfile, #FAILED,  #SC\_POL1,  #RC\_REFUSED) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_REFUSED | PROC\_REQ10, PF\_REQ13 |

###### Test Sequence N°4 – Nominal Case: POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DisableProfile  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ10, PF\_REQ13, PF\_REQ19 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DisableProfile, #WARNING, #SC\_POL2, #RC\_OBJ\_EXIST) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_OBJ\_EXIST | PROC\_REQ10, PF\_REQ13 |

## ES2 (MNO – SM-DP): DeleteProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ12
* PF\_REQ14, PF\_REQ20

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
* #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT

###### Test Environment

ES3-DeleteISDP

ES2-DeleteProfile

ES2-DeleteProfile

**SM-DP-UT**

**MNO1-S**

**SM-SR-S**

###### TC.ES2.DP.1: DeleteProfile

###### Test Purpose

*To ensure Profile can be deleted by the SM-DP through the SM-SR when a MNO requests it. Some error cases are also defined:*

* *the POL2 of the impacted Profiles does not allow this operation*
* *the target Profile cannot be Disabled (in case of the disabling of the Profile SHALL be performed before the deletion)*

###### Referenced Requirements

* PROC\_REQ12
* PF\_REQ14, PF\_REQ20

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DeleteISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ12, PF\_REQ14, PF\_REQ20 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP(  ES3-DeleteISDP) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | The Status is equal to  #SUCCESS | PROC\_REQ12, PF\_REQ14 |

###### Test Sequence N°2 – Error Case: Incompatible POL2

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DeleteISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ12, PF\_REQ14, PF\_REQ20 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DeleteISDP, #FAILED, #SC\_POL2,  #RC\_REFUSED) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_REFUSED | PROC\_REQ12, PF\_REQ14 |

###### Test Sequence N°3 – Error Case: Automatic Disabling Not Allowed

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DeleteISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ12, PF\_REQ14, PF\_REQ20 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DeleteISDP, #FAILED, #SC\_EUICC, #RC\_REFUSED) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_REFUSED | PROC\_REQ12, PF\_REQ14 |

###### Test Sequence N°4 – Error Case: ISD-P identified by its AID does not exist on the targeted eUICC

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS},  #ICCID1\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-DeleteISDP  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The ICCID is equal to   #ICCID1\_RPS | PROC\_REQ12, PF\_REQ14, PF\_REQ20 |
| 3 | SM-SR-S → SM-DP-UT | SEND\_ERROR\_RESP(  ES3-DeleteISDP, #WARNING, #SC\_ISDP, #RC\_NOT\_PRESENT) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_ISDP 2. The Reason code is equal to #RC\_NOT\_PRESENT | PROC\_REQ12, PF\_REQ14 |

## ES3 (SM-DP – SM-SR): GetEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ14

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-GetEIS

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.GEIS.1: GetEIS

###### Test Purpose

*To ensure EIS can be retrieved by the SM-SR when a SM-DP requests it. An error case is also defined:*

* *the EID is unknown to the SM-SR*

###### Referenced Requirements

* PM\_REQ14

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3- GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned is equal to   #EIS\_ES3\_RPS | PM\_REQ14 |

###### Test Sequence N°2 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3- GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to   #RC\_ID\_UNKNOWN | PM\_REQ14 |

## ES3 (SM-DP – SM-SR): AuditEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ15

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT

###### Test Environment

ES3-AuditEIS

ES3-AuditEIS

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.AEIS.1: AuditEIS

###### Test Purpose

*To ensure the EIS audit can be performed by the SM-SR if the EID is known to the SM-SR.*

###### Referenced Requirements

* PM\_REQ15

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-AuditEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the ES3- AuditEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to   #RC\_UNKNOWN | PM\_REQ15 |

## ES3 (SM-DP – SM-SR): CreateISDP

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ16

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-CreateISDP

ES3-CreateISDP

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.CISDP.1: CreateISDP

###### Test Purpose

*To ensure the ISDP creation is well implemented on SM-SR. Only error cases are defined:*

* *the eUICC has not enough free memory to execute the creation of the new ISD-P with the required amount of memory*
* *the ICCID is already allocated to another Profile*

###### Referenced Requirements

* PM\_REQ16

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Not Enough Memory

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is not present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-CreateISDP, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS, #MNO1\_ID\_RPS, #BIG\_MEM\_RPS,  #MORE\_TODO\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-CreateISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_MEMORY | PM\_REQ16 |

###### Test Sequence N°2 – Error Case: Already In Use

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Enabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-CreateISDP, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS, #MNO1\_ID\_RPS, #SMALL\_MEM\_RPS, #NO\_MORE\_TODO\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-CreateISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_ALREADY\_USED | PM\_REQ16 |

## ES3 (SM-DP – SM-SR): SendData

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ17

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-SendData

ES3-SendData

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.SDATA.1: SendData

###### Test Purpose

*To ensure the SendData method can be used by the SM-DP except if:*

* *the ISD-P is unknown to the SM-SR or*
* *the ISD-P is known to the SM-SR but installed on another eUICC than the one identified by the SM-DP*

###### Referenced Requirements

* PM\_REQ17

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown ISD-P

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is not present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-SendData, #VIRTUAL\_EID\_RPS, #SD\_ISDP2\_RPS, #DATA\_RPS,  #MORE\_TODO\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-SendData  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SD\_AID 2. The Reason code is equal to #RC\_UNKNOWN | PM\_REQ17 |

###### VOID

## ES3 (SM-DP – SM-SR): UpdatePolicyRules

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PROC\_REQ16
* PM\_REQ19, PM\_REQ22

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES3-UpdatePolicyRules

ES4-GetEIS

**MNO1-S**

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.UPR.1: UpdatePolicyRules

###### Test Purpose

*To ensure the SM-SR can update the Policy Rules (POL2) according the parameters sent by the SM-DP. To make sure that the POL2 have been set on SM-SR side, the EIS is retrieved just after updating the rules.*

###### Referenced Requirements

* PROC\_REQ16
* PM\_REQ19, PM\_REQ22

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-UpdatePolicyRules, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS, #POL2\_DIS\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-UpdatePolicyRules  response | The Status is equal to  #SUCCESS | PM\_REQ19, PROC\_REQ16 |
| 3 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4- GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned is equal to #EIS\_ES4\_RPS except that POL2 of #ICCID1 is equal to #POL2\_DIS\_RPS | PM\_REQ19, PM\_REQ22, PROC\_REQ16 |

## ES3 (SM-DP – SM-SR): UpdateSubscriptionAddress

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ20, PM\_REQ22

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES3-UpdateSubscriptionAddress

ES4-GetEIS

**MNO1-S**

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.USA.1: UpdateSubscriptionAddress

###### Test Purpose

*To ensure Subscription Address can be updated by the SM-SR when a SM-DP requests it. To make sure that the Subscription Address has been set on SM-SR side, the EIS is retrieved just after updating the address.*

###### Referenced Requirements

* PM\_REQ20, PM\_REQ22

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-UpdateSubscriptionAddress, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  #NEW\_ADDR\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-UpdateSubscriptionAddress  response | The Status is equal to  #SUCCESS | PM\_REQ20 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4- GetEIS  response | 1. The Status is equal to #SUCCESS 2. The EIS returned is equal to #EIS\_ES4\_RPS except that the Subscription Address of #ICCID1 is equal to #SUB\_ADDR3\_RPS | PM\_REQ20, PM\_REQ22 |

## ES3 (SM-DP – SM-SR): UpdateConnectivityParameters

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ21

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-UpdateConnectivityParameters

ES3-UpdateConnectivityParameters

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.UCP.1: UpdateConnectivityParameters

###### Test Purpose

*To ensure the UpdateConnectivityParameters method can be performed by the SM-SR except if:*

* *the EID is unknown to the SM-SR or*
* *the Profile identified by the ICCID is unknown*

###### Referenced Requirements

* PM\_REQ21

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-UpdateConnectivityParameters, #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS,  #CON\_PARAM\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-UpdateConnectivityParameters  response | 1. The Status is equal to #FAILED 2. The Subject code   is equal to #SC\_EID   1. The Reason code is equal to #RC\_UNKNOWN | PM\_REQ21 |

###### Test Sequence N°2 – Error Case: Unknown Profile ICCID

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is not present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-UpdateConnectivityParameters, #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS,  #CON\_PARAM\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-UpdateConnectivityParameters  response | 1. The Status is equal to #FAILED 2. The Subject code is equal to #SC\_PROFILE\_ICCID 3. The Reason code is equal to #RC\_UNKNOWN | PM\_REQ21 |

## ES3 (SM-DP – SM-SR): EnableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ18

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-EnableProfile

ES3-EnableProfile

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.EP.1: EnableProfile

###### Test Purpose

*To ensure a Profile can be Enabled by the SM-SR, when an SM-DP requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the Profile identified by its ICCID is in Disabled state*
* *the POL2 of the target Profile and the POL2 of the currently Enabled Profile allows the enabling*

###### Referenced Requirements

* PF\_REQ18

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to   #SC\_EID   1. The Reason code is equal to   #RC\_UNKNOWN | PF\_REQ18 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to   #SC\_PROFILE\_ICCID   1. The Reason code is equal to   #RC\_INVALID\_DEST | PF\_REQ18 |

###### Test Sequence N°3 – Error Case: Already Enabled Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Enabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_NOT\_ALLOWED | PF\_REQ18 |

###### Test Sequence N°4 – Error Case: Incompatible Enabled Profile POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID2 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Enabled state

The POL2 of the Profile identified by the #ICCID2 is “Disabling of this Profile not allowed”

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to   #RC\_REFUSED | PF\_REQ18 |

## ES3 (SM-DP – SM-SR): DisableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ19

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-DisableProfile

ES3-DisableProfile

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.DISP.1: DisableProfile

###### Test Purpose

*To ensure a Profile can be Disabled by the SM-SR, when an SM-DP requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the Profile identified by its ICCID is in Enabled state*
* *the POL2 of the target Profile allows the disabling*

###### Referenced Requirements

* PF\_REQ19

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to   #SC\_EID   1. The Reason code is equal to   #RC\_UNKNOWN | PF\_REQ19 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to   #RC\_INVALID\_DEST | PF\_REQ19 |

###### Test Sequence N°3 – Error Case: Already Disabled Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to   #RC\_NOT\_ALLOWED | PF\_REQ19 |

###### Test Sequence N°4 – Error Case: Incompatible POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The POL2 of the Profile identified by the #ICCID1 is “Disabling of this Profile not allowed”

The Profile identified by the #ICCID1 is in Enabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to   #RC\_REFUSED | PF\_REQ19 |

## ES3 (SM-DP – SM-SR): DeleteISDP

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ20

###### Test Cases

###### General Initial Conditions

* #SM\_DP\_S\_ID and #SM\_DP\_S\_ACCESSPOINT well known to the SM-SR-UT
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES3-DeleteISDP

ES3-DeleteISDP

**SM-SR-UT**

**SM-DP-S**

###### TC.ES3.DISDP.1: DeleteISDP

###### Test Purpose

*To ensure a Profile can be deleted by the SM-SR, when an SM-DP requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the SM-DP is authorized to delete the target Profile by the MNO owning the target Profile*
* *the POL2 of the target Profile allows the deletion*
* *the target Profile is not the Profile having the Fall-back Attribute*

###### Referenced Requirements

* PF\_REQ20

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DeleteISDP, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DeleteISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to   #RC\_UNKNOWN | PF\_REQ20 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DeleteISDP,  #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DeleteISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to   #SC\_PROFILE\_ICCID   1. The Reason code is equal to   #RC\_INVALID\_DEST | PF\_REQ20 |

###### Test Sequence N°3 – Error Case: Incompatible POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The POL2 of the Profile identified by the #ICCID1 is “Deletion of this Profile not allowed”

The Profile identified by the #ICCID1 is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DeleteISDP,  #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DeleteISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_REFUSED | PF\_REQ20 |

###### Test Sequence N°5 – Error Case: Fall-back Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID

The Profile identified by the #ICCID1 has the Fall-back Attribute

The Profile identified by the #ICCID1 is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ(  ES3-DeleteISDP, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → SM-DP-S | Send the  ES3-DeleteISDP  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to   #RC\_REFUSED | PF\_REQ20 |

## ES4 (MNO – SM-SR): GetEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ22

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-GetEIS

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.GEIS.1: GetEIS

###### Test Purpose

*To ensure EIS can be retrieved by the SM-SR when a MNO requests it.*

###### Referenced Requirements

* PM\_REQ22

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4- GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned is equal to   #EIS\_ES4\_RPS | PM\_REQ22 |

###### Test Sequence N°2 – Error Case: Not Allowed to Manage the EIS

|  |  |
| --- | --- |
|  | This test case is defined as FFS pending further clarification in the GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7). |

## ES4 (MNO – SM-SR): UpdatePolicyRules

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ22, PM\_REQ23

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-UpdatePolicyRules

ES4-GetEIS

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.UPR.1: UpdatePolicyRules

###### Test Purpose

*To ensure the SM-SR can update the Policy Rules (POL2) according the parameters sent by the MNO. To make sure that the POL2 have been set on SM-SR side, the EIS is retrieved just after updating the rules.*

###### Referenced Requirements

* PM\_REQ22, PM\_REQ23

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-UpdatePolicyRules, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  #POL2\_DIS\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-UpdatePolicyRules  response | The Status is equal to  #SUCCESS | PM\_REQ23 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4- GetEIS  response | 1. The Status is equal to #SUCCESS 2. The EIS returned is equal to #EIS\_ES4\_RPS except that POL2 of #ICCID1 is equal to #POL2\_DIS\_RPS | PM\_REQ22, PM\_REQ23 |

## ES4 (MNO – SM-SR): UpdateSubscriptionAddress

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ22, PM\_REQ24

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-UpdateSubscriptionAddress

ES4-GetEIS

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.USA.1: UpdateSubscriptionAddress

###### Test Purpose

*To ensure Subscription Address can be updated by the SM-SR when a MNO requests it. To make sure that the Subscription Address has been set on SM-SR side, the EIS is retrieved just after updating the address. An error case is also defined:*

* *the MNO is not allowed to manage the Subscription Address*

###### Referenced Requirements

* PM\_REQ22, PM\_REQ24

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS (i.e. the Profile identified by #ICCID1 is present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-UpdateSubscriptionAddress, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  #NEW\_ADDR\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-UpdateSubscriptionAddress  response | The Status is equal to  #SUCCESS | PM\_REQ24 |
| 3 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #VIRTUAL\_EID\_RPS) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4- GetEIS  response | 1. The Status is equal to #SUCCESS 2. The EIS returned is equal to #EIS\_ES4\_RPS except that the Subscription Address of #ICCID1 is equal to #SUB\_ADDR3\_RPS | PM\_REQ22, PM\_REQ24 |

###### Test Sequence N°2 – Error Case: Not Allowed

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by #VIRTUAL\_EID and is not owned by MNO1-S (i.e. the MNO-ID is not equal to #MNO1\_S\_ID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-UpdateSubscriptionAddress, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS,  #NEW\_ADDR\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-UpdateSubscriptionAddress  response | 1. The Status is equal to #FAILED 2. The Subject code is equal to #SC\_SUB\_ADDR 3. The Reason code is equal to #RC\_NOT\_ALLOWED | PM\_REQ24 |

## ES4 (MNO – SM-SR): AuditEIS

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PM\_REQ25

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-AuditEIS

ES4-AuditEIS

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.AEIS.1: AuditEIS

###### Test Purpose

*To ensure the EIS audit can be performed by the SM-SR when MNO requests it, except if:*

* *the Profile identified by the ICCID in the list does not belong to the MNO*

###### Referenced Requirements

* PM\_REQ25

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Profile does not Belong to MNO

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by #VIRTUAL\_EID and is not owned by MNO1-S (i.e. the MNO-ID is not equal to #MNO1\_S\_ID)

The Profile identified by the #ICCID1 is Enabled

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS,  #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE 2. The Reason code is equal to   #RC\_NOT\_ALLOWED | PM\_REQ25 |

## ES4 (MNO – SM-SR): EnableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ24

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-EnableProfile

ES4-EnableProfile

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.EP.1: EnableProfile

###### Test Purpose

*To ensure a Profile can be Enabled by the SM-SR, when an MNO requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the Profile identified by its ICCID is in Disabled state*
* *the POL2 of the target Profile and the POL2 of the currently Enabled Profile allows the enabling*
* *the target Profile is owned by the requesting MNO*

###### Referenced Requirements

* PF\_REQ24

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to   #RC\_UNKNOWN | PF\_REQ24 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_INVALID\_DEST | PF\_REQ24 |

###### Test Sequence N°3 – Error Case: Already Enabled Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Enabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_NOT\_ALLOWED | PF\_REQ24 |

###### Test Sequence N°4 – Error Case: Incompatible Enabled Profile POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID2 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Enabled state

The POL2 of the Profile identified by the #ICCID2 is “Disabling of this Profile not allowed”

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-enableProfile, #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_REFUSED | PF\_REQ24 |

###### Test Sequence N°5 – Error Case: Bad Profile Owner

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by #VIRTUAL\_EID and is not owned by MNO1-S (i.e. the MNO-ID is not equal to #MNO1\_S\_ID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to   #SC\_PROFILE\_ICCID   1. The Reason code is equal to   #RC\_NOT\_ALLOWED | PF\_REQ24 |

## ES4 (MNO – SM-SR): DisableProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ25

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-DisableProfile

ES4-DisableProfile

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.DISP.1: DisableProfile

###### Test Purpose

*To ensure a Profile can be Disabled by the SM-SR, when an MNO requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the Profile identified by its ICCID is in Enabled state*
* *the POL2 of the target Profile allows the disabling*
* *the target Profile is owned by the requesting MNO*

###### Referenced Requirements

* PF\_REQ25

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_UNKNOWN | PF\_REQ25 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_INVALID\_DEST | PF\_REQ25 |

###### Test Sequence N°3 – Error Case: Already Disabled Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID and is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_NOT\_ALLOWED | PF\_REQ25 |

###### Test Sequence N°4 – Error Case: Incompatible POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The POL2 of the Profile identified by the #ICCID1 is “Disabling of this Profile not allowed”

The Profile identified by the #ICCID1 is in Enabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to   #RC\_REFUSED | PF\_REQ25 |

###### Test Sequence N°6 – Error Case: Bad Profile Owner

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by #VIRTUAL\_EID and is not owned by MNO1-S (i.e. the MNO-ID is not equal to #MNO1\_S\_ID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to   #RC\_NOT\_ALLOWED | PF\_REQ25 |

## ES4 (MNO – SM-SR): DeleteProfile

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* PF\_REQ26

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO1-S and the SM-SR-UT

###### Test Environment

ES4-DeleteProfile

ES4-DeleteProfile

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.DP.1: DeleteProfile

###### Test Purpose

*To ensure a Profile can be Disabled by the SM-SR, when an MNO requests it, only if:*

* *the SM-SR is responsible for the management of the targeted eUICC*
* *the Profile identified by its ICCID is loaded on the targeted eUICC*
* *the POL2 of the target Profile allows the deletion*
* *the target Profile is not the Profile having the Fall-back Attribute*
* *the target Profile is owned by the requesting MNO*

###### Referenced Requirements

* PF\_REQ26

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Unknown eUICC

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_UNKNOWN | PF\_REQ26 |

###### Test Sequence N°2 – Error Case: Invalid Destination

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS2\_ES1\_RPS (i.e. the ISD-P identified by #ISDP3\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

The eUICC identified by the #VIRTUAL\_EID2 is provisioned on the SM-SR-UT with the #EIS3\_ES1\_RPS (i.e. the ISD-P identified by #ISDP2\_RPS is only present)

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_INVALID\_DEST | PF\_REQ26 |

###### Test Sequence N°3 – Error Case: Incompatible POL2

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The POL2 of the Profile identified by the #ICCID1 is “Deletion of this Profile not allowed”

The Profile identified by the #ICCID1 is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to   #RC\_REFUSED | PF\_REQ26 |

###### Test Sequence N°4 – Error Case: Bad Profile Owner

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by #VIRTUAL\_EID and is not owned by MNO1-S (i.e. the MNO-ID is not equal to #MNO1\_S\_ID)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #VIRTUAL\_EID\_RPS,  #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to   #RC\_NOT\_ALLOWED | PF\_REQ26 |

###### Test Sequence N°5 – Error Case: Fall-back Profile

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT (e.g. using #EIS\_ES1\_RPS)

The Profile identified by the #ICCID1 is installed on the eUICC identified by

#VIRTUAL\_EID

The Profile identified by the #ICCID1 has the Fall-back Attribute

The Profile identified by the #ICCID1 is in Disabled state

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #VIRTUAL\_EID\_RPS, #ICCID1\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_PROFILE\_ICCID 2. The Reason code is equal to #RC\_REFUSED | PF\_REQ26 |

## ES4 (MNO – SM-SR): PrepareSMSRChange

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* EUICC\_REQ35

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT

###### Test Environment

ES4-PrepareSMSRChange

**SM-SR-UT**

**MNO1-S**

###### TC.ES4.PSMSRC.1: PrepareSMSRChange

###### Test Purpose

*To ensure the method PrepareSMSRChange is well implemented on the SM-SR.*

*An error case is also defined:*

* *the SM-SR is not capable of managing the eUICC identified by this EID*

###### Referenced Requirements

* EUICC\_REQ35

###### Initial Conditions

* None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS,  #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35 |

###### Test Sequence N°2 – Error Case: SM-SR Not Capable of Managing the eUICC

###### Initial Conditions

No setting has been initialized on SM-SR-UT to accept the SM-SR change

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS, #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUN\_PROV 2. The Reason code is equal to #RC\_COND\_USED | EUICC\_REQ35 |

###### Test Sequence N°3 – Error Case: The new SM-SR does not know the current SM-SR

###### Initial Conditions

SM-SR-UT does not know #CUR\_SR\_S\_ID

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS, #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR 2. The Reason code is equal to #RC\_ID\_UNKNOWN | EUICC\_REQ35 |

## ES4 (MNO – SM-SR): SMSRchange

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* EUICC\_REQ36, EUICC\_REQ39

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC

ES4-SMSRChange

**SM-SR-S**

**SM-SR-UT**

**MNO1-S**

Note that the function ES4-PrepareSMSRChange SHALL NOT be performed by the simulators (in the schema above, this is only an informative message).

In the following test cases, the Initiator Role (see GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) section 2.3.1) is assumed to be played by the MNO1-S.

###### TC.ES4.SMSRC.1: SMSRChange

###### Test Purpose

*To ensure the method SMSRChange can be performed by the SM-SR except if:*

* *the ECASD certificate is expired or*
* *the new SM-SR is not capable of managing the eUICC identified by this EID or*
* *the preparation step has not been performed for the eUICC*
* *the targeted SM-SR is unknown*

###### Referenced Requirements

* EUICC\_REQ36, EUICC\_REQ39

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS
* The variable {SM\_DP\_ID\_RPS} SHALL be set to #SM\_DP\_S\_ID\_RPS

###### Test Sequence N°1 – Error Case: Invalid ECASD

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS | EUICC\_REQ36, EUICC\_REQ39 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_ECASD,  #RC\_EXPIRED) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_ECASD 2. The Reason code is equal to #RC\_EXPIRED | EUICC\_REQ36 |

###### Test Sequence N°2 – Error Case: Condition of Use Not Satisfied

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS | EUICC\_REQ36, EUICC\_REQ39 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_FUN\_PROV, #RC\_COND\_USED) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUN\_PROV 2. The Reason code is equal to #RC\_COND\_USED | EUICC\_REQ36 |

###### Test Sequence N°3 – Error Case: Preparation Step Not Performed

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS | EUICC\_REQ36, EUICC\_REQ39 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED,  #SC\_EID,  #RC\_ID\_UNKNOWN) |  |  |
| 4 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_ID\_UNKNOWN | EUICC\_REQ36 |

###### Test Sequence N°4 – Error Case: Unknown Targeted SM-SR

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR- UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS, #TGT\_SR\_S\_UNK\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR 2. The Reason code is equal to #RC\_UNKNOWN | EUICC\_REQ36 |

## ES7 (SM-SR – SM-SR): HandoverEUICC

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* EUICC\_REQ35, EUICC\_REQ39

###### Test Cases

###### General Initial Conditions

* All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)
* #EUM\_S\_PK\_ECDSA well known to the SM-SR-UT

###### Test Environment

**SM-SR-S**

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC

ES7-HandoverEUICC

ES4-SMSRChange

**SM-SR-UT**

**MNO1-S**

Note that the function ES4-SMSRChange SHALL NOT be performed by the simulators (in the schema above, they are only informative messages).

###### TC.ES7.HEUICC.1: HandoverEUICC

###### Test Purpose

*To ensure the method HandoverEUICC is well implemented on the SM-SR. Only error case is defined:*

* *the ECASD certificate is expired*

###### Referenced Requirements

* EUICC\_REQ35, EUICC\_REQ39

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT
* None

###### Test Sequence N°1 – Error Case: Invalid ECASD

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS, #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_REQ(  ES7-HandoverEUICC, #EIS\_EXPIREDCASD\_RPS) |  |  |
| 4 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_ECASD 2. The Reason code is equal to #RC\_EXPIRED | EUICC\_REQ39 |

###### Test Sequence N°2 – Error Case: One MNO owning a profile on this eUICC is unknown by the new SM-SR

###### Initial Conditions

#MNO1\_S\_ID is well known to the SM-SR-UT

#MNO2\_S\_ID is unknown to the SM-SR-UT

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS, #CUR\_SR\_S\_ID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35 |
| 3 | SM-SR-S→ SM-SR-UT | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS |  |
| 4 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EXT\_RES 2. The Reason code is equal to #RC\_ID\_UNKNOWN | EUICC\_REQ39 |

## ES7 (SM-SR – SM-SR): AuthenticateSMSR

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40

###### Test Cases

###### General Initial Conditions

* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT

###### Test Environment

ES4-SMSRChange

ES7-HandoverEUICC

ES7-AuthenticateSMSR

ES7-HandoverEUICC

ES4-SMSRChange

**SM-SR-S**

**SM-SR-UT**

**MNO1-S**

###### TC.ES7.ASMSR.1: AuthenticateSMSR

###### Test Purpose

*To ensure the method AuthenticateSMSR is well implemented on the SM-SR. Only error case is defined:*

* *SM-SR certificate expired*

###### Referenced Requirements

* EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS

###### Test Sequence N°1 – Error Case: Invalid SM-SR Certificate

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

{SM\_DP\_ID\_RPS} has been set to #SM\_DP\_S\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS | EUICC\_REQ36, EUICC\_REQ39 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR, #VIRTUAL\_EID\_RPS, #EXPIRED\_SM\_SR\_CERTIFICATE) |  |  |
| 4 | SM-SR-UT→ SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR\_CERT 2. The Reason code is equal to #RC\_EXPIRED | EUICC\_REQ40 |
| 5 | SM-SR-S→ SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_SM\_SR\_CERT, #RC\_EXPIRED) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR\_CERT 2. The Reason code is equal to #RC\_EXPIRED | EUICC\_REQ39 |

###### Test Sequence N°2 – Error Case: SM-SR certificate signature cannot be verified

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS,  #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_ES7\_RPS | EUICC\_REQ36, EUICC\_REQ39 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR, #VIRTUAL\_EID\_RPS, #INVALID\_SM\_SR\_CERTIFICATE) |  |  |
| 4 | SM-SR-UT→ SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR\_CERT 2. The Reason code is equal to #RC\_VERIFICATION\_FA ILED | EUICC\_REQ40 |
| 5 | SM-SR-S→ SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_SM\_SR\_CERT,  #RC\_VERIFICATION\_FAILED) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR\_CERT 2. The Reason code is equal to #RC\_VERIFICATION\_FA ILED | EUICC\_REQ39 |

###### Test Sequence N°3 – Error Case: The target SMSRid is unknown

###### Initial Conditions

The eUICC identified by the #VIRTUAL\_EID is provisioned on the SM-SR-UT with the #EIS\_ES1\_RPS

{SM\_SR\_ID\_RPS} has been set to #SM\_SR\_UT\_ID\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #VIRTUAL\_EID\_RPS,  #TGT\_UK\_SR\_S\_ID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_SM\_SR 2. The Reason code is equal to #RC\_UNKNOWN | EUICC\_REQ39 |

## ES7 (SM-SR – SM-SR): CreateAdditionalKeySet

###### Conformance Requirements

###### References

* GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* EUICC\_REQ35, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13

###### Test Cases

###### General Initial Conditions

* All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)
* #MNO1\_S\_ID is well known to the SM-SR-UT
* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
* The eUICC identified by #VIRTUAL\_EID is not provisioned on the SM-SR-UT

###### Test Environment

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC

ES7-AuthenticateSMSR

ES7-AuthenticateSMSR

ES7-CreateAdditionalKeyset

ES7-CreateAdditionalKeyset

ES7-HandoverEUICC

ES4-SMSRChange

**SM-SR-UT**

**SM-SR-S**

**MNO1-S**

Note that the function ES4-SMSRChange SHALL NOT be performed by the simulators (in the schema above, they are only informative messages).

###### TC.ES7.CAK.1: CreateAdditionalKeyset

###### Test Purpose

*To ensure the method CreateAdditionalKeyset is well implemented on the SM-SR. This test proposes to simulate that an invalid receipt has been generated by the eUICC. In this case, the new SM-SR SHALL send a corresponding error code to the former SM-SR through the method HandoverEUICC.*

###### Referenced Requirements

* EUICC\_REQ35, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13

###### Initial Conditions

* None

###### Test Sequence N°1 – Error Case: Invalid Receipt

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS, #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT  → MNO1-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35 PROC\_REQ13 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_REQ(  ES7-HandoverEUICC, #EIS\_ES7\_RPS) |  |  |
| 4 | SM-SR- UT→ SM- SR-S | Send the  ES7-AuthenticateSMSR  request | 1. The EID parameter is equal to #VIRTUAL\_EID\_RPS 2. The smsrCertificate parameter is present and contain all mandatory TLVs 3. Tag ‘73’ of the SM-SR certificate contains tags ‘C8’ and ‘C9’ (tag ‘C8’ is set to ‘02’) | EUICC\_REQ40 PROC\_REQ13 |
| 5 | SM-SR-S→ SM-SR-UT | SEND\_SUCCESS\_RESP(  ES7-AuthenticateSMSR,  {RC})  The {RC} is randomly generated (16 bytes long) |  |  |
| 6 | SM-SR- UT→ SM- SR-S | Send the  ES7-CreateAdditionalKeyset  request | 1. All mandatory input parameters are present 2. The EID parameter is equal to #VIRTUAL\_EID\_RPS 3. scenarioParameter SHALL be set to ‘09’, ‘0B’, ‘0D’ or ‘0F’ 4. hostId parameter SHALL be set only if scenarioParameter indicates that Host and Card ID are included in the key derivation process (i.e. bit3 is set to 1) | EUICC\_REQ38 PROC\_REQ13 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | SM-SR-S→ SM-SR-UT | SEND\_SUCCESS\_RESP(  ES7-CreateAdditionalKeyset,  {DR}, {RECEIPT})  The {DR} is randomly generated (16 bytes long)  The {RECEIPT} is randomly generated (16 bytes long)  See Note |  |  |
| 8 | SM-SR- UT→ SM- SR-S | Send the  ES7-HandoverEUICC  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_CERT\_REQ 2. The Reason code is equal to #RC\_VERIFICATION\_FAIL ED | EUICC\_REQ39 PROC\_REQ13 |
| *Note: The {DR} SHALL be generated and passed as an output parameter only if the scenarioParameter set in the ES7-CreateAdditionalKeyset request indicates that the derivation random SHALL be included in the key derivation process (i.e. bit2 set to 1)* | | | | |

# System Behaviour Testing

## General Overview

This section focuses on the implementation of the system according to the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7). The aim is to verify the functional behaviour of the system.

## eUICC Behaviour

## Device – eUICC

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + EUICC\_REQ10, EUICC\_REQ11

###### Test Cases

###### General Initial Conditions

* + - * + None

###### TC.ECASD.1: EIDRetrieval

###### Test Purpose

*To ensure the Device can retrieve the EID by reading the ECASD information.*

###### Referenced Requirements

* + - * + EUICC\_REQ10, EUICC\_REQ11

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | [SELECT\_ECASD] |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | ATS | SW=’9000’ | EUICC\_REQ10, EUICC\_REQ11 |
| 4 | DS → eUICC-UT | [GET\_DATA\_5A] |  |  |
| 5 | eUICC-UT → DS | TAG ‘5A’ returned | 1. TAG ‘5A’ content:    1. is equal to #EID    2. starts with the byte ‘89’    3. is 16 bytes long   2- SW=’9000’  3- Using the TAG content as a decimal integer, the remainder of the division by 97 SHALL be equal to 1 | EUICC\_REQ10 |
| *Note: On this test, the basic channel 00 is used but it is assumed that a logical channel can be used* | | | | |

## LOCKED State Unsupported by ISD-R and ISD-P

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ1, EUICC\_REQ6, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Test Cases

###### General Initial Conditions

* + - * + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### TC.LOCKISDR.1: LockISDR

###### Test Purpose

*To ensure ISD-R cannot be locked. After trying to lock the ISD-R, an audit is performed to make sure that the lifecycle state of the security domain remains unchanged.*

###### Referenced Requirements

* + - * + PF\_REQ7
        + EUICC\_REQ1, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Error Case: Unable to Lock the ISD-R

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [LOCK\_ISDR]) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_6985]   (see Note 1) | EUICC\_REQ1, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ22 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_E3\_ISDP\_3F] (i.e. the ISD-R is not LOCKED) | EUICC\_REQ1, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, PF\_REQ7 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6D00’ or ‘6A86’ or ‘6A81’* | | | | |

###### TC.LOCKISDP.1: LockISDP

###### Test Purpose

*To ensure an ISD-P cannot be locked. After trying to lock the ISD-P, an audit is performed to make sure that the lifecycle state of the security domain remains unchanged.*

###### Referenced Requirements

PF\_REQ7

EUICC\_REQ6, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

None

###### Test Sequence N°1 – Error Case: Unable to Lock an ISD-P

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [LOCK\_DEFAULT\_ISDP]) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_6985]   (see Note 1) | EUICC\_REQ6, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_ISDP\_ENABLED]) |  | EUICC\_REQ22 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal   to [R\_AB\_E3\_ISDP\_3F] | EUICC\_REQ6, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, PF\_REQ7 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6D00’ or ‘6A86’ or ‘6A81’* | | | | |

## Components and Visibility

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PM\_REQ1, PM\_REQ2, PM\_REQ5
        + EUICC\_REQ2, EUICC\_REQ3, EUICC\_REQ8, EUICC\_REQ9, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Test Cases

###### General Initial Conditions

* + - * + None

###### TC.CV.1: ComponentVisibility

###### Test Purpose

*To ensure Profile Component cannot have any visibility to components outside its ISD-P and that an ISD-P SHALL NOT have any visibility of, or access to, any other ISD-P.*

###### Referenced Requirements

* + - * + PM\_REQ2
        + EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: No Visibility for the MNO-SD to the ISD-R

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, [GET\_STATUS\_ISDR])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88] | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22; PM\_REQ2 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°2 – Nominal Case: No Visibility for an ISD-P to another ISD-P

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID and #ISD\_P\_AID1 are present on the eUICC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [GET\_ISDP1])) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. INITIALIZE UPDATE and EXTERNAL AUTHENTICATE   commands are successfully executed (i.e. SW=’9000’)   1. SW=’6A88’ for the GET STATUS command (see Note 1) | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, PM\_REQ2 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’ or ‘6D00’* | | | | |

###### TC.CV.2: ISDRVisibility

###### Test Purpose

*To ensure any component outside the ISD-P cannot have any visibility to Profile Components. In this test case, the aim is to verify that the ISD-R cannot have any visibility on the MNO-SD.*

###### Referenced Requirements

PM\_REQ1

EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case: No Visibility for the ISD-R to the MNO-SD

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID present on the eUICC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [GET\_MNO\_SD]) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_6A88] | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, PM\_REQ1 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### TC.CV.3: ISDPNotEnabled

###### Test Purpose

*To ensure the applications or the file system within a Disabled Profile cannot be selected. In this test case, a new Profile including an applet and a file is dynamically downloaded: the selection of these two components SHALL be only possible when the Profile state is updated to Enabled.*

###### Referenced Requirements

EUICC\_REQ8, EUICC\_REQ9

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

#ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process *ES8-EstablishISDPKeySet* has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

TP-Destination-Address has been set on #ISD\_R\_AID with #DEST\_ADDR

###### Test Sequence N°1 - Nominal Case using CAT\_TP: Applet Selectable Only on an Enabled Profile

###### Initial Conditions

Applet3 (defined in [A.3](#_bookmark262)) is not present on the Profile linked to the

#DEFAULT\_ISD\_P\_AID

#PE\_APPLET3 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | Execute the test sequence defined in section [4.2.18.2.1.1](#_bookmark129) (TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE (including #PE\_APPLET3) under the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 4 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 5 | Execute the test sequence defined in section [4.2.19.2.1.1](#_bookmark134) (TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS) from step 2 to step 6 in order to set the SMS Connectivity Parameters in the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 6 | DS → eUICC-UT | [SELECT\_APPLET3] |  |  |
| 7 | eUICC-UT → DS | ATS | SW=’6A82’ | EUICC\_REQ9 |
| 8 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 9 | Execute the test sequence defined in section [4.2.4.2.1.1](#_bookmark70) (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 9 in order to Enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.13.2.1.1](#_bookmark109) (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 1 to step 16 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| 11 | DS → eUICC-UT | [SELECT\_APPLET3] |  |  |
| 12 | eUICC-UT → DS | ATS | SW=’9000’ | EUICC\_REQ9 |

###### Test Sequence N°2 - Nominal Case using HTTPS: Applet Selectable Only on an Enabled Profile

###### Initial Conditions

Applet3 (defined in [A.3](#_bookmark262)) is not present on the Profile linked to the

#DEFAULT\_ISD\_P\_AID

#PE\_APPLET3 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | Execute the test sequence defined in section [4.2.18.2.2.1](#_bookmark131) (TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS) from step 3 to step  8 in order to download the #PROFILE\_PACKAGE (including  #PE\_APPLET3) under the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 4 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | Execute the test sequence defined in section [4.2.19.2.1.1](#_bookmark134) (TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS) from step 2 to step 6 in order to set the SMS Connectivity Parameters in the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 6 | DS → eUICC-UT | [SELECT\_APPLET3] |  |  |
| 7 | eUICC-UT → DS | ATS | SW=’6A82’ | EUICC\_REQ9 |
| 8 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 9 | Execute the test sequence defined in section [4.2.4.2.1.1](#_bookmark70)  (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 9 in order to Enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.13.2.1.1](#_bookmark109) (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 1 to step 16 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| 11 | DS → eUICC-UT | [SELECT\_APPLET3] |  |  |
| 12 | eUICC-UT → DS | ATS | SW=’9000’ | EUICC\_REQ9 |

###### Test Sequence N°3 - Nominal Case using CAT\_TP: File Selectable Only on an Enabled Profile

###### Initial Conditions

Elementary File with the identifier '1122' is not present on the Profile linked to the

#DEFAULT\_ISD\_P\_AID

#PE\_EF1122 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 3 | Execute the test sequence defined in section [4.2.18.2.1.1](#_bookmark129) (TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP) from step 3 to step  8 in order to download the #PROFILE\_PACKAGE (including  #PE\_EF1122) under the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 4 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 5 | Execute the test sequence defined in section [4.2.19.2.1.1](#_bookmark134) (TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS) from step 2 to step 6 in order to set the SMS Connectivity Parameters in the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 6 | DS → eUICC-UT | [SELECT\_FILE\_1122] |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | eUICC-UT → DS | ATS | SW=’6A82’ | EUICC\_REQ8 |
| 8 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 9 | Execute the test sequence defined in section [4.2.4.2.1.1](#_bookmark70)  (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 9 in order to Enable the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.13.2.1.1](#_bookmark109) (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 1 to step 16 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| 11 | DS → eUICC-UT | [SELECT\_FILE\_1122] |  |  |
| 12 | eUICC-UT → DS | ATS | SW=’9000’ | EUICC\_REQ8 |

###### Test Sequence N°4 - Nominal Case using HTTPS: File Selectable Only on an Enabled Profile

###### Initial Conditions

Elementary File with the identifier '1122' is not present on the Profile linked to the

#DEFAULT\_ISD\_P\_AID

#PE\_EF1122 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | Execute the test sequence defined in section [4.2.18.2.2.1](#_bookmark131) (TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS) from step 3 to step  8 in order to download the #PROFILE\_PACKAGE (including  #PE\_EF1122) under the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 4 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |
| 5 | Execute the test sequence defined in section [4.2.19.2.1.1](#_bookmark134) (TC.ES8.UCP.1:UpdateConnectivityParameters\_SMS) from step 2 to step 6 in order to set the SMS Connectivity Parameters in the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 6 | DS → eUICC-UT | [SELECT\_FILE\_1122] |  |  |
| 7 | eUICC-UT → DS | ATS | SW=’6A82’ | EUICC\_REQ8 |
| 8 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 9 | Execute the test sequence defined in section [4.2.4.2.1.1](#_bookmark70) (TC.ES5.EP.1:EnableProfile\_SMS) from step 2 to step 9 in order to Enable the #ISD\_P\_AID1 | | All steps successfully executed |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | Execute the test sequence defined in section [4.2.13.2.1.1](#_bookmark109) (TC.ES5.NOTIFPE.1:Notification\_SMS) from step 1 to step 16 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |
| 11 | DS → eUICC-UT | [SELECT\_FILE\_1122] |  |  |
| 12 | eUICC-UT → DS | ATS | SW=’9000’ | EUICC\_REQ8 |

###### TC.CV.4: TarAllocation

###### Test Purpose

*To ensure it is possible to allocate the same TAR within distinct Profiles. In this test case, an applet is installed through the MNO-SD on the Enabled Profile. Then, another applet with the same TAR is installed during the downloading of a new Profile. An error case is also defined to make sure that a Profile Component cannot use the reserved ISD-R TAR.*

###### Referenced Requirements

EUICC\_REQ3

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

Applet1 and Applet2 (defined in [Annex A](#_bookmark253)) are not present on the default Profile identified by #DEFAULT\_ISD\_P\_AID

###### Test Sequence N°1 - Nominal Case using CAT\_TP: Same TAR within Two Profiles

###### Initial Conditions

Applet1 and Applet2 (defined in [Annex A](#_bookmark253)) are not present on the Profile identified by

#ISD\_P\_AID1

#PE\_APPLET1 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET2}; [INSTALL\_APPLET2])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | PROACTIVE COMMAND: SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 8 | Execute the test sequence defined in section [4.2.3.2.2.1](#_bookmark65) (TC.ES5.CISDP.2:CreateISDP\_CAT\_TP) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 9 | Execute the test sequence defined in section [4.2.17.2.2.1](#_bookmark124) (TC.ES8.EISDPK.2:EstablishISDPkeyset\_CAT\_TP) from step 3 to step 6 in order to personalize the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.18.2.1.1](#_bookmark129) (TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE (including #PE\_APPLET1) under the #ISD\_P\_AID1 | | All steps successfully executed | EUICC\_REQ3 |
| 11 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### Test Sequence N°2 - Nominal Case using HTTPS: Same TAR within Two Profiles

###### Initial Conditions

Applet1 and Applet2 (defined in [Annex A](#_bookmark253)) are not present on the Profile identified by

#ISD\_P\_AID1

#PE\_APPLET1 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET2}; [INSTALL\_APPLET2])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | PROACTIVE COMMAND: SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 8 | Execute the test sequence defined in section [4.2.3.2.3.1](#_bookmark67) (TC.ES5.CISDP.3:CreateISDP\_HTTPS) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 9 | Execute the test sequence defined in section [4.2.17.2.3.1](#_bookmark126) (TC.ES8.EISDPK.3:EstablishISDPkeyset\_HTTPS) from step 3 to step 6 in order to personalize the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.18.2.2.1](#_bookmark131) (TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE (including #PE\_APPLET1) under the #ISD\_P\_AID1 | | All steps successfully executed | EUICC\_REQ3 |
| 11 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

###### Test Sequence N°3 - Error Case: Unauthorized ISD-R TAR

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET1})  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, [INSTALL\_TAR\_ISDR])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’6985’ for the   INSTALL command (see Note 1) | EUICC\_REQ3 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’* | | | | |

###### TC.CV.5: AIDAllocation

###### Test Purpose

*To ensure it is possible to allocate the same AID within distinct Profiles. In this test case, an applet is installed through the MNO-SD on the Enabled Profile. Then, another applet with the same AID is installed during the downloading of a new Profile. An error case is also defined to make sure that a Profile Component cannot use the reserved ECASD AID.*

###### Referenced Requirements

EUICC\_REQ2

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

Applet3 (defined in [A.3](#_bookmark262)) is not present on the default Profile identified by

#DEFAULT\_ISD\_P\_AID

###### Test Sequence N°1 - Nominal Case using CAT\_TP: Same AID within Two Profiles

###### Initial Conditions

Applet3 (defined in [A.3](#_bookmark262)) is not present on the Profile identified by #ISD\_P\_AID1

#PE\_APPLET3 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET3}; [INSTALL\_APPLET3])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 8 | Execute the test sequence defined in section [4.2.3.2.2.1](#_bookmark65) (TC.ES5.CISDP.2:CreateISDP\_CAT\_TP) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 9 | Execute the test sequence defined in section [4.2.17.2.2.1](#_bookmark124) (TC.ES8.EISDPK.2:EstablishISDPkeyset\_CAT\_TP) from step 3 to step 6 in order to personalize the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.18.2.1.1](#_bookmark129) (TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE (including #PE\_APPLET3) under the #ISD\_P\_AID1 | | All steps successfully executed | EUICC\_REQ2 |
| 11 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |

###### Test Sequence N°2 - Nominal Case using HTTPS: Same AID within Two Profiles

###### Initial Conditions

Applet3 (defined in [A.3](#_bookmark262)) is not present on the Profile identified by #ISD\_P\_AID1

#PE\_APPLET3 defined in section [B.7.3](#_bookmark279) SHALL be added to the

#PROFILE\_PACKAGE

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET3}; [INSTALL\_APPLET3])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 8 | Execute the test sequence defined in section [4.2.3.2.3.1](#_bookmark67) (TC.ES5.CISDP.3:CreateISDP\_HTTPS) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 9 | Execute the test sequence defined in section [4.2.17.2.3.1](#_bookmark126) (TC.ES8.EISDPK.3:EstablishISDPkeyset\_HTTPS) from step 3 to step 6 in order to personalize the #ISD\_P\_AID1 | | All steps successfully executed |  |
| 10 | Execute the test sequence defined in section [4.2.18.2.2.1](#_bookmark131) (TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE (including #PE\_APPLET3) under the #ISD\_P\_AID1 | | All steps successfully executed | EUICC\_REQ2 |
| 11 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | |

###### Test Sequence N°3 - Error Case: Unauthorized ECASD AID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR,  {LOAD\_APPLET3})  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’9000’ for all commands |  |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, [INSTALL\_AID\_ECASD])  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. SW=’6985’ for the INSTALL   command (see Note 1) | EUICC\_REQ2 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’* | | | | |

###### TC.CV.6: MNOSDDefinition

###### Test Purpose

*To ensure the MNO-SD AID and TAR can be freely allocated during the Profile definition. In this test case, a GET STATUS is sent to the MNO-SD to retrieve its information.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

PM\_REQ5

###### Initial Conditions

#DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, [GET\_MNO\_ISD])  Use #MNO\_SCP80\_ENC\_KEY,  #MNO\_SCP80\_AUTH\_KEY, #MNO\_SCP80\_DATA\_ENC\_KEY |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to [R\_AB\_MNO\_SD] | PM\_REQ5, EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

## Security and Responsibility

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ1
        + SEC\_REQ6
        + EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ19, EUICC\_REQ20, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ54, EUICC\_REQ55, EUICC\_REQ56, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61

###### Test Cases

###### General Initial Conditions

* + - * + None

###### TC.SAR.1: SecurityError\_SMS

###### Test Purpose

*To ensure a SMS SHALL be rejected by the eUICC (i.e. no POR returned) when:*

* + - * + *the security level does not meet the one expected by the ISD-R*
        + *the SM-SR is not authenticated*

###### Referenced Requirements

* + - * + EUICC\_REQ20

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Error Case: Low Security Level

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #BAD\_SPI, #ISD\_R\_TAR,  [GET\_DEFAULT\_ISDP]) |  |  |
| 3 | eUICC-UT → DS | *NO PROACTIVE COMMAND PENDING* | No SMS POR sent SW=’9000’ | EUICC\_REQ20 |

###### Test Sequence N°2 – Error Case: eUICC cannot Authenticate the SM-SR

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DEFAULT\_ISDP])  Do not use the  #SCP80\_ENC\_KEY,  #SCP80\_AUTH\_KEY, #SCP80\_DATA\_ENC\_KEY  see Note |  |  |
| 3 | eUICC-UT → DS | *NO PROACTIVE COMMAND PENDING* | No SMS POR sent SW=’9000’ | EUICC\_REQ20 |
| *Note: The correct ISD-R SCP80 keys SHALL NOT be used. Other values with same length can be freely chosen.* | | | | |

###### TC.SAR.2: ISDRResponsibility

###### Test Purpose

*To ensure only ISD-R can create an ISD-P.*

###### Referenced Requirements

PF\_REQ1

EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

None

###### Test Sequence N°1 - Error Case: ISD-P Cannot Create another ISD-P

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #DEFAULT\_ISD\_P\_TAR, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN,  [INSTALL\_ISDP])) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. INITIALIZE UPDATE and EXTERNAL AUTHENTICATE   commands are successfully executed (i.e. SW=’9000’)   1. The SW is ‘6985’ for the INSTALL command (see Note 1) | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, PF\_REQ1 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note 1: The SW MAY be also ‘6A80’, ‘6A88’ or ‘6D00’* | | | | |

###### TC.SAR.3: ReplayAttack

###### Test Purpose

*To ensure the communication between the SM-SR and the eUICC is protected against replay attacks. In this test case, the same secured packet is sent twice to make sure that only the first one is accepted by the eUICC.*

###### Referenced Requirements

SEC\_REQ6

EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22

###### Initial Conditions

None

###### Test Sequence N°1 - Error Case: Same Secured Packet Not Accepted

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR, [GET\_DEFAULT\_ISDP]) |  | EUICC\_REQ22 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #SCP80\_AUTH\_KEY 3. The response data is in   expanded format with definite length | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| 7 | DS → eUICC-UT | Send exactly the same SMS as the previous one |  | EUICC\_REQ22 |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE | see Note |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. Verify the cryptographic   checksum using  #SCP80\_AUTH\_KEY   1. No response data is returned 2. The status code is equal to ‘02’ - Counter low | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, SEC\_REQ6 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: Depending on the implementation, the eUICC MAY decide to not send back a POR (i.e. SW ‘9000’ on the ENVELOPE command). Therefore, the steps 8, 9, 10 and 11 SHALL be considered as optional.* | | | | |

###### TC.SAR.4: HTTPSRestrictions

###### Test Purpose

*To ensure the following HTTPS restrictions are well configured on the ISD-R:*

*TLS 1.2 SHALL only be supported meaning that the ‘i’ parameter is set to ‘04’*

*session resumption SHALL NOT be supported*

*several parallel sessions SHALL NOT be supported*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ43, EUICC\_REQ45, EUICC\_REQ46, EUICC\_REQ47, EUICC\_REQ54, EUICC\_REQ55, EUICC\_REQ56

###### Initial Conditions

None

###### Test Sequence N°1 - Nominal Case: TLS 1.2 only Supported by ISD- R

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.1 [[15]](#_bookmark20) SHALL be supported

Only the cipher-suite TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL be accepted

Note: the cipher-suite TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 cannot be used here as it SHALL be only negotiated using TLS version 1.2

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [OPEN\_SCP81\_SESSION]) |  | EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ54 |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:* SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The SCP80 status code is equal to ‘00’ – POR OK | EUICC\_REQ21 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 7 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL |  |  |
| 8 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | 1. The bearer description is equal to #BEARER\_DESCRIPTION 2. The buffer size is equal to #BUFFER\_SIZE 3. The NAN is equal to   #NAN\_VALUE   1. The port is equal to   #TCP\_PORT   1. The IP is equal to   #IP\_VALUE | EUICC\_REQ13, EUICC\_REQ14, EUICC\_REQ42 |
| 10 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |
| 11 | *For readability reason, the proactive commands are not fully specified in the next steps.*  *The BIP communication between the DS and the eUICC-UT SHALL be compliant with the* [*Annex F.*](#_bookmark285) *The TLS records used here after SHALL be compliant with the Annex H.* | | | |
| 12 | eUICC-UT → DS | TLS\_CLIENT\_HELLO |  | EUICC\_REQ14, EUICC\_REQ43 |
| 13 | DS → eUICC-UT | TLS\_1\_1\_SERVER\_HELLO  and TLS\_1\_1\_SERVER\_HELLO\_DONE |  |  |
| 14 | eUICC-UT → DS | TLS\_ALERT\_PROTOCOL\_VERSION |  | EUICC\_REQ55 |
| 15 | eUICC-UT → DS | *PROACTIVE COMMAND:*  CLOSE CHANNEL | The HTTP session is closed. | EUICC\_REQ55 |
| 16 | DS → eUICC-UT | TERMINAL RESPONSE |  |  |

###### Test Sequence N°2 - Nominal Case: No TLS Session Resumption

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

* Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported
* Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

* The following Pre-Shared Key SHALL be defined:
  + PSK identifier: #PSK\_ID
  + PSK value: #SCP81\_PSK

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section 4.2.1.1 | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC- UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [OPEN\_SCP81\_WITH\_RETRY]) |  |  |
| 3 | Execute the generic sub-sequence “Open HTTPS session on ISD-R” defined in section 4.2.1.5 from step 2 to step 9 | | | |
| 4 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The TLS\_CLIENT\_HELLO does not contain a SessionTicket extension  (SessionTicket extension type = 0x0023) | EUICC\_REQ56 |
| 5 | Execute the generic sub-sequence “Open HTTPS session on ISD-R” defined in section 4.2.1.5 from step 11 to step 14 | | | |
| 6 | DS → eUICC- UT | RESET | IP communication is broken ATR returned by eUICC |  |
| 7 | DS → eUICC- UT | [TERMINAL\_PROFILE] | Toolkit initialization |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  OPEN CHANNEL | See Note |  |
| 9 | DS → eUICC- UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  OPEN CHANNEL | The ISD-R makes first attempt for resuming the HTTP administration session |  |
| 11 | DS → eUICC- UT | TERMINAL RESPONSE |  |  |
| 12 | eUICC-UT → DS | TLS\_CLIENT\_HELLO | The TLS\_CLIENT\_HELLO  contains an empty Session Identifier (i.e. the previous TLS session is not reused) | EUICC\_REQ56 |
| 13 | Execute the generic sub-sequence “Open HTTPS session on ISD-R” defined in section 4.2.1.5 from step 11 to step 14 | | | |
| 14 | Close HTTPS session as described in section 4.2.1.7 | | | |
| Note: The OPEN CHANNEL command MAY be triggered by a TIMER EXPIRATION if the eUICC supports TIMER MANAGEMENT. | | | | |

###### Test Sequence N°3 - Nominal Case: No HTTPS Sessions in Parallel

###### Initial Conditions

The HTTPS server SHALL be configured as follow:

* Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported
* Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

* The following Pre-Shared Key SHALL be defined:
  + PSK identifier: #PSK\_ID
  + PSK value: #SCP81\_PSK

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | |
| 3 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_R\_TAR,  [OPEN\_SCP81\_SESSION]) |  | EUICC\_REQ22, EUICC\_REQ42, EUICC\_REQ54 |
| 4 | eUICC-UT → DS | *NO OPEN CHANNEL COMMAND PENDING* | A new HTTPS session SHALL NOT be open (see Note) | EUICC\_REQ56 |
| *Note: Depending on the implementation, a SMS POR MAY be returned by the eUICC with an incorrect SW (e.g. ‘9300’).* | | | | |

###### TC.SAR.5: SCP03t\_ErrorManagement

###### Test Purpose

*To ensure SCP03t is well implemented on the eUICC. This test case proposes to check that a dedicated error (e.g. reference data not found, error in length, security error) is returned when incorrect SCP03t command is sent.*

*Note that all the following error cases propose to send small SCP03t scripts over SMS. Depending on the eUICC implementation, it MAY be necessary to run these tests only over HTTPS or CAT\_TP.*

###### Referenced Requirements

EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ59, EUICC\_REQ60, EUICC\_REQ61

###### Initial Conditions

#ISD\_P\_AID1 present on the eUICC and personalized with SCP03 keys

The process ES8-EstablishISDPKeySet has been used

{SCP\_KENC}, {SCP\_KMAC}, {SCP\_KDEK} have been set

###### Test Sequence N°1 – Error Case: Incorrect Length in INITIALIZE UPDATE

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #SCP03\_KVN, #PE\_HEADER))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK}  Change the length value of the INITIALIZE UPDATE TLV command  before sending the script (e.g. with ‘11’ instead of ‘0A’) |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_SCP03T\_IU\_01]   See Note | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ59 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: Instead of using the SCP03t error tag (0x9F44), the eUICC MAY return the Bad format TLV tag (i.e. 0x90) indicating “Wrong length found” (i.e. 0x02) as defined in ETSI TS 102 226 [6].* | | | | |

###### Test Sequence N°2 – Error Case: Incorrect Parameter in INITIALIZE UPDATE

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #BAD\_SCP03\_KVN, #PE\_HEADER))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK} |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_SCP03T\_IU\_03] | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ59 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

###### Test Sequence N°3 – Error Case: Incorrect Length in EXTERNAL AUTHENTICATE

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #SCP03\_KVN, #PE\_HEADER))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK}  Change the length value of the EXTERNAL AUTHENTICATE TLV  command (TAG ‘85’) before sending the script (e.g. with ‘19’ instead of ‘11’) |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_SCP03T\_EA\_01]   See Note | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ60 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: Instead of using the SCP03t error tag (0x9F45), the eUICC MAY return the Bad format TLV tag (i.e. 0x90) indicating “Wrong length found” (i.e. 0x02) as defined in ETSI TS 102 226 [6].* | | | | |

###### Test Sequence N°4 – Error Case: Incorrect Security in EXTERNAL AUTHENTICATE

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #SCP03\_KVN, #PE\_HEADER))  Do not use the SCP03 keys  {SCP\_KENC}, {SCP\_KMAC} and  {SCP\_KDEK}  see Note |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal   to [R\_AB\_SCP03T\_EA\_02] | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ60 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: The correct ISD-P SCP03 keys SHALL NOT be used. Other values with same length can be freely chosen.* | | | | |

###### Test Sequence N°5 – Error Case: Incorrect Length in Profile TLV Command

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #SCP03\_KVN, #PE\_HEADER))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK}  Change the length value of the Profile data TLV command (TAG ‘86’) before sending the script |  |  |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_SCP03T\_01]   See Note | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ61 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: Instead of using the SCP03t error tag (0x9F46), the eUICC MAY return the Bad format TLV tag (i.e. 0x90) indicating “Wrong length found” (i.e. 0x02) ad defined in ETSI TS 102 226 [6].* | | | | |

###### Test Sequence N°6 – Error Case: Incorrect Security in Profile TLV Command

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 2 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #ISD\_P\_TAR1, SCP03T\_SCRIPT(  #SCP03\_KVN, #PE\_HEADER))  Use the SCP03 keys {SCP\_KENC},  {SCP\_KMAC} and {SCP\_KDEK}  Corrupt a block of ciphered data in the Profile data TLV command (TAG ‘86’) before sending the script |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:*  SEND SHORT MESSAGE |  |  |
| 4 | DS → eUICC-UT | FETCH |  |  |
| 5 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #SCP80\_ENC\_KEY 2. The response data is equal to [R\_AB\_SCP03T\_02] | EUICC\_REQ13, EUICC\_REQ19, EUICC\_REQ21, EUICC\_REQ22, EUICC\_REQ61 |
| 6 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

## Confidential Setup of MNO Secure Channel Keys

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)

###### Requirements

* + - * + SEC\_REQ20

###### Test Cases

###### General Initial Conditions

* + - * + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)

###### TC.CSMNOSCK.1: Scenario#2.B

###### Test Purpose

*To ensure MNO can update the OTA Keys on its Profile using the scenario #2.B as defined in GlobalPlatform Card Specification v.2.2.1 - UICC Configuration* [*[13]*](#_bookmark18)*.*

###### Referenced Requirements

* + - * + SEC\_REQ20

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by the eUICC |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | DS → eUICC-UT | [SELECT\_CASD] |  |  |
| 3 | eUICC-UT → DS | ATS | SW=’9000’ | SEC\_REQ20 |
| 4 | DS → eUICC-UT | [GET\_DATA\_CASD\_CERT] |  |  |
| 5 | eUICC-UT → DS | DGI ‘7F21’ returned | 1. The returned DGI ‘7F21’contains the TLV certificate [R\_CASD\_SC2B] 2. The {PK\_CASD\_CT} SHALL be   recovered from the signature using the #EUM\_PK\_CA\_AUT | SEC\_REQ20 |
| 6 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, STORE\_MNO\_KEYS\_2B(  {PK\_CASD\_CT})) Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_9000] | SEC\_REQ20 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |
| *Note: After the execution of this test, all the MNO-SD keysets SHOULD be deleted except the one identified by*  *#MNO\_SCP80\_KVN* | | | | |

###### TC.CSMNOSCK.2: Scenario#3

###### Test Purpose

*To ensure MNO can update the OTA Keys on its Profile using the scenario #3 as defined in GlobalPlatform Card Specification v.2.2 Amendment E: Security Upgrade for Card Content Management* [*[13]*](#_bookmark18)*.*

###### Referenced Requirements

SEC\_REQ20

###### Initial Conditions

None

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | DS → eUICC-UT | RESET | ATR returned by the eUICC |  |
| 2 | DS → eUICC-UT | [SELECT\_CASD] |  |  |
| 3 | eUICC-UT → DS | ATS | SW=’9000’ | SEC\_REQ20 |
| 4 | DS → eUICC-UT | [GET\_DATA\_CASD\_CERT] |  |  |
| 5 | eUICC-UT → DS | DGI ‘7F21’ returned | 1. The returned DGI ‘7F21’ contains the TLV certificate [R\_CASD\_SC3] 2. The {PK\_CASD\_CT} SHALL   be retrieved from the TAG ‘7F49’ | SEC\_REQ20 |
| 6 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | |
| 7 | DS → eUICC-UT | ENVELOPE\_SMS\_PP( #SPI\_VALUE, #MNO\_TAR, STORE\_MNO\_KEYS\_3())  Use #MNO\_SCP80\_ENC\_KEY, #MNO\_SCP80\_AUTH\_KEY,  #MNO\_SCP80\_DATA\_ENC\_KEY |  |  |
| 8 | eUICC-UT → DS | *PROACTIVE COMMAND PENDING:* SEND SHORT MESSAGE |  |  |
| 9 | DS → eUICC-UT | FETCH |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | eUICC-UT → DS | *PROACTIVE COMMAND:*  SEND SHORT MESSAGE | 1. Decrypt the response packet with the #MNO\_SCP80\_ENC\_KEY 2. Verify the cryptographic checksum using #MNO\_SCP80\_AUTH\_KEY 3. The response data is equal to   [R\_AB\_RECEIPT]   1. Calculate ShS from   #SM\_ESK\_ECKA and  {PK\_CASD\_CT}   1. Derive keyset from ShS and retrieve the {SCP\_KENC},   {SCP\_KMAC} and {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | SEC\_REQ20 |
| 11 | DS → eUICC-UT | TERMINAL RESPONSE | SW=’9000’ |  |

## Full Profile Installation Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ7, PROC\_REQ19, PROC\_REQ2, EUICC\_REQ51\_1

###### Test Cases

###### General Initial Conditions

* + - * + ISD-P #ISD\_P\_AID1 not present on the eUICC
        + #DEFAULT\_ISD\_P\_AID in Enabled state (SHALL be the initial state of the eUICC)
        + No POL1 is defined on the #DEFAULT\_ISD\_P\_AID

###### TC.FPIP.1: ProfileDownloadAndEnabling

###### Test Purpose

*To ensure a Profile can be fully downloaded using only one OTA session and Enabled. Here are the different steps that are executed:*

* + - * + *ISD-P creation*
        + *ISD-P keys establishment with scenario #3*
        + *Download and installation of a Profile*
        + *Profile enabling*

*The test sequences below propose to execute these steps using either CAT\_TP or HTTPS. Between each step related to the Profile Downloading process, no operation is performed on the eUICC during a delay of 30 seconds in order to simulate exchanges related to the off-card interfaces.*

###### Referenced Requirements

* + - * + PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ7, PROC\_REQ19, PROC\_REQ21

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 – Nominal Case: Using CAT\_TP

###### Initial Conditions

CAT\_TP Connectivity Parameters have been set on #ISD\_R\_AID with

#UDP\_PORT, #CAT\_TP\_PORT and #IP\_VALUE

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | | |
| 2 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | | |
| 3 | Execute the test sequence defined in section [4.2.3.2.2.1](#_bookmark65) (TC.ES5.CISDP.2:CreateISDP\_CAT\_TP) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | All steps executed | successfully | PROC\_REQ1 |
| *Maintain open the CAT\_TP session for 30 seconds by sending an ACK NUL every 10 seconds (as defined in steps 4 and 5)* | | | | | |
| 4 | DS → eUICC-UT | ACK\_NUL |  | |  |
| 5 | eUICC-UT → DS | ACK\_NO\_DATA |  | |  |
| *Third ACK NUL sent (Timer of 30 seconds reached)* | | | | | |
| 6 | Execute the test sequence defined in section [4.2.17.2.2.1](#_bookmark124) (TC.ES8.EISDPK.2:EstablishISDPkeyset\_CAT\_TP) from step 3 to step 4 in order to start the personalization of the #ISD\_P\_AID1 | | All steps executed | successfully | PROC\_REQ2 |
| 7 | Maintain open the CAT\_TP session for 30 seconds by executing steps 4 and 5 of this sequence | | | | |
| 8 | Execute the test sequence defined in section [4.2.17.2.2.1](#_bookmark124) (TC.ES8.EISDPK.2:EstablishISDPkeyset\_CAT\_TP) from step 5 to step 6 in order to finish the personalization of the #ISD\_P\_AID1 | | All steps executed | successfully | PROC\_REQ2 |
| 9 | Maintain open the CAT\_TP session for 30 seconds by executing steps 4 and 5 of this sequence | | | | |
| 10 | Execute the test sequence defined in section [4.2.18.2.1.1](#_bookmark129) (TC.ES8.DAI.1:DownloadAndInstallation\_CAT\_TP) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE under the #ISD\_P\_AID1 | | All steps executed | successfully | PROC\_REQ3 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 11 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 12 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 13 | Execute the test sequence defined in section [4.2.19.2.2.1](#_bookmark136) (TC.ES8.UCP.2:UpdateConnectivityParameters\_CAT\_TP) from step 3 to step 4 in order to set the CAT\_TP Connectivity Parameters in the #ISD\_P\_AID1 | | All steps successfully executed | PROC\_REQ19 |
| 14 | Close CAT\_TP session as described in section [4.2.1.4](#_bookmark54) | | | |
| 15 | Open CAT\_TP session on ISD-R as described in section [4.2.1.2](#_bookmark52) | | | |
| 16 | Execute the test sequence defined in section [4.2.4.2.2.1](#_bookmark72) (TC.ES5.EP.2:EnableProfile\_CAT\_TP) from step 3 to step 8 in order to Enable the #ISD\_P\_AID1 | | All steps successfully executed | PROC\_REQ7 |
| 17 | Execute the test sequence defined in section [4.2.13.2.2](#_bookmark110) (TC.ES5.NOTIFPE.2:Notification\_CAT\_TP) from step 1 to step 18 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed |  |

###### Test Sequence N°2 – Nominal Case: Using HTTPS

###### Initial Conditions

HTTPS Connectivity Parameters have been set on #ISD\_R\_AID with #TCP\_PORT, #IP\_VALUE, #ADMIN\_HOST, #AGENT\_ID, #PSK\_ID, #SCP81\_KVN, #SCP81\_KEY\_ID and #ADMIN\_URI

The HTTPS server SHALL be configured as follow:

Only the version TLS Protocol 1.2 [[8]](#_bookmark13) SHALL be supported

Only the cipher-suites TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 and TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 as defined in RFC 5487 [[9]](#_bookmark14) SHALL

be accepted

The following Pre-Shared Key SHALL be defined:

PSK identifier: #PSK\_ID

PSK value: #SCP81\_PSK

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | | **Expected result** | **REQ** |
| 1 | Initialization sequence as described in section [4.2.1.1](#_bookmark51) | | | | |
| 2 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | | |
| 3 | Execute the test sequence defined in section [4.2.3.2.3.1](#_bookmark67) (TC.ES5.CISDP.3:CreateISDP\_HTTPS) from step 3 to step 4 in order to create the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ1 |
| 4 | DS → eUICC-UT | | TLS\_APPLICATION  containing the result of  HTTPS\_EMPTY\_CONTENT() |  | EUICC\_REQ51  \_1 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | | **Expected result** | **REQ** |
| 5 | eUICC-UT → DS | | TLS\_APPLICATION with  empty body | 1. Decrypt the TLS record with the #SCP81\_PSK using the cipher-suite negotiated during the TLS handshake 2. The POST URI is equal to #POST\_URI 3. The different headers are equal to   #HOST #X\_ADMIN\_PROTOCOL  #X\_ADMIN\_FROM\_ISD\_R   1. The HTTP body is empty | EUICC\_REQ51  \_1 |
| 6 | Execute the test sequence defined in section [4.2.17.2.3.1](#_bookmark126) (TC.ES8.EISDPK.3:EstablishISDPkeyset\_HTTPS) from step 3 to step 4 in order to start the personalization of the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ2 |
| 7 | Execute steps 4 and 5 of this sequence | | | | EUICC\_REQ51  \_1 |
| 8 | Execute the test sequence defined in section [4.2.17.2.3.1](#_bookmark126) (TC.ES8.EISDPK.3:EstablishISDPkeyset\_HTTPS) from step 5 to step 6 in order to finish the personalization of the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ2 |
| 9 | Execute steps 4 and 5 of this sequence | | | | EUICC\_REQ51  \_1 |
| 10 | Execute the test sequence defined in section [4.2.18.2.2.1](#_bookmark131) (TC.ES8.DAI.2:DownloadAndInstallation\_HTTPS) from step 3 to step 8 in order to download the #PROFILE\_PACKAGE under the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ3 |
| 11 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | | |
| 12 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | | |
| 13 | Execute the test sequence defined in section [4.2.19.2.3.1](#_bookmark138) (TC.ES8.UCP.3:UpdateConnectivityParameters\_HTTPS) from step 3 to step 4 in order to set the HTTPS Connectivity Parameters in the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ19 |
| 14 | Close HTTPS session as described in section [4.2.1.7](#_bookmark57) | | | | |
| 15 | Open HTTPS session on ISD-R as described in section [4.2.1.5](#_bookmark55) | | | | |
| 16 | Execute the test sequence defined in section [4.2.4.2.3.1](#_bookmark74) (TC.ES5.EP.3:EnableProfile\_HTTPS) from step 3 to step 8 in order to Enable the #ISD\_P\_AID1 | | | All steps successfully executed | PROC\_REQ7 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 17 | Execute the test sequence defined in section [4.2.13.2.3.1](#_bookmark111) (TC.ES5.NOTIFPE.3:Notification\_HTTPS) from step 1 to step 19 in order to manage the different notifications exchanged with the eUICC and to make sure that the Profile linked to the #ISD\_P\_AID1 is now Enabled | | All steps successfully executed | PROC\_REQ21 |

## Platform Behaviour

## eUICC Identity Check

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + SEC\_REQ15
        + PROC\_REQ1
        + PM\_REQ11, PM\_REQ14
        + EUICC\_REQ35, EUICC\_REQ39

###### Test Cases

###### General Initial Conditions

* + - * + None

###### TC.EUICCIC.1: eUICCEligibilitySMDP

###### Test Purpose

*To ensure SM-DP is able to check the validity of an eUICC. In case of a bad ECASD in the eUICC, the SM-DP SHALL be able to refuse the download of the Profile.*

###### Test Environment

ES2-DownloadProfile

ES2-DownloadProfile

**SM-DP-UT**

**MNO1-S**

ES3-GetEIS

###### Referenced Requirements

**SM-SR-S**

* + - * + SEC\_REQ15
        + PROC\_REQ1
        + PM\_REQ11, PM\_REQ14

###### Initial Conditions

* + - * + The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
        + The variable {SM\_DP\_ID\_RPS} SHALL be set to #SM\_DP\_UT\_ID\_RPS
        + #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
        + #SM\_SR\_S\_ID and #SM\_SR\_S\_ACCESSPOINT well known to the SM-DP-UT
        + #EUM\_S\_PK\_ECDSA well known to the SM-DP-UT
        + The Profile #ICCID1 is well known to the SM-DP-UT

###### Test Sequence N°1 – Error Case: Invalid Signature in ECASD Certificate

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #VIRTUAL\_EID\_RPS,  {SM\_SR\_ID\_RPS}, #ICCID1\_RPS,  #EP\_FALSE\_RPS) |  |  |
| 2 | SM-DP-UT → SM-SR-S | Send the  ES3-GetEIS  request | The EID parameter is equal to  #VIRTUAL\_EID\_RPS | PROC\_REQ1, PM\_REQ11, PM\_REQ14 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-S → SM-DP-UT | SEND\_SUCCESS\_RESP( ES3-GetEIS,  #EIS\_BADCASDSIGN\_RPS) |  |  |
| 4 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to #FAILED 2. The Subject code   is equal to #SC\_ECASD | PM\_REQ11, SEC\_REQ15 |

##### VOID

###### TC.EUICCIC.2: eUICCEligibilitySMSR

###### Test Purpose

*To ensure SM-SR is able to check the validity of an eUICC. In case of a bad ECASD in the eUICC, the SM-SR SHALL be able to refuse the change of a SM-SR.*

###### Test Environment

**SM-SR-S**

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC

ES7-HandoverEUICC

ES4-SMSRChange

**SM-SR-UT**

**MNO1-S**

Note that the function ES4-SMSRChange SHALL NOT be performed by the simulators (in the schema above, they are only informative messages).

###### Referenced Requirements

* SEC\_REQ15
* EUICC\_REQ35, EUICC\_REQ39

###### Initial Conditions

* The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_S\_ID\_RPS
* The variable {SM\_DP\_ID\_RPS} SHALL be set to #SM\_DP\_S\_ID\_RPS
* #MNO1\_S\_ID and #MNO2\_S\_ID well known to the SM-SR-UT (because Profiles related to these operators are present in the EIS)
* The eUICC identified by the #VIRTUAL\_EID is not provisioned on the SM-SR-UT
* #EUM\_S\_PK\_ECDSA well known to the SM-SR-UT
* All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)

###### Test Sequence N°1 – Error Case: Invalid Signature in ECASD Certificate

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #VIRTUAL\_EID\_RPS,  #CUR\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → MNO1-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35 |
| 3 | SM-SR-S→ SM-SR-UT | SEND\_REQ(  ES7-HandoverEUICC, #EIS2\_BADCASDSIGN\_RPS) |  |  |
| 4 | SM-SR-UT→ SM-SR-S | Send the  ES7-HandoverEUICC  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_ECASD | EUICC\_REQ39, SEC\_REQ15 |

##### VOID

## Profile Download and Installation Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ7, PROC\_REQ20
        + PM\_REQ3, PM\_REQ4, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PM\_REQ22, PM\_REQ25
        + PF\_REQ2, PF\_REQ3, PF\_REQ4, PF\_REQ7, PF\_REQ18, PF\_REQ27
        + EUICC\_REQ27, EUICC\_REQ29, EUICC\_REQ42, EUICC\_REQ53

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT
        + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT

A direct connection exists between the MNO1-S and the SM-SR-UT

* + - * + The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS
        + #SM\_SR\_ID and #SM\_SR\_ACCESSPOINT well known to the SM-DP-UT
        + #SM\_DP\_ID and #SM\_DP\_ACCESSPOINT well known to the SM-SR-UT
        + The Profile identified by #ICCID is owned by MNO2-S and is in Enabled state
        + The SM-SR-UT is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the current Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)

* + - * + SM-DP-UT is responsible for downloading and installation of the Profile identified by

#NEW\_ICCID

A Profile similar to #PROFILE\_PACKAGE SHALL be stored on the SM-DP-UT and compatible with the eUICC

The Profile SHALL be associated with the Subscription Address #NEW\_MSISDN

###### TC.PROC.DIP.1: DownloadAndInstallProfile

###### Test Purpose

*To ensure that the Profile download and installation procedure is properly implemented on the SM-DP and the SM-SR. After the Profile download execution, an audit request is sent to the SM-SR to make sure that the Profile has been downloaded. The OTA capabilities set during the eUICC registration allow the use of CAT\_TP or HTTPS during the download process.*

###### Test Environment

**Device**

ES2-DownloadProfile

ES3-GetEIS

ES3-CreateISDP

ES5-CreateISDP

ES3-CreateISDP

ES3-SendData(ES8-EstablishISDPKeyset)

ES3-SendData(ES8-DownloadAndInstallation)

ES3-ProfileDownloadCompleted

ES2-DownloadProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Referenced Requirements

* + - * + EUICC\_REQ42, EUICC\_REQ53
        + PROC\_REQ1, PROC\_REQ2, PROC\_REQ3
        + PM\_REQ3, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PM\_REQ22, PM\_REQ25
        + PF\_REQ2, PF\_REQ3, PF\_REQ7

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 - Nominal Case: Using CAT\_TP

###### Initial Conditions

The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

the #EIS\_RPS SHALL be adapted to indicate that the eUICC does not support HTTPS

the capabilities #CATTP\_CAP\_RPS SHALL be used in the #EIS\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS, #EP\_FALSE\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #SUCCESS   1. The ICCID returned is equal to #NEW\_ICCID\_RPS | PROC\_REQ1,P ROC\_REQ2,PR OC\_REQ3, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PF\_REQ2, PF\_REQ3, EUICC\_REQ53 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned contains the new Profile information (i.e. identified by #NEW\_ICCID) 2. The new Profile information has a state equal to Disabled 3. The new Profile information has the SM-DP identifier set to #SM-DP-ID 4. The new Profile information has an ISD-P RID equal to #ISD\_P\_RID 5. The new Profile information has an ISD-P PIX that starts with #ISD\_P\_PIX\_PREFIX 6. The new Profile information has a MNO-ID equal to #MNO1\_S\_ID 7. The new Profile information has the Subscription Address equal to #NEW\_MSISDN | PM\_REQ3, PM\_REQ22 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS parameter is equal to that received in step 5 except that:    1. the free memory of the new Profile is updated (i.e. lower than that received in step 5)    2. the remaining memory is updated (i.e. lower than that received in step 5) | PM\_REQ25, PF\_REQ2, PF\_REQ7 |

###### Test Sequence N°2 - Nominal Case: Using HTTPS

###### Initial Conditions

The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

the #EIS\_RPS SHALL be adapted to indicate that the eUICC does not support CAT\_TP

the capabilities #HTTPS\_CAP\_RPS SHALL be used in the #EIS\_RPS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ( ES2-  DownloadProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS, #EP\_FALSE\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | | **REQ** |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #SUCCESS   1. The ICCID returned is equal to #NEW\_ICCID\_RPS | PROC\_REQ1,P ROC\_REQ2,PR OC\_REQ3, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PF\_REQ2, PF\_REQ3, EUICC\_REQ42 | |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  | |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned contains the new Profile information (i.e. identified by #NEW\_ICCID) 2. The new Profile information has a state equal to Disabled 3. The new Profile information has the SM-DP identifier set to #SM-DP-ID 4. The new Profile information   has an ISD-P RID equal to  #ISD\_P\_RID   1. The new Profile information has an ISD-P PIX that starts with #ISD\_P\_PIX\_PREFIX 2. The new Profile information has a MNO-ID equal to #MNO1\_S\_ID 3. The new Profile information has the Subscription Address equal to #NEW\_MSISDN | PM\_REQ3, PM\_REQ22 | |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS,  #NEW\_ICCID\_RPS) |  |  | |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS parameter is equal to that received in step 5 except that:    1. the free memory of the new Profile is updated (i.e. lower than that received in step 5)    2. the remaining memory is updated (i.e. lower than that received in step 5) | PM\_REQ25, PF\_REQ2, PF\_REQ7 |

###### TC.PROC.DIP.2: DownloadAndInstallAndEnableProfile

###### Test Purpose

*To ensure that the Profile download process followed by the Enable procedure is properly implemented on the SM-DP and the SM-SR. After the Profile download execution, an audit request is sent to the SM-SR to make sure that the Profile has been Enabled. An error case is also described to illustrate the platforms behaviour in case of enabling error.*

###### Referenced Requirements

PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ7, PROC\_REQ20

PM\_REQ4, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PM\_REQ22, PM\_REQ25

PF\_REQ2, PF\_REQ3, PF\_REQ4, PF\_REQ7, PF\_REQ18, PF\_REQ27

EUICC\_REQ27, EUICC\_REQ29

###### Initial Conditions

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The Profile identified by #NEW\_ICCID SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

#MNO2\_S\_ID well known to the SM-SR-UT

#MNO2\_S\_ACCESSPOINT well known to the SM-SR-UT

A direct connection exists between the MNO2-S and the SM-SR-UT

The SMS mode is the default way (priority order 1) to send the notification

###### Test Sequence N°1 - Nominal Case

###### Test Environment

**Device**

ES2-DownloadProfile

ES3-GetEIS

ES3-CreateISDP

ES5-CreateISDP

ES3-CreateISDP

ES3-SendData(ES8-EstablishISDPKeyset)

ES3-SendData(ES8-DownloadAndInstallation)

ES3-ProfileDownloadCompleted

ES3-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification ES5-HandleNotificationConfirmation*

ES3-EnableProfile

ES2-DownloadProfile

*ES4-HandleProfileDisabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

POL1 and POL2 of the Profile identified by #ICCID do not contain any rules

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not required

POL2 MAY be adapted on the #EIS\_RPS

POL1 MAY be adapted in the eUICC

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS}, #NEW\_ICCID\_RPS,  #EP\_TRUE\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #SUCCESS   1. The ICCID returned is equal to #NEW\_ICCID\_RPS | PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ7, PROC\_REQ20, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ14, PM\_REQ16, PM\_REQ17, PM\_REQ18, PF\_REQ2, PF\_REQ3, PF\_REQ4, PF\_REQ18, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileDisabledNo tification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ27, PROC\_REQ7 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS returned contains the new Profile information (i.e. identified by #NEW\_ICCID) 2. The new Profile information has a state equal to Enabled 3. The new Profile information has the SM-DP identifier set to #SM-DP-ID 4. The new Profile information has an ISD-P RID equal to #ISD\_P\_RID 5. The new Profile information has an ISD-P PIX that starts with #ISD\_P\_PIX\_PREFIX 6. The new Profile information has a MNO-ID equal to #MNO1\_S\_ID 7. The new Profile information has the Subscription Address equal to #NEW\_MSISDN | PM\_REQ4, PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:    1. the free memory of the new Profile is updated (i.e. lower than that received in step 6)    2. the remaining memory is updated (i.e. lower than that received in step 6) | PM\_REQ25, PF\_REQ2, PF\_REQ7 |

###### Test Sequence N°2 – Error Case: POL1 Refuses Profile Disabling

###### Test Environment

**Device**

**eUICC**

ES2-DownloadProfile

ES3-GetEIS

ES3-CreateISDP

ES5-CreateISDP

ES3-CreateISDP

ES3-SendData(ES8-EstablishISDPKeyset)

ES3-SendData(ES8-DownloadAndInstallation)

ES3-ProfileDownloadCompleted

ES3-EnableProfile

ES5-EnableProfile

*POL1 refuses disabling*

ES5-EnableProfile

ES3-EnableProfile

ES2-DownloadProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

POL1 of the Profile identified by #ICCID contains the rule “Disabling not Allowed”

POL2 of the Profile identified by #ICCID does not contain any rules

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DownloadProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS, #EP\_TRUE\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DownloadProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_ISDR 2. The Reason code is equal to #RC\_EXECUTION\_ERROR 3. The euiccResponseData is present and contains the POR generated by the eUICC (i.e. SW=’69E1’) | PROC\_REQ1, PROC\_REQ2, PROC\_REQ3, PROC\_REQ8, PM\_REQ8, PM\_REQ9, PM\_REQ11, PM\_REQ12, PM\_REQ17, PM\_REQ18, PF\_REQ2, PF\_REQ3, PF\_REQ4, PF\_REQ18, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS,  {SM\_SR\_ID\_RPS}) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ4, PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 except that:    1. the free memory of the new Profile is updated (i.e. lower than that received in step 5)    2. the remaining memory is updated (i.e. lower than that received in step 5) | PM\_REQ25, PF\_REQ2, PF\_REQ7 |

## Profile Enabling Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ2, PF\_REQ4, PF\_REQ6, PF\_REQ7, PF\_REQ12, PF\_REQ15, PF\_REQ17, PF\_REQ18, PF\_REQ21, PF\_REQ23, PF\_REQ24, PF\_REQ27, PF\_REQ29
        + PROC\_REQ5, PROC\_REQ6, PROC\_REQ7, PROC\_REQ8, PROC\_REQ20
        + PM\_REQ22, PM\_REQ26
        + EUICC\_REQ27, EUICC\_REQ29

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
* A direct connection exists between the MNO1-S and the SM-SR-UT
* #MNO2\_S\_ID well known to the SM-SR-UT
* The Profile identified by #ICCID is owned by MNO2-S and is in Enabled state
* The Profile identified by #NEW\_ICCID is owned by MNO1-S and is in Disabled state
  + To download the new Profile (e.g. #PROFILE\_PACKAGE), the test sequence defined in section [5.3.2.2.1.1](#_bookmark231) MAY be used
* The SM-SR-UT is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)
  + It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s

network related to the default Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)

* The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

* The SMS mode is the default way (priority order 1) to send the notification

Note: To facilitate the execution of the test cases, the default Enabled Profile and the Profile to be Enabled MAY use the same Connectivity Parameters (i.e. the two Profiles are linked to the same MNO’s network).

###### TC.PROC.PE.1: ProfileEnablingByMNO

###### Test Purpose

*To ensure a Profile can be Enabled by the SM-SR when the MNO requests it, different Policy Rules are used and an error case, using bad Connectivity Parameters, is described to make sure that the roll-back process is well implemented. In case of a successful enabling process, an audit request is sent to the SM-SR to make sure that the Profile has been Enabled.*

###### Referenced Requirements

* PF\_REQ2, PF\_REQ4, PF\_REQ6, PF\_REQ7, PF\_REQ24, PF\_REQ27, PF\_REQ29
* PROC\_REQ5, PROC\_REQ6, PROC\_REQ20
* PM\_REQ22, PM\_REQ26
* EUICC\_REQ27, EUICC\_REQ29

###### Initial Conditions

* #MNO2\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO2-S and the SM-SR-UT

###### Test Sequence N°1 – Nominal Case: Empty POL1 and POL2

###### Test Environment

**Device**

ES4-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES4-EnableProfile

*ES4-HandleProfileDisabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 and POL2 of the Profile identified by #ICCID do not contain any rules and MAY need to be adapted on the #EIS\_RPS and in the eUICC as follow:

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ4, PF\_REQ24, PROC\_REQ5, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileDisabledNo tification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ27, PROC\_REQ5 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 - Nominal Case: POL1 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES4-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES4-EnableProfile

*ES4-HandleProfileDeletedNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 of the Profile identified by #ICCID contains only the rule “Delete when Disabling” (POL1 MAY need to be adapted on the eUICC)

POL2 of the Profile identified by #ICCID does not contain any rules (POL2 MAY need to be adapted on the #EIS\_RPS)

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL1 | PF\_REQ2, PF\_REQ4, PF\_REQ24, PROC\_REQ5, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileDeletedNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ29, PROC\_REQ5 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 - Nominal Case: POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES4-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES5-DeleteProfile

ES4-EnableProfile

*ES4-HandleProfileDeletedNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s

network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 of the Profile identified by #ICCID does not contain any rules (POL1 MAY need to be adapted on the eUICC)

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

POL2 of the Profile identified by #ICCID contains only the rule “Profile deletion is mandatory when it is disabled” (POL2 MAY need to be adapted on the #EIS\_RPS)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL2 | PF\_REQ2, PF\_REQ4, PF\_REQ6, PF\_REQ24, PROC\_REQ5, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileDeletedNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ29, PROC\_REQ5 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°4 – Error Case: Bad Connectivity Parameters

###### Test Environment

**Device**

ES4-EnableProfile

ES5-EnableProfile

*Network attachment fails*

*Enable #ICCID Profile ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES4-EnableProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain inconsistent Connectivity Parameters (e.g. #NAN\_VALUE, #LOGIN, #PWD)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-EnableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** | |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_INACCESSIBLE | PF\_REQ2, PF\_REQ4, PF\_REQ24, PROC\_REQ6, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 | |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  | |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 | |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  | |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### TC.PROC.PE.2: ProfileEnablingViaSMDP

###### Test Purpose

*To ensure a Profile can be Enabled by the SM-DP and the SM-SR when the MNO requests it, different Policy Rules are used and an error case, using bad Connectivity Parameters, is described to make sure that the roll-back process is well implemented. In case of successful enabling process, an audit request is sent to the SM-SR to make sure that the Profile has been Enabled.*

###### Referenced Requirements

PF\_REQ2, PF\_REQ4, PF\_REQ6, PF\_REQ7, PF\_REQ12, PF\_REQ15, PF\_REQ17, PF\_REQ18, PF\_REQ21, PF\_REQ23

PROC\_REQ7, PROC\_REQ8, PROC\_REQ20

PM\_REQ22, PM\_REQ26

EUICC\_REQ27, EUICC\_REQ29

###### Initial Conditions

#MNO2\_S\_ACCESSPOINT is unknown to the SM-SR-UT

#MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT

#MNO2\_S\_ID and #MNO2\_S\_ACCESSPOINT well known to the SM-DP-UT

The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS

#SM\_SR\_ID and #SM\_SR\_ACCESSPOINT well known to the SM-DP-UT

#SM\_DP\_ID and #SM\_DP\_ACCESSPOINT well known to the SM-SR-UT

The Profile identified by #ICCID is linked to the SM-DP identified by #SM\_DP\_ID

(the #EIS\_RPS MAY need to be adapted on the SM-SR-UT)

###### Test Sequence N°1 – Nominal Case: Empty POL1 and POL2

###### Test Environment

**Device**

ES2-EnableProfile

ES3-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES3-EnableProfile

ES2-EnableProfile

**MNO2-S**

*ES3-HandleProfileDisabledNotification*

*ES2-HandleProfileDisabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 and POL2 of the Profile identified by #ICCID do not contain any rules and MAY need to be adapted on the #EIS\_RPS and in the eUICC as follow:

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
|  |  |  | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ4, PF\_REQ12, PF\_REQ18, PF\_REQ21, PROC\_REQ7, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
|  |  | Send the |  |
| 3 | SM-DP-UT → MNO1-S | ES2-EnableProfile |  |
|  |  | response |  |
|  |  | Send the | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ15, PROC\_REQ7 |
| 4 | SM-DP-UT → MNO2-S | ES2-  HandleProfileDisabledNo tification |  |
|  |  | notification |  |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 – Nominal Case: POL1 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES2-EnableProfile

ES3-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES3-EnableProfile

ES2-EnableProfile

**MNO2-S**

*ES3-HandleProfileDeletedNotification*

*ES2-HandleProfileDeletedNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 of the Profile identified by #ICCID contains only the rule “Profile deletion is mandatory when it is disabled” (POL1 MAY need to be adapted on the eUICC)

POL2 of the Profile identified by #ICCID does not contain any rules (POL2 MAY need to be adapted on the #EIS\_RPS)

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL1 | PF\_REQ2, PF\_REQ4, PF\_REQ12, PF\_REQ18, PF\_REQ23, PROC\_REQ7, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-DP-UT → MNO2-S | Send the  ES2-  HandleProfileDeletedNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ17, PROC\_REQ7 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 – Nominal Case: POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

###### Initial Conditions

**Device**

ES2-EnableProfile

ES3-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES5-DeleteProfile

ES3-EnableProfile

ES2-EnableProfile

**MNO2-S**

*ES3-HandleProfileDeletedNotification*

*ES2-HandleProfileDeletedNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain correct Connectivity Parameters (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

The SM-SR-UT is able to communicate with the network linked to the new Profile of the eUICC (identified by #NEW\_ICCID)

It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the new Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

POL1 of the Profile identified by #ICCID does not contain any rules (POL1 MAY need to be adapted on the eUICC)

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

POL2 of the Profile identified by #ICCID contains only the rule “Profile deletion is mandatory when it is disabled” (POL2 MAY need to be adapted on the #EIS\_RPS)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL1 | PF\_REQ2, PF\_REQ4, PF\_REQ6, PF\_REQ12, PF\_REQ18, PF\_REQ23, PROC\_REQ7, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-DP-UT → MNO2-S | Send the  ES2-  HandleProfileDeletedNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ17, PROC\_REQ7 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°4 – Error Case: Bad Connectivity Parameters

###### Test Environment

**Device**

ES2-EnableProfile

ES3-EnableProfile

ES5-EnableProfile

*Network attachment fails*

*Enable #ICCID Profile*

*ES5-HandleDefaultNotification*

ES3-EnableProfile

*ES5-HandleNotificationConfirmation*

ES2-EnableProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The Profile downloaded, identified by #NEW\_ICCID, SHALL be adapted to contain inconsistent Connectivity Parameters (e.g. #NAN\_VALUE, #LOGIN, #PWD)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-EnableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-EnableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_INACCESSIBLE | PF\_REQ2, PF\_REQ4, PF\_REQ12, PF\_REQ18, PROC\_REQ8, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO1-S are present among which that identified by #NEW\_ICCID   3. the Profile identified by   #ICCID is not present   * 1. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

## Profile Disabling Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ2, PF\_REQ5, PF\_REQ6, PF\_REQ7, PF\_REQ13, PF\_REQ16, PF\_REQ19, PF\_REQ22, PF\_REQ25, PF\_REQ28
        + PROC\_REQ9, PROC\_REQ10, PROC\_REQ20
        + PM\_REQ22, PM\_REQ26
        + EUICC\_REQ27, EUICC\_REQ29

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
* A direct connection exists between the MNO1-S and the SM-SR-UT
* #MNO2\_S\_ID well known to the SM-SR-UT
* The Profile identified by #ICCID is owned by MNO2-S, is in Disabled state and has the Fall-back Attribute
  + The Profile MAY need to be adapted to have the Fall-back Attribute
* The Profile identified by #NEW\_ICCID is owned by MNO1-S and is in Enabled state
  + To Enable the new Profile (e.g. #PROFILE\_PACKAGE), the test sequence defined in section [5.3.3.2.1.1](#_bookmark235) MAY be used
* The SM-SR-UT is able to communicate with the network linked to the Enabled Profile (identified by #NEW\_ICCID)
  + It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s

network related to the Enabled Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

* The SM-SR-UT is able to communicate with the network linked to the Profile with the Fall-back Attribute (identified by #ICCID)
  + It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s

network related to the Profile with the Fall-back Attribute (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)

* The eUICC identified by #EID has been provisioned on the SM-SR-UT using the

#EIS\_RPS

* The SMS mode is the default way (priority order 1) to send the notification

Note: To facilitate the execution of the test cases, the Profile with the Fall-back Attribute and the Profile to be Disabled MAY use the same Connectivity Parameters (i.e. the two Profiles are linked to the same MNO’s network).

###### TC.PROC.DIS.1: ProfileDisablingByMNO

###### Test Purpose

*To ensure a Profile can be Disabled by the SM-SR when the MNO requests it, different Policy Rules are used. After the Profile disabling, an audit request is sent to the SM-SR to make sure that the Profile has been Disabled. Some error cases are also described:*

* *the Profile with the Fall-back Attribute contains bad Connectivity Parameters*
* *the Profile to be Disabled contains the POL1 “Disabling not Allowed”*

###### Referenced Requirements

* PF\_REQ2, PF\_REQ5, PF\_REQ6, PF\_REQ7, PF\_REQ25, PF\_REQ28
* PROC\_REQ9, PROC\_REQ20
* PM\_REQ22, PM\_REQ26
* EUICC\_REQ27, EUICC\_REQ29

###### Initial Conditions

* #MNO2\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO2-S and the SM-SR-UT

###### Test Sequence N°1 - Nominal Case: Empty POL1 and POL2

###### Test Environment

**Device**

ES4-DisableProfile

ES5-DisableProfile

*ES5*-*HandleDefaultNotification*

*ES5*-*HandleNotificationConfirmation*

ES4-DisableProfile

**MNO2-S**

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

POL1 and POL2 of the Profile identified by #NEW\_ICCID do not contain any rules

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ5, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9, PROC\_REQ20 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ28, PROC\_REQ9 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 - Nominal Case: POL1 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES4-DisableProfile

ES5-DisableProfile

*ES5*-*HandleDefaultNotification*

*ES5*-*HandleNotificationConfirmation*

ES4-DisableProfile

*ES4-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID contains the rule “Profile deletion is mandatory when it is disabled”

POL2 of the Profile identified by #NEW\_ICCID allows disabling

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_OBJ\_EXIST | PF\_REQ2, PF\_REQ5, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9, PROC\_REQ20 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ28, PROC\_REQ9 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 - Nominal Case: POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES4-DisableProfile

ES5-DisableProfile

*ES5*-*HandleDefaultNotification ES5*-*HandleNotificationConfirmation*

ES5-DeleteProfile

ES4-DisableProfile

*ES4-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**SM-SR-UT**

**MNO1-S**

**eUICC**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID does not contain any rules

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

POL2 of the Profile identified by #NEW\_ICCID contains the rule “Profile deletion is mandatory when it is disabled”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile,  #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_OBJ\_EXIST | PF\_REQ2, PF\_REQ5, PF\_REQ6, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9, PROC\_REQ20 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ28, PROC\_REQ9 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°4 - Nominal Case: POL1 with “Deletion not Allowed” and POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment



**Device**

ES4-DisableProfile

ES5-DisableProfile

*ES5*-*HandleDefaultNotification*

*ES5*-*HandleNotificationConfirmation*

ES5-DeleteProfile

ES4-DisableProfile

fails

*ES4-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID forbids deletion

Disabling of the Profile is allowed

Deletion of the Profile is not allowed

POL2 of the Profile identified by #NEW\_ICCID contains the rule “Profile deletion is mandatory when it is disabled”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | The Status is equal to  #SUCCESS (see Note1) | PF\_REQ2, PF\_REQ5, PF\_REQ6, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9, PROC\_REQ20 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES4-  HandleProfileEnabledNo tification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ28, PROC\_REQ9 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |
| *Note 1: Even if a DELETE command is sent by the SM-SR and fails (because of POL1), the status of the disabling process SHALL be successful.* | | | | |

###### Test Sequence N°5 - Error Case: Bad Connectivity Parameters

###### Test Environment

**Device**

ES4-DisableProfile

ES5-DisableProfile

*Network attachment fails*

*Enable #NEW\_ICCID Profile*

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES4-DisableProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

The Profile, identified by #ICCID, SHALL be adapted to contain inconsistent Connectivity Parameters (e.g. #NAN\_VALUE, #LOGIN, #PWD)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_INACCESSIBLE | PF\_REQ2, PF\_REQ5, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9, PROC\_REQ20 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°6 - Error Case: POL1 with “Disabling not Allowed”

###### Test Environment

**Device**

ES4-DisableProfile

ES5-DisableProfile

ES4-DisableProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID contains the rule “Disabling not Allowed”

POL2 of the Profile identified by #NEW\_ICCID does not contain any rules

Disabling of the Profile is allowed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DisableProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_REFUSED 3. The euiccResponseData is present and contains the POR generated by the eUICC (i.e. SW=’69E1’) | PF\_REQ2, PF\_REQ5, PF\_REQ25, PF\_REQ28, EUICC\_REQ27, EUICC\_REQ29, PROC\_REQ9 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### TC.PROC.DIS.2: ProfileDisablingViaSMDP

###### Test Purpose

*To ensure a Profile can be Disabled by the SM-DP and the SM-SR when the MNO requests it. After the Profile disabling, an audit request is sent to the SM-SR to make sure that the Profile has been Disabled. An error case is also described:*

*the Profile with the Fall-back Attribute contains bad Connectivity Parameters*

###### Referenced Requirements

PF\_REQ2, PF\_REQ5, PF\_REQ7, PF\_REQ13, PF\_REQ16, PF\_REQ19, PF\_REQ22

PROC\_REQ10, PROC\_REQ20

PM\_REQ22, PM\_REQ26

EUICC\_REQ27, EUICC\_REQ29

###### Initial Conditions

#MNO2\_S\_ACCESSPOINT is unknown to the SM-SR-UT

#MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT

#MNO2\_S\_ID and #MNO2\_S\_ACCESSPOINT well known to the SM-DP-UT

The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS

#SM\_SR\_ID and #SM\_SR\_ACCESSPOINT well known to the SM-DP-UT

#SM\_DP\_ID and #SM\_DP\_ACCESSPOINT well known to the SM-SR-UT

The Profile identified by #ICCID is linked to the SM-DP identified by #SM\_DP\_ID

(the #EIS\_RPS MAY need to be adapted on the SM-SR-UT)

###### Test Sequence N°1 – Nominal Case: Empty POL1 and POL2

###### Test Environment

**Device**

ES2-DisableProfile

ES3-DisableProfile

ES5-DisableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES2-DisableProfile ES3-DisableProfile

*ES3-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

POL1 and POL2 of the Profile identified by #NEW\_ICCID do not contain any rules

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | The Status is equal to  #SUCCESS | PF\_REQ5, PF\_REQ13, PF\_REQ19, PF\_REQ22, PROC\_REQ10, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES2-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ16, PROC\_REQ10 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 – Nominal Case: POL1 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES2-DisableProfile

ES3-DisableProfile

ES5-DisableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES2-DisableProfile ES3-DisableProfile

*ES3-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID contains the rule “Profile deletion is mandatory when it is disabled”

POL2 of the Profile identified by #NEW\_ICCID allows disabling

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_OBJ\_EXIST | PF\_REQ5, PF\_REQ13, PF\_REQ19, PF\_REQ22, PROC\_REQ10, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES2-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ16, PROC\_REQ10 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 – Nominal Case: POL2 with “Profile Deletion is Mandatory when it is Disabled”

###### Test Environment

**Device**

ES2-DisableProfile

ES3-DisableProfile

ES5-DisableProfile

*ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES5-DeleteProfile

ES2-DisableProfile ES3-DisableProfile

*ES3-HandleProfileEnabledNotification*

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**MNO2-S**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID does not contain any rules

Disabling of the Profile is allowed

“Profile deletion is mandatory when it is disabled” is not set

POL2 of the Profile identified by #NEW\_ICCID contains the rule “Profile deletion is mandatory when it is disabled”

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_POL2 2. The Reason code is equal to #RC\_OBJ\_EXIST | PF\_REQ5, PF\_REQ13, PF\_REQ19, PF\_REQ22, PROC\_REQ10, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |
| 4 | SM-SR-UT → MNO2-S | Send the  ES2-  HandleProfileEnabledNot ification  notification | 1. The EID parameter is equal to #EID\_RPS 2. The ICCID is equal to   #ICCID\_RPS   1. The completion timestamp is present | PF\_REQ16, PROC\_REQ10 |
| 5 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 6 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 |
| 7 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 6 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 6) | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°4 – Error Case: Bad Connectivity Parameters

###### Test Environment

**Device**

ES2-DisableProfile

ES3-DisableProfile

ES5-DisableProfile

*Network attachment fails*

*Enable #NEW\_ICCID Profile ES5-HandleDefaultNotification*

*ES5-HandleNotificationConfirmation*

ES3-DisableProfile

ES2-DisableProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Initial Conditions

The Profile, identified by #ICCID, SHALL be adapted to contain inconsistent Connectivity Parameters (e.g. #NAN\_VALUE, #LOGIN, #PWD)

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DisableProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DisableProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EUICC 2. The Reason code is equal to #RC\_INACCESSIBLE | PF\_REQ5, PF\_REQ13, PF\_REQ19, PF\_REQ22, PROC\_REQ10, PROC\_REQ20, EUICC\_REQ27, EUICC\_REQ29 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Enabled | PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

## Profile Deletion Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ2, PF\_REQ6, PF\_REQ7, PF\_REQ14, PF\_REQ20, PF\_REQ26
        + PROC\_REQ11, PROC\_REQ12
        + PM\_REQ22, PM\_REQ26

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO1\_S\_ACCESSPOINT well known to the SM-SR-UT
* A direct connection exists between the MNO1-S and the SM-SR-UT
* #MNO2\_S\_ID well known to the SM-SR-UT
* The Profile identified by #ICCID is owned by MNO2-S and is in Enabled state
* The Profile identified by #NEW\_ICCID is owned by MNO1-S and is in Disabled state
  + To download the new Profile (e.g. #PROFILE\_PACKAGE), the test sequence defined in section [5.3.2.2.1.1](#_bookmark231) MAY be used
* The SM-SR-UT is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)
  + It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the default Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)
* The eUICC identified by #EID has been initially provisioned on the SM-SR-UT using the #EIS\_RPS

###### TC.PROC.DEL.1: ProfileDeletionByMNO

###### Test Purpose

*To ensure a Profile can be deleted by the SM-SR when the MNO requests it. After the Profile deletion, an audit request is sent to the SM-SR to make sure that the Profile has been deleted. An error case with a POL1 defined with “Deletion not allowed” is also described.*

###### Test Environment

**Device**

ES4-DeleteProfile

ES5-DeleteProfile

ES4-DeleteProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**MNO1-S**

###### Referenced Requirements

* PF\_REQ2, PF\_REQ6, PF\_REQ7, PF\_REQ26
* PROC\_REQ11
* PM\_REQ22, PM\_REQ26

###### Initial Conditions

* The Profile identified by #ICCID is the Profile with the Fall-back Attribute

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

POL1 and POL2 of the Profile identified by #NEW\_ICCID do not contain any rules

Deletion of the Profile is allowed

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** | |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  | |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ6, PF\_REQ26, PROC\_REQ11 | |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  | |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 | |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  | |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 except that:   a. the remaining memory is updated (i.e. bigger than that received in step 5) | | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 - Error Case: POL1 with “Deletion not Allowed”

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID contains the rule “Deletion not Allowed”

POL2 of the Profile identified by #NEW\_ICCID does not contain any rules

Deletion of the Profile is allowed

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** | |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  | |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 3 | SM-SR-UT → MNO1-S | Send the  ES4-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_REFUSED 3. The euiccResponseData is present and contains the POR generated by the eUICC (i.e. SW=’69E1’) | PF\_REQ2, PF\_REQ6, PF\_REQ26, PROC\_REQ11 | |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  | |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 | |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  | |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 - Error Case: ISD-P not present on the eUICC

###### Initial Conditions

The Profile identified by #NEW\_ICCID is no more present in the eUICC (even though it is present in the EIS known to the SM-SR-UT)

POL2 of the Profile identified by #NEW\_ICCID do not contain any rules in the EIS

Deletion of the Profile is allowed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-DeleteProfile, #EID\_RPS, #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
|  |  |  | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_ISDP 2. The Reason code is equal to #RC\_NOT\_PRESENT 3. The euiccResponseData MAY be present. If any, it SHALL contain the POR generated by the eUICC (i.e. SW=’6A88’ or SW=’6A82’) | PF\_REQ2, PF\_REQ6, PF\_REQ26, PROC\_REQ11 |
|  |  | Send the |  |
| 3 | SM-SR-UT → MNO1-S | ES4-DeleteProfile |  |
|  |  | response |  |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is no more present | PM\_REQ22 |

###### TC.PROC.DEL.1: ProfileDeletionViaSMDP

###### Test Purpose

*To ensure a Profile can be deleted by the SM-DP and the SM-SR when the MNO requests it. After the Profile deletion, an audit request is sent to the SM-SR to make sure that the Profile has been deleted. An error case with a POL1 defined with “Deletion not allowed” is also described.*

###### Test Environment

**Device**

ES2-DeleteProfile

ES3-DeleteISDP

ES5-DeleteProfile

ES3-DeleteISDP

ES2-DeleteProfile

ES4-GetEIS

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

**eUICC**

**SM-SR-UT**

**SM-DP-UT**

**MNO1-S**

###### Referenced Requirements

PF\_REQ2, PF\_REQ6, PF\_REQ7, PF\_REQ14, PF\_REQ20

PROC\_REQ12

PM\_REQ22, PM\_REQ26

###### Initial Conditions

#MNO1\_S\_ID and #MNO1\_S\_ACCESSPOINT well known to the SM-DP-UT

The variable {SM\_SR\_ID\_RPS} SHALL be set to #SM\_SR\_UT\_ID\_RPS

#SM\_SR\_ID and #SM\_SR\_ACCESSPOINT well known to the SM-DP-UT

###### Test Sequence N°1 - Nominal Case

###### Initial Conditions

POL1 and POL2 of the Profile identified by #NEW\_ICCID do not contain any rules

Deletion of the Profile is allowed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ6, PF\_REQ14, PF\_REQ20, PROC\_REQ12 | |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  | |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is not present | PM\_REQ22 | |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  | |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 except:   a. the remaining memory is updated (i.e. bigger than that received in step 5) | | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°2 - Error Case: POL1 with “Deletion not Allowed”

###### Initial Conditions

POL1 of the Profile identified by #NEW\_ICCID contains the rule “Deletion not Allowed”

POL2 of the Profile identified by #NEW\_ICCID does not contain any rules

* Deletion of the Profile is allowed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_POL1 2. The Reason code is equal to #RC\_REFUSED 3. The euiccResponseData   is not present | PF\_REQ2, PF\_REQ6, PF\_REQ14, PF\_REQ20, PROC\_REQ12 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by   #NEW\_ICCID is  Disabled | PM\_REQ22 |
| 6 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → MNO1-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to that received in step 5 | PF\_REQ2, PF\_REQ7, PM\_REQ26 |

###### Test Sequence N°3 - Error Case: ISD-P not present on the eUICC

###### Initial Conditions

The Profile identified by #NEW\_ICCID is no more present in the eUICC (even though it is present in the EIS known to the SM-SR-UT)

POL2 of the Profile identified by #NEW\_ICCID do not contain any rules in the EIS

* Deletion of the Profile is allowed

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO1-S → SM-DP-UT | SEND\_REQ(  ES2-DeleteProfile, #EID\_RPS,  {SM\_SR\_ID\_RPS},  #NEW\_ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-DP-UT → MNO1-S | Send the  ES2-DeleteProfile  response | 1. The Status is equal to   #WARNING   1. The Subject code is equal to #SC\_ISDP 2. The Reason code is equal to #RC\_NOT\_PRESENT 3. The euiccResponseData   is not present | PF\_REQ2, PF\_REQ6, PF\_REQ14, PF\_REQ20, PROC\_REQ12 |
| 4 | MNO1-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 5 | SM-SR-UT → MNO1-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. the Profile identified by #NEW\_ICCID is no more present | PM\_REQ22 |

## Master Delete Process

|  |  |
| --- | --- |
|  | As no interface is defined between the MNO, the SM-DP and the SM-SR in the GSMA Remote Provisioning Architecture for Embedded UICC - Technical  Specification [[2]](#_bookmark7), this section is FFS. Only test cases that allow testing the eUICC are defined (see section [4.2.9](#_bookmark92)). |

## SM-SR Change Process

###### Conformance Requirements

###### References

* + - * + GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6)
        + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PF\_REQ2, PF\_REQ7
        + EUICC\_REQ24, EUICC\_REQ25, EUICC\_REQ33, EUICC\_REQ34, EUICC\_REQ35, EUICC\_REQ36, EUICC\_REQ37, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40
        + PM\_REQ22, PM\_REQ25
        + PROC\_REQ13
        + SEC\_REQ19

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO2\_S\_ID well known to the SM-SR-UT
        + The Profile identified by #ICCID is owned by MNO2-S and is in Enabled state
        + The SM-SR-UT is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)
* It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s network related to the default Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)

###### TC.PROC.SMSRCH.1: SMSRChange

###### Test Purpose

*To ensure the SM-SR can be changed when the MNO requests it. In this test case, the switch is from the SM-SR-UT to the SM-SR-S.*

###### Test Environment

**MNO2-S**

**SM-DP-UT**

**SM-SR-UT**

**SM-SR-S**

|  |  |  |
| --- | --- | --- |
| **Device** | | |
|  | **eUICC** |  |

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC

ES7-AuthenticateSMSR

ES5-EstablishISDRKeySet

ES7-AuthenticateSMSR

ES7-CreateAdditionalKeyset

ES5-EstablishISDRKeySet

ES7-CreateAdditionalKeyset

ES5-FinaliseISDRhandover

ES7-HandoverEUICC

ES4-SMSRChange

*ES3-HandleSMSRChangeNotification*

*ES2-HandleSMSRChangeNotification*

ES4-GetEIS

Note that the functions ES4-PrepareSMSRChange and ES5-FinaliseISDRhandover SHALL NOT be performed by the simulators (in the schema above, they are only informative messages).

In this test case, the Initiator Role (see GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) section 2.3.1) is assumed to be played by the MNO2-S.

###### Referenced Requirements

* + - * + PF\_REQ2
        + EUICC\_REQ24, EUICC\_REQ33, EUICC\_REQ34, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40
        + PM\_REQ22
        + PROC\_REQ13
        + SEC\_REQ19

###### Initial Conditions

* + - * + #MNO2\_S\_ACCESSPOINT is unknown to the SM-SR-UT
        + #MNO2\_S\_ID and #MNO2\_S\_ACCESSPOINT well known to the SM-DP-UT
        + The eUICC identified by #EID has been initially provisioned on the SM-SR-UT using the #EIS\_RPS
        + All Profiles present in the #EIS\_RPS SHALL contain an smdp-id equal to

#SM\_DP\_ID

* + - * + All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)

###### Test Sequence N°1 – Nominal Case: No DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_RPS except that  the ISD-R keys values are empty | EUICC\_REQ36, EUICC\_REQ39, PROC\_REQ13 |
| 3 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR, #EID\_RPS, #VALID\_SR\_CERTIF\_RPS) |  |  |
| 4 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 5 | SM-SR-UT → SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #SUCCESS   1. The Random Challenge is present (i.e. {RC}) | PF\_REQ2, EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13 |
| 6 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-CreateAdditionalKeyset, #EID\_RPS,  #KEY\_VERSION\_RPS, #INIT\_SEQ\_COUNTER\_RPS, #ECC\_KEY\_LENGTH\_RPS, #SC3\_NO\_DR\_RPS, #EPHEMERAL\_PK\_RPS,  #SIGNATURE\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 8 | SM-SR-UT → SM-SR-S | Send the  ES7-CreateAdditionalKeyset  response | 1. The Status is equal to   #SUCCESS   1. The derivation random is not present 2. The receipt (i.e.   {RECEIPT}) is present 4- Calculate ShS from  #SM\_ESK\_ECKA and  #PK\_ECASD\_ECKA   1. Derive keyset from ShS and retrieve the   {SCP\_KENC},  {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tag ‘A6’) | PF\_REQ2, EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, PROC\_REQ13 |
| 9 | SM-SR-S → SM-SR-UT | SEND\_SUCCESS\_RESP(  ES7-HandoverEUICC) |  |  |
| 10 | SM-SR-UT → MNO2-S | Send the  ES4-SMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ36, PROC\_REQ13 |
| 11 | SM-SR-S → SM-DP-UT | SEND\_NOTIF( ES3-  HandleSMSRChangeNotification, #EIS\_RPS,  #TIMESTAMP\_RPS)  Note: The #EIS\_RPS shall:   * not contain the ISD-R keysets * At most contain Profiles related to the #SM\_DP\_ID |  |  |
| 12 | SM-DP-UT → MNO2-S | Send the  ES2-  HandleSMSRChangeNotification  notification | 1. The EIS parameter is equal to #EIS\_RPS except that:    1. The ISD-R information is not provided    2. At most Profiles owned by the MNO2-Sare present 2. The completion timestamp is equal to #TIMESTAMP\_RPS | EUICC\_REQ33, EUICC\_REQ34, PROC\_REQ13 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 13 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 14 | SM-SR-UT → MNO2-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_ID\_UNKNOWN | PM\_REQ22, SEC\_REQ19 |

###### Test Sequence N°2 – Nominal Case: DR, No Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to #EIS\_RPS except that the ISD-R keys values are empty | EUICC\_REQ36, EUICC\_REQ39, PROC\_REQ13 |
| 3 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR, #EID\_RPS, #VALID\_SR\_CERTIF\_RPS) |  |  |
| 4 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 5 | SM-SR-UT → SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #SUCCESS   1. The Random Challenge is present (i.e. {RC}) | EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 6 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-CreateAdditionalKeyset, #EID\_RPS,  #KEY\_VERSION\_RPS, #INIT\_SEQ\_COUNTER\_RPS, #ECC\_KEY\_LENGTH\_RPS, #SC3\_DR\_RPS, #EPHEMERAL\_PK\_RPS,  #SIGNATURE\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → SM-SR-S | Send the  ES7-CreateAdditionalKeyset  response | 1. The Status is equal to   #SUCCESS   1. The derivation random is present (i.e.   {DR})   1. The receipt (i.e.   {RECEIPT}) is present 4- Calculate ShS from  #SM\_ESK\_ECKA and  #PK\_ECASD\_ECKA   1. Derive keyset from ShS and {DR} and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, PROC\_REQ13 |
| 9 | SM-SR-S → SM-SR-UT | SEND\_SUCCESS\_RESP(  ES7-HandoverEUICC) |  |  |
| 10 | SM-SR-UT → MNO2-S | Send the  ES4-SMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ36, PROC\_REQ13 |
| 11 | SM-SR-S → SM-DP-UT | SEND\_NOTIF( ES3-  HandleSMSRChangeNotification, #EIS\_RPS,  #TIMESTAMP\_RPS)  Note: The #EIS\_RPS shall:   * Not contain the ISD-R keysets * At most contain Profiles related to   #SM\_DP\_ID |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 12 | SM-DP-UT → MNO2-S | Send the  ES2-  HandleSMSRChangeNotification  notification | 1. The EIS parameter is equal to #EIS\_RPS except that:    1. The ISD-R information is not provided    2. At most Profiles owned by the MNO2- S are present 2. The completion timestamp is equal to #TIMESTAMP\_RPS | EUICC\_REQ33, EUICC\_REQ34, PROC\_REQ13 |
| 13 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 14 | SM-SR-UT → MNO2-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_ID\_UNKNOWN | PM\_REQ22, SEC\_REQ19 |

###### Test Sequence N°3 – Nominal Case: DR, Host ID

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to #EIS\_RPS except that  the ISD-R keys values are empty | EUICC\_REQ36, EUICC\_REQ39, PROC\_REQ13 |
| 3 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR,  #EID\_RPS, #VALID\_SR\_CERTIF\_RPS) |  |  |
| 4 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 5 | SM-SR-UT → SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #SUCCESS   1. The Random Challenge is present (i.e. {RC}) | EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13 |
| 6 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-CreateAdditionalKeyset, #EID\_RPS,  #KEY\_VERSION\_RPS, #INIT\_SEQ\_COUNTER\_RPS, #ECC\_KEY\_LENGTH\_RPS, #SC3\_DR\_HOST\_RPS, #HOST\_ID\_RPS, #EPHEMERAL\_PK\_RPS,  #SIGNATURE\_RPS) |  |  |
| 7 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 8 | SM-SR-UT → SM-SR-S | Send the  ES7-CreateAdditionalKeyset  response | 1. The Status is equal to   #SUCCESS   1. The derivation random is present (i.e.   {DR})   1. The receipt (i.e.   {RECEIPT}) is present 4- Calculate ShS from  #SM\_ESK\_ECKA and  #PK\_ECASD\_ECKA   1. Derive keyset from ShS and {DR} and retrieve the {SCP\_KENC},   {SCP\_KMAC} and  {SCP\_KDEK}   1. Verify the {RECEIPT} (i.e. it SHALL be generated by calculating a MAC across the tags ‘A6’ and ‘85’) | EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, PROC\_REQ13 |
| 9 | SM-SR-S → SM-SR-UT | SEND\_SUCCESS\_RESP(  ES7-HandoverEUICC) |  |  |
| 10 | SM-SR-UT → MNO2-S | Send the  ES4-SMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ36, PROC\_REQ13 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 11 | SM-SR-S → SM-DP-UT | SEND\_NOTIF( ES3-  HandleSMSRChangeNotification, #EIS\_RPS,  #TIMESTAMP\_RPS)  Note: The #EIS\_RPS shall:   * Not contain the ISD-R keysets * At most contain Profiles related to   #SM\_DP\_ID |  |  |
| 12 | SM-DP-UT → MNO2-S | Send the  ES2-  HandleSMSRChangeNotification  notification | 1. The EIS parameter is equal to #EIS\_RPS except that:    1. The ISD-R information is not provided    2. At most Profiles owned by the MNO2-S are present 2. The completion timestamp is equal to #TIMESTAMP\_RPS | EUICC\_REQ33, EUICC\_REQ34, PROC\_REQ13 |
| 13 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 14 | SM-SR-UT → MNO2-S | Send the  ES4-GetEIS  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_EID 2. The Reason code is equal to #RC\_ID\_UNKNOWN | PM\_REQ22, SEC\_REQ19 |

###### TC.PROC.SMSRCH.2: SMSRChange

###### Test Purpose

*To ensure the SM-SR can be changed when the MNO requests it. In this test case, the switch is from the SM-SR-TP to SM-SR-UT.*

###### Test Environment

**Device**

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC ES7-AuthenticateSMSR

ES5-EstablishISDRKeySet

ES7-AuthenticateSMSR

ES7-CreateAdditionalKeyset

ES5-EstablishISDRKeySet

ES7-CreateAdditionalKeyset

ES5-FinaliseISDRhandover

ES7-HandoverEUICC

ES4-SMSRChange

*ES4-HandleSMSRChangeNotification*

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

ES4-EnableProfile

ES5-EnableProfile

*ES5-HandleDefaultNotification*

ES4-EnableProfile

*ES5-HandleNotificationConfirmation*

**MNO1-S**

**eUICC**

**SM-SR-UT**

**MNO2-S**

**SM-SR-TP**

In this test case, the Initiator Role (see GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) section 2.3.1) is assumed to be played by the MNO2-S.

Note: To facilitate the execution of the test cases, the default Enabled Profile and the Profile to be Enabled MAY use the same Connectivity Parameters (i.e. the two Profiles are linked to the same MNO’s network).

###### Referenced Requirements

* PF\_REQ2, PF\_REQ7
* EUICC\_REQ25, EUICC\_REQ35, EUICC\_REQ36, EUICC\_REQ37, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40
* PM\_REQ25
* PROC\_REQ13

###### Initial Conditions

* #MNO1\_S\_ID well known to the SM-SR-TP
* #MNO2\_S\_ID well known to the SM-SR-TP
* #MNO2\_S\_ACCESSPOINT well known to the SM-SR-UT
  + A direct connection exists between the MNO2-S and the SM-SR-UT
* The eUICC identified by #EID has been initially provisioned on the SM-SR-TP using the #EIS\_RPS
* All Profiles present in the #EIS\_RPS SHALL NOT contain any smdp-id
* The SM-SR-TP is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)
  + It means that the SM-SR-TP knows the Connectivity Parameters of the MNO’s network related to the default Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)
* All necessary settings have been initialized on SM-SR-TP to accept the SM-SR change (i.e. business agreement…)
* The Profile identified by #NEW\_ICCID is owned by MNO1-S and is in Disabled state
  + To download the new Profile (e.g. #PROFILE\_PACKAGE), the test sequence defined in section [5.3.2.2.1.1](#_bookmark231) MAY be used
* POL1 and POL2 of the Profile identified by #ICCID do not contain any rules and MAY need to be adapted on the #EIS\_RPS and in the eUICC as follow:
  + Disabling of the Profile is allowed
  + “Profile deletion is mandatory when it is disabled” is not set
* The SM-SR-UT is able to communicate with the network linked to the Profile identified by #NEW\_ICCID
  + It means that the SM-SR-TP knows the Connectivity Parameters of the MNO’s network related to the Disabled Profile (i.e. #MNO1\_CON\_NAN, #MNO1\_CON\_LOGIN, #MNO1\_CON\_PWD)

###### Test Sequence N°1 – Nominal Case

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-PrepareSMSRChange, #EID\_RPS, #CUR\_SR\_ID\_RPS)  see Note 1 |  |  |
| 2 | SM-SR-UT → MNO2-S | Send the  ES4-PrepareSMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ35, PROC\_REQ13 |
| 3 | MNO2-S → SM-SR-TP | SEND\_REQ(  ES4-SMSRChange,  #EID\_RPS, #TGT\_SR\_UT\_ID\_RPS) |  |  |
| 4 | *Wait until a response is received (the SM-SR-TP and SM-SR-UT treatments MAY take several minutes)* | | | |
| 5 | SM-SR-TP → MNO2-S | Send the  ES4-SMSRChange  response | The Status is equal to  #SUCCESS | EUICC\_REQ25, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13, PF\_REQ2 |
| 6 | SM-SR-UT → MNO2-S | SEND\_NOTIF( ES4-  HandleSMSRChangeNotification, #EIS\_RPS,  #TIMESTAMP\_RPS)  Note: The #EIS\_RPS shall:   * Not contain the ISD-R information * Only contain Profiles owned by the MNO2-S |  | EUICC\_REQ37 |
| 7 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS) |  |  |
| 8 | *Wait until a response is received (the SM-SR-UT treatments MAY take several minutes)* | | | |
| 9 | SM-SR-UT → MNO2-S | Send the  ES4-AuditEIS  response | 1. The Status is equal to   #SUCCESS   1. The EIS is equal to   #EIS\_RPS except that:   * 1. the ISD-R information is not present   2. only Profiles related to the MNO2-S are present | PM\_REQ25, PROC\_REQ13, PF\_REQ7, PF\_REQ2 |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 10 | MNO1-S → SM- SR-UT | SEND\_REQ(  ES4-EnableProfile, #EID\_RPS, #NEW\_ICCID\_RPS)  See Note 2 |  |  |
| 11 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 12 | SM-SR-UT → MNO1-S | Send the  ES4-EnableProfile  response | The Status is equal to  #SUCCESS |  |
| *Note 1: In the #CUR\_SR\_ID\_RPS, the SM-SR identifier is the SM-SR-TP one (not the SM-SR-UT one)*  *Note 2: Before performing this operation, the SM-SR-UT SHOULD use the ES5-UpdateSMSRAddressingParameters method to set the #SM\_SR\_DEST\_ADDR (and optionally the #SM\_SR\_UDP\_IP, #SM\_SR\_UDP\_PORT, #SM\_SR\_TCP\_IP, #SM\_SR\_TCP\_PORT, #SM\_SR\_HTTP\_URI and #SM\_SR\_HTTP\_HOST).* | | | | |

###### TC.PROC.SMSRCH.3: SMSRChange

###### Test Purpose

*To ensure the SM-SR change process is correctly implemented when an error occurs during the procedure.*

*To make sure that the audit trail contains an audit operation in the function ES7- HandoverEUICC*, *an audit request is sent on the current SM-SR before launching the SM- SR change process.*

*As the SM-SR change fails, the eUICC SHALL be associated to the same SM-SR (i.e. SM- SR-UT).*

###### Test Environment

**Device**

ES4-AuditEIS

ES5-eUICCCapabilityAudit

ES4-AuditEIS

ES4-PrepareSMSRChange ES4-SMSRChange

ES7-HandoverEUICC

ES7-HandoverEUICC

ES4-SMSRChange

ES4-GetEIS

**eUICC**

**SM-SR-S**

**SM-SR-UT**

**MNO2-S**

Note that the function ES4-PrepareSMSRChange SHALL NOT be performed by the simulators (in the schema above, this is only an informative message).

In this test case, the Initiator Role (see GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) section 2.3.1) is assumed to be played by the MNO2-S.

###### Referenced Requirements

* PF\_REQ2, PF\_REQ7
* EUICC\_REQ36, EUICC\_REQ39
* PM\_REQ22, PM\_REQ25
* PROC\_REQ13

###### Initial Conditions

* The eUICC identified by #EID has been initially provisioned on the SM-SR-UT using the #EIS\_RPS
* All Profiles present in the #EIS\_RPS SHALL NOT contain any smdp-id
* All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)

###### Test Sequence N°1 – Error Case: Unable to manage the eUICC

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-AuditEIS, #EID\_RPS, #ICCID\_RPS) |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → MNO2-S | Send the  ES4-AuditEIS  response | The Status is equal to  #SUCCESS | PF\_REQ2, PF\_REQ7, PM\_REQ25 |
| 4 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange,  #EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 5 | SM-SR-UT → SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to #EIS\_RPS  except that:   1. the audit trail is present and contains the operation #AUDIT\_OPERATION\_RPS   (i.e. other records MAY be present)   1. the last audit date is present and equal to   {CURRENT\_DATE}   1. the ISD-R keys values are empty | EUICC\_REQ36, EUICC\_REQ39, PROC\_REQ13 |
| 6 | SM-SR-S → SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_FUN\_PROV, #RC\_COND\_USED) |  |  |
| 7 | SM-SR-UT → MNO2-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUN\_PROV 2. The Reason code is equal to #RC\_COND\_USED | EUICC\_REQ36, PROC\_REQ13 |
| 8 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 9 | SM-SR-UT → MNO2-S | Send the  ES4-GetEIS  response | The Status is equal to  #SUCCESS | PM\_REQ22, PROC\_REQ13 |

###### TC.PROC.SMSRCH.4: SMSRChange

###### Test Purpose

*To ensure the SM-SR change process is correctly implemented when an error occurs during the procedure. In this particular test case, a conditional parameter (i.e. HostID) is missing in the input parameters of the method ES7-CreateAdditionalKeyset. As the SM-SR change fails, the eUICC SHALL be associated to the same SM-SR (i.e. SM-SR-UT).*

###### Test Environment

ES4-PrepareSMSRChange

ES4-SMSRChange

ES7-HandoverEUICC ES7-AuthenticateSMSR

ES5-EstablishISDRKeySet

ES7-AuthenticateSMSR

ES7-CreateAdditionalKeyset

ES7-CreateAdditionalKeyset

ES7-HandoverEUICC

ES4-SMSRChange

ES4-GetEIS

**SM-SR-S**

**SM-SR-UT**

**MNO2-S**

|  |  |  |
| --- | --- | --- |
| **Device** | | |
|  | **eUICC** |  |

Note that the function ES4-PrepareSMSRChange SHALL NOT be performed by the simulators (in the schema above, this is only an informative message).

In this test case, the Initiator Role (see GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) section 2.3.1) is assumed to be played by the MNO2-S.

###### Referenced Requirements

PF\_REQ2

EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, EUICC\_REQ40

PM\_REQ22

PROC\_REQ13

###### Initial Conditions

The eUICC identified by #EID has been initially provisioned on the SM-SR-UT using the #EIS\_RPS

All Profiles present in the #EIS\_RPS SHALL NOT contain any smdp-id

All necessary settings have been initialized on SM-SR-UT to accept the SM-SR change (i.e. business agreement…)

###### Test Sequence N°1 – Error Case: Missing Host ID parameter

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-SMSRChange, #EID\_RPS, #TGT\_SR\_S\_ID\_RPS) |  |  |
| 2 | SM-SR-UT → SM-SR-S | Send the  ES7-HandoverEUICC  request | The EIS is equal to  #EIS\_RPS except that  the ISD-R keys values are empty | EUICC\_REQ36, EUICC\_REQ39, PROC\_REQ13 |
| 3 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-AuthenticateSMSR, #EID\_RPS,  #VALID\_SR\_CERTIF\_RPS) |  |  |
| 4 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 5 | SM-SR-UT → SM-SR-S | Send the  ES7-AuthenticateSMSR  response | 1. The Status is equal to   #SUCCESS   1. The Random Challenge is present (i.e. {RC}) | PF\_REQ2, EUICC\_REQ24, EUICC\_REQ36, EUICC\_REQ39, EUICC\_REQ40, PROC\_REQ13 |
| 6 | SM-SR-S → SM-SR-UT | SEND\_REQ(  ES7-CreateAdditionalKeyset, #EID\_RPS,  #KEY\_VERSION\_RPS, #INIT\_SEQ\_COUNTER\_RPS, #ECC\_KEY\_LENGTH\_RPS, #SC3\_DR\_HOST\_RPS, #EPHEMERAL\_PK\_RPS,  #SIGNATURE\_RPS) |  |  |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 7 | SM-SR-UT → SM-SR-S | Send the  ES7-CreateAdditionalKeyset  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUNCTION 2. The Reason code is equal to #RC\_COND\_PARAM 3. derivationRandom is empty 4. The receipt is empty | EUICC\_REQ36, EUICC\_REQ38, EUICC\_REQ39, PROC\_REQ13 |
| 8 | SM-SR-S → SM-SR-UT | SEND\_ERROR\_RESP(  ES7-HandoverEUICC, #FAILED, #SC\_FUN\_PROV,  #RC\_COND\_PARAM) |  |  |
| 9 | SM-SR-UT → MNO2-S | Send the  ES4-SMSRChange  response | 1. The Status is equal to   #FAILED   1. The Subject code is equal to #SC\_FUNCTION 2. The Reason code is equal to #RC\_COND\_PARAM | EUICC\_REQ36, PROC\_REQ13 |
| 10 | MNO2-S → SM-SR-UT | SEND\_REQ(  ES4-GetEIS, #EID\_RPS) |  |  |
| 11 | SM-SR-UT → MNO2-S | Send the  ES4-GetEIS  response | The Status is equal to  #SUCCESS | PM\_REQ22, PROC\_REQ13 |

## Update Connectivity Parameters Process

###### Conformance Requirements

###### References

* + - * + GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7)

###### Requirements

* + - * + PROC\_REQ19
        + PM\_REQ21

###### Test Cases

###### General Initial Conditions

* + - * + #MNO1\_S\_ID well known to the SM-SR-UT
        + #MNO2\_S\_ID well known to the SM-SR-UT
        + The Profile identified by #ICCID is owned by MNO2-S and is in Enabled state
        + The SM-SR-UT is able to communicate with the network linked to the default Enabled Profile of the eUICC (identified by #ICCID)
* It means that the SM-SR-UT knows the Connectivity Parameters of the MNO’s

network related to the default Enabled Profile (i.e. #MNO2\_CON\_NAN, #MNO2\_CON\_LOGIN, #MNO2\_CON\_PWD)

* + - * + The eUICC identified by #EID has been initially provisioned on the SM-SR-UT using the #EIS\_RPS

###### TC.PROC.UCP.1: UpdateConnectivityParameters

###### Test Purpose

*To ensure the Connectivity Parameters can be updated by the SM-SR when the SM-DP requests it.*

###### Test Environment

**SM-SR-UT**

**Device**

ES3-UpdateConnectivityParameters(ES8-UpdateConnectivityParameters)

**eUICC**

**SM-DP-S**

###### Referenced Requirements

* + - * + PROC\_REQ19
        + PM\_REQ21

###### Initial Conditions

* + - * + None

###### Test Sequence N°1 - Nominal Case: Update SMS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ( ES3-  UpdateConnectivityParameters, #EID\_RPS,  #ICCID\_RPS, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN, [STORE\_SMS\_PARAM\_MNO2]))  see Note 1 |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → SM-DP-S | Send the  ES3-  UpdateConnectivityParameters  response | The Status is equal to  #SUCCESS | PROC\_REQ19, PM\_REQ21 |
| *Note 1: The C-APDUs generated by the method SCP03\_SCRIPT SHALL be set into the RPS element*  *<connectivityParameters>* | | | | |

###### Test Sequence N°2 - Nominal Case: Update CAT\_TP Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ( ES3-  UpdateConnectivityParameters, #EID\_RPS,  #ICCID\_RPS, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN, [STORE\_CATTP\_PARAM\_MNO2]))  see Note 1 |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → SM-DP-S | Send the  ES3-  UpdateConnectivityParameters  response | The Status is equal to  #SUCCESS | PROC\_REQ19, PM\_REQ21 |
| *Note 1: The C-APDUs generated by the method SCP03\_SCRIPT SHALL be set into the RPS element*  *<connectivityParameters>* | | | | |

###### Test Sequence N°3 - Nominal Case: Update HTTPS Parameters

###### Initial Conditions

None

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Step** | **Direction** | **Sequence / Description** | **Expected result** | **REQ** |
| 1 | SM-DP-S → SM-SR-UT | SEND\_REQ( ES3-  UpdateConnectivityParameters, #EID\_RPS,  #ICCID\_RPS, SCP03\_SCRIPT(  #DEFAULT\_ISD\_P\_SCP03\_KVN, [STORE\_HTTPS\_PARAM\_MNO2]))  see Note 1 |  |  |
| 2 | *Wait until a response is received (the SM-SR-UT treatment MAY take several minutes)* | | | |
| 3 | SM-SR-UT → SM-DP-S | Send the  ES3-  UpdateConnectivityParameters  response | The Status is equal to  #SUCCESS | PROC\_REQ19, PM\_REQ21 |
| *Note 1: The C-APDUs generated by the method SCP03\_SCRIPT SHALL be set into the RPS element*  *<connectivityParameters>* | | | | |

# Test Specifications

Some test specifications related to the eUICC ecosystem have been developed by external organisations (e.g. SIMAlliance). These organisations defined their own requirements for test benches, test applicability and pass criteria.

This section lists the test specifications that relate to the GSMA Remote Provisioning Architecture for Embedded UICC - Technical Specification [[2]](#_bookmark7).

## SIMAlliance eUICC Profile Package Test Specification

The eUICC SHALL take test cases defined in the SIMAlliance eUICC Profile Package: Interoperable Format Test Specification [17] in order to check its compliance with the SIMAlliance eUICC Profile Package: Interoperable Format Technical Specification [16].

All the mandatory test cases are applicable according to the applicability of the referred SIMAlliance test specification.

eUICC Manufacturers SHALL declare that the following SIMAlliance options are supported by the eUICC:

* + - O\_JAVACARD



# Annex A Reference Applications

The following Annex provides clarification on the applications to be used to execute some test cases.

## Applet1

## Description

This applet defines an application which implements uicc.toolkit.ToolkitInterface. The event *EVENT\_FORMATTED\_SMS\_PP\_ENV* is set in the Toolkit Registry entry of the applet.

## AID

* + - * Executable Load File AID: A0 00 00 05 59 10 10 01
      * Executable Module AID: A0 00 00 05 59 10 10 01 11 22 33

## Source Code (Java Card)

**package** com.gsma.euicc.test.applet1;

**import** javacard.framework.AID; **import** javacard.framework.APDU; **import** javacard.framework.Applet;

**import** javacard.framework.ISOException; **import** javacard.framework.Shareable; **import** uicc.toolkit.ToolkitException; **import** uicc.toolkit.ToolkitInterface; **import** uicc.toolkit.ToolkitRegistrySystem; **import** uicc.usim.toolkit.ToolkitConstants;

/\*\*

\* GSMA Test Toolkit Applet1

\*/

**public class** Applet1 **extends** Applet **implements** ToolkitConstants, ToolkitInterface {

/\*\*

\* Default Applet constructor

\*/

**public** Applet1() {

// nothing to do

}

/\*\*

* Create an instance of the applet, the Java Card runtime environment will
* call this static method first.
* **@param** bArray the array containing installation parameters
* **@param** bOffset the starting offset in bArray
* **@param** bLength the length in bytes of the parameter data in bArray
* **@throws** ISOException if the install method failed
* **@see** javacard.framework.Applet

\*/

**public static void** install(**byte**[] bArray, **short** bOffset, **byte** bLength)

**throws** ISOException {

Applet1 applet1 = **new** Applet1(); **byte** aidLen = bArray[bOffset]; **if** (aidLen == (**byte**) 0) {

applet1.register();

} **else** {

applet1.register(bArray, (**short**) (bOffset + 1), aidLen);

}

applet1.registerEvent();

}

/\*

* (non-Javadoc)
* @see Applet#process(javacard.framework.APDU)

\*/

**public void** process(APDU apdu) **throws** ISOException {

// nothing to do

}

/\*

* (non-Javadoc)
* @see Applet#getShareableInterfaceObject(javacard.framework.AID, byte)

\*/

**public** Shareable getShareableInterfaceObject(AID clientAID, **byte** param) {

**if** ((param == (**byte**) 0x01) && (clientAID == **null**)) {

**return** ((Shareable) **this**);

}

**return null**;

}

/\*

* (non-Javadoc)
* @see uicc.toolkit.ToolkitInterface#processToolkit(short)

\*/

**public void** processToolkit(**short** event) **throws** ToolkitException {

// nothing to do

}

/\*\*

* Registration to the event EVENT\_FORMATTED\_SMS\_PP\_ENV

\*/

**private void** registerEvent() { ToolkitRegistrySystem.*getEntry*()

.setEvent(*EVENT\_FORMATTED\_SMS\_PP\_ENV*);

}

}

## Applet2

## Description

This applet is a clone of Applet1 except that the package AID and the applet AID are different.

## AID

* + - * Executable Load File AID: A0 00 00 05 59 10 10 02
      * Executable Module AID: A0 00 00 05 59 10 10 02 11 22 33

## Source Code (Java Card)

This source code is exactly the same as the Applet1 defined in Annex [A.1](#_bookmark254) except that the package name SHALL be com.gsma.euicc.test.applet2.

## Applet3

## Description

This applet defines a “simple” application.

## AID

* + - * Executable Load File AID: A0 00 00 05 59 10 10 03
      * Executable Module AID: A0 00 00 05 59 10 10 03 44 55 66

## Source Code (Java Card)

**package** com.gsma.euicc.test.applet3;

**import** javacard.framework.APDU;

**import** javacard.framework.Applet;

**import** javacard.framework.ISOException;

/\*\*

\* GSMA Test Applet3

\*/

**public class** Applet3 **extends** Applet {

/\*\*

\* Default Applet constructor

\*/

**public** Applet3() {

// nothing to do

}

/\*\*

* Create an instance of the applet, the Java Card runtime environment will
* call this static method first.
* **@param** bArray the array containing installation parameters
* **@param** bOffset the starting offset in bArray
* **@param** bLength the length in bytes of the parameter data in bArray
* **@throws** ISOException if the install method failed
* **@see** javacard.framework.Applet

\*/

**public static void** install(**byte**[] bArray, **short** bOffset, **byte** bLength)

**throws** ISOException {

Applet3 applet3 = **new** Applet3(); **byte** aidLen = bArray[bOffset]; **if** (aidLen == (**byte**) 0) {

applet3.register();

} **else** {

applet3.register(bArray, (**short**) (bOffset + 1), aidLen);

}

}

/\*

* (non-Javadoc)
* @see Applet#process(javacard.framework.APDU)

\*/

**public void** process(APDU apdu) **throws** ISOException {

// nothing to do

}

}

# Annex B Constants

## Hexadecimal Constants

Here are the hexadecimal constants values used in this document:

|  |  |
| --- | --- |
| **Constant name** | **Value in hexadecimal string** |
| ADMIN\_HOST | 6C 6F 63 61 6C 68 6F 73 74 |
| ADMIN\_URI | 2F 67 73 6D 61 2F 61 64 6D 69 6E 61 67 65 6E 74 |
| AGENT\_ID | 2F 2F 73 65 2D 69 64 2F 65 69 64 2F #EID 3B  2F 2F 61 61 2D 69 64 2F 61 69 64 2F 41 30 30 30  30 30 30 35 35 39 2F 31 30 31 30 46 46 46 46 46  46 46 46 38 39 30 30 30 30 30 31 30 30 |
| BAD\_SCP03\_KVN | 35 |
| BAD\_SPI | 12 29 |
| BAD\_TOKEN | 01 02 03 |
| BEARER\_DESCRIPTION | 02 00 00 03 00 00 02 |
| BUFFER\_SIZE | 05 78 |
| CASD\_AID | A0 00 00 01 51 53 50 43 41 53 44 00 |
| CAT\_TP\_PORT | 04 00 |
| DATA | 22 0E 80 50 30 00 08 01 02 03 04 01 02 03 04 00 |
| DCS | F6 |
| DEST\_ADDR | 05 85 02 82 F2 |
| DEST\_ADDR2 | 05 85 03 83 F3 |
| DEST\_ADDR3 | 05 85 03 83 F4 |
| DIALING\_NUMBER | 33 86 99 42 11 F0 |
| DIALING\_NUMBER\_INITIAL | 33 86 99 00 00 F0 |
| ECASD\_AID | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 02 00 |
| ECASD\_TAR | 00 00 02 |
| FIRST\_SCRIPT | 01 |
| HOST\_ID | 47 53 4D 41 5F 48 4F 53 54 5F 49 44 |
| ICCID1 | 89 01 99 99 00 00 44 77 78 78 |
| ICCID2 | 89 01 99 99 00 00 44 77 78 79 |
| IP\_VALUE | 7F 00 00 01 |
| IP\_VALUE2 | 7F 00 00 02 |
| ISD\_P\_AID1 | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 10 00  see Note 1 |
| ISD\_P\_ID1 | 00 00 10  see Note 3 |
| ISD\_P\_AID2 | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 11 00 |
| ISD\_P\_AID3 | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 12 00 |
| ISD\_P\_AID\_UNKNOWN | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 99 00 |
| ISD\_P\_ATTRIBUTE | 53 |
| ISD\_P\_MOD\_AID | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 0E 00 |

|  |  |
| --- | --- |
| **Constant name** | **Value in hexadecimal string** |
| ISD\_P\_PIX\_PREFIX | 10 10 FF FF FF FF 89 |
| ISD\_P\_PKG\_AID | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 0D 00 |
| ISD\_P\_PROV\_ID | 47 53 4D 41 |
| ISD\_P\_RID | A0 00 00 05 59 |
| ISD\_P\_SDIN | 49 53 44 50 53 44 49 4E |
| ISD\_P\_SIN | 49 53 44 50 |
| ISD\_P\_TAR1 | 00 00 10  see Note 1 |
| ISD\_R\_AID | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 01 00 |
| ISD\_R\_TAR | 00 00 01 |
| KEY | 11 22 33 44 55 66 77 88 99 10 11 12 13 14 15 16 |
| KEY\_USAGE | 00 80 |
| LAST\_SCRIPT | 03 |
| LOGIN | 04 6C 6F 67 69 6E |
| MEMORY\_QUOTA | 00 00 20 00 |
| MNO\_AGENT\_ID | 2F 2F 73 65 2D 69 64 2F 65 69 64 2F #EID 3B  2F 2F 61 61 2D 69 64 2F 61 69 64 2F #MNO\_SD\_AID |
| NEW\_SCP81\_PSK | 18 94 D8 3C 1F BF 38 27 92 76 B7 0F 8F 02 61 16 |
| NAN\_VALUE | 09 47 53 4D 41 65 55 49 43 43 |
| PID | 11 |
| PSK\_DEK | 01 02 03 04 05 06 07 08 01 02 03 04 05 06 07 08 |
| PWD | 04 70 61 73 73 77 6F 72 64 |
| RESERVED\_ISD\_P\_AID | A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 0F 00 |
| SC3\_DR | 0B |
| SC3\_DR\_HOST | 0F |
| SC3\_NO\_DR | 09 |
| SCP03\_KVN | 30 |
| SCP80\_NEW\_KVN | 0E  see Note 2 |
| SPI\_VALUE | 16 39 |
| SPI\_VALUE\_NO\_POR | 16 00 |
| SPI\_NOTIF | 02 00 |
| SUB\_SCRIPT | 02 |
| TCP\_PORT | 1F 41 |
| TOKEN\_ID | 01 |
| TON\_NPI | 91 |
| UDP\_PORT | 1F 40 |
| VIRTUAL\_EID | 89 00 10 12 01 23 41 23 40 12 34 56 78 90 12 24 |
| VIRTUAL\_EID2 | 89 00 15 67 01 02 03 04 05 06 07 08 09 10 11 52 |
| VIRTUAL\_SDIN | 00 00 00 00 01 02 03 04 05 06 07 08 |
| VIRTUAL\_SIN | 01 02 03 04 |

|  |  |
| --- | --- |
| **Constant name** | **Value in hexadecimal string** |
| *Note 1: SHALL be different from the Profiles already installed on the eUICC. This constant depends on the eUICC*  *Note 2: SHALL NOT be initialized by default on the eUICC (different than #SCP80\_KVN)*  *Note 3: SHALL correspond to the identifier of #ISD\_P\_AID1 (i.e. digits 15 to 20 of PIX of ISD-P)* | |

###### Table 8: Hexadecimal Constants

## ASCII Constants

Here are the ASCII constants values used in this document:

|  |  |
| --- | --- |
| **Constant name** | **Value in ASCII** |
| BIG\_MEM | 9999999 |
| CONTENT\_TYPE | Content-Type: application/vnd.globalplatform.card- content-mgt-response;version=1.0 |
| EUM\_S\_ID | 10.11.12 |
| FAILED | Failed |
| HOST | Host: localhost |
| HTTP\_CODE\_200 | HTTP/1.1 200 |
| HTTP\_CODE\_204 | HTTP/1.1 204 |
| IMSI1 | 234101943787656 |
| IMSI2 | 234101943787657 |
| IMSI3 | 234101943787658 |
| MNO1\_S\_ID | 1.2.3 |
| MNO2\_S\_ID | 11.22.33 |
| MSISDN1 | 447112233445 |
| MSISDN2 | 447112233446 |
| MSISDN3 | 447112233447 |
| POST\_URI | POST /gsma/adminagent HTTP/1.1 |
| POST\_URI\_NOTIF | POST /gsma/adminagent?msg=#NOTIF\_PROFILE\_CHANGE HTTP/1.1 |
| POST\_URI\_NOTIF2 | POST /gsma/adminagent?msg=#NOTIF\_PROFILE\_CHANGE2 HTTP/1.1 |
| PROFILE1\_TYPE | GENERIC PROFILE1 3G |
| PROFILE2\_TYPE | GENERIC PROFILE2 3G |
| PSK\_ID | 8001028110#EID4F10#ISD\_R\_AID8201#SCP81\_KEY\_ID8301#SCP81\_KVN  see Note 2 |
| RC\_ALREADY\_USED | 3.3 |
| RC\_COND\_PARAM | 2.3 |
| RC\_COND\_USED | 3 |
| RC\_EXECUTION\_ERROR | 4.2 |
| RC\_EXPIRED | 6.3 |
| RC\_ID\_UNKNOWN | 1.1 |
| RC\_INACCESSIBLE | 5.1 |
| RC\_INVALID | 2.1 |

|  |  |
| --- | --- |
| **Constant name** | **Value in ASCII** |
| RC\_INVALID\_DEST | 3.4 |
| RC\_MEMORY | 4.8 |
| RC\_NOT\_ALLOWED | 1.2 |
| RC\_OBJ\_EXIST | 3.6 |
| RC\_REFUSED | 3.8 |
| RC\_UNKNOWN | 3.9 |
| RC\_NOT\_PRESENT | 4.6 |
| RC\_VERIFICATION\_FAILED | 6.1 |
| SC\_CERT\_REQ | 8.5.1 |
| SC\_ECASD | 8.5.2 |
| SC\_EID | 8.1.1 |
| SC\_EIS | 8.6 |
| SC\_EUICC | 8.1 |
| SC\_FUN\_PROV | 1.2 |
| SC\_EXT\_RES | 1.4 |
| SC\_FUNCTION | 1.6 |
| SC\_SD\_AID | 8.3.1 |
| SC\_ISDP | 8.3 |
| SC\_ISDR | 8.4 |
| SC\_POL1 | 8.2.2 |
| SC\_POL2 | 8.2.3 |
| SC\_PROFILE\_ICCID | 8.2.1 |
| SC\_PROFILE | 8.2 |
| SC\_SM\_SR | 8.7 |
| SC\_SM\_SR\_CERT | 8.7.1 |
| SC\_SR\_CERTIF | 8.5.3 |
| SC\_SUB\_ADDR | 8.2.6 |
| SMALL\_MEM | 999 |
| SM\_DP\_S\_ID | 4.5.6 |
| SM\_SR\_S\_ID | 7.8.9 |
| SUCCESS | Executed-Success |
| TRANSFER\_ENCODING | Transfer-Encoding: chunked |
| UNKNOWN\_SM\_SR\_ID | 8888.9999.1111  see Note 1 |
| WARNING | Executed-WithWarning |
| X\_ADMIN\_FROM\_ISD\_R | X-Admin-From: //se-id/eid/#EID;//aa- id/aid/A000000559/1010FFFFFFFF8900000100 |
| X\_ADMIN\_FROM\_MNO | X-Admin-From: //se-id/eid/#EID;//aa-id/aid/#MNO\_SD\_AID |
| X\_ADMIN\_NEXT\_URI | X-Admin-Next-URI: /gsma/adminagent |
| X\_ADMIN\_PROTOCOL | X-Admin-Protocol: globalplatform-remote-admin/1.0 |
| X\_ADMIN\_STATUS\_OK | X-Admin-Script-Status: ok |

|  |  |
| --- | --- |
| **Constant name** | **Value in ASCII** |
| *Note 1: This value SHALL be unknown to all platforms under test.*  *Note 2: This Pre-Shared Key identity string SHALL be configured by default in the ISD-R.* | |

###### Table 9: ASCII Constants

## eUICC Settings

Here are the different settings that SHALL be given by the eUICC Manufacturer to execute the test cases defined in this document.

|  |  |
| --- | --- |
| **eUICC setting name** | **Description** |
| CARD\_RECOGNITION\_DATA | Value of the TLV ‘66’ - Card recognition data. |
| DEFAULT\_ISD\_P\_AID | The AID of the default ISD-P pre-installed on the eUICC (this ISD-P SHALL be Enabled). |
| DEFAULT\_ISD\_P\_ID | The Identifier of the default ISD-P (digits 15 to 20 of PIX of ISD-P) pre- installed on the eUICC (this corresponds to the #DEFAULT\_ISD\_P\_AID). |
| DEFAULT\_ISD\_P\_SCP03\_KDEK | The SCP03 DEK key of the default ISD-P pre-installed on the eUICC. |
| DEFAULT\_ISD\_P\_SCP03\_KENC | The SCP03 ENC key of the default ISD-P pre-installed on the eUICC. |
| DEFAULT\_ISD\_P\_SCP03\_KMAC | The SCP03 MAC key of the default ISD-P pre-installed on the eUICC. |
| DEFAULT\_ISD\_P\_SCP03\_KVN | The SCP03 KVN of the default ISD-P pre-installed on the eUICC. |
| DEFAULT\_ISD\_P\_TAR | The TAR of the default ISD-P pre-installed on the eUICC. |
| ECASD\_CERTIFICATE | Value of the TLV ‘7F21’ - ECASD certificate (i.e. CERT.ECASD.ECKA). |
| CASD\_CERTIFICATE\_SC2B | Value of the TLV ‘7F21’ - CASD certificate (of the default Enabled Profile) allowing to confidentialy setup keys using scenario #2.B. |
| CASD\_CERTIFICATE\_SC3 | Value of the TLV ‘7F21’ - CASD certificate (of the default Enabled Profile) allowing to confidentialy setup keys using scenario #3. |
| EID | Content of the TLV ‘5A’ available on the ECASD. |
| EUM\_OID | EUM\_OID (i.e. value of the tag ‘42’ – CA Identifier of the ECASD certificate) *Note: When present in the ECASD, this value SHALL be encoded as a value part of the DER\_TLV\_OID (e.g. 0x2B….).*  *When present in the EIS, this value SHALL be encoded as a dotted number notation (e.g. “1.3.6….“.).* |
| EUM\_SUBJECT\_KEY\_ID | Subject Key Identifier of the EUM Certificate (i.e. value of the tag ‘C9’ of the ECASD certificate) |
| EUM\_PK\_ECDSA | Public key of the EUM used for ECDSA. |
| EUM\_PK\_CA\_AUT | Public key of the EUM used to verify the MNO CASD certificate. |
| ISD\_R\_SIN | Content of the TLV ‘42’ available on the ISD-R. |
| ISD\_R\_SDIN | Content of the TLV ‘45’ available on the ISD-R. |
| PROFILE\_PACKAGE | A Profile Package that contains all Profile Elements allowing the testing of the download and the network attachment processes. This Profile SHOULD follow the description defined in Annex [B.7.](#_bookmark273) |
| MNO\_PSK\_ID | The Pre-Shared Key identity string related to the SCP81 keyset initialized on the MNO-SD. (optional: depends if O\_MNO\_HTTPS is supported) |
| MNO\_SCP80\_AUTH\_KEY | The value of the SCP80 message authentication key initialized on the default MNO-SD. (key identifier 02) |

|  |  |
| --- | --- |
| **eUICC setting name** | **Description** |
| MNO\_SCP80\_DATA\_ENC\_KEY | The value of the SCP80 data encryption key initialized on the default MNO- SD. (key identifier 03) |
| MNO\_SCP80\_ENC\_KEY | The value of the SCP80 encryption key initialized on the default MNO-SD. (key identifier 01) |
| MNO\_SCP80\_KVN | The key version number of the SCP80 keyset initialized on the default MNO-SD. |
| MNO\_SCP81\_KEY\_ID | The key identifier of the PSK in the SCP81 keyset initialized on the MNO- SD. (optional: depends if O\_MNO\_HTTPS is supported) |
| MNO\_SCP81\_KVN | The key version number of the SCP81 keyset initialized on the MNO-SD. (optional: depends if O\_MNO\_HTTPS is supported) |
| MNO\_SCP81\_PSK | The value of the Pre-Shared Key initialized on the MNO-SD. (optional: depends if O\_MNO\_HTTPS is supported) |
| MNO\_SD\_AID | The MNO ISD AID of the default Profile pre-installed on the eUICC. |
| MNO\_TAR | The TAR of the default MNO-SD (SHOULD be ‘B2 01 00’). |
| PK\_ECASD\_ECKA | Public Key of the ECASD used for ECKA (i.e. PK.ECASD.ECKA). |
| SCP80\_DATA\_ENC\_KEY | The value of the SCP80 data encryption key initialized on the ISD-R.  (key identifier 03) |
| SCP80\_ENC\_KEY | The value of the SCP80 encryption key initialized on the ISD-R.  (key identifier 01) |
| SCP80\_KVN | The key version number of the SCP80 keyset initialized on the ISD-R. |
| SCP80\_AUTH\_KEY | The value of the SCP80 message authentication key initialized on the ISD-  R. (key identifier 02) |
| SCP81\_KEY\_ID | The key identifier of the PSK in the SCP81 keyset initialized on the ISD-R. (optional: depends if O\_HTTPS is supported) |
| SCP81\_KVN | The key version number of the SCP81 keyset initialized on the ISD-R. (optional: depends if O\_HTTPS is supported) |
| SCP81\_PSK | The value of the Pre-Shared Key initialized on the ISD-R. (optional: depends if O\_HTTPS is supported) |

###### Table 10: eUICC Settings

## Platforms Settings

Here are the different platforms’ settings that SHALL be used to execute the test cases defined in this document. The corresponding values SHALL be given either by the test tool provider, the platform under test or the CI.

|  |  |
| --- | --- |
| **Platform setting name** | **Description** |
| ECASD\_BAD\_SIGN\_CERT | A certificate CERT.ECASD.ECKA with an invalid signature of a simulated eUICC. The TLV ‘7F21’ SHALL contain:  93 01 09  42 {L} #EUM\_OID  5F 20 10 #VIRTUAL\_EID  95 02 00 80  5F 25 04 20 00 01 01  5F 24 04 21 45 01 01  45 0C #VIRTUAL\_SDIN  73 {L}  C0 01 01 |

|  |  |
| --- | --- |
| **Platform setting name** | **Description** |
|  | C1 01 01  C2 01 01  C9 14 #EUM\_SUBJECT\_KEY\_ID 7F 49 {L} #PK\_ECASD\_S\_ECKA  5F 37 {L} {SIGNATURE}  This signature SHALL NOT be generated using the #EUM\_S\_SK\_ECDSA.  see Note 1 |
| EUM\_S\_ACCESSPOINT | The EUM-S access point allowing SM-SR-UT to communicate with a EUM simulator.  see Note 1 |
| EUM\_S\_CERT\_ID\_ECDSA | The certificate subject name of the EUM-S used for ECDSA.  The use of the certificate subject name in the EIS implicitly means that all platforms under test (i.e. SM-DP-UT and SM-SR-UT) know the #EUM\_S\_PK\_ECDSA (this public key is part of the #EUM\_S\_CERT\_ECDSA).  see Note 1 |
| EUM\_S\_PK\_ECDSA | Public key of the EUM-S used for ECDSA.  see Note 1 |
| EUM\_S\_SK\_ECDSA | Private key of the EUM-S used for ECDSA.  see Note 1 |
| EUM\_S\_CERT\_ECDSA | X.509 Certificate of the EUM-S used for ECDSA. Subject name of this certificate is set to #EUM\_S\_CERT\_ID\_ECDSA. |
| EXPIRED\_ECASD\_CERT | An expired certificate CERT.ECASD.ECKA of a simulated eUICC. The TLV ‘7F21’ SHALL contain:  93 01 09  42 {L} #EUM\_OID  5F 20 10 #VIRTUAL\_EID  95 02 00 80  5F 25 04 20 00 01 01  5F 24 04 20 00 02 02  45 0C #VIRTUAL\_SDIN  73 {L}  C0 01 01  C1 01 01  C2 01 01  C9 14 #EUM\_SUBJECT\_KEY\_ID 7F 49 {L} #PK\_ECASD\_S\_ECKA  5F 37 {L} {SIGNATURE}  This signature SHALL be generated using the #EUM\_S\_SK\_ECDSA.  see Note 1 |
| EXPIRED\_SM\_SR\_CERTIFICATE | An expired certificate CERT.SR.ECDSA of a simulated SM-SR. The TLV ‘7F21’ SHALL contain:  93 01 01  42 {L} #CI\_OID  5F 20 01 01  95 01 82  5F 24 04 20 00 01 01  73 {L}  C8 01 02  C9 14 #CI\_SUBJECT\_KEY\_ID 7F 49 {L} #SM\_PK\_ECDSA  5F 37 {L} {SIGNATURE} |

|  |  |
| --- | --- |
| **Platform setting name** | **Description** |
|  | This signature SHALL be generated using the #SK\_CI\_ECDSA.  This TLV ‘7F21’ SHALL be part of the DGI ‘7F21’. see Note 1 |
| KEY\_SECURED | The #KEY encrypted with a transport key (as defined in GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7)). The transport key value and the related algorithm can be freely chosen by  the SM-SR-UT.  see Note 2 |
| INVALID\_SM\_DP\_CERTIFICATE | An invalid certificate CERT.DP.ECDSA of a simulated SM-DP (TLV ‘7F21’). The #SK\_CI\_ECDSA SHALL NOT be used to generate the signature. The content of the TLV is the same as #VALID\_SM\_DP\_CERTIFICATE.  see Note 1 |
| INVALID\_SM\_SR\_CERTIFICATE | An invalid certificate CERT.DP.ECDSA of a simulated SM-DP (TLV ‘7F21’). The #SK\_CI\_ECDSA SHALL NOT be used to generate the signature. The content of the TLV is the same as #VALID\_SM\_SR\_CERTIFICATE.  see Note 1 |
| MNO1\_S\_ACCESSPOINT | The MNO1-S access point allowing platforms under test to communicate with a MNO simulator.  see Note 1 |
| MNO2\_S\_ACCESSPOINT | The MNO2-S access point allowing platforms under test to communicate with a MNO simulator.  see Note 1 |
| PF\_ICCID\_TO\_DOWNLOAD | The ICCID of a single profile of type PF\_PROFILE\_TYPE\_TO\_DOWNLOAD, for which the SM-DP-UT can deliver a Profile Package |
| PF\_PROFILE\_TYPE\_TO\_DOWNLOAD | A profile type that is known by the SM-DP-UT; the SM-DP can provide one and only one profile package for this profile type, and the ICCID of the corresponding profile would be PF\_ICCID\_TO\_DOWNLOAD. |
| PK\_ECASD\_S\_ECKA | Public Key of a virtual ECASD used for ECKA (i.e. PK.ECASD.ECKA).  see Note 1 |
| SK\_CI\_ECDSA | The CI private key used for signing data to generate the SM-SR and the SM-DP certificates (i.e. SK.CI.ECDSA).  see Note 3 |
| SM\_DP\_ACCESSPOINT | The SM-DP-UT access point allowing communication. This value depends on the transport protocol used by the SM-DP-UT.  see Note 2 |
| SM\_DP\_ID | The SM-DP-UT identifier. see Note 2 |
| SM\_DP\_S\_ACCESSPOINT | The SM-SR-S access point allowing platforms under test to communicate with a SM-DP simulator.  see Note 1 |
| SM\_EPK\_ECKA | Ephemeral Public Key of a simulated SM-SR (i.e. ePK.SR.ECKA), SM-DP (i.e. ePK.DP.ECKA) or MNO used for ECKA.  see Note 1 |
| SM\_ESK\_ECKA | Ephemeral Private Key of a simulated SM-SR (i.e. eSK.SR.ECKA), SM-DP (i.e. eSK.DP.ECKA) or MNO used for ECKA.  see Note 1 |
| SM\_PK\_ECDSA | Public Key of a simulated SM-SR (i.e. PK.SR.ECDSA) or SM-DP (i.e. PK.DP.ECDSA) for verifying signatures.  see Note 1 |
| SM\_SK\_ECDSA | Private Key of a simulated SM-SR (i.e. SK.SR.ECDSA) or SM-DP (i.e. SK.DP.ECDSA) for creating signatures.  see Note 1 |
| SM\_SR\_ACCESSPOINT | The SM-SR-UT access point allowing communication. This value depends on the transport protocol used by the SM-SR-UT.  see Note 2 |
| SM\_SR\_ID | The SM-SR-UT identifier. |

|  |  |
| --- | --- |
| **Platform setting name** | **Description** |
|  | see Note 2 |
| SM\_SR\_S\_ACCESSPOINT | The SM-SR-S access point allowing platforms under test to communicate with a SM-SR simulator.  see Note 1 |
| VALID\_SM\_DP\_CERTIFICATE | A valid certificate CERT.DP.ECDSA of a simulated SM-DP. The TLV ‘7F21’ SHALL contain:  93 01 02  42 {L} #CI\_OID  5F 20 01 02  95 01 82  5F 24 04 21 45 01 01  73 {L}  C8 01 01  C9 14 #CI\_SUBJECT\_KEY\_ID 7F 49 {L} #SM\_PK\_ECDSA  5F 37 {L} {SIGNATURE}  This signature SHALL be generated using the #SK\_CI\_ECDSA.  see Note 1 |
| VALID\_SM\_SR\_CERTIFICATE | A valid certificate CERT.SR.ECDSA of a simulated SM-SR. The TLV ‘7F21’ SHALL contain:  93 01 01  42 {L} #CI\_OID  5F 20 01 01  95 01 82  5F 24 04 21 45 01 01  73 {L}  C8 01 02  C9 14 #CI\_SUBJECT\_KEY\_ID 7F 49 {L} #SM\_PK\_ECDSA  5F 37 {L} {SIGNATURE}  This signature SHALL be generated using the #SK\_CI\_ECDSA.  see Note 1 |
| VIRTUAL\_ECASD\_CERT | A valid certificate CERT.ECASD.ECKA of a simulated eUICC. The TLV ‘7F21’ SHALL contain:  93 01 09  42 {L} #EUM\_OID  5F 20 10 #VIRTUAL\_EID  95 02 00 80  5F 25 04 20 00 01 01  5F 24 04 21 45 01 01  45 0C #VIRTUAL\_SDIN  73 {L}  C0 01 01  C1 01 01  C2 01 01  C9 #EUM\_SUBJECT\_KEY\_ID 7F 49 {L} #PK\_ECASD\_S\_ECKA  5F 37 {L} {SIGNATURE}  This signature SHALL be generated using the #EUM\_S\_SK\_ECDSA.  see Note 1 |
| CI\_SUBJECT\_KEY\_ID | Subject Key Identifier of the CI Root Certificate (20 bytes long). see Note 3 |
| CI\_OID | OID of the root CI see Note 3 |

|  |  |
| --- | --- |
| **Platform setting name** | **Description** |
| *Note 1: SHALL be generated by the test tool*  *Note 2: SHALL be given by the platform under test Note 3: SHALL be given by the CI* | |

###### Table 11: Platforms Settings

## RPS Elements

Here are the different RPS elements that SHALL be used to execute the test cases defined in this document.

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
| AUDIT\_OPERATION\_RPS | <Record>  #EID\_RPS #SM\_SR\_UT\_ID\_RPS  <OperationDate>{CURRENT\_DATE}</OperationDate>  <OperationType>0500</OperationType>  <RequesterId>#MNO2\_S\_ID</RequesterId>  <OperationExecutionStatus> #SUCCESS  </OperationExecutionStatus>  <Isd-p-aid>#DEFAULT\_ISD\_P\_AID</Isd-p-aid> #ICCID\_RPS  </Record> |
| BIG\_MEM\_RPS | <RequiredMemory>#BIG\_MEM</RequiredMemory> |
| CATTP\_CAP\_RPS | <CattpSupport>TRUE</CattpSupport>  <CattpVersion>6.13.0</CattpVersion>  <HttpSupport>FALSE</HttpSupport>  <SecurePacketVersion>12.1.0</SecurePacketVersion>  <RemoteProvisioningVersion>3.2.0</RemoteProvisioningVersion> |
| CON\_PARAM\_RPS | <connectivityParameters>  222F80E288002A3A0727A1253507#BEARER\_DESCRIPTION4709#NAN\_VALU E0D05#LOGIN0D08#PWD  </connectivityParameters>  see Note 6 |
| CUR\_SR\_ID\_RPS | <CurrentSmSrid>#SM\_SR\_ID</CurrentSmSrid> |
| CUR\_SR\_S\_ID\_RPS | <CurrentSmSrid>#SM\_SR\_S\_ID</CurrentSmSrid> |
| DATA\_RPS | <Data>#DATA</Data>  see Note 6 |
| DEFAULT\_ISDP\_RPS | <Isd-p-aid>#DEFAULT\_ISD\_P\_AID</Isd-p-aid> |
| ECASD\_BADSIGN\_RPS | <Aid>#ECASD\_AID</Aid>  <Tar>#ECASD\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ECASD</Role>  <Keyset>  <Version>116</Version>  <Type>CA</Type> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | <Certificate>  <Index>4</Index>  <CAId>#EUM\_OID</CAId>  <Value>#ECASD\_BAD\_SIGN\_CERT</Value>  </Certificate>  </Keyset> |
| ECASD\_RPS | <Aid>#ECASD\_AID</Aid>  <Tar>#ECASD\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ECASD</Role>  <Keyset>  <Version>116</Version>  <Type>CA</Type>  <Certificate>  <Index>4</Index>  <CAId>#EUM\_OID</CAId>  <Value>#VIRTUAL\_ECASD\_CERT</Value>  </Certificate>  </Keyset> |
| ECC\_KEY\_LENGTH\_RPS | <ECCKeyLength>ECC-256</ECCKeyLength> |
| EID\_RPS | <Eid>#EID</Eid> |
| EIS\_BADCASDSIGN\_RPS  (ES3 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid>#ISD\_P\_MOD\_AID</Isd-p-module-aid>  <Ecasd>#ECASD\_BADSIGN\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig)  #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | #PROFILE1\_RPS *-- Optional*  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  </Eis>  see Note 1 and Note 8 |
| EIS\_BADEUMSIGN\_RPS  (ES1 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig)  #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS #PROFILE2\_RPS  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  </Eis>  see Note 2 |
| EIS\_ES1\_RPS  (ES1 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature  [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS #PROFILE2\_RPS  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  <AdditionalProperties>  <Property key=”a key” value=”a value”/>  </AdditionalProperties>  </Eis>  see Note 1 |
| EIS\_ES2\_RPS  (ES2 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid>#ISD\_P\_MOD\_AID</Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) #SIGNED\_INFO\_RPS  <ds:SignatureValue> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS}  #PROFILE1\_RPS *-- Optional*  <AdditionalProperties>  <Property key=”a key” value=”a value”/>  </AdditionalProperties>  </Eis>  see Note 1 |
| EIS\_ES3\_RPS  (ES3 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid>#ISD\_P\_MOD\_AID</Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature  [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS}  #PROFILE1\_RPS *-- Optional*  #PROFILE2\_RPS *-- Optional*  <Isdr-r>#ISD\_R\_ES3\_RPS</Isdr-r>  <AdditionalProperties>  <Property key=”a key” value=”a value”/>  </AdditionalProperties> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | </Eis>  see Note 1 and Note 8 |
| EIS\_ES4\_RPS  (ES4 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig)  #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS  <AdditionalProperties>  <Property key=”a key” value=”a value”/>  </AdditionalProperties>  </Eis>  see Note 1 |
| EIS\_ES7\_RPS  (ES7 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature  [xmlns:ds="http://www.w3.org/2000/09/xmldsig](http://www.w3.org/2000/09/xmldsig)"> #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EnumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS #PROFILE2\_RPS  <Isdr-r>#ISD\_R\_ES7\_RPS</Isdr-r>  <AdditionalProperties>  <Property key=”a key” value=”a value”/>  </AdditionalProperties>  </Eis>  see Note 1 |
| EIS\_EXPIREDCASD\_RPS  (ES7 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#EXPIREDECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature  [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) #SIGNED\_INFO\_RPS  <ds:SignatureValue> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS #PROFILE2\_RPS  <Isdr-r>#ISD\_R\_ES7\_RPS</Isdr-r>  </Eis>  see Note 1 |
|  | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo> |
| EIS2\_ES1\_RPS | <EumSignature |
| (ES1 interface) | [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) |
|  | #SIGNED\_INFO\_RPS |
|  | <ds:SignatureValue> |
|  | {SIGNATURE} |
|  | </ds:SignatureValue> |
|  | #KEY\_INFO\_RPS |
|  | </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE3\_RPS  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  </Eis> |
|  | see Note 3 |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  |  |
|  | <Eis>  <EumSignedInfo> #VIRTUAL\_EID2\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid> #ISD\_P\_MOD\_AID  </Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo> |
| EIS3\_ES1\_RPS | <EumSignature |
| (ES1 interface) | [xmlns:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) |
|  | #SIGNED\_INFO\_RPS |
|  | <ds:SignatureValue> |
|  | {SIGNATURE} |
|  | </ds:SignatureValue> |
|  | #KEY\_INFO\_RPS |
|  | </EumSignature> |
|  | <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 800000  </AvailableMemoryForProfiles>  {SM\_SR\_ID\_RPS} #PROFILE1\_RPS  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  </Eis> |
|  | see Note 3 |
| EP\_FALSE\_RPS | <EnableProfile>FALSE</EnableProfile> |
| EP\_TRUE\_RPS | <EnableProfile>TRUE</EnableProfile> |
| EPHEMERAL\_PK\_RPS | <EphemeralPublicKey>#SM\_EPK\_ECKA</EphemeralPublicKey> |
| EUICC\_RESP1\_RPS | <EuiccResponseData>[R\_AB\_6985]</EuiccResponseData> |
| EXPIREDECASD\_RPS | <Aid>#ECASD\_AID</Aid>  <Tar>#ECASD\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ECASD</Role>  <Keyset>  <Version>116</Version>  <Type>CA</Type>  <Certificate> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | <Index>4</Index>  <CAId>#EUM\_OID</CAId>  <Value>#EXPIRED\_ECASD\_CERT</Value>  </Certificate>  </Keyset> |
| FULL\_CAP\_RPS | <CattpSupport>TRUE</CattpSupport>  <CattpVersion>6.13.0</CattpVersion>  <HttpSupport>TRUE</HttpSupport>  <HttpVersion>1.1.3</HttpVersion>  <SecurePacketVersion>12.1.0</SecurePacketVersion>  <RemoteProvisioningVersion>3.2.0</RemoteProvisioningVersion> |
| HOST\_ID\_RPS | <HostId>#HOST\_ID</HostId> |
| HTTPS\_CAP\_RPS | <CattpSupport>FALSE</CattpSupport>  <HttpSupport>TRUE</HttpSupport>  <HttpVersion>1.1.3</HttpVersion>  <SecurePacketVersion>12.1.0</SecurePacketVersion>  <RemoteProvisioningVersion>3.2.0</RemoteProvisioningVersion> |
| ICCID\_RPS | <Iccid>#ICCID</Iccid> |
| ICCID1\_RPS | <Iccid>#ICCID1</Iccid> |
| ICCID2\_RPS | <Iccid>#ICCID2</Iccid> |
| INIT\_SEQ\_COUNTER\_RPS | <InitialSequenceCounter>0</InitialSequenceCounter> |
| INVALID\_EIS\_RPS  (ES1 interface) | <Eis>  <EumSignedInfo> #VIRTUAL\_EID\_RPS  <Eum-Id>#EUM\_S\_ID</Eum-Id>  <ProductionDate>2014-01-01T09:30:47Z</ProductionDate>  <PlatformType>JavaCard Operating System</PlatformType>  <PlatformVersion>3.0.1</PlatformVersion>  <Isd-p-loadfile-aid> #ISD\_P\_PKG\_AID  </Isd-p-loadfile-aid>  <Isd-p-module-aid>#ISD\_P\_MOD\_AID</Isd-p-module-aid>  <Ecasd>#ECASD\_RPS</Ecasd>  <EuiccCapabilities> #FULL\_CAP\_RPS  </EuiccCapabilities>  </EumSignedInfo>  <EumSignature  xmlns[:ds="http://www.w3.org/2000/09/xmldsig">](http://www.w3.org/2000/09/xmldsig) #SIGNED\_INFO\_RPS  <ds:SignatureValue>  {SIGNATURE}  </ds:SignatureValue> #KEY\_INFO\_RPS  </EumSignature>  <RemainingMemory>750000</RemainingMemory>  <AvailableMemoryForProfiles> 500  </AvailableMemoryForProfiles> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | {SM\_SR\_ID\_RPS} #PROFILE1\_RPS #PROFILE2\_RPS  <Isdr-r>#ISD\_R\_RPS</Isdr-r>  </Eis>  see Note 1 |
| ISD\_R\_ES3\_RPS | <Aid>#ISD\_R\_AID</Aid>  <Tar>#ISD\_R\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ISD-R</Role>  <Keyset>  <version>1</version>  </Keyset> |
| ISD\_R\_ES7\_RPS | <Aid>#ISD\_R\_AID</Aid>  <Tar>#ISD\_R\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ISD-R</Role>  <Keyset>  <version>1</version>  <Type>SCP80</Type>  <Cntr>1</Cntr>  <Key kcv=””>  <Index>1</Index>  <KeyComponent type=”88” value=””>  </KeyComponent>  </Key>  <Key kcv=””>  <Index>2</Index>  <KeyComponent type=”88” value=””>  </KeyComponent>  </Key>  <Key kcv=””>  <Index>3</Index>  <KeyComponent type=”88” value=””>  </KeyComponent>  </Key>  </Keyset> |
| ISD\_R\_RPS | <Aid>#ISD\_R\_AID</Aid>  <Tar>#ISD\_R\_TAR</Tar>  <Sin>#VIRTUAL\_SIN</Sin>  <Sdin>#VIRTUAL\_SDIN</Sdin>  <Role>ISD-R</Role>  <Keyset>  <version>1</version>  <Type>SCP80</Type>  <Cntr>1</Cntr>  <Key kcv=”{KEY\_KCV}”>  <Index>1</Index>  <KeyComponent  type=”88” value=”#KEY\_SECURED”>  </KeyComponent>  </Key>  <Key kcv=”{KEY\_KCV}”>  <Index>2</Index>  <KeyComponent  type=”88” value=”#KEY\_SECURED”>  </KeyComponent> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | </Key>  <Key kcv=”{KEY\_KCV}”>  <Index>3</Index>  <KeyComponent  type=”88” value=”#KEY\_SECURED”>  </KeyComponent>  </Key>  </Keyset> |
| ISDP2\_RPS | <Isd-p-aid>#ISD\_P\_AID2</Isd-p-aid> |
| ISDP3\_RPS | <Isd-p-aid>#ISD\_P\_AID3</Isd-p-aid> |
| KEY\_INFO\_RPS | <ds:KeyInfo>  <ds:X509Data>  <ds:X509SubjectName>  #EUM\_S\_CERT\_ID\_ECDSA  </ds:X509SubjectName>  </ds:X509Data>  </ds:KeyInfo> |
| KEY\_VERSION\_RPS | <KeyVersionNumber>#SCP80\_KVN</KeyVersionNumber>  see Note 4 |
| MNO1\_ID\_RPS | <Mno-id>#MNO1\_S\_ID</Mno-id> |
| MNO2\_ID\_RPS | <Mno-id>#MNO2\_S\_ID</Mno-id> |
| MORE\_TODO\_RPS | <MoreToDo>TRUE</MoreToDo> |
| NEW\_ADDR\_RPS | <newSubscriptionAddress>  <Msisdn>#MSISDN3</Imsi>  <Imsi>#IMSI3</Imsi>  </newSubscriptionAddress> |
| NEW\_ICCID\_RPS | <Iccid>#NEW\_ICCID</Iccid> |
| NO\_MORE\_TODO\_RPS | <MoreToDo>FALSE</MoreToDo> |
| PF\_ICCID\_TO\_DOWNLOAD\_RPS | <Iccid>#PF\_ICCID\_TO\_DOWNLOAD</Iccid> |
| PF\_PROFILE\_TYPE\_TO\_DOWNLOAD\_RPS | <ProfileType>#PF\_PROFILE\_TYPE\_TO\_DOWNLOAD</ProfileType> |
| POL2\_DEL\_RPS | <pol2>  <Rule>  <Subject>PROFILE</Subject>  <Action>DELETE</Action>  <Qualification>Not allowed</Qualification>  </Rule>  </pol2> |
| POL2\_DIS\_RPS | <pol2>  <Rule>  <Subject>PROFILE</Subject>  <Action>DISABLE</Action>  <Qualification>Not allowed</Qualification>  </Rule>  </pol2> |
| POL2\_EMPTY\_RPS | <pol2/> |
| PROF\_TYPE1\_RPS | <ProfileType>#PROFILE\_TYPE1</ProfileType> |
| PROF\_TYPE2\_RPS | <ProfileType>#PROFILE\_TYPE2</ProfileType> |
| PROFILE1\_RPS | <ProfileInfo>  #ICCID1\_RPS |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | #ISDP2\_RPS #MNO1\_ID\_RPS  <FallbackAttribute>TRUE</FallbackAttribute> #SUB\_ADDR1\_RPS  <State>Disabled</State>  {SM\_DP\_ID\_RPS} #PROF\_TYPE1\_RPS  <AllocatedMemory>300000</AllocatedMemory>  <FreeMemory>50000</FreeMemory> #POL2\_DEL\_RPS  </ProfileInfo> |
| PROFILE2\_RPS | <ProfileInfo>  #ICCID2\_RPS #ISDP3\_RPS #MNO2\_ID\_RPS  <FallbackAttribute>FALSE</FallbackAttribute> #SUB\_ADDR2\_RPS  <State>Enabled</State>  {SM\_DP\_ID\_RPS} #PROF\_TYPE2\_RPS  <AllocatedMemory>100000</AllocatedMemory>  <FreeMemory>50000</FreeMemory> #POL2\_DEL\_RPS  </ProfileInfo> |
| PROFILE3\_RPS | <ProfileInfo>  #ICCID2\_RPS #ISDP3\_RPS #MNO2\_ID\_RPS  <FallbackAttribute>TRUE</FallbackAttribute> #SUB\_ADDR2\_RPS  <State>Enabled</State>  {SM\_DP\_ID\_RPS} #PROF\_TYPE2\_RPS  <AllocatedMemory>100000</AllocatedMemory>  <FreeMemory>50000</FreeMemory> #POL2\_DEL\_RPS  </ProfileInfo> |
| SC3\_DR\_HOST\_RPS | <ScenarioParameter>#SC3\_DR\_HOST</ScenarioParameter> |
| SC3\_DR\_RPS | <ScenarioParameter>#SC3\_DR</ScenarioParameter> |
| SC3\_NO\_DR\_RPS | <ScenarioParameter>#SC3\_NO\_DR</ScenarioParameter> |
| SD\_ISDP2\_RPS | <sd-aid>#ISD\_P\_AID2</sd-aid> |
| SIGNATURE\_RPS | <Signature>{SIGNATURE}</Signature>  see Note 5 |
| SIGNED\_INFO\_RPS | <ds:SignedInfo>  <ds:CanonicalizationMethod [Algorithm=”http://www.w3.org/2001/10/xml](http://www.w3.org/2001/10/xml-exc-c14n)-exc-c14n”/> |

|  |  |
| --- | --- |
| **RPS element name** | **Value** |
|  | <ds:SignatureMethod [Algorithm=”http://www.w3.org/2001/04/xmldsig](http://www.w3.org/2001/04/xmldsig-)- more#ecdsa-sha256”/>  <ds:Reference>  <ds:DigestMethod [Algorithm=”http://www.w3.org/2001/04/xmlenc#sha](http://www.w3.org/2001/04/xmlenc#sha) 256”/>  <ds:DigestValue>{DIGEST}</ds:DigestValue>  </ds:Reference>  </ds:SignedInfo> |
| SM\_DP\_S\_ID\_RPS | <Smdp-id>#SM\_DP\_S\_ID</Smdp-id> |
| SM\_DP\_UT\_ID\_RPS | <Smdp-id>#SM\_DP\_ID</Smdp-id> |
| SM\_SR\_S\_ID\_RPS | <SmSr-id>#SM\_SR\_S\_ID</SmSr-id> |
| SM\_SR\_UT\_ID\_RPS | <SmSr-id>#SM\_SR\_ID</SmSr-id> |
| SMALL\_MEM\_RPS | <RequiredMemory>#SMALL\_MEM</RequiredMemory> |
| SUB\_ADDR1\_RPS | <SubscriptionAddress>  <Msisdn>#MSISDN1</Imsi>  <Imsi>#IMSI1</Imsi>  </SubscriptionAddress> |
| SUB\_ADDR2\_RPS | <SubscriptionAddress>  <Msisdn>#MSISDN2</Imsi>  <Imsi>#IMSI2</Imsi>  </SubscriptionAddress> |
| SUB\_ADDR3\_RPS | <SubscriptionAddress>  <Msisdn>#MSISDN3</Imsi>  <Imsi>#IMSI3</Imsi>  </SubscriptionAddress> |
| TGT\_SR\_S\_ID\_RPS | <Target-SmSr-id>#SM\_SR\_S\_ID</Target-SmSr-id> |
| TGT\_SR\_S\_UNK\_ID\_RPS | <Target-SmSr-id>#UNKNOWN\_SM\_SR\_ID</Target-SmSr-id> |
| TGT\_SR\_UT\_ID\_RPS | <Target-SmSr-id>#SM\_SR\_ID</Target-SmSr-id> |
| TGT\_UK\_SR\_S\_ID\_RPS | <Target-SmSr-id>#UNKNOWN\_SM\_SR\_ID</Target-SmSr-id> |
| TIMESTAMP\_RPS | <completionTimestamp>{CURRENT\_DATE}</completionTimestamp> |
| VALID\_SR\_CERTIF\_RPS | <smsrCertificate> '7F21'{L}#VALID\_SM\_SR\_CERTIFICATE  </smsrCertificate> |
| VIRTUAL\_EID\_RPS | <Eid>#VIRTUAL\_EID</Eid> |
| VIRTUAL\_EID2\_RPS | <Eid>#VIRTUAL\_EID2</Eid> |
| *Note 1: The {SIGNATURE} SHALL be generated with the #EUM\_S\_SK\_ECDSA Note 2: The {SIGNATURE} SHALL NOT be generated with the #EUM\_S\_SK\_ECDSA Note 3: The {SIGNATURE} SHALL be generated with the #EUM\_S\_SK\_ECDSA Note 4: The #SCP80\_KVN SHALL be converted in Integer*  *Note 5: The {SIGNATURE} SHALL use the {RC} (see the method STORE\_ISDR\_KEYS defined in* [*Annex D*](#_bookmark281) *to have more details on the way to generate the signature)*  *Note 6: As this RPS element is used to execute non-nominal tests, the content of the C-APDUs SHOULD NOT be executed on the eUICC (i.e. the C-APDUs do not have to be relevant)*  *Note 7: Void*  *Note 8: The ISD-R definition SHALL NOT contain the keyset information.* | |

###### Table 12: RPS Elements

## Profiles Information

Here is the different Profiles information used to execute the test cases defined in the section

5.3 of this Test Plan. This information is related to:

* the Profiles pre-installed on the eUICC
* the Profile that is dynamically loaded on the eUICC

The different values SHALL be either provided by the eUICC Manufacturer or the MNO owning the new Profile.

|  |  |
| --- | --- |
| **Profile information** | **Description** |
| EIS\_RPS | The eUICC Information Set (RPS format) related to the eUICC. The different data SHALL be consistent with the state of the eUICC after the manufacturing. The eUICC Manufacturer SHALL give, at least, these values:   * EID (i.e. #EID) * EUM Identifier * production date * platform type * platform version * remaining memory * available memory for Profiles * all Profiles pre-installed information with (for each one)   + ICCID (i.e. #ICCID if the Profile is Enabled)   + ISD-P AID (i.e. #DEFAULT\_ISD\_P\_AID if the Profile is Enabled)   + MSISDN (i.e. #MSISDN if the Profile is Enabled)   + Fall-back Attribute   + state   + Profile type   + allocated memory   + POL2 * ISD-R information with   + AID (i.e. #ISD\_R\_AID)   + SIN   + SDIN   + SCP80 and/or SCP81 keysets information * ECASD information with   + AID (i.e. #ECASD\_AID)   + SIN   + SDIN   + certificate (i.e. #ECASD\_CERTIFICATE) * eUICC capabilities   + supported CAT\_TP version and/or supported HTTPS version     - depends if O\_HTTPS and O\_CAT\_TP are supported   + supported secured packet version   + supported remote provisioning version * The EUM X.509 certificate containg the #EUM\_PK\_ECDSA |

|  |  |
| --- | --- |
| **Profile information** | **Description** |
|  | The tool provider SHALL format the data (i.e. RPS) and add:   * the SM-SR-UT Identifier (i.e. #SM\_SR\_ID) * the SM-DP-UT Identifier (i.e. #SM\_DP\_ID) if required * the ISD-P Executable Load File AID (i.e. #ISD\_P\_PKG\_AID) * the ISD-P Executable Module AID (i.e. #ISD\_P\_MOD\_AID) * the MNO Identifier of the pre-installed Profiles (i.e. #MNO2\_S\_ID SHALL be set on the default Enabled Profile) * the signature using the #EUM\_S\_PK\_ECDSA |
| ICCID | The ICCID of the default Profile pre-installed on the eUICC. |
| MSISDN | The MSISDN of the default Profile pre-installed on the eUICC. A network connectivity SHALL be available with this mobile subscription. |
| NEW\_ICCID | The ICCID of the new Profile dynamically downloaded on the eUICC. This ICCID SHALL NOT be present on the #EIS\_RPS. |
| NEW\_MSISDN | The MSISDN of the new Profile dynamically downloaded on the eUICC. This MSISDN SHALL NOT be present on the #EIS\_RPS. A network connectivity SHALL be available with this mobile subscription. |
| MNO1\_CON\_NAN | The NAN, of the new Profile dynamically downloaded on the eUICC, which allows MNO’s network connection. |
| MNO1\_CON\_LOGIN | The NAN related login, of the new Profile dynamically downloaded on the eUICC, which allows MNO’s network connection. |
| MNO1\_CON\_PWD | The NAN related password, of the new Profile dynamically downloaded on the eUICC, which allows MNO’s network connection. |
| MNO1\_CON\_TON\_NPI | The TON and NPI of the MNO that owns the new Profile dynamically downloaded on the eUICC. |
| MNO1\_CON\_DIAL\_NUM | The dialing number of the MNO that owns the new Profile dynamically downloaded on the eUICC. |
| MNO2\_CON\_NAN | The NAN, of the Enabled Profile pre-installed on the eUICC, which allows MNO’s network connection. |
| MNO2\_CON\_LOGIN | The NAN related login, of the Enabled Profile pre-installed on the eUICC, which allows MNO’s network connection. |
| MNO2\_CON\_PWD | The NAN related password, of the Enabled Profile pre-installed on the eUICC, which allows MNO’s network connection. |
| MNO2\_CON\_TON\_NPI | The TON and NPI of the MNO that owns the Enabled Profile pre-installed on the eUICC. |
| MNO2\_CON\_DIAL\_NUM | The dialing number of the MNO that owns the Enabled Profile pre-installed on the eUICC. |
| SM\_SR\_DEST\_ADDR | The destination address of the SM-SR-UT. |
| SM\_SR\_UDP\_IP | The UDP IP of the SM-SR-UT related to the CAT\_TP implementation. |
| SM\_SR\_UDP\_PORT | The UDP port of the SM-SR-UT related to the CAT\_TP implementation. |
| SM\_SR\_TCP\_IP | The TCP IP of the SM-SR-UT related to the HTTPS implementation. |
| SM\_SR\_TCP\_PORT | The TCP port of the SM-SR-UT related to the HTTPS implementation. |
| SM\_SR\_HTTP\_URI | The URI of the SM-SR-UT related to the HTTPS implementation. |
| SM\_SR\_HTTP\_HOST | The HOST of the SM-SR-UT related to the HTTPS implementation. |

###### Table 13: Profiles Information

## Profile Package Description

Here is a description of the Profile Package content that SHOULD be used during the testing of the Profile download process (see section [4.2.18](#_bookmark127)). Some parts of this PEs list MAY be adapted according to the eUICC implementation.

This Profile, defined in [Table 14**: Profile Package Content**](#_bookmark275), 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
* RFM application

The parameters below have been chosen to personalize the Profile:

* Profile type: "GSMA Profile Package"
* ICCID: '89019990001234567893'
* IMSI: 234101943787656
* MNO-SD AID / TAR: 'A000000151000000' / 'B20100'
* UICC RFM application AID / TAR: 'A00000055910100001' / 'B00000'
* USIM RFM application AID / TAR: ' A00000055910100002' / 'B00020'
* Executable Load File AID for SD: 'A0000001515350'
* Executable Module AID for SD: 'A000000151000000'
* SSD AID / TAR: 'A00000055910100102736456616C7565' / '6C7565'
* All access rules are defined in the [Table 15](#_bookmark278)

Note that all these parameters MAY be freely adapted if necessary.

### Profile Package Content

The #PROFILE\_PACKAGE SHOULD be the result of the concatenation of the different PEs described below (respecting the order).

|  |  |
| --- | --- |
| **ASN.1 format** | **DER TLV format** |
| **PE\_HEADER** | |
| headerValue ProfileElement ::= header : { major-version 2,  minor-version 1,  profileType "GSMA Profile Package", iccid '89019990001234567893'H,  eUICC-Mandatory-services { usim NULL,  milenage NULL, javacard NULL  },  eUICC-Mandatory-GFSTEList {  *-- see Note 1*  id-MF, id-USIM  },  *-- These SMS Connectivity Parameters MAY be freely changed*  connectivityParameters 'A0090607#TON\_NPI#DIALING\_NUMBER'H  } | A0 4F  80 01 02  81 01 01  82 14 47534D412050726F66696C65205061636B616765  83 0A 89019990001234567893  A5 06  81 00  84 00  8B 00  A6 10  06 06 67810F010201  06 06 67810F010204  87 0B A0090607913386994211F0 |
| **PE\_MF** | |
| mfValue ProfileElement ::= mf : { mf-header {  mandated NULL, identification 1  },  templateID id-MF, mf {  fileDescriptor : {  pinStatusTemplateDO '01020A'H | B0 8201F8 A0 05  80 00  81 01 01  81 06 67810F010201  A2 07  A1 05  C6 03 01020A |

}

},

ef-pl { fileDescriptor : {

A3 05

A1 03

*-- EF PL modified to use Access Rule 15 within EF ARR*

securityAttributesReferenced '0F'H

}

},

ef-iccid {

8B 01 0F

A4 0C

*-- swapped ICCID: 98109909002143658739*

fillFileContent '98109909002143658739'H

},

ef-dir { fileDescriptor {

83 0A 98109909002143658739

A5 27

A1 09

*-- Shareable Linear Fixed File*

*-- 4 records, record length: 38 bytes*

fileDescriptor '42210026'H, efFileSize '98'H

},

82 04 42210026

80 01 98

*-- USIM AID: A0000000871002FF33FF018900000100*

fillFileContent '61184F10A0000000871002FF33FF01890000010050045553494D'H

},

ef-arr { fileDescriptor {

83 1A

61184F10A0000000871002FF33FF01890000010050045553494D

A6 82019E A1 0A

*-- Shareable Linear Fixed File*

*-- 15 records, record length: 37 bytes*

*-- ARR created with content defined in Annex B.7.2*

*-- plus one additional record for use with EF PL*

fileDescriptor '42210025'H, efFileSize '022B'H

},

82 04 42210025

80 02 022B

*-- see Table 15 to see the access rules definitions*

fillFileContent '#ACCESS\_RULE1'H, fillFileOffset 10, fillFileContent '#ACCESS\_RULE2'H, fillFileOffset 15, fillFileContent '#ACCESS\_RULE3'H,

83 1B #ACCESS\_RULE1

82 01 0A

83 16 #ACCESS\_RULE2

82 01 0F

83 0B #ACCESS\_RULE3

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  |  | fillFileOffset 26, | 82 01 | 1A |
|  |  | fillFileContent '#ACCESS\_RULE4'H, | 83 0A | #ACCESS\_RULE4 |
|  |  | fillFileOffset 27, | 82 01 | 1B |
|  |  | fillFileContent '#ACCESS\_RULE5'H, | 83 16 | #ACCESS\_RULE5 |
|  |  | fillFileOffset 15, | 82 01 | 0F |
|  |  | fillFileContent '#ACCESS\_RULE6'H, | 83 16 | #ACCESS\_RULE6 |
|  |  | fillFileOffset 15, | 82 01 | 0F |
|  |  | fillFileContent '#ACCESS\_RULE7'H, | 83 21 | #ACCESS\_RULE7 |
|  |  | fillFileOffset 4, | 82 01 | 04 |
|  |  | fillFileContent '#ACCESS\_RULE8'H, | 83 21 | #ACCESS\_RULE8 |
|  |  | fillFileOffset 4, | 82 01 | 04 |
|  |  | fillFileContent '#ACCESS\_RULE9'H, | 83 1B | #ACCESS\_RULE9 |
|  |  | fillFileOffset 10, | 82 01 | 0A |
|  |  | fillFileContent '#ACCESS\_RULE10'H, | 83 10 | #ACCESS\_RULE10 |
|  |  | fillFileOffset 21, | 82 01 | 15 |
|  |  | fillFileContent '#ACCESS\_RULE11'H, | 83 15 | #ACCESS\_RULE11 |
|  |  | fillFileOffset 16, | 82 01 | 10 |
|  |  | fillFileContent '#ACCESS\_RULE12'H, | 83 10 | #ACCESS\_RULE12 |
|  |  | fillFileOffset 21, | 82 01 | 15 |
|  |  | fillFileContent '#ACCESS\_RULE13'H, | 83 16 | #ACCESS\_RULE13 |
|  |  | fillFileOffset 15, | 82 01 | 0F |
|  |  | fillFileContent '#ACCESS\_RULE14'H, | 83 0B | #ACCESS\_RULE14 |
|  |  | fillFileOffset 26, | 82 01 | 1A |
|  |  | fillFileContent '8001019000800102A010A40683010195 | 83 25 | 8001019000800102A010A40683010195 |
|  |  | 0108A406830102950108800158A40683 |  | 0108A406830102950108800158A40683 |
|  |  | 010A950108'H |  | 010A950108 |
|  | } |  |  |  |
| } |  |  |  |  |
| **PE\_PUK** | | | | |
| pukVal ProfileElement ::= pukCodes : { puk-Header {  mandated NULL, identification 2  },  pukCodes { | | | A3 3F |  |
| A0 05 |  |
| 80 00 |  |
| 81 01 | 02 |
| A1 36 |  |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| { | | | |  |  | 30 11 |  |  |
| keyReference pukAppl1, | | | | 80 01 | 01 |
| pukValue '3030303030303030'H, | | | | 81 08 | 3030303030303030 |
|  | } | } | *-- maxNumOfAttemps:9, retryNumLeft:9*  maxNumOfAttemps-retryNumLeft 153  },  {  keyReference pukAppl2, pukValue '3132333435363738'H  },  {  keyReference secondPUKAppl1, pukValue '3932393435363738'H,  *-- maxNumOfAttemps:8, retryNumLeft:8*  maxNumOfAttemps-retryNumLeft 136  } | 82 02 | 0099 |  |
| 30 0D |  |  |
| 80 01 | 02 |  |
| 81 08 | 3132333435363738 |  |
| 30 12 |  |  |
| 80 02 | 0081 |  |
| 81 08 | 3932393435363738 |  |
| 82 02 | 0088 |  |
| **PE\_PIN** | | | | | | | | |
|  | | | |  |  |  | | |
|  | pinVal ProfileElement ::= pinCodes : { pin-Header {  mandated NULL, identification 3  },  pinCodes pinconfig : {  {  keyReference pinAppl1, pinValue '31323334FFFFFFFF'H,  unblockingPINReference pukAppl1  },  {  keyReference pinAppl2, pinValue '30303030FFFFFFFF'H,  unblockingPINReference pukAppl2  },  { | | | A2 41 | |  |
| A0 05 | |
| 80 00 | |
| 81 01 03 | |
| A1 38 A0 36 | |
| 30 10 | |
| 80 01 01 | |
| 81 08 31323334FFFFFFFF | |
| 82 01 01 | |
| 30 10 | |
| 80 01 02 | |
| 81 08 30303030FFFFFFFF | |
| 82 01 02 | |
| 30 10 | |

|  |  |
| --- | --- |
| keyReference adm1,  pinValue '35363738FFFFFFFF'H,  pinAttributes 1  }  }  } | 80 01 0A  81 08 35363738FFFFFFFF  83 01 01 |
| **PE\_USIM** | |
| usimValue ProfileElement ::= usim : { usim-header {  mandated NULL, identification 4  },  templateID id-USIM, 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 {  *-- Service Dialling Numbers, Short Message Storage…*  fillFileContent '0A2E178CE73204000000000000'H  },  ef-spn {  *-- ASCII format: "GSMA eUICC"*  fillFileContent '0247534D41206555494343FFFFFFFFFFFF'H | B3 7C  A0 05  80 00  81 01 04  81 06 67810F010204 A2 1D  A1 1B  83 02 7FF1  84 10 A0000000871002FF33FF018900000100  C6 03 01810A  A3 0B  83 09 082943019134876765  A4 06  A1 04  C7 02 2F06  A8 0F  83 0D 0A2E178CE73204000000000000  AD 13  83 11 0247534D41206555494343FFFFFFFFFFFF |

|  |  |
| --- | --- |
| },  ef-est {  *-- Services deactivated*  fillFileContent '00'H  },  ef-acc {  *-- Access class 4*  fillFileContent '0040'H  },  ef-ecc {  *-- Emergency Call Code 911*  fillFileContent '19F1FF01'H  }  } | AE 03  83 01 00  B2 04  83 02 0040  B6 06  83 04 19F1FF01 |
| **PE\_USIM\_PIN** | |
| usimPin ProfileElement ::= pinCodes : { pin-Header {  mandated NULL, identification 05  },  pinCodes pinconfig : {  {  keyReference secondPINAppl1, pinValue '39323338FFFFFFFF'H  unblockingPINReference secondPUKAppl1,  *-- PIN is Enabled*  pinAttributes 1,  *-- maxNumOfAttemps:2, retryNumLeft:2*  maxNumOfAttemps-retryNumLeft 34  }  }  } | A2 25  A0 05  80 00  81 01 05  A1 1C A0 1A  30 18  80 02 0081  81 08 39323338FFFFFFFF  82 02 0081  83 01 01  84 01 22 |
| **PE\_NAA** | |
| akaParamValue ProfileElement ::= akaParameter : { | A4 3A |

|  |  |  |  |
| --- | --- | --- | --- |
| } | aka-header { mandated NULL, identification 6  },  algoConfiguration algoParameter : { algorithmID milenage,  *-- RES and MAC 64 bits, CK and IK 128 bits*  algorithmOptions '01'H,  key '000102030405060708090A0B0C0D0E0F'H, opc '0102030405060708090A0B0C0D0E0F00'H,  *-- rotationConstants uses default: '4000204060'H*  *-- xoringConstants uses default value*  authCounterMax '010203'H  }  *-- sqnOptions uses default: '02'H*  *-- sqnDelta uses default: '000010000000'H*  *-- sqnAgeLimit uses default: '000010000000'H*  *-- sqnInit uses default: all bytes zero* | A0 05 | |
| 80 00 | |
| 81 01 06 | |
| A1 31 A1 2F | |
| 80 01 01 | |
| 81 01 01 | |
| 82 10 000102030405060708090A0B0C0D0E0F | |
| 83 10 0102030405060708090A0B0C0D0E0F00 | |
| 86 03 010203 | |
| **PE\_MNO\_SD** | | | |
| mnoSdValue ProfileElement ::= securityDomain : { sd-Header {  mandated NULL, identification 7  },  instance {  applicationLoadPackageAID 'A0000001515350'H, classAID 'A000000151535041'H,  instanceAID 'A000000151000000'H,  applicationPrivileges '82FC80'H,  *-- Personalised*  lifeCycleState '0F'H,  *-- SCP80 supported*  applicationSpecificParametersC9 '81028000'H,  -- *other parameters MAY be necessary*  applicationParameters { | | A6 820104 |  |
| A0 05 |  |
| 80 00 |  |
| 81 01 | 07 |
| A1 3E |  |
| 4F 07 | A0000001515350 |
| 4F 08 | A000000151535041 |
| 4F 08 | A000000151000000 |
| 82 03 | 82FC80 |
| 83 01 | 0F |
| C9 04 | 81028000 |
| EA 11 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| *-- TAR: B20100, MSL: 12*  uiccToolkitApplicationSpecificParametersField '0100000100000002011203B2010000'H  }  },  keyList {  {  *-- C-ENC + R-ENC*  keyUsageQualifier '38'H,  *-- ENC key* keyIdentifier '01'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '112233445566778899AABBCCDDEEFF10'H  }  }  },  {  *-- C-MAC + R-MAC*  keyUsageQualifier '34'H,  *-- MAC key* keyIdentifier '02'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '112233445566778899AABBCCDDEEFF10'H  }  }  },  {  *-- C-DEK + R-DEK* |  |  | 80 0F  0100000100000002011203B2010000  A2 81BA  30 22  95 01 38  82 01 01  83 01 01  30 17  30 15  80 01 80  86 10 112233445566778899AABBCCDDEEFF10  30 22  95 01 34  82 01 02  83 01 01  30 17  30 15  80 01 80  86 10 112233445566778899AABBCCDDEEFF10  30 22 |

|  |  |  |  |
| --- | --- | --- | --- |
| keyUsageQualifier 'C8'H,  *-- data ENC key* keyIdentifier '03'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '112233445566778899AABBCCDDEEFF10'H  }  }  },  *-- AES Token Key (as an example)*  *-- This value MAY be freely changed*  keyUsageQualifier '81'H,  *-- MAY be used by SD*  keyAccess '01'H,  *-- Key Id 01*  keyIdentifier '01'H, keyVersionNumber '70'H, keyCompontents {  {  *-- AES (16 bytes key length)*  *-- This value MAY be freely changed*  keyType '88'H,  *-- This value MAY be freely changed*  keyData 'CDFE56B7B72FAE6A047341F003D7A48D'H  }  }  },  {  *-- Receipt (the AES scheme SHALL be supported)*  keyUsageQualifier '44'H,  *-- MAY be used by SD* |  |  | 95 01 C8  82 01 03  83 01 01  30 17  30 15  80 01 80  86 10 112233445566778899AABBCCDDEEFF10  30 25  95 01 81  96 01 01  82 01 01  83 01 70  30 17  30 15  80 01 88  86 10 CDFE56B7B72FAE6A047341F003D7A48D  30 25  95 01 44  96 01 01 |

|  |  |  |  |
| --- | --- | --- | --- |
| keyAccess '01'H,  *-- Key Id 01*  keyIdentifier '01'H, keyVersionNumber '71'H, keyCompontents {  {  *-- AES (16 bytes key length)*  keyType '88'H,  *-- This value MAY be freely changed*  keyData '11121314212223243132333441424344'H  }  }  }  }  } | 82 | 01 01 |  |
| 83 | 01 71 |  |
| 30 | 17 |  |
|  | 30 15 |  |
|  | 80 01 | 88 |
|  | 86 10 | 11121314212223243132333441424344 |
| **PE\_SSD** | | | |
| ssdValue ProfileElement ::= securityDomain : { | A6 81C0 | | |
| sd-Header { | A0 05 | | |
| mandated NULL, | 80 00 | | |
| identification 8 | 81 01 08 | | |
| }, |  | | |
| instance { | A1 49 | | |
| applicationLoadPackageAID 'A0000001515350'H, | 4F 07 A0000001515350 | | |
| classAID 'A000000151535041'H, | 4F 08 A000000151535041 | | |
| instanceAID 'A00000055910100102736456616C7565'H, | 4F 10 A00000055910100102736456616C7565 | | |
| *-- by default extradited under MNO-SD* |  | | |
| *-- Privileges: Security Domain + Trusted Path* |  | | |
| applicationPrivileges '808000'H, | 82 03 808000 | | |
| *-- Personalized* |  | | |
| lifeCycleState '0F'H, | 83 01 0F | | |
| *-- SCP80 supported, extradiction supported* |  | | |
| applicationSpecificParametersC9 '810280008201F0'H, | C9 07 810280008201F0 | | |
| applicationParameters { | EA 11 | | |
| *-- TAR: 6C7565, MSL: 12* |  | | |

|  |  |  |  |
| --- | --- | --- | --- |
| uiccToolkitApplicationSpecificParametersField '01000001000000020112036C756500'H  }  },  keyList {  {  *-- C-ENC + R-ENC*  keyUsageQualifier '38'H,  keyIdentifier '01'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '11223344556677881122334455667788'H  }  }  },  {  *-- C-MAC + R-MAC*  keyUsageQualifier '34'H,  *-- MAC key* keyIdentifier '02'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '11223344556677881122334455667788'H  }  }  },  {  *-- C-DEK + R-DEK*  keyUsageQualifier 'C8'H, |  |  | 80 0F  01000001000000020112036C756500  A2 6C  30 22  95 01 38  82 01 01  83 01 01  30 17  30 15  80 01 80  86 10 11223344556677881122334455667788  30 22  95 01 34  82 01 02  83 01 01  30 17  30 15  80 01 80  86 10 11223344556677881122334455667788  30 22  95 01 C8 |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *-- data ENC key* keyIdentifier '03'H, keyVersionNumber '01'H, keyCompontents {  {  *-- DES mode implicitly known (as an example)*  keyType '80'H,  *-- This value MAY be freely changed*  keyData '11223344556677881122334455667788'H  }  }  }  }  } | | 82 01 03 | |  |  |
| 83 01 01 | |  |  |
| 30 17 | |  |  |
| 30 15 | |  |  |
| 80 01 | | 80 |  |
| 86 10 | | 11223344556677881122334455667788 |  |
| **PE\_RFM\_UICC** | | | | | |
| rfmUicc ProfileElement ::= rfm : { rfm-header {  identification 11  },  *-- Instance AID*  instanceAID ' A00000055910100001'H,  tarList { 'B00000'H  },  *-- cryptographic checksum + counter higher*  minimumSecurityLevel '12'H,  *-- full access*  uiccAccessDomain '00'H,  *-- full access*  uiccAdminAccessDomain '00'H  } | | A7 20 | | | |
| A0 03 | | | |
| 81 01 0B | | | |
| 4F 09 A00000055910100001 | | | |
| A0 05 04 03 | | | |
| B00000 | | | |
| 81 01 12 | | | |
| 04 01 00 | | | |
| 04 01 00 | | | |
| **PE\_RFM\_USIM** | | | | | |
|  | |  | | | |
|  | rfmUsim ProfileElement ::= rfm : {  rfm-header { |  | A7 40 | |  |
| A0 03 | |

|  |  |
| --- | --- |
| identification 12  },  *-- Instance AID*  instanceAID 'A00000055910100002'H,  tarList { 'B00020'H  },  *-- cryptographic checksum + counter higher*  minimumSecurityLevel '12'H,  *-- full access*  uiccAccessDomain '00'H,  *-- full access* uiccAdminAccessDomain '00'H, adfRFMAccess {  adfAID 'A0000000871002FF33FF018900000100'H,  *-- UICC access condition: ADM1*  adfAccessDomain '02000100'H,  *-- UICC access condition: ADM1*  adfAdminAccessDomain '02000100'H  }  } | 81 01 0C  4F 09 A00000055910100002  A0 05  04 03 B00020  81 01 12  04 01 00  04 01 00  30 1E  80 10 A0000000871002FF33FF018900000100  81 04 02000100  82 04 02000100 |
| **PE\_END** | |
| endValue ProfileElement ::= end : { end-header {  mandated NULL, identification 99  }  } | AA 07  A0 05  80 00  81 01 63 |
| *Note: The rule related to the usage of curly brackets defined in section* [*2.2.3*](#_bookmark35) *SHALL NOT apply for the elements described in the column “ASN.1 format” of this table. Note 1: The following OIDs are used:*  *id-MF OBJECT IDENTIFIER ::=*  *{joint-iso-itu-t(2) international-organizations(23) simalliance(143) euicc-profile(1) template(2) mf(1)} id-USIM OBJECT IDENTIFIER ::=* | |

*{joint-iso-itu-t(2) international-organizations(23) simalliance(143) euicc-profile(1) template(2) usim(4)}*

*These OIDs allow identifying the templates used to accelerate the creation of the file system in the Profile as defined in the SIMAlliance Profile Package specification* [*[16].*](#_bookmark21)

###### Table 14: Profile Package Content

### Access Rules

Here are the access rules used in the Profile Package content defined in [Profile Package **Content**](#_bookmark276).

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **Access rule name** | **File access conditions** | | | | | | **Hexadecimal value** |
| **READ** | **UPDATE** | **INCREASE** | **ACTIVATE** | **DEACTIVATE** | **DELETE** |
| ACCESS\_RULE1 | ALWAYS | PIN1 | NEVER | ADM1 | ADM1 | ADM1 | 8001019000  800102A406830101950108  800158A40683010A950108 |
| ACCESS\_RULE2 | PIN1 | ADM1 | NEVER | ADM1 | ADM1 | ADM1 | 800101A406830101950108  80015AA40683010A950108 |
| ACCESS\_RULE3 | ADM1 | ADM1 | NEVER | ADM1 | ADM1 | ADM1 | 80015BA40683010A950108 |
| ACCESS\_RULE4 | ALWAYS | NEVER | NEVER | NEVER | NEVER | ADM1 | 8001019000  80015A9700 |
| ACCESS\_RULE5 | PIN1 | PIN1 | NEVER | ADM1 | ADM1 | ADM1 | 800103A406830101950108  800158A40683010A950108 |
| ACCESS\_RULE6 | PIN1 | ADM1 | NEVER | PIN1 | ADM1 | ADM1 | 800111A406830101950108  80014AA40683010A950108 |
| ACCESS\_RULE7 | PIN1 | PIN1 | PIN1 | ADM1 | ADM1 | ADM1 | 800103A406830101950108  800158A40683010A950108  840132A406830101950108 |
| ACCESS\_RULE8 | PIN1 | PIN2 | NEVER | ADM1 | ADM1 | ADM1 | 800101A406830101950108  800102A406830181950108  800158A40683010A950108 |
| ACCESS\_RULE9 | ALWAYS | PIN1 | NEVER | PIN1 | PIN1 | ADM1 | 8001019000  80011AA406830101950108 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  | 800140A40683010A950108 |
| ACCESS\_RULE10 | ALWAYS | ADM1 | NEVER | ADM1 | ADM1 | ADM1 | 8001019000  80015AA40683010A950108 |
| ACCESS\_RULE11 | ALWAYS | NEVER | NEVER | ADM1 | ADM1 | NEVER | 8001019000  800118A40683010A950108  8001429700 |
| ACCESS\_RULE12 | PIN1 | NEVER | NEVER | NEVER | NEVER | NEVER | 800101A406830101950108  80015A9700 |
| ACCESS\_RULE13 | PIN1 | PIN1 | NEVER | PIN1 | ADM1 | ADM1 | 800113A406830101950108  800148A40683010A950108 |
| **Access rule name** | **MF/ADF/DF access conditions** | | | | | | **Hexadecimal value** |
| **DELETE**  **self** | **TERMINATE** | **ACTIVATE** | **DEACTIVATE** | **CREATE DF** | **CREATE EF** |
| ACCESS\_RULE14 | ADM1 | NEVER | ADM1 | ADM1 | ADM1 | ADM1 | 80015EA40683010A950108 |
| *Note: These access rules strictly follow the definition provided in the SIMAlliance Profile Package specification* [*[16]*](#_bookmark21) *(section 9.9)* | | | | | | | |

###### Table 15: Access Rules

### Additional Profile Elements

Here are additional Profile Elements that SHALL be added to the Profile Package content defined above in order to execute the tests defined in section [5.2](#_bookmark202):

* + - * #PE\_APPLET1: This PE allows loading and instantiating the Applet 1 defined in section [A.1](#_bookmark254)
      * #PE\_APPLET3: This PE allows loading and instantiating the Applet 3 defined in section [A.3](#_bookmark262)
      * #PE\_EF1122: This PE allows creating an EF with the identifier ‘1122’. This transparent file is 16 bytes long, activated and present under the MF ‘3F00’

|  |  |
| --- | --- |
| **ASN.1 format** | **DER TLV format** |
| **PE\_APPLET1** | |
| applet1 ProfileElement ::= application : { app-Header {  mandated NULL, identification 9  },  loadBlock {  loadPackageAID 'A000000559101001'H,  loadBlockObject '{LFDB\_APPLET1}'H  },  instanceList {  {  applicationLoadPackageAID 'A000000559101001'H, classAID 'A000000559101001112233'H, instanceAID 'A00000055910100111223301'H,  applicationPrivileges '000000'H,  *-- Selectable by default*  applicationSpecificParametersC9 '00'H, applicationParameters {  uiccToolkitApplicationSpecificParametersField  *-- TAR: 112233*  '0100000000000311223300'H  }  }  }  }  see Note 1 | A8 {L}  A0 05  80 00  81 01 09  A1 {L}  4F 08 A000000559101001 C4 {L} {LFDB\_APPLET1}  A2 3E  30 3C  4F 08 A000000559101001  4F 0B A000000559101001112233  4F 0C A00000055910100111223301  82 03 000000  C9 01 00 EA 0D  80 0B  0100000000000311223300 |

|  |  |
| --- | --- |
| **PE\_APPLET3** | |
| applet3 ProfileElement ::= application : { app-Header {  mandated NULL, identification 10  },  loadBlock {  loadPackageAID 'A000000559101003'H,  loadBlockObject '{LFDB\_APPLET3}'H  },  instanceList {  {  applicationLoadPackageAID 'A000000559101003'H, classAID 'A000000559101003445566'H, instanceAID 'A00000055910100344556601'H,  applicationPrivileges '000000'H, applicationSpecificParametersC9 '00'H  }  }  }  see Note 1 | A8 {L}  A0 05  80 00  81 01 0A  A1 {L}  4F 08 A000000559101003 C4 {L} {LFDB\_APPLET3}  A2 2F  30 2D  4F 08 A000000559101003  4F 0B A000000559101003445566  4F 0C A00000055910100344556601  82 03 000000  C9 01 00 |
| **PE\_EF1122** | |

|  |  |
| --- | --- |
| ef1122 ProfileElement ::= genericFileManagement : { gfm-header {  mandated NULL, identification 22  },  fileManagementCMD {  {  createFCP {  *-- Transparent File* fileDescriptor '0121'H, fileID '1122'H,  *-- reference to the #ACCESS\_RULE1* securityAttributesReferenced '2F0601'H, efFileSize '10'H,  shortEFID ''H  },  fillFileContent '1122334455'H  }  }  }  see Note 2 | A1 26  A0 05  80 00  81 01 16  A1 1D  30 1B  62 12  82 02 0121  83 02 1122  8B 03 2F0601  80 01 10  88 00  81 05 1122334455 |
| *Note: The rule related to the usage of curly brackets defined in section* [*2.2.3*](#_bookmark35) *SHALL NOT apply for the elements described in the column “ASN.1 format”. Note 1: This PE SHALL be added just after the #PE\_SSD.*  *Note 2: This PE SHALL be added just after the #PE\_PIN.* | |

###### Table 16: Additional Profile Elements

# Annex C Dynamic Content

Here are the different dynamic values used in the test cases defined in this document. These values SHOULD be either calculated by the test tools or generated dynamically by an entity under test.

|  |  |
| --- | --- |
| **Variable name** | **Description** |
| ACK\_NUM | CAT\_TP PDU acknowledgment number (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| CARD\_CHALLENGE | Pseudo-random value (8 bytes long). |
| CARD\_CRYPTOGRAM | Card cryptogram as defined in GlobalPlatform Card Specification - Amendment D  [[11]](#_bookmark16) (8 bytes long). |
| CC | Cryptographic Checksum as defined in ETSI TS 102 225 [[4]](#_bookmark9) (8 bytes long). |
| CNTR | Counter coded on 5 bytes as defined in ETSI TS 102 225 [[4].](#_bookmark9) |
| COMMAND\_SCRIPT | List of commands to execute formatted in expanded format as defined in ETSI TS 102 226 [[6].](#_bookmark11) |
| CPI | Command Packet Identifier as defined in ETSI TS 102 225 [[4].](#_bookmark9) |
| CS | CAT\_TP PDU checksum (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| CURRENT\_DATE | The current date formatted as specified by W3C: YYYY-MM-DDThh:mm:ssTZD |
| DATA | CAT\_TP PDU data as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| DATA\_LENGTH | CAT\_TP PDU data length as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| DEST\_PORT | CAT\_TP PDU destination port (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| DIGEST | SHA-256 of the data to sign. |
| DR | Derivation Random as defined in GlobalPlatform Card Specification v.2.2 Amendment E [[12]](#_bookmark17) (Confidential Setup of Secure Channel Keys using ECKA). |
| FUNC\_CALL\_ID | Identification of a function call. This identifier enables to manage function call retry policies. As consequence, it SHALL be unique. |
| FUNCTION\_REC\_ID | Depending of the direction of the test step, this value SHALL be either:   * #SM\_DP\_ID or * #SM\_SR\_ID or * #SM\_DP\_S\_ID or * #SM\_SR\_S\_ID or * #MNO1\_S\_ID or * #MNO2\_S\_ID or * #EUM\_S\_ID |
| FUNCTION\_REQ\_ID | Depending of the direction of the test step, this value SHALL be either:   * #SM\_DP\_ID or * #SM\_SR\_ID or * #SM\_DP\_S\_ID or * #SM\_SR\_S\_ID or * #MNO1\_S\_ID or * #MNO2\_S\_ID or * #EUM\_S\_ID |
| HL | CAT\_TP PDU header length (1 byte) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| HOST\_CHALLENGE | Random value (8 bytes long). |
| HOST\_CRYPTOGRAM | Host cryptogram as defined in GlobalPlatform Card Specification - Amendment D  [[11]](#_bookmark16) (8 bytes long). |

|  |  |
| --- | --- |
| **Variable name** | **Description** |
| IDENTIFICATION\_DATA | CAT\_TP off-card entity identification data as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| KEY\_DIV\_DATA | Key diversification data as defined in GlobalPlatform Card Specification - Amendment D [[11]](#_bookmark16) (10 bytes long). |
| KEY\_KCV | The Key Check Value of the #KEY. |
| KEY\_LENGTH | Symmetric key length that SHALL be at least 16 bytes long. |
| KEYS\_ENCRYPTED | Encrypted secure channel keys used during the confidential setup. The value of each plain key is #KEY. |
| KIC | SC80 Key and algorithm Identifier for ciphering as defined in ETSI TS 102 225 [[4].](#_bookmark9) |
| KID | SCP80 Key and algorithm Identifier for RC/CC/DS as defined in ETSI TS 102 225 [[4].](#_bookmark9) |
| L | Exact length of the corresponding tag or of the remaining data. |
| LC | Exact length of a command data. |
| LFDB\_APPLET1 | Load File Data Block of the Applet1 defined in [Annex A.](#_bookmark253) |
| LFDB\_APPLET3 | Load File Data Block of the Applet3 defined in [Annex A.](#_bookmark253) |
| LOAD\_APPLET1 | List of C-APDUs that allows loading the Applet1 defined in [Annex A.](#_bookmark253) The script is composed of one INSTALL FOR LOAD and several LOAD commands. |
| LOAD\_APPLET2 | List of C-APDUs that allows loading the Applet2 defined in [Annex A.](#_bookmark253) The script is composed of one INSTALL FOR LOAD and several LOAD commands. |
| LOAD\_APPLET3 | List of C-APDUs that allows loading the Applet3 defined in [Annex A.](#_bookmark253) The script is composed of one INSTALL FOR LOAD and several LOAD commands. |
| MAC | C-MAC as defined in GlobalPlatform Card Specification – Amendment D [[11].](#_bookmark16) |
| MAX\_PDU\_SIZE | CAT\_TP maximum PDU size (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| MAX\_SDU\_SIZE | CAT\_TP maximum SDU size (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| NB\_APP | Number of applications installed. |
| NEW\_SCP81\_PSK KCV | Key check value of the #NEW\_SCP81\_PSK |
| NON\_VOLATILE\_MEMORY | Non volatile memory available. |
| NOTIF\_NUMBER | The notification sequence number as defined in GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2].](#_bookmark7) |
| PCNTR | Padding Counter coded on 1 byte as defined in ETSI TS 102 225 [[4].](#_bookmark9) |
| PK\_CASD\_CT | Symmetric or asymmetric key (depending of the implementation choice) of the MNO CASD. |
| PROFILE\_PART1 | The first part of the Profile Elements list defined by #PROFILE\_PACKAGE. This part of the Profile Package SHALL be split according the eUICC capabilities. |
| PROFILE\_PARTi | An intermediate part of the Profile Elements list defined by #PROFILE\_PACKAGE. Each middle part of the Profile Package SHALL be split according the eUICC capabilities. |
| PROFILE\_PARTn | The last part of the Profile Elements list defined by #PROFILE\_PACKAGE. This part of the Profile Package SHALL be split according the eUICC capabilities. |
| PSK\_DEK KCV | Key check value of the #PSK\_DEK |
| RC | Random Challenge as defined in GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2].](#_bookmark7) |
| REASON\_CODE | CAT\_TP reason code as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| RECEIPT | Receipt as defined in GlobalPlatform Card Specification v.2.2 Amendment E [[12]](#_bookmark17) (Confidential Setup of Secure Channel Keys using ECKA). |
| REL\_MESSAGE\_ID | Identifier of the initial message request. |
| REQ\_MESSAGE\_ID | Identifier of the message to send. It SHALL be unique and composed of the domain portion of the tool provider and an integer (or a date). |

|  |  |
| --- | --- |
| **Variable name** | **Description** |
| SCP\_KDEK | The new SCP DEK key generated on the ISD-R or the ISD-P. |
| SCP\_KENC | The new SCP ENC key generated on the ISD-R or the ISD-P. |
| SCP\_KMAC | The new SCP MAC key generated on the ISD-R or the ISD-P. |
| SCP03\_SEQ\_NUM | The SCP03 sequence number (3 bytes long). |
| SEQ\_NUM | CAT\_TP PDU sequence number (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| SIGNATURE | A signature used for key set establishment. |
| SM\_SR\_ID\_RPS | The SM-SR identifier structure used in off-card interfaces. Depending of the test, this value SHALL be either:   * #SM\_SR\_UT\_ID\_RPS or * #SM\_SR\_S\_ID\_RPS |
| SM\_DP\_ID\_RPS | The SM-DP identifier structure used in off-card interfaces. Depending of the test, this value SHALL be either:   * #SM\_DP\_UT\_ID\_RPS or * #SM\_DP\_S\_ID\_RPS |
| SRC\_PORT | CAT\_TP PDU source port (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |
| TOKEN\_KEY | The AES token key value (key version number = ‘70’) of the ISD-P (16 bytes long). |
| TOKEN\_VALUE | The token generated with the {TOKEN\_KEY} (16 bytes long). |
| UDH | User Data Header as defined in 3GPP TS 23.040 [[5].](#_bookmark10) |
| VOLATILE\_MEMORY | Volatile memory available. |
| WIN\_SIZE | CAT\_TP PDU window size port (2 bytes long) as defined in ETSI TS 102 127 [[7].](#_bookmark12) |

###### Table 17: Dynamic Content

# Annex D Methods

Here are the methods’ descriptions used in this document:

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
| *ENVELOPE\_SMS\_PP* | Generate an SMS envelope.  Parameters:   * *SPI* * *TAR* * *COMMAND1; COMMAND2…* (i.e. APDUs or TLVs) * *CHAINING\_OPT* (optional parameter)   Here is the content of the envelope SMS-PP download to send:  '80 C2 00 00 {LC} D1 {L}  82 02 83 81  86 02 80 01  8B {L}  40 05 81 12 50 F3 96 F6 22 22 22 22 22 22 22  {L} {UDH}' +  SCP80\_PACKET(*SPI*, *TAR*, *COMMAND1;COMMAND2*…, *CHAINING\_OPT)*  See [Annex C](#_bookmark280) for the definition of {UDH}.  The method SCP80\_PACKET is defined below.  If the SMS content length is higher than the SMS maximum size, it SHALL be split into several envelopes: SMS concatenation SHALL be used.  Note that the first Transport Layer Protocol values present under the tag ‘8B’ (referenced by the 3GPP TS 23.040 specification [[5]](#_bookmark10)) are informative: they MAY be freely adapted by the test tool provider if needed. |
| *HTTPS\_CONTENT* | Generate an HTTPS POST message containing APDU commands. This method is used to ask the ISD-R or the MNO-SD to execute some scripts.  Parameters:   * *APDU1; APDU2…*   Here is the TLS record (TLS\_APPLICATION) content (in ASCII) to send:  #HTTP\_CODE\_200 #X\_ADMIN\_PROTOCOL  Content-Type: application/vnd.globalplatform.card-content-  mgt;version=1.0 |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | #X\_ADMIN\_NEXT\_URI  {COMMAND\_SCRIPT}  {COMMAND\_SCRIPT} SHALL be:  'AE 80' +  '22 {L}' + *APDU1* +  '22 {L}' + *APDU2* + … + '00 00' |
| *HTTPS\_CONTENT\_ISDP* | Generate an HTTPS POST message containing some commands (i.e. ADPUs or TLVs) to the ISD-P.  Parameters:   * *ISD\_P\_TARGETED\_AID* * *COMMAND1; COMMAND2…*(i.e. APDUs or TLVs) * *CHAINING\_OPT* (optional parameter)   Here is the TLS record (TLS\_APPLICATION) content (in ASCII) to send:  #HTTP\_CODE\_200 #X\_ADMIN\_PROTOCOL  Content-Type: application/vnd.globalplatform.card-content- mgt;version=1.0  #X\_ADMIN\_NEXT\_URI   1. Admin-Targeted-Application: *ISD*\_*P*\_*TARGETED*\_*AID*   {COMMAND\_SCRIPT}   * + If the commands list is composed of APDUs:   {COMMAND\_SCRIPT} SHALL contain the list of APDUs formatted using the expanded format with indefinite length as defined in ETSI TS 102 226 [[6].](#_bookmark11)  If *CHAINING\_OPT* is not set, the {COMMAND\_SCRIPT} SHALL be:  'AE 80' +  '22 {L}' + *COMMAND1* +  '22 {L}' + *COMMAND2* + … + '00 00'  If *CHAINING\_OPT* is set, the {COMMAND\_SCRIPT} SHALL be:  'AE 80' +  '83 01' + *CHAINING\_OPT* + '22 {L}' + *COMMAND1* +  '22 {L}' + *COMMAND2* + … + '00 00' |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | * If the commands list is composed of TLVs (e.g. SCP03t commands):   {COMMAND\_SCRIPT} SHALL contain the list of TLVs formatted using the expanded format with indefinite length as defined in ETSI TS 102 226 [[6].](#_bookmark11)  If *CHAINING\_OPT* is not set, the {COMMAND\_SCRIPT} SHALL be:  'AE 80' + *COMMAND1* + *COMMAND2* + … + '00 00'  If *CHAINING\_OPT* is set, the {COMMAND\_SCRIPT} SHALL be:  'AE 80' +  '83 01' + *CHAINING\_OPT* +  *COMMAND1* +  *COMMAND2* + … +  '00 00' |
| *HTTPS\_EMPTY\_CONTE NT* | Generate an HTTPS POST message sent by the SM-SR containing no command but instructing to not close the HTTP session.  #HTTP/1.1 204 #X\_ADMIN\_PROTOCOL #X\_ADMIN\_NEXT\_URI |
| *SCP03\_SCRIPT* | Generate an SCP03 script with the APDUs in parameters.  Parameters:   * *KVN* * *APDU1; APDU2;…;APDUn*   Here is the SCP03 script to generate:  '80 50' + *KVN* + '00 08 {HOST\_CHALLENGE} 00'  '84 82 33 00 10 {HOST\_CRYPTOGRAM} {MAC}' '{APDU1\_SECURED}'  '{APDU2\_SECURED}'  '…' '{APDUn\_SECURED}'  See [Annex C](#_bookmark280) for the definition of {HOST\_CHALLENGE}, {HOST\_CRYPTOGRAM} and  {MAC}.  The {APDUx\_SECURED} is the command *APDUx* secured according GlobalPlatform Card Specification - Amendment D [[11].](#_bookmark16)  If it is not defined differently in the test step, these following SCP03 keys SHALL be used: |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | * #DEFAULT\_ISD\_P\_SCP03\_KENC * #DEFAULT\_ISD\_P\_SCP03\_KMAC * #DEFAULT\_ISD\_P\_SCP03\_KDEK   In order to retrieve the SCP03 sequence counter (i.e. {SCP03\_SEQ\_NUM}), it is assumed that a INITIALIZE UPDATE APDU command MAY be used every time it is necessary. |
| *SCP03\_SUB\_SCRIPT* | Generate the next part of an SCP03 script.  Parameters:   * *APDU1; APDU2;…APDUn*   Here is the SCP03 script to generate:  '{APDU1\_SECURED}' '{APDU2\_SECURED}'  '…' '{APDUn\_SECURED}'  The {APDUx\_SECURED} is the command *APDUx* secured according GlobalPlatform Card Specification - Amendment D [[11].](#_bookmark16)  The SCP03 session keys of the previous generated script SHALL be used. |
| *SCP03T\_SCRIPT* | Generate an SCP03t script with the PEs in parameters encoded in TLV structures using DER.  Parameters:   * *KVN* * *PE\_TLVs*   The *PE\_TLVs* SHALL be split in several parts: each of these sub-parts (named PE\_TLV1, PE\_TLV2 … PE\_TLVn here after) SHALL have a size which does not exceed 1007 bytes (considering that the maximum length of a SCP03t TLV command SHALL be 1020 bytes).  Here is the SCP03t script to generate:  '84 0A' + *KVN* + '00 {HOST\_CHALLENGE}' '85 11 33 {HOST\_CRYPTOGRAM} {MAC}'  '86 {L} {PE\_TLV1\_SECURED}'  '86 {L} {PE\_TLV2\_SECURED}'  '…'  '86 {L} {PE\_TLVn\_SECURED}'  See [Annex C](#_bookmark280) for the definition of {HOST\_CHALLENGE}, {HOST\_CRYPTOGRAM} and  {MAC}. |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | The {PE\_TLVx\_SECURED} is the PE\_TLVx secured according GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7) (section 4.1.3.3).  If it is not defined differently in the test step, these following SCP03 keys SHALL be used:   * #DEFAULT\_ISD\_P\_SCP03\_KENC * #DEFAULT\_ISD\_P\_SCP03\_KMAC * #DEFAULT\_ISD\_P\_SCP03\_KDEK   In order to retrieve the SCP03 sequence counter (i.e. {SCP03\_SEQ\_NUM}), it is assumed that a INITIALIZE UPDATE TLV command MAY be used every time it is necessary. |
| *SCP03T\_SUB\_SCRIPT* | Generate the next part of an SCP03t script.  Parameters:   * *PE\_TLVs*   The *PE\_TLVs* SHALL be split in several parts: each of these sub-parts (named PE\_TLV1, PE\_TLV2 … PE\_TLVn here after) SHALL have a size which does not exceed 1007 bytes (considering that the maximum length of a SCP03t TLV command SHALL be 1020 bytes).  Here is the SCP03t script to generate:  '86 {L} {PE\_TLV1\_SECURED}'  '86 {L} {PE\_TLV2\_SECURED}'  '…'  '86 {L} {PE\_TLVn\_SECURED}'  The {PE\_TLVx\_SECURED} is the PE\_TLVx secured according GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7) (section 4.1.3.3).  The SCP03 session keys of the previous generated script SHALL be used. |
| *SCP80\_PACKET* | Generate an SCP80 secured packet with the commands (i.e. ADPUs or TLVs) in parameters.  Parameters:   * *SPI* * *TAR* * *COMMAND1; COMMAND2…*(i.e. APDUs or TLVs) * *CHAINING\_OPT* (optional parameter)   Here is the content of the command packet to generate:  '{CPI} {L} 15' + *SPI* + '{KIC} {KID}' + *TAR* + '{CNTR} {PCNTR} {CC}  {COMMAND\_SCRIPT}' |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | See [Annex C](#_bookmark280) for the definition of {CPI}, {KIC}, {KID}, {CNTR}, {PCNTR} and  {CC}.  For KIC and KID, if the KVN to use is ‘06’ (for example), the value SHALL be ‘62’ (AES in CBC mode). The KVN used SHALL be either #SCP80\_KVN or #MNO\_SCP80\_KVN (depending of the targeted SD).  Note that if the TAR is equal to #MNO\_TAR, the algorithm used MAY be also Triple DES in outer-CBC depending of the Profile (i.e. KIC and KID SHALL be adapted in consequence).  {CNTR} SHALL be incremented each time this function is called.   * If the commands list is composed of one TLV which is either [OPEN\_SCP81\_SESSION] or [OPEN\_SCP81\_MNO\_SESSION] (i.e. SCP81   administration session triggering parameters):  {COMMAND\_SCRIPT} SHALL contain the TLV command.   * If the commands list is composed of APDUs:   {COMMAND\_SCRIPT} SHALL contain the list of APDUs formatted using the expanded format with definite length as defined in ETSI TS 102 226 [[6].](#_bookmark11)  If *CHAINING\_OPT* is not set, the {COMMAND\_SCRIPT} SHALL be:  'AA {L}' +  '22 {L}' + *COMMAND1* + '22 {L}' + *COMMAND2* …  If *CHAINING\_OPT* is set, the {COMMAND\_SCRIPT} SHALL be:  'AA {L}' +  '83 01' + *CHAINING\_OPT* + '22 {L}' + *COMMAND1* + '22 {L}' + *COMMAND2* …   * If the commands list is composed of TLVs (e.g. SCP03t commands):   {COMMAND\_SCRIPT} SHALL contain the list of TLVs formatted using the expanded format with definite length as defined in ETSI TS 102 226 [[6].](#_bookmark11)  If *CHAINING\_OPT* is not set, the {COMMAND\_SCRIPT} SHALL be:  'AA {L}' + *COMMAND1* + *COMMAND2* … |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | If *CHAINING\_OPT* is set, the {COMMAND\_SCRIPT} SHALL be:  'AA {L}' +  '83 01' + *CHAINING\_OPT* +  *COMMAND1* +  *COMMAND2* …  In any cases, this packet SHALL be secured according the SPI value.  If it is not defined differently in the test step, these following SCP80 keys SHALL be used:   * #SCP80\_ENC\_KEY * #SCP80\_AUTH\_KEY * #SCP80\_DATA\_ENC\_KEY |
| *SEND\_ERROR\_RESP* | Send a secured error response message for a given request using network to an off- card entity.  Parameters:   * *FUNCTION\_NAME* * *STATUS* * *SUBJECT\_CODE* * *REASON\_CODE* * *OUT\_DATA1, OUT\_DATA2…* (optional parameter) Here is the content of the response to answer:   <?xml version="1.0" encoding="UTF-8"?>  <RPSMessage [xmlns="http://namespaces.gsma.org/esim](http://namespaces.gsma.org/esim-messaging/1)-[messaging/1"](http://namespaces.gsma.org/esim-messaging/1) [xmlns:xsi="http://www.w3.org/2001/XMLSchema](http://www.w3.org/2001/XMLSchema-instance)-instance" MessageVersion="1.0.0">  <RPSHeader>  <SenderEntity>  <EntityId>{FUNCTION\_REQ\_ID}</EntityId>  </SenderEntity>  <SenderName>{TOOL\_NAME}</SenderName>  <ReceiverEntity>  <EntityId>{FUNCTION\_REC\_ID}</EntityId>  </ReceiverEntity>  <MessageId>{REQ\_MESSAGE\_ID}</MessageId>  <RelatesTo>{REL\_MESSAGE\_ID}</RelatesTo>  <MessageType>*FUNCTION\_NAME*</MessageType>  <MessageDate>{CURRENT\_DATE}</MessageDate> |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | </RPSHeader>  <RPSBody>  <*FUNCTION\_NAME*>  <ProcessingStart>{CURRENT\_DATE}</ProcessingStart>  <ProcessingEnd>{CURRENT\_DATE}</ProcessingEnd>  <FunctionExecutionStatus>  <Status>*STATUS*</Status>  <StatusCodeData>  <Subject>*SUBJECT\_CODE*</Subject>  <Reason>*REASON\_CODE*</Reason>  </StatusCodeData>  </FunctionExecutionStatus>  *OUT\_DATA1 OUT\_DATA2*  …  </*FUNCTION\_NAME*>  </RPSBody>  </RPSMessage>  See [Annex C](#_bookmark280) for the definition of {CURRENT\_DATE}, {FUNCTION\_REQ\_ID} and  {FUNCTION\_REC\_ID}.  The mapping of this function into message SHALL be compliant with the [Annex A](#_bookmark253) of the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2].](#_bookmark7)  To transport the message, the technology of the entity under test SHALL be used (mail, file, Web Services…).  Depending of the receiver of this message, the endpoint SHALL be either the  #SM\_DP\_ACCESSPOINT or the #SM\_SR\_ACCESSPOINT. |
| *SEND\_NOTIF* | Send a secured notification message using network to an off-card entity.  Parameters:   * *NOTIF\_NAME* * *IN\_DATA1; IN\_DATA2…*   Here is the message to send:  <?xml version="1.0" encoding="UTF-8"?>  <RPSMessage [xmlns="http://namespaces.gsma.org/esim](http://namespaces.gsma.org/esim-messaging/1)-[messaging/1"](http://namespaces.gsma.org/esim-messaging/1) [xmlns:xsi="http://www.w3.org/2001/XMLSchema](http://www.w3.org/2001/XMLSchema-instance)-instance" MessageVersion="1.0.0"> |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | <RPSHeader>  <SenderEntity>  <EntityId>{FUNCTION\_REQ\_ID}</EntityId>  <EntityName>{TOOL\_NAME}</EntityName>  </SenderEntity>  <SenderName>{TOOL\_NAME}</SenderName>  <ReceiverEntity>  <EntityId>{FUNCTION\_REC\_ID}</EntityId>  </ReceiverEntity>  <MessageId>{MESSAGE\_ID}</MessageId>  <MessageType>*NOTIF\_NAME*</MessageType>  <MessageDate>{CURRENT\_DATE}</MessageDate>  </RPSHeader>  <RPSBody>  <*NOTIF\_NAME*>  <FunctionCallIdentifier>  {FUNC\_CALL\_ID}  </FunctionCallIdentifier>  *IN\_DATA1 IN\_DATA2*  …  </*NOTIF\_NAME*>  </RPSBody>  </RPSMessage>  See [Annex C](#_bookmark280) for the definition of {CURRENT\_DATE}, {FUNCTION\_REQ\_ID} and  {FUNCTION\_REC\_ID}.  To transport the message, the technology of the entity under test SHALL be used (mail, file, Web Services…).  Depending of the receiver of this message, the endpoint SHALL be either the  #SM\_DP\_ACCESSPOINT or the #SM\_SR\_ACCESSPOINT. |
| *SEND\_REQ* | Send a secured request message using network to an off-card entity.  Parameters:   * *FUNCTION\_NAME* * *IN\_DATA1; IN\_DATA2…* |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | Here is the content of the request to send:  <?xml version="1.0" encoding="UTF-8"?>  <RPSMessage [xmlns="http://namespaces.gsma.org/esim](http://namespaces.gsma.org/esim-messaging/1)-[messaging/1"](http://namespaces.gsma.org/esim-messaging/1) [xmlns:xsi="http://www.w3.org/2001/XMLSchema](http://www.w3.org/2001/XMLSchema-instance)-instance" MessageVersion="1.0.0">  <RPSHeader>  <SenderEntity>  <EntityId>{FUNCTION\_REQ\_ID}</EntityId>  <EntityName>{TOOL\_NAME}</EntityName>  </SenderEntity>  <SenderName>{TOOL\_NAME}</SenderName>  <ReceiverEntity>  <EntityId>{FUNCTION\_REC\_ID}</EntityId>  </ReceiverEntity>  <MessageId>{MESSAGE\_ID}</MessageId>  <MessageType>*FUNCTION\_NAME*</MessageType>  <MessageDate>{CURRENT\_DATE}</MessageDate>  </RPSHeader>  <RPSBody>  <*FUNCTION\_NAME*>  <FunctionCallIdentifier>  {FUNC\_CALL\_ID}  </FunctionCallIdentifier>  *IN\_DATA1 IN\_DATA2*  …  </*FUNCTION\_NAME*>  </RPSBody>  </RPSMessage>  See [Annex C](#_bookmark280) for the definition of {CURRENT\_DATE}, {FUNC\_CALL\_ID},  {FUNCTION\_REQ\_ID} and {FUNCTION\_REC\_ID}.  The mapping of this function into message SHALL be compliant with the Annex A of the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2].](#_bookmark7)  To transport the message, the technology of the entity under test SHALL be used (mail, file, Web Services…). |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | Depending of the receiver of this message, the endpoint SHALL be either the  #SM\_DP\_ACCESSPOINT or the #SM\_SR\_ACCESSPOINT.  If needed, the attribute ResponseEndpoint MAY be used. |
| *SEND\_SUCCESS\_RESP* | Send a secured success response message for a given request using network to an off-card entity.  Parameters:   * *FUNCTION\_NAME* * *OUT\_DATA1; OUT\_DATA2…* (optional parameter) Here is the content of the response to answer:   <?xml version="1.0" encoding="UTF-8"?>  <RPSMessage [xmlns="http://namespaces.gsma.org/esim](http://namespaces.gsma.org/esim-messaging/1)-[messaging/1"](http://namespaces.gsma.org/esim-messaging/1) [xmlns:xsi="http://www.w3.org/2001/XMLSchema](http://www.w3.org/2001/XMLSchema-instance)-instance" MessageVersion="1.0.0">  <RPSHeader>  <SenderEntity>  <EntityId>{FUNCTION\_REQ\_ID}</EntityId>  </SenderEntity>  <SenderName>{TOOL\_NAME}</SenderName>  <ReceiverEntity>  <EntityId>{FUNCTION\_REC\_ID}</EntityId>  </ReceiverEntity>  <MessageId>{REQ\_MESSAGE\_ID}</MessageId>  <RelatesTo>{REL\_MESSAGE\_ID}</RelatesTo>  <MessageType>*FUNCTION\_NAME*</MessageType>  <MessageDate>{CURRENT\_DATE}</MessageDate>  </RPSHeader>  <RPSBody>  <*FUNCTION\_NAME*>  <ProcessingStart>{CURRENT\_DATE}</ProcessingStart>  <ProcessingEnd>{CURRENT\_DATE}</ProcessingEnd>  <FunctionExecutionStatus>  <Status>#SUCCESS</Status>  </FunctionExecutionStatus>  *OUT\_DATA1*  *OUT\_DATA2* |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | …  </*FUNCTION\_NAME*>  </RPSBody>  </RPSMessage>  See [Annex C](#_bookmark280) for the definition of {CURRENT\_DATE}, {FUNCTION\_REQ\_ID} and  {FUNCTION\_REC\_ID}.  The mapping of this function into message SHALL be compliant with the Annex A of the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2].](#_bookmark7)  To transport the message, the technology of the entity under test SHALL be used (mail, file, Web Services…).  Depending of the receiver of this message, the endpoint SHALL be either the  #SM\_DP\_ACCESSPOINT or the #SM\_SR\_ACCESSPOINT. |
| *STORE\_ISDP\_KEYS* | Generate the APDU command allowing the creation or the update of the ISD-P keys (scenario#3 based on ECKA EG (ElGamal) scheme as defined in GlobalPlatform Card Specification Amendment E [[12]](#_bookmark17)).  Parameters:   * *SC3\_PARAM* * *RANDOM\_CHALLENGE*   Here is the content of the APDU to generate:   * CLA = 80 * INS = E2   - P1 = 89  - P2 = 01   * LC = {LC} * Data =   '3A 02 {L}  A6 {L}  90 02 03' + *SC3\_PARAM* +  '95 01 10  80 01 88  81 01 10  82 01 01   * 1. 01 #SCP03\_KVN   91 00   * 1. {L} #HOST\_ID (present only if *SC3\_PARAM*=#SC3\_DR\_HOST)   7F 49 {L} #SM\_EPK\_ECKA'  5F 37 {L} {SIGNATURE}  - LE = 00 |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | The following TLV-encoded data SHALL be signed with #SM\_SK\_ECDSA to generate the {SIGNATURE}:  '3A 02 {L}  A6 {L}  90 02 03' + *SC3\_PARAM +*  '95 01 10  80 01 88  81 01 10  82 01 01   1. 01 #SCP03\_KVN   91 00   1. {L} #HOST\_ID (present only if *SC3\_PARAM*=#SC3\_DR\_HOST)   7F 49 {L} #SM\_EPK\_ECKA  00 85 {L}' + *RANDOM\_CHALLENGE* |
| *STORE\_ISDR\_KEYS* | Generate the APDU command allowing the creation or the update of the ISD-R keys (scenario#3 based on ECKA EG (ElGamal) scheme as defined in GlobalPlatform Card Specification Amendment E [[12]](#_bookmark17)).  Parameters:   * *SC3\_PARAM* * *RANDOM\_CHALLENGE*   Here is the content of the APDU to generate:   * CLA = 80 * INS = E2   - P1 = 89  - P2 = 01   * LC = {LC} * Data =   '3A 02 {L}  A6 {L}  90 02 03' + *SC3\_PARAM* +  '95 01 10 *-- Key Usage*  80 01 88 *-- Key Type*  81 01 10 *-- Key Length*  82 01 01 *-- Key Identifier*  83 01 #SCP80\_KVN *-- Key Version Number*  91 00 *-- Initial Sequence Counter*  84 {L} #HOST\_ID (present only if *SC3\_PARAM*=#SC3\_DR\_HOST)  7F 49 {L} #SM\_EPK\_ECKA'  5F 37 {L} {SIGNATURE}  - LE = 00 |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | The following TLV-encoded data SHALL be signed with #SM\_SK\_ECDSA to generate the {SIGNATURE}:  '3A 02 {L}  A6 {L}  90 02 03' + *SC3\_PARAM* +  '95 01 10  80 01 88  81 01 10  82 01 01   1. 01 #SCP80\_KVN   91 00   1. {L} #HOST\_ID (present only if *SC3\_PARAM*=#SC3\_DR\_HOST)   7F 49 {L} #SM\_EPK\_ECKA  00 85 {L}' + *RANDOM\_CHALLENGE* |
| *STORE\_MNO\_KEYS\_2B* | Generate the APDU command that allows updating the MNO keys using the scenario#2.B as defined in GlobalPlatform Card Specification v.2.2.1 - UICC Configuration [[13].](#_bookmark18)  Parameters:   * *CASD\_PUBLIC\_KEY*   Here is the content of the APDU to generate:   * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {LC} * Data =   00 A6 18  A6 16  90 01 04  95 01 10  80 01 80 (MNO-SD SHALL be configured with 3DES keys)  81 01 10  83 01 #MNO\_SCP80\_KVN  91 05 00 00 00 00 01  80 10 {L} {KEYS\_ENCRYPTED}  The {KEYS\_ENCRYPTED} SHALL be encrypted with the *CASD\_PUBLIC KEY*. |
| *STORE\_MNO\_KEYS\_3* | Generate the APDU command that allows updating the MNO keys using the scenario#3 based on ECKA EG (ElGamal) scheme as defined in GlobalPlatform Card Specification Amendment E [[12].](#_bookmark17)  Parameters: |

|  |  |
| --- | --- |
| **Method name** | **Explanation** |
|  | * *None*   Here is the content of the APDU to generate:   * CLA = 80 * INS = E2   - P1 = 89  - P2 = 00   * LC = {LC} * Data =   00 A6 1C  A6 1A  90 02 03 01  95 01 10   * 1. 01 80 (or '88' if the MNO-SD is configured with AES keys)   2. 01 10   3. 01 01   4. 01 #MNO\_SCP80\_KVN   91 05 00 00 00 00 01  7F 49 {L} #SM\_EPK\_ECKA  - LE = 00 |

###### Table 18: Methods

# Annex E Commands and Responses

Here are all the commands and responses used in this document.

## Commands

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| BAD\_MASTER\_DEL\_ISDP1 | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40  - LC = 33   * Data =   4F 10 #ISD\_P\_AID1 B6 1A  42 04 #ISD\_P\_SIN  45 08 #ISD\_P\_SDIN  5F 20 04 #ISD\_P\_PROV\_ID  93 01 #TOKEN\_ID  9E 03 #BAD\_TOKEN  - LE = 00 |
| BAD\_STORE\_POL1 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 06  - Data = 3A 06 03 81 01 07 |
| DELETE\_ISDP1 | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40  - LC = 12   * Data = 4F 10 #ISD\_P\_AID1   - LE = 00 |
| DELETE\_ISDP\_UNKNOWN | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40  - LC = 12   * Data = 4F 10 #ISD\_P\_AID\_UNKNOWN   - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| DELETE\_SCP80\_KEYSETS | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 00  - LC = 05   * Data =   F2 03 #SCP03\_KVN 01 03  - LE = 00 |
| DELETE1\_KEYSETS | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 00  - LC = 05  - Data = F2 03 #SCP80\_KVN 01 03  - LE = 00 |
| DELETE2\_KEYSETS | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 00   * LC = 0A * Data =   F2 03 #SCP80\_KVN 01 03  F2 03 #SCP81\_KVN 01 05  - LE = 00 |
| DISABLE\_ISDP1 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 15   * Data = 3A 04 12 4F 10 #ISD\_P\_AID1 |
| ENABLE\_ISDP1 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 15   * Data = 3A 03 12 4F 10 #ISD\_P\_AID1 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| GET\_DATA\_5A | * CLA = 80 * INS = CA   - P1 = 00  - P2 = 5A  - LE = 00 |
| GET\_DATA\_BF30\_CERT | * CLA = 80 * INS = CA * P1 = BF   - P2 = 30  - LC = 04  - Data = 5C 02 7F 21  - LE = 00 |
| GET\_DATA\_BF30\_REC | * CLA = 80 * INS = CA * P1 = BF   - P2 = 30  - LC = 03  - Data = 5C 01 66  - LE = 00 |
| GET\_DATA\_C1 | * CLA = 80 * INS = CA   - P1 = 00  - P2 = C1  - LE = 00 |
| GET\_DATA\_CASD\_CERT | * CLA = 80 * INS = CA   - P1 = 7F  - P2 = 21  - LE = 00 |
| GET\_DATA\_E0 | * CLA = 80 * INS = CA   - P1 = 00  - P2 = E0  - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| GET\_DATA\_FF21 | * CLA = 80 * INS = CA * P1 = FF   - P2 = 21  - LE = 00 |
| GET\_DEFAULT\_ISDP | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 17   * Data = 4F 10 #DEFAULT\_ISD\_P\_AID 5C 03 4F 9F 70   - LE = 00 |
| GET\_FALLBACK | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 09   * Data =   4F 00  #ISD\_P\_ATTRIBUTE 01 01  5C 02 4F #ISD\_P\_ATTRIBUTE  - LE = 00 |
| GET\_ISDP1 | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 17   * Data = 4F 10 #ISD\_P\_AID1 5C 03 4F 9F 70   - LE = 00 |
| GET\_ISDP1\_MEM | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 19  - Data = 4F 10 #ISD\_P\_AID1 5C 05 4F 9F 70 8F 91  - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| GET\_ISDP\_DISABLED | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02   * LC = 0B   - Data = 4F 00 9F 70 01 1F 5C 03 4F 9F 70  - LE = 00 |
| GET\_ISDP\_ENABLED | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02   * LC = 0B   - Data = 4F 00 9F 70 01 3F 5C 03 4F 9F 70  - LE = 00 |
| GET\_ISDP\_LIST | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 07  - Data = 4F 00 5C 03 4F 9F 70  - LE = 00 |
| GET\_MNO\_ISD | * CLA = 80 * INS = F2   - P1 = 80  - P2 = 02  - LC = 07  - Data = 4F 00 5C 03 4F 9F 70  - LE = 00 |
| GET\_MNO\_SD | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02   * LC = {L} * Data = 4F {L} #MNO\_SD\_AID 5C 01 4F   - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| GET\_STATUS\_ISDR | * CLA = 80 * INS = F2   - P1 = 40  - P2 = 02  - LC = 12   * Data = 4F 10 #ISD\_R\_AID   - LE = 00 |
| INSTALL\_AID\_ECASD | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00   * LC = 2C * Data =   08 A0 00 00 05 59 10 10 03  0B A0 00 00 05 59 10 10 03 44 55 66  10 #ECASD\_AID  01 00  02 C9 00  00  -LE = 00 |
| INSTALL\_TAR\_ISDR | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00  - LC = 37   * Data =   08 A0 00 00 05 59 10 10 01  0B A0 00 00 05 59 10 10 01 11 22 33  0C A0 00 00 05 59 10 10 01 11 22 33 01  01 00  11 EA 0D 80 0B 01 00 00 00 00 00 03 #ISD\_R\_TAR  00  C9 00  00  -LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| INSTALL\_APPLET2 | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00  - LC = 37   * Data =   08 A0 00 00 05 59 10 10 02  0B A0 00 00 05 59 10 10 02 11 22 33  0C A0 00 00 05 59 10 10 02 11 22 33 01  01 00  11 EA 0D 80 0B 01 00 00 00 00 00 03 11 22 33 00  C9 00  00  -LE = 00 |
| INSTALL\_APPLET3 | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00  - LC = 28   * Data =   08 A0 00 00 05 59 10 10 03  0B A0 00 00 05 59 10 10 03 44 55 66  0C A0 00 00 05 59 10 10 03 44 55 66 01  01 00  02 C9 00  00  -LE = 00 |
| INSTALL\_ISDP | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00   * LC = 3F * Data =   10 #ISD\_P\_PKG\_AID  10 #ISD\_P\_MOD\_AID  10 #ISD\_P\_AID1  03 80 C0 00  06 C9 04 81 02 03 70  00  -LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| INSTALL\_ISDP\_MEM | * CLA = 80 * INS = E6   - P1 = 0C  - P2 = 00  - LC = 47   * Data =   10 #ISD\_P\_PKG\_AID  10 #ISD\_P\_MOD\_AID  10 #ISD\_P\_AID1  03 80 C0 00  0E EF 06 83 04 #MEMORY\_QUOTA C9 04 81 02 03 70  00  - LE = 00 |
| INSTALL\_PERSO\_RES\_ISDP | * CLA = 80 * INS = E6   - P1 = 20  - P2 = 00  - LC = 16   * Data = 00 00 10 #RESERVED\_ISD\_P\_AID 00 00 00   - LE = 00 |
| INSTALL\_PERSO\_ISDP1 | * CLA = 80 * INS = E6   - P1 = 20  - P2 = 00  - LC = 16  - Data = 00 00 10 #ISD\_P\_AID1 00 00 00  - LE = 00 |
| LOCK\_DEFAULT\_ISDP | * CLA = 80 * INS = F0   - P1 = 40  - P2 = 80  - LC = 10   * Data = #DEFAULT\_ISD\_P\_AID |
| LOCK\_ISDR | * CLA = 80 * INS = F0   - P1 = 80  - P2 = 7F  - LC = 10   * Data = #ISD\_R\_AID |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| MASTER\_DEL\_ISDP1 | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40  - LC = 40   * Data =   4F 10 #ISD\_P\_AID1 B6 1A  42 04 #ISD\_P\_SIN  45 08 #ISD\_P\_SDIN  5F 20 04 #ISD\_P\_PROV\_ID  93 01 #TOKEN\_ID  9E 10 {TOKEN\_VALUE}  - LE = 00 |
| MASTER\_DEL\_ISDP1\_INV\_SDIN | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40   * LC = {L} * Data =   4F 10 #ISD\_P\_AID1 B6 {L}  42 {L} #ISD\_P\_SIN  45 {L} #ISD\_P\_RID  5F 20 {L} #ISD\_P\_PROV\_ID  93 01 #TOKEN\_ID  9E 10 {TOKEN\_VALUE}  - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| MASTER\_DEL\_ISDP1\_INV\_SIN | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40   * LC = {L} * Data =   4F 10 #ISD\_P\_AID1 B6 {L}  42 {L} #ISD\_P\_RID  45 {L} #ISD\_P\_SDIN  5F 20 {L} #ISD\_P\_PROV\_ID  93 01 #TOKEN\_ID  9E 10 {TOKEN\_VALUE}  - LE = 00 |
| MASTER\_DEL\_ISDP1\_RID | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40   * LC = {L} * Data =   4F 10 #ISD\_P\_AID1 B6 {L}  42 04 #ISD\_P\_SIN  45 08 #ISD\_P\_SDIN  5F 20 05 #ISD\_P\_RID  93 01 #TOKEN\_ID  9E 10 {TOKEN\_VALUE}  - LE = 00 |
| MASTER\_DEL\_ISDP1\_NO\_PROV\_ID | * CLA = 80 * INS = E4   - P1 = 00  - P2 = 40   * LC = {L} * Data =   4F 10 #ISD\_P\_AID1 B6 {L}  42 04 #ISD\_P\_SIN  45 08 #ISD\_P\_SDIN  93 01 #TOKEN\_ID  9E 10 {TOKEN\_VALUE}  - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| NOTIF\_CONFIRMATION | * CLA = 80 * INS = E2   - P1 = 89  - P2 = 00  - LC = 07   * Data = 3A 08 04 4E 02 {NOTIF\_NUMBER}   - LE = 00 |
| NOTIF\_PROFILE\_CHANGE | E1 {L}  4C 10 #EID  4D 01 02  4E 02 {NOTIF\_NUMBER}  2F 10 #ISD\_P\_AID1  see Note 1 |
| NOTIF\_PROFILE\_CHANGE2 | E1 {L}  4C 10 #EID  4D 01 02  4E 02 {NOTIF\_NUMBER}  2F 10 #DEFAULT\_ISD\_P\_AID  see Note 1 |
| NOTIF\_ROLL\_BACK | E1 {L}  4C 10 #EID  4D 01 03  4E 02 {NOTIF\_NUMBER}  2F 10 #DEFAULT\_ISD\_P\_AID  see Note 1 |
| OPEN\_CHANNEL\_FOR\_BIP | * CLA = 80 * INS = EC   - P1 = 01  - P2 = 01  - LC = 25   * Data =   35 07 #BEARER\_DESCRIPTION  3C 03 01 #UDP\_PORT  39 02 #BUFFER\_SIZE  47 0A #NAN\_VALUE  3E 05 21 #IP\_VALUE |
| OPEN\_CHANNEL\_FOR\_CATTP | * CLA = 80 * INS = EC   - P1 = 01  - P2 = 02  - LC = 05   * Data = 3C 03 00 #CAT\_TP\_PORT |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| OPEN\_SCP81\_MNO\_SESSION | 81 {L}  83 {L}  84 25  35 07 #BEARER\_DESCRIPTION  39 02 #BUFFER\_SIZE  47 0A #NAN\_VALUE  3C 03 02 #TCP\_PORT  3E 05 21 #IP\_VALUE  89 {L}  8A 09 #ADMIN\_HOST  8B {L} #MNO\_AGENT\_ID  8C 10 #ADMIN\_URI  85 {L}  {L} #MNO\_PSK\_ID  02 #MNO\_SCP81\_KVN #MNO\_SCP81\_KEY\_ID |
| OPEN\_SCP81\_SESSION | 81 {L}  83 {L}  84 25  35 07 #BEARER\_DESCRIPTION  39 02 #BUFFER\_SIZE  47 0A #NAN\_VALUE  3C 03 02 #TCP\_PORT  3E 05 21 #IP\_VALUE  89 {L}  8A 09 #ADMIN\_HOST  8B {L} #AGENT\_ID  8C 10 #ADMIN\_URI |
| OPEN\_SCP81\_WITH\_RETRY | 81 {L}  83 {L}  84 25  35 07 #BEARER\_DESCRIPTION  39 02 #BUFFER\_SIZE  47 0A #NAN\_VALUE  3C 03 02 #TCP\_PORT  3E 05 21 #IP\_VALUE  86 {L}  00 02 A5 03 00 00 10  89 {L}  8A 09 #ADMIN\_HOST  8B {L} #AGENT\_ID  8C 10 #ADMIN\_URI |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| PUTKEY\_SCP81 | * CLA = 80 * INS = D8   - P1 = 00  - P2 = 81   * LC = {L}   #SCP81\_KVN 85  11  10 #NEW\_SCP81\_PSK (see Note 2)  03  {NEW\_SCP81\_PSK KCV} (see Note 3) 88  11  10 #PSK\_DEK (see Note 4)  03  {PSK\_DEK KCV} (see Note 3)  - LE = 00 |
| SELECT\_APPLET3 | * CLA = 00 * INS = A4   - P1 = 04  - P2 = 00   * LC = 0C   - Data = A0 00 00 05 59 10 10 03 44 55 66 01  - LE = 00 |
| SELECT\_CASD | * CLA = 00 * INS = A4   - P1 = 04  - P2 = 00   * LC = 0C * Data = #CASD\_AID   - LE = 00 |
| SELECT\_ECASD | * CLA = 00 * INS = A4   - P1 = 04  - P2 = 00  - LC = 10   * Data = #ECASD\_AID   - LE = 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| SELECT\_FILE\_1122 | * CLA = 00 * INS = A4   - P1 = 00  - P2 = 04  - LC = 02  - Data = 11 22  - LE = 00 |
| SET\_FALLBACK | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 15   * Data = 3A 05 12 4F 10 #ISD\_P\_AID1 |
| STORE\_CATTP\_PARAM | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 16   * Data =   3A 07 13 A4 11  3C 03 01 #UDP\_PORT  3C 03 00 #CAT\_TP\_PORT  3E 05 21 #IP\_VALUE |
| STORE\_CATTP\_PARAM\_MNO | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 2D * Data =   3A 07 2A A2 28  35 07 #BEARER\_DESCRIPTION  47 0A #NAN\_VALUE  0D 06 #LOGIN  0D 09 #PWD |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_CATTP\_PARAM\_MNO2 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A2 {L}  35 07 #BEARER\_DESCRIPTION  47 {L} #MNO2\_CON\_NAN  0D {L} #MNO2\_CON\_LOGIN  0D {L} #MNO2\_CON\_PWD |
| STORE\_DP\_CERTIF | * CLA = 80 * INS = E2   - P1 = 09  - P2 = 00   * LC = {LC} * Data = 3A 01 {L} #VALID\_SM\_DP\_CERTIFICATE   - LE = 00 |
| STORE\_HTTPS\_PARAM | * CLA = 80 * INS = E2   - P1 = 90  - P2 = 00   * LC = {L} * Data =   A5 {L}  84 {L}  3C 03 02 #TCP\_PORT  3E 05 21 #IP\_VALUE  39 02 #BUFFER\_SIZE  89 {L}  8A 09 #ADMIN\_HOST  8B {L} #AGENT\_ID  8C 10 #ADMIN\_URI |
| STORE\_HTTPS\_PARAM\_MNO | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 2D * Data =   3A 07 2A A1 28  35 07 #BEARER\_DESCRIPTION  47 0A #NAN\_VALUE  0D 06 #LOGIN  0D 09 #PWD |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_HTTPS\_PARAM\_MNO2 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A1 {L}  35 07 #BEARER\_DESCRIPTION  47 {L} #MNO2\_CON\_NAN  0D {L} #MNO2\_CON\_LOGIN  0D {L} #MNO2\_CON\_PWD |
| STORE\_INVALID\_DP\_CERTIF | * CLA = 80 * INS = E2   - P1 = 09  - P2 = 00   * LC = {LC} * Data = 3A 01 {L} #INVALID\_SM\_DP\_CERTIFICATE   - LE = 00 |
| STORE\_INVALID\_SR\_CERTIF | * CLA = 80 * INS = E2   - P1 = 09  - P2 = 00   * LC = {LC} * Data = 3A 01 {L} #INVALID\_SM\_SR\_CERTIFICATE   - LE = 00 |
| STORE\_POL1\_DEL\_AUTO | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 06  - Data = 3A 06 03 81 01 04 |
| STORE\_POL1\_DEL\_DIS | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 06  - Data = 3A 06 03 81 01 03 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_POL1\_DIS | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 06  - Data = 3A 06 03 81 01 01 |
| STORE\_POL1\_NO\_RULE | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 06  - Data = 3A 06 03 81 01 00 |
| STORE\_PROV\_ID | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 0A   - Data = 00 70 07 5F 20 04 #ISD\_P\_PROV\_ID |
| STORE\_SDIN | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 0D * Data = 00 70 0A 45 08 #ISD\_P\_SDIN |
| STORE\_SIN | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 09  - Data = 00 70 06 42 04 #ISD\_P\_SIN |
| STORE\_SMS\_PARAM | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 0C * Data =   3A 07 09 A3 07 81 05 #DEST\_ADDR |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_SMS\_PARAM\_ISDPS | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A3 {L}  81 05 #DEST\_ADDR A2 {L}  81 03 #DEFAULT\_ISD\_P\_ID  82 {L} #DEST\_ADDR2 A2 {L}  81 03 #ISD\_P\_ID1  82 {L} #DEST\_ADDR3 |
| STORE\_SMS\_PARAM\_ISDP | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A3 {L}  81 05 #DEST\_ADDR A2 {L}  81 03 #DEFAULT\_ISD\_P\_ID  82 {L} #DEST\_ADDR2 |
| STORE\_SMSCATTP\_PARAM | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 38   * Data =   3A 07 35  A2 28  35 07 #BEARER\_DESCRIPTION  47 0A #NAN\_VALUE  0D 06 #LOGIN  0D 09 #PWD  A0 09 06 07 #TON\_NPI #DIALING\_NUMBER |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_HTTPSSMS\_PARAM | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00  - LC = 38   * Data =   3A 07 35  A1 28  35 07 #BEARER\_DESCRIPTION  47 0A #NAN\_VALUE  0D 06 #LOGIN  0D 09 #PWD  A0 09 06 07 #TON\_NPI #DIALING\_NUMBER |
| STORE\_SMS\_PARAM\_MNO | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = 0E * Data =   3A 07 0B A0 09  06 07 #TON\_NPI #DIALING\_NUMBER |
| STORE\_SMS\_PARAM\_MNO1 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A0 {L}  06 07 #TON\_NPI #DIALING\_NUMBER   1. 01 #PID 2. 01 #DCS |
| STORE\_SMS\_PARAM\_MNO2 | * CLA = 80 * INS = E2   - P1 = 88  - P2 = 00   * LC = {L} * Data =   3A 07 {L} A0 {L}  06 {L} #MNO2\_CON\_TON\_NPI #MNO2\_CON\_DIAL\_NUM |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| STORE\_SR\_CERTIF | * CLA = 80 * INS = E2   - P1 = 09  - P2 = 00   * LC = {LC} * Data = 3A 01 {L} #VALID\_SM\_SR\_CERTIFICATE   - LE = 00 |
| TERMINAL\_PROFILE | * CLA = 80 * INS = 10   - P1 = 00  - P2 = 00   * LC = 1F * Data =   FF FF FF FF FF FF 1F FF FF 03 02 FF FF 9F FF EF DF  FF  0F FF 0F FF FF 0F FF 03 00 3F 7F FF 03 |
| *Note 1: The AID tag that allows identifying the ISD-P MAY be either ‘2F’ or ‘AF’. The different TLV data objects within the tag ‘E1’ MAY be returned with a different order. Moreover, the TLV notification MAY also contain proprietary tags. However, the entire TLV SHALL fit into one SMS-MO if the notification is sent over SMS, and SHALL NOT exceed the size of 240 bytes if sent by HTTPs or CAT\_TP.*  *Note 2: #NEW\_SCP81\_PSK SHALL be encrypted as defined in GlobalPlatform Amendment B [18]*  *Note 3: Key check value (KCV) of #NEW\_SCP81\_PSK and #PSK\_DEK SHALL be computed as defined in [2]*  *Note 4: #PSK\_DEK SHALL be encrypted with the session KEK key of the key set used to open the SCP session as defined in [3]* | |

###### Table 19: Commands

## Responses

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_009000 | AB 09  80 02 00 01  23 03 00 90 00  see Note 2 |
| R\_AB\_PUTKEY | AB {L}  80 02 00 01  23 {L} ... 90 00 *–- any response data MAY be returned*  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_026982 | AB 08  80 02 00 02  23 02 69 82  see Note 2 |
| R\_AB\_026A80 | AB 0D  80 02 00 02  23 03 00 90 00  23 02 6A 80  see Note 2 |
| R\_AB\_029000 | AB 0D  80 02 00 02  23 03 00 90 00  23 02 90 00  see Note 2 |
| R\_AB\_02RC | AB {L}  80 02 00 02  23 {L} 85 {L} {RC}  90 00  see Note 2 |
| R\_AB\_02RECEIPT | AB {L}  80 02 00 02  23 {L} 86 {L} {RECEIPT}  90 00  see Note 2 |
| R\_AB\_02RECEIPT\_DR | AB {L}  80 02 00 02  23 {L} 85 {L} {DR} 86 {L} {RECEIPT}  90 00  see Note 2 |
| R\_AB\_036982 | AB 0D  80 02 00 03  23 03 00 90 00  23 02 69 82  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_03RC | AB {L}  80 02 00 03  23 03 00 90 00  23 {L} 85 {L} {RC}  90 00  see Note 2 |
| R\_AB\_6985 | AB 08  80 02 00 01  23 02 69 85  see Note 2 |
| R\_AB\_69E1 | AB 08  80 02 00 01  23 02 69 E1  see Note 2 |
| R\_AB\_6A88 | AB 08  80 02 00 01  23 02 6A 88  see Note 2 |
| R\_AB\_9000 | AB 08  80 02 00 01  23 02 90 00  see Note 2 |
| R\_AB\_BF30\_ECASD | AB {L}  80 02 00 01  23 {L}  BF 30 {L}  7F 21 {L}  7F 21 {L} #ECASD\_CERTIFICATE  90 00  see Note 2 |
| R\_AB\_BF30\_REC | AB {L}  80 02 00 01  23 {L}  BF 30 {L}  66 {L} #CARD\_RECOGNITION\_DATA  90 00  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_E0\_SCP80 | AB 1C  80 02 00 01  23 16  E0 12  C0 04 01 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 02 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 03 #SCP80\_KVN 88 {KEY\_LENGTH}  90 00  see Note 1  see Note 2 |
| R\_AB\_E0\_SCP80\_SCP81 | AB 22  80 02 00 01  23 1C  E0 18  C0 04 01 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 02 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 03 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 #SCP81\_KEY\_ID #SCP81\_KVN 85 {KEY\_LENGTH}  90 00  see Note 1  see Note 2  see Note 5 |
| R\_AB\_E3\_ISDP\_3F | AB 20  80 02 00 01  23 1A  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 3F  90 00  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_E3\_ISDP\_LIST1 | AB 3C  80 02 00 02  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 3F  90 00  23 1A  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 1F  90 00  see Note 2 |
| R\_AB\_E3\_ISDP\_LIST2 | AB 3C  80 02 00 02  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 1F  90 00  23 1A  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 3F  90 00  see Note 2 |
| R\_AB\_E3\_ISDP\_LIST3 | AB 38  80 02 00 01  23 32  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 1F  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 3F  90 00  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_E3\_ISDP1\_07 | AB 20  80 02 00 01  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 07  90 00  see Note 2 |
| R\_AB\_E3\_ISDP1\_0F | AB 20  80 02 00 01  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 0F  90 00  see Note 2 |
| R\_AB\_E3\_ISDP1\_1F | AB 20  80 02 00 01  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 1F  90 00  see Note 2 |
| R\_AB\_E3\_ISDP1\_3F | AB 20  80 02 00 01  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 3F  90 00  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_E3\_ISDP1\_E1 | AB 1F  80 02 00 01  23 19  E3 15  4F 10 #ISD\_P\_AID1  #ISD\_P\_ATTRIBUTE 01 01  90 00  see Note 2 |
| R\_AB\_E3\_ISDP1\_MEM | AB 2C  80 02 00 01  23 26  E3 22  4F 10 #ISD\_P\_AID1  9F 70 01 07  8F 04 #MEMORY\_QUOTA  91 04 #MEMORY\_QUOTA  90 00  see Note 2  see Note 4 |
| R\_AB\_FF21 | AB {L}  80 02 00 01  23 {L}  FF 21 {L}   1. {L} {NB\_APP} 2. {L} {NON\_VOLATILE\_MEMORY} 3. {L} {VOLATILE\_MEMORY}   90 00  see Note 2 |
| R\_AB\_MNO\_SD | AB {L}  80 02 00 01  23 {L}  E3 {L}  4F {L} #MNO\_SD\_AID  9F 70 01 0F  90 00  see Note 2  see Note 3 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_NOTIF | AB 0A  80 02 00 01  23 04  80 00  90 00  see Note 2 |
| R\_AB\_NOTIF1 | AB 1C  80 02 00 01  23 16  80 12 4F 10 #DEFAULT\_ISD\_P\_AID  90 00  see Note 2 |
| R\_AB\_NOTIF2 | AB 1C  80 02 00 01  23 16  80 12 4F 10 #ISD\_P\_AID1  90 00  see Note 2 |
| R\_AB\_RECEIPT | AB {L}  80 02 00 01  23 {L} 86 {L} {RECEIPT}  90 00  see Note 2 |
| R\_AB\_SCP03T\_01 | AB 2C  80 02 00 03 [R\_SCP03T\_INITUP\_OK] [R\_SCP03T\_EXTAUTH\_OK] 9F 46 01 01  see Note 2 |
| R\_AB\_SCP03T\_02 | AB 2C  80 02 00 03 [R\_SCP03T\_INITUP\_OK] [R\_SCP03T\_EXTAUTH\_OK] 9F 46 01 02  see Note 2 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AB\_SCP03T\_EA\_01 | AB 2A  80 02 00 02 [R\_SCP03T\_INITUP\_OK] 9F 45 01 01  see Note 2 |
| R\_AB\_SCP03T\_EA\_02 | AB 2A  80 02 00 02 [R\_SCP03T\_INITUP\_OK] 9F 45 01 02  see Note 2 |
| R\_AB\_SCP03T\_IU\_01 | AB 08  80 02 00 01  9F 44 01 01  see Note 2 |
| R\_AB\_SCP03T\_IU\_03 | AB 08  80 02 00 01  9F 44 01 03  see Note 2 |
| R\_AF\_009000 | AF 80  23 03 00 90 00  00 00 |
| R\_AF\_029000 | AF 80  23 03 00 90 00  23 02 90 00  00 00 |
| R\_AF\_02RC | AF 80  23 03 00 90 00  23 {L} 85 {L} {RC} 90 00  00 00 |
| R\_AF\_6A88 | AF 80  23 02 6A 88  00 00 |
| R\_AF\_9000 | AF 80  23 02 90 00  00 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AF\_BF30\_CERT | AF 80  23 {L}  BF 30 {L}  7F 21 {L}  7F 21 {L} #ECASD\_CERTIFICATE  90 00  00 00 |
| R\_AF\_BF30\_REC | AF 80  23 {L}  BF 30 {L}  66 {L} #CARD\_RECOGNITION\_DATA  90 00  00 00 |
| R\_AF\_E0\_SCP80\_SCP81 | AF 80  23 1C  E0 18  C0 04 01 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 02 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 03 #SCP80\_KVN 88 {KEY\_LENGTH}  C0 04 #SCP81\_KEY\_ID #SCP81\_KVN 85 {KEY\_LENGTH}  90 00  00 00  see Note 1  see Note 5 |
| R\_AF\_E3\_ISDP\_3F | AF 80  23 1A  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 3F  90 00  00 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AF\_E3\_ISDP\_LIST3 | AF 80  23 32  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 1F  E3 16  4F 10 #DEFAULT\_ISD\_P\_AID  9F 70 01 3F  90 00  00 00 |
| R\_AF\_E3\_ISDP1\_07 | AF 80  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 07  90 00  00 00 |
| R\_AF\_E3\_ISDP1\_0F | AF 80  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 0F  90 00  00 00 |
| R\_AF\_E3\_ISDP1\_1F | AF 80  23 1A  E3 16  4F 10 #ISD\_P\_AID1  9F 70 01 1F  90 00  00 00 |
| R\_AF\_E3\_ISDP1\_E1 | AF 80  23 19  E3 15  4F 10 #ISD\_P\_AID1  #ISD\_P\_ATTRIBUTE 01 01  90 00  00 00 |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| R\_AF\_FF21 | AF 80  23 {L}  FF 21 {L}  81 {L} {NB\_APP}  82 {L} {NON\_VOLATILE\_MEMORY}  83 {L} {VOLATILE\_MEMORY}  90 00  00 00 |
| R\_AF\_NOTIF | AF 80  23 04  80 00  90 00  00 00 |
| R\_AF\_RC | AF 80  23 {L} 85 {L} {RC}  90 00  00 00 |
| R\_AF\_RECEIPT | AF 80  23 {L} 86 {L} {RECEIPT}  90 00  00 00 |
| R\_CASD\_SC2B | 7F 21 {L} #CASD\_CERTIFICATE\_SC2B 90 00 |
| R\_CASD\_SC3 | 7F 21 {L} #CASD\_CERTIFICATE\_SC3 90 00 |
| R\_PROF\_PKG\_OK | 30 07  A0 05  30 03  80 01 00 |
| R\_SCP03T\_EMPTY | 86 00 |
| R\_SCP03T\_EXTAUTH\_OK | 85 00 |
| R\_SCP03T\_INITUP\_OK | 84 20  {KEY\_DIV\_DATA} #SCP03\_KVN 03 70  {CARD\_CHALLENGE}  {CARD\_CRYPTOGRAM}  {SCP03\_SEQ\_NUM} |

|  |  |
| --- | --- |
| **Name** | **Content in hexadecimal string** |
| *Note 1: Key Information Data Structure – Extended as defined in GlobalPlatform Card Specification* [*[3]*](#_bookmark8) *MAY also be returned. The order of the tags ‘C0’ (i.e. key information data) SHALL NOT be checked.*  *Note 2: In this table, the expanded remote responses using definite length contain a number of executed commands (i.e. value of the BER-TLV tag ‘80’) coded on 2 bytes (i.e. short number) as an example. But, it MAY be also coded on ‘01’ byte as defined in ETSI TS 102 226* [*[6].*](#_bookmark11) *As a consequence, the expected response scripting template tag (i.e. ‘AB’) SHALL be adapted according the eUICC implementation.*  *Note 3: Depending on the support of the GlobalPlatform Amendment C specification* [*[14]*](#_bookmark19) *in the Profile linked to the MNO-SD, the lifecycle state MAY be encoded with two bytes instead of one (that is, the contactless activation state SHALL be encoded in the second byte). In addition, other tags (e.g. ‘C5’ – Privileges) MAY be returned in the R- APDU as the tag ‘5C’ (i.e. tag list) present in the related GET STATUS command MAY NOT be supported by the MNO-SD. The content of the tag ‘9F70’ – Lifecycle state is set with ‘0F’ (i.e. SECURED) as an example: it SHALL NOT be checked in the response.*  *Note 4: The values of the tags ‘8F’ (i.e. cumulative granted non-volatile Memory) and ‘91’ (cumulative remaining non-volatile memory) MAY be also encoded in 2 bytes. In addition, they MAY be lower or equal to #MEMORY\_QUOTA.*  *Note 5: Other keys with an identifier from 1 to 5 MAY be also present under the keyset identified by #SCP81\_KVN.* | |

###### Table 20: Responses

# Annex F Bearer Independent Protocol

Here is a sequence explaining the BIP communication between the Device and the eUICC.

|  |  |
| --- | --- |
| **Direction** | **Sequence / Description** |
|  | *TRIGGERING EVT* |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* OPEN CHANNEL |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:* OPEN CHANNEL |
| Device → eUICC | TERMINAL RESPONSE |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* SEND DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:*  SEND DATA containing the data to send to the off-card entity |
| Device → eUICC | TERMINAL RESPONSE |
| *Several SEND DATA commands MAY be used to send the complete data* | |
| Device → eUICC | ENVELOPE EVENT DOWNLOAD |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* RECEIVE DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:* RECEIVE DATA |
| Device → eUICC | TERMINAL RESPONSE containing the data sent by the off-card entity |
| *Several RECEIVE DATA commands MAY be used to retrieve the complete data* | |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* SEND DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:*  SEND DATA containing the data to send to the off-card entity |
| Device → eUICC | TERMINAL RESPONSE |
| *Several SEND DATA commands MAY be used to send the complete data* | |
| Device → eUICC | ENVELOPE EVENT DOWNLOAD |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* RECEIVE DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:* RECEIVE DATA |
| Device → eUICC | TERMINAL RESPONSE containing the data sent by the off-card entity |

|  |  |
| --- | --- |
| **Direction** | **Sequence / Description** |
| *Several RECEIVE DATA commands MAY be used to retrieve the complete data* | |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* SEND DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:*  SEND DATA containing the data to send to the off-card entity |
| Device → eUICC | TERMINAL RESPONSE |
| *Several SEND DATA commands MAY be used to send the complete data* | |
| Device → eUICC | ENVELOPE EVENT DOWNLOAD |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* RECEIVE DATA |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:* RECEIVE DATA |
| Device → eUICC | TERMINAL RESPONSE containing the message sent by the off-card entity to close the session |
| *Before closing the channel, the card MAY send a confirmation* | |
| eUICC → Device | *PROACTIVE COMMAND PENDING:* CLOSE CHANNEL |
| Device → eUICC | FETCH |
| eUICC → Device | *PROACTIVE COMMAND:* CLOSE CHANNEL |
| Device → eUICC | TERMINAL RESPONSE |
| *Note: It is assumed that some proactive commands TIMER MANAGEMENT or MORE TIME MAY be sent by the eUICC at any time* | |

###### Table 21: BIP Exchanges

# Annex G CAT\_TP PDUs

Here are the different CAT\_TP PDUs that SHALL be used by the CAT\_TP entities during a test sequence. The values in square brackets depend on the context and the CAT\_TP implementation. The other values need to be checked.

|  |  |
| --- | --- |
| **PDU** | **Value in hexadecimal string** |
| ACK\_DATA | 40 00 00 12  {SRC\_PORT}  {DEST\_PORT}  {DATA\_LENGTH}  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS}  {DATA}  Or  44 00 00 12  {SRC\_PORT}  {DEST\_PORT}  {DATA\_LENGTH}  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS}  {DATA}  See [Annex C](#_bookmark280) for the definition of {SRC\_PORT}, {DEST\_PORT}, {SEQ\_NUM}, {ACK\_NUM},  {WIN\_SIZE}, and {CS}.  {DATA} is either a command packet or a response packet as defined in ETSI TS 102 225 [[4].](#_bookmark9)  If the data length is higher to the Maximum PDU size, the ACK\_DATA SHALL be segmented (1st byte = ‘44’) and the data SHALL be split in several PDUs.  The command packet length SHALL NOT be higher than the Maximum SDU size. |
| ACK\_NO\_DATA | 40 00 00 12  {SRC\_PORT}  {DEST\_PORT} 00 00  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS}  See [Annex C](#_bookmark280) for the definition of {SRC\_PORT}, {DEST\_PORT}, {SEQ\_NUM}, {ACK\_NUM},  {WIN\_SIZE}, and {CS}. |
| ACK\_NUL | 48 00 00 12 |

|  |  |
| --- | --- |
| **PDU** | **Value in hexadecimal string** |
|  | {SRC\_PORT}  {DEST\_PORT} 00 00  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS}  See Annex C for the definition of {SRC\_PORT}, {DEST\_PORT}, {SEQ\_NUM}, {ACK\_NUM},  {WIN\_SIZE}, and {CS}. |
| RST | 10 00 00 13  {SRC\_PORT}  {DEST\_PORT} 00 00  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS}  {REASON\_CODE}  See [Annex C](#_bookmark280) for the definition of {SRC\_PORT}, {DEST\_PORT}, {SEQ\_NUM}, {ACK\_NUM},  {WIN\_SIZE}, {CS} and {REASON\_CODE}. |
| SYN | 80 00 00 {HL}  {SRC\_PORT} #CAT\_TP\_PORT 00 00  {SEQ\_NUM} 00 00  {WIN\_SIZE}  {CS}  {MAX\_PDU\_SIZE}  {MAX\_SDU\_SIZE}  #EID (optional: it MAY contain another value)  See [Annex C](#_bookmark280) for the definition of {HL}, {SRC\_PORT}, {SEQ\_NUM}, {WIN\_SIZE}, {CS},  {MAX\_PDU\_SIZE} and {MAX\_SDU\_SIZE}.  {WIN\_SIZE} SHALL be taken into account by the off-card entity.  {MAX\_SDU\_SIZE} and {MAX\_PDU\_SIZE} SHALL be taken into account by the off-card entity. |
| SYN\_ACK | C0 00 00 {HL} #CAT\_TP\_PORT  {DEST\_PORT} 00 00  {SEQ\_NUM}  {ACK\_NUM}  {WIN\_SIZE}  {CS} |

|  |  |
| --- | --- |
| **PDU** | **Value in hexadecimal string** |
|  | {MAX\_PDU\_SIZE}  {MAX\_SDU\_SIZE}  {IDENTIFICATION\_DATA}  See [Annex C](#_bookmark280) for the definition of {HL}, {DEST\_PORT}, {SEQ\_NUM}, {ACK\_NUM},  {WIN\_SIZE}, {CS}, {MAX\_PDU\_SIZE} and {MAX\_SDU\_SIZE}.  {IDENTIFICATION\_DATA} is the off-card entity identification data which can be freely chosen. |

###### Table 22: CAT\_TP PDUs

# Annex H TLS Records

Here are the different TLS records that SHALL be used by the TLS entities. All values defined in the tables below are hexadecimal strings. The values in square brackets depend on the context and the TLS implementation. The other values need to be checked.

|  |  |  |  |
| --- | --- | --- | --- |
| **TLS\_CLIENT\_HELLO** | | | |
| Content type: Handshake | | 16 | |
| Version: TLS 1.2 | | 03 03 | |
| Length | | {L} | |
| Protocol message | Message type: ClientHello | 01 | |
| Length | {L} | |
| Version: TLS 1.2 | 03 | 03 |
| Random value | AA | BB CC ……01 02 |
| Session id length | 00 | |
| Cipher suite length | {L} | |
| TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 | 00 | AE |
| TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 | 00 | A8 |
| Compression length | 01 | |
| Compression method: no compression | 00 | |
| Extension message length | 00 | 05 |
| Extension-type: max fragment length | 00 | 01 |
| Extension data length | 00 | 01 |
| Max fragment length: 2^9 | 01 | |
| *Note 1: TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 and/or TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256*  *SHALL be present. Other cipher suites MAY be present. Note 2: The TLS record length is coded with 2 bytes.*  *Note 3: The protocol message length is coded with 3 bytes. Note 4: The cipher suites length is coded with 2 bytes.*  *Note 5: The random value present in the table above is informative.* | | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **TLS\_SERVER\_HELLO** | | | |
| Content type: Handshake | | 16 | |
| Version: TLS 1.2 | | 03 03 | |
| Length | | {L} | |
| Protocol message | Message type: ServerHello | 02 | |
| Length | {L} | |
| Version: TLS 1.2 | 03 | 03 |
| Random value | AA | BB CC ……01 02 |
| Session id length | {L} | |
| Session id | AA | BB CC … |
| TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 | 00 | A8 |
| Compression method: no compression | 00 | |
| Extension message length | 00 | 05 |
| Extension-type: max fragment length | 00 | 01 |
| Extension data length | 00 | 01 |
| Max fragment length: 2^9 | 01 | |
| *Note 1: The cipher suite MAY be also TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256. Note 2: The TLS record length is coded with 2 bytes.*  *Note 3: The protocol message length is coded with 3 bytes.*  *Note 4: The random value and the session ID present in the table above are informative.* | | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_SERVER\_HELLO\_DONE** | | |
| Content type: Handshake | | 16 |
| Version: TLS 1.2 | | 03 03 |
| Length | | 00 04 |
| Protocol message | Message type: ServerHelloDone | 0E |

|  |  |  |
| --- | --- | --- |
|  | Length | 00 00 00 |
| *Note: this TLS record MAY be concatenated to the TLS\_SERVER\_HELLO message* | | |

|  |  |  |  |
| --- | --- | --- | --- |
| **TLS\_1\_1\_SERVER\_HELLO** | | | |
| Content type: Handshake | | 16 | |
| Version: TLS 1.1 | | 03 02 | |
| Length | | {L} | |
| Protocol message | Message type: ServerHello | 02 | |
| Length | {L} | |
| Version: TLS 1.1 | 03 | 02 |
| Random value | AA | BB CC ……01 02 |
| Session id length | {L} | |
| Session id | AA | BB CC … |
| TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 | 00 | AE |
| Compression method: no compression | 00 | |
| Extension message length | 00 | 05 |
| Extension-type: max fragment length | 00 | 01 |
| Extension data length | 00 | 01 |
| Max fragment length: 2^9 | 01 | |
| *Note 1: The TLS record length is coded with 2 bytes.*  *Note 2: The protocol message length is coded with 3 bytes.*  *Note 3: The random value and the session ID present in the table above are informative.* | | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_1\_1\_SERVER\_HELLO\_DONE** | | |
| Content type: Handshake | | 16 |
| Version: TLS 1.1 | | 03 02 |
| Length | | 00 04 |
| Protocol message | Message type: ServerHelloDone | 0E |
| Length | 00 00 00 |
| *Note: this TLS record MAY be concatenated to the TLS\_1\_1\_SERVER\_HELLO message* | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_CLIENT\_KEY\_EXCHANGE** | | |
| Content type: Handshake | | 16 |
| Version: TLS 1.2 | | 03 03 |
| Length | | {L} |
| Protocol message | Message type: ClientKeyExchange | 10 |
| Length | {L} |
| PSK Identity length | {L} |
| PSK Identity | #PSK\_ID |
| *Note 1: The TLS record length is coded with 2 bytes.*  *Note 2: The protocol message length is coded with 3 bytes. Note 3: The PSK Identity length is coded with 2 bytes.* | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_CHANGE\_CIPHER\_SPEC** | | |
| Content type: ChangeCipherSpec | | 14 |
| Version: TLS 1.2 | | 03 03 |
| Length | | 00 01 |
| Protocol message | Message type: ChangeCipherSpec | 01 |

|  |  |
| --- | --- |
| **TLS\_FINISHED** | |
| Content type: Handshake | 16 |
| Version: TLS 1.2 | 03 03 |

|  |  |  |
| --- | --- | --- |
| Length | | {L} |
| Protocol message | Message type: Finished | 14 |
| Length | {L} |
| Ciphered data | AA BB CC … |
| *Note 1: The TLS record length is coded with 2 bytes.*  *Note 2: The protocol message length is coded with 3 bytes.*  *Note 3: The ciphered data present in the table above is informative.* | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_APPLICATION** | | |
| Content type: Application | | 17 |
| Version: TLS 1.2 | | 03 03 |
| Length | | {L} |
| Protocol message | Ciphered data | AA BB CC … |
| MAC | AA BB CC … |
| Padding | 01 |
| *Note 1: The ciphered data contains the HTTP content. Note 2: The TLS record length is coded with 2 bytes.*  *Note 3: The ciphered data, the MAC and the padding present in the table above are informative.* | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_ALERT\_CLOSE\_NOTIFY** | | |
| Content type: Handshake | | 15 |
| Version: TLS 1.2 | | 03 03 |
| Length | | {L} |
| Protocol message | Alert level : Warning | 01 |
| Alert description: Close notify | 00 |
| MAC | AA BB … |
| Padding | 01 |
| *Note 1: The TLS record length is coded with 2 bytes.*  *Note 2: The MAC and the padding present in the table above are informative.* | | |

|  |  |  |
| --- | --- | --- |
| **TLS\_ALERT\_PROTOCOL\_VERSION** | | |
| Content type: Handshake | | 15 |
| Version: TLS 1.2 | | 03 03 |
| Length | | {L} |
| Protocol message | Alert level : Fatal | 02 |
| Alert description: Protocol version | 46 |
| MAC | AA BB … |
| Padding | 01 |
| *Note 1: The TLS record length is coded with 2 bytes.*  *Note 2: The MAC and the padding present in the table above are informative.* | | |

# Annex I Initial States

Here are all the initial states of the different entities under test. Each initial state is an extract of the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7). As consequence, each cross-reference present in the table below (i.e. column Initial state) does not refer to documents listed in the section [1.5](#_bookmark5) of this Test Plan. The column “Chapter” refers to the section where the initial state is defined in the document GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7).

|  |  |
| --- | --- |
| **Chapter** | **Initial state** |
| 2.2.1.1 | There SHALL be only one ISD-R on an eUICC.  The ISD-R SHALL be installed and first personalized by the EUM during eUICC manufacturing. The ISD-R SHALL be Associated with itself.  After eUICC manufacturing, the ISD-R SHALL be in life-cycle state PERSONALIZED as defined in GlobalPlatform Card Specification [6], section 5.3. The ISD-R privileges SHALL be granted according to Annex C. |
| 2.2.1.2 | There SHALL be only one ECASD on an eUICC.  The ECASD SHALL be installed and personalized by the EUM during the eUICC manufacturing. The ECASD SHALL be Associated with the ISD-R.  After eUICC manufacturing, the ECASD SHALL be in life-cycle state PERSONALIZED as defined in GlobalPlatform Card Specification [6], section 5.3.  The ECASD SHALL be personalized by the EUM during eUICC manufacturing with:   * PK.CI.ECDSA * SK.ECASD.ECKA * CERT.ECASD.ECKA for eUICC Authentication and key establishment * EID |
| 2.2.1.3 | At least one ISD-P with a Profile SHALL be installed and first personalized by the EUM during eUICC manufacturing to allow future eUICC connectivity. |
| 2.2.3 | The RID of the Executable Load File, the Executable Module and the Application of the ISD- R and the ECASD SHALL be set to 'A000000559' (as defined in ISO/IEC 7816-5:2004).  The ISD- R Executable Load File AID and the ISD-R Executable Module AID can be freely selected by the EUM. The ISD-R application AID SHALL be ‘A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 01 00' as defined into Annex H. The ECASD Executable Load File AID and the ECASD Executable Module AID can be freely selected by the EUM. |
| 2.2.5.1 | To enable SCP80, the ISD-R SHALL be personalized before issuance by the EUM with at least one key set, with a Key Version Number between ‘01’ to ‘0F’ following GlobalPlatform Card Specification UICC Configuration [7]. |
| 2.2.5.1 | To enable SCP81, the ISD-R SHALL be personalized with at least one key set, with a Key Version Number between ‘40’ to ‘4F’ following GlobalPlatform Secure Element Configuration[34]. |
| 2.3 | * Every SM-SR and SM-DP SHALL be certified according to a GSMA agreed certification scheme. * The eUICC SHALL be certified according to the GSMA eUICC Protection Profile. * The eUICC Manufacturer SHALL be SAS certified. |

|  |  |
| --- | --- |
| **Chapter** | **Initial state** |
| 2.3.1 | The Certificate Issuer (CI) Role issues the certificates for the eUICC Remote Provisioning  System and acts as a trusted third party for the purpose of mutual authentication of the entities of the system. The CI provides:   * A self-signed Root Certificate used to verify certificates issued and signed by the CI. * A public key (PK.CI.ECDSA), part of that Root Certificate, used on the eUICC to verify certificates issued by the CI. * A certificate (CERT.DP.ECDSA, signed by the CI) to authenticate the SM-DP. This certificate is used in the “Load and Install Profile” procedure. * A certificate (CERT.SR.ECDSA, signed by the CI) to authenticate the SM-SR. This certificate is used in the “SM-SR change” procedure. * A certificate, signed by the CI, to authenticate the EUM. This certificate is used in the "Download and Install Profile" and in the “SM-SR change” procedures. |
| 2.3.2 | The following certificates SHALL be signed and issued by the CI:   * Self-signed Root Certificate * EUM Certificates * SM-SR Certificates * SM-DP Certificates |
| 2.3.2 | The following certificates SHALL be signed and issued by the EUM:   * eUICC Certificates |
| 2.3.2 | The following certificate and key SHALL be stored in the eUICC:   * the eUICC Certificate * the Root public key |
| 2.3.2 | The eUICC Certificate is part of the EIS (eUICC Information Set) which is stored in the SM-SR and/or at EUM level. This certificate contains:   * the PK.ECASD.ECKA used for ElGamal Elliptic Curves key agreement as defined in GlobalPlatform Card Specification Amendment E [11] * the EID * the technical reference of the product, which allows the Common Criteria (CC) certification report to be identified by Common Criteria certification body |
| 3.1.5 | The notification of “First network attachment” has been generated by the eUICC and confirmed by the SM-SR. |
| Annex B | In case Web Services is used, the section "Binding to SOA environment" is normative and implementation SHALL comply with the requirements provided in this section. |
| Annex B / 2 | This specification mandates usage of SOAP v1.2 as the minimal version and specified in [40]. |
| Annex B / 2.1.2 | WS-MakeConnection SHALL be used in asynchronous scenarios when the receiving party of a request cannot initiate a connection to the sending party (due to network security constraints for example). |

|  |  |
| --- | --- |
| **Chapter** | **Initial state** |
| Annex B / 2.2 | To secure the messages being sent between Function requester and Function provider, one of the two following mechanisms SHALL be used:   1. Relying on mutual authenticated transport level security (Transport Layer Security, TLS) 2. Relying on transport level security (TLS) with only server side authentication and WS- Security standards   This specification mandates usage of TLS v 1.2 defined in RFC 5246 [15] to allow appropriate algorithm and key length as defined in section 2.4.1 |
| Annex B / 4 | In case Web Services are used, the following WSDL files (provided within the SGP.02 WSDL package) SHALL be used:   * ES1\_SMSR.wsdl * ES2\_MNO.wsdl * ES2\_SMDP.wsdl * ES3\_SMDP.wsdl * ES3\_SMSR.wsdl * ES4\_MNO.wsdl * ES4\_SMSR.wsdl * ES7\_SMSR\_Provider.wsdl * ES7\_SMSR\_Requester.wsdl |

###### Table 23: Initial States

GSM Association Non-Confidential

SGP.11 Remote Provisioning Architecture for Embedded UICC Test Specification

# Annex J Requirements

Each requirement in the tables below is an extract of either the GSMA Embedded SIM Remote Provisioning Architecture [[1]](#_bookmark6) or the GSMA Remote Provisioning Architecture for Embedded UICC-Technical Specification [[2]](#_bookmark7).

## Format of the Requirements Table

The columns in Table 21 and 22 have the following meaning:

|  |  |
| --- | --- |
| **Column** | **Meaning** |
| ID | Requirement identifier used in the test cases defined in this Test Plan. This identifier is unique and formatted as follow “XXX\_REQYYY” with   * XXX: a prefix related to the corresponding functional group * YYY: a number |
| Source | The cross-reference to the source document where the requirement is specified. All cross-references are described in the section [1.5](#_bookmark5) of this Test Plan. |
| Chapter | The chapter in the source document where the requirement is specified. |
| Support | The following common notations are used for the support column: M mandatory: SHALL be supported by the implementation  C conditional: the support of the requirement depends of the support of other requirement(s)  O optional: MAY be supported or not by the implementation |
| Description | An extract of the source document that describes the requirement. Some of these descriptions are adapted for readability reason. All cross-references present in this column do not refer to the ones present in this document (i.e. section [1.5](#_bookmark5)) but refer to cross-references defined in the corresponding source document.  The notes in *italic and underline* SHALL be considered as remarks or comments related to the requirement. |
| Functional group | Functional group of the corresponding requirement. A functional group MAY be:   * Platform Management * eUICC Management * Profile Management * Procedure Flow * Security |

###### Table 24 Format of the Tables of Requirements

## Requirements in Scope

Here are all the requirements’ descriptions that are covered by this Test Plan.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ1 | [[2]](#_bookmark7) | 2.2.1.1 | M | The LOCKED state SHALL NOT be supported by the ISD-R. | eUICC Management |
| PF\_REQ1 | [[2]](#_bookmark7) | 2.2.1.1 | M | The ISD-R SHALL only be able to perform Platform Management functions on ISD-Ps. | Platform Management |
| PM\_REQ1 | [[2]](#_bookmark7) | 2.2.1.3 | M | No component outside the ISD-P SHALL have visibility or access to any Profile Component with the exception of the ISD-R, which SHALL have read access to POL1 | Profile Management |
| PM\_REQ2 | [[2]](#_bookmark7) | 2.2.1.3 | M | A Profile Component SHALL NOT have any visibility of, or access to, components outside its ISD-P. An ISD-P SHALL NOT have any visibility of, or access to, any other ISD-P. | Profile Management |
| EUICC\_REQ2 | [[2]](#_bookmark7) | 2.2.1.3 | M | It SHALL be possible to allocate the same AID within different Profiles.  A Profile Component SHALL NOT use the reserved ISD-R, ISD-P and ECASD AIDs. | eUICC Management |
| EUICC\_REQ3 | [[2]](#_bookmark7) | 2.2.1.3 | M | It SHALL be possible to allocate the same TAR within distinct Profiles.  A Profile Component SHALL NOT use the reserved ISD-R, ISD-P and ECASD TARs. | eUICC Management |
| EUICC\_REQ4 | [[2]](#_bookmark7) | 2.2.1.3 | M | After execution of the procedure described in section 3.1.1 (ISD-P creation), the ISD-P SHALL be in SELECTABLE state | eUICC Management |
| EUICC\_REQ5 | [[2]](#_bookmark7) | 2.2.1.3 | M | After execution of the procedure described in section 3.1.2 (Key Establishment with Scenario#3- Mutual Authentication), the ISD-P SHALL be in PERSONALIZED state | eUICC Management |
| PM\_REQ3 | [[2]](#_bookmark7) | 2.2.1.3 | M | After execution of the procedure described in section 3.1.3 (Download and Installation of the Profile) or 3.4 (Profile Disabling), the ISD-P SHALL be in the DISABLED state.  The ISD-P can also transition to the DISABLED state as the result of the enabling of another ISD- P as described in section 3.2, or the activation of the fall-back mechanism. | Profile Management |
| PM\_REQ4 | [[2]](#_bookmark7) | 2.2.1.3 | M | After execution of the procedure described in section 3.2 (Profile Enabling), the ISD-P SHALL be in the ENABLED state. The ISD-P can also transition to the ENABLED state as the result of the activation of the fall-back mechanism. | Profile Management |
| EUICC\_REQ6 | [[2]](#_bookmark7) | 2.2.1.3 | M | The LOCKED state SHALL NOT be supported by an ISD-P. | eUICC Management |
| EUICC\_REQ7 | [[2]](#_bookmark7) | 2.2.1.3 | M | When an ISD-P is not in Enabled state, the eUICC SHALL ensure that Remote management of any Profile Component is not possible via the ES6 interface | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ8 | [[2]](#_bookmark7) | 2.2.1.3 | M | When an ISD-P is not in Enabled state, the eUICC SHALL ensure that the file system within the Profile cannot be selected by the Device or any application on the eUICC | eUICC Management |
| EUICC\_REQ9 | [[2]](#_bookmark7) | 2.2.1.3 | M | When an ISD-P is not in Enabled state, the eUICC SHALL ensure that the applications (including NAAs and Security Domains) within the Profile cannot be selected, triggered or deleted. | eUICC Management |
| EUICC\_REQ10 | [[2]](#_bookmark7) | 2.2.2 | M | The EID SHALL be stored within the ECASD and can be retrieved by the Device at any time using the standard GlobalPlatform GET DATA command by targeting the ECASD as specified in GlobalPlatform Card Specification [6] as follows:   * Select the ECASD using the SELECT command with the AID value defined in section 2.2.3, * Send a ‘GET DATA’ command to the ECASD with the data object tag '5A' to get the EID.   The EID SHALL have the format described in section 2.2.2. | eUICC Management |
| EUICC\_REQ11 | [[2]](#_bookmark7) | 2.2.3 | M | The ECASD application AID SHALL be ‘A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 02 00' as  defined into Annex H. | eUICC Management |
| EUICC\_REQ12 | [[2]](#_bookmark7) | 2.2.3 | M | The ISD-P application SHALL be installed by SM-SR during the “Profile Download and Installation” procedure.  The ISD-P Executable Load File AID SHALL be ‘A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 0D  00' as defined into Annex H.  The ISD-P Executable Module AID SHALL be ‘A0 00 00 05 59 10 10 FF FF FF FF 89 00 00 0E  00' as defined into Annex H.  The ISD-P application AID SHALL be coded according to Annex 8. The SM-SR SHALL allocate the ISD-P application AID in the range defined in Annex H. | eUICC Management |
| PM\_REQ5 | [[2]](#_bookmark7) | 2.2.3 | M | The MNO-SD application AID and TAR(s) can be freely allocated by the MNO during Profile definition. | Profile Management |
| EUICC\_REQ13 | [[2]](#_bookmark7) | 2.2.5.1 | M | The eUICC SHALL support SCP80 (defined in ETSI 102 225 [4] and ETSI 102 226 [5]). | eUICC Management |
| EUICC\_REQ14 | [[2]](#_bookmark7) | 2.2.5.1 | C | The eUICC MAY support SCP81 (as defined in ETSI TS 102 226)  *Note: If EUICC\_REQ18 is not supported, this requirement SHALL be supported* | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ15 | [[2]](#_bookmark7) | 2.2.5.2 | M | To enable SCP03 and SCP03t, the ISD-P SHALL be personalized with at least one key set, with a Key Version number between ‘30’ to ‘3F’ (see GlobalPlatform Secure Element Configuration [34]). | eUICC Management |
| EUICC\_REQ16 | [[2]](#_bookmark7) | 2.3 | M | For the eUICC interfaces, the Platform Management commands (ES5) and the OTA Platform commands (ES6) SHALL be protected by either a SCP80 or SCP81 secure channel with security level defined in section 2.4. | eUICC Management |
| EUICC\_REQ17 | [[2]](#_bookmark7) | 2.3 | M | The Profile Management commands (ES8) SHALL be at least protected by a SCP03 security level as detailed in section 2.5. | eUICC Management |
| EUICC\_REQ18 | [[2]](#_bookmark7) | 2.4.1 | C | The eUICC MAY support CAT\_TP  *Note: If EUICC\_REQ14 is not supported, this requirement SHALL be supported* | eUICC Management |
| PF\_REQ2 | [[2]](#_bookmark7) | 2.4.1 | M | The SM-SR SHALL support SMS, HTTPS and CAT\_TP. | Platform Management |
| EUICC\_REQ19 | [[2]](#_bookmark7) | 2.4.3 | M | The eUICC SHALL support the sending of secure packet over SMS as defined in 3GPP TS  31.115 [13]. The eUICC SHALL support RAM over SMS as defined in ETSI TS 102 226 [5]. The eUICC SHALL comply with 3GPP TS 31.111 [27] and 3GPP TS 31.116 [28].  Except for the notification described in section 3.15.1, concerning the security level, the SMS (MT or MO) SHALL make use of a CC with a length of 64 bits using AES CMAC mode, ciphering using AES in CBC mode and counter value higher (SPI1=’16’).. | eUICC Management |
| EUICC\_REQ20 | [[2]](#_bookmark7) | 2.4.3 | M | Procedures for the PoR SHALL follow ETSI TS 102 225 **Error! Reference source not found.** a nd 3GPP TS 31.115 **Error! Reference source not found.** with the following precisions:   * In the case that an incoming SMS for the ISD-R does not meet the security level described in “EUICC\_REQ19”, it must be rejected by the eUICC and no PoR SHALL be sent back * When the eUICC cannot authenticate the SM-SR, it SHALL NOT send any PoR and discard the command packet with no further action being taken | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ54 | [[2]](#_bookmark7) | 2.4.3 | M | SPI2 SHALL be set to:   * ‘00’: no PoR (this value SHALL only be used for the notification described in section   3.15.1 and optionally for the SMS for HTTPS session triggering described in section 2.4.3.1),   * or to ‘39’: PoR with CC and encryption. | eUICC Management |
| EUICC\_REQ21 | [[2]](#_bookmark7) | 2.4.3 | M | When a PoR is returned, the SMS SHALL make use of a CC with a length of 64 bits using AES CMAC mode, ciphering using AES in CBC mode and SHALL be sent using SMS-SUBMIT mode. | eUICC Management |
| EUICC\_REQ21\_1 | [[2]](#_bookmark7) | 2.4.3 | M | The SM-SR MAY choose to request a PoR or not for this special SMS, and set the SPI2 byte of the SMS accordingly. | eUICC Management |
| EUICC\_REQ22 | [[2]](#_bookmark7) | 2.4.3.3 | M | The commands sent to the eUICC within a secure script in SMS SHALL be formatted as an expanded remote command structure as defined in ETSI TS 102 226 [5]. As a consequence, the eUICC SHALL provide the answer as an expanded remote response structure. | eUICC Management |
| EUICC\_REQ23 | [[2]](#_bookmark7) | 2.5 | M | The eUICC SHALL support the Secure Channel Protocol 03 (SCP03) as defined in GlobalPlatform Card Specification Amendment D [10], as well as the variant SCP03t defined in this specification (see section **Error! Reference source not found.**), with:   * AES in CBC mode with key length of 128 bits, referred as AES-128 * Use of C-MAC, C-DECRYPTION R-MAC and R-ENCRYPTION for SCP03 (set in reference control parameter P1 of the EXTERNAL AUTHENTICATE command) and for SCP03t * Use of mode i=’70’, meaning use of pseudo-random card challenge, R-MAC and R- ENCRYPTION support   As a result the SM-DP and its ISD-P are mutually authenticated, all commands sent from the SM- DP to the ISD-P are signed and encrypted, and all responses sent by the ISD-P to the SM-DP are also signed and encrypted. | eUICC Management |
| EUICC\_REQ28 | [[2]](#_bookmark7) | 4.1.1.11 | M | ES5: HandleDefaultNotification  A protocol priority order for default notification MAY be defined for every Profile during profile installation or download, and updated using the functions defined in 4.1.2.2 and 4.1.3.4. This protocol priority order specifies which protocols to use, and in which order, among SMS, HTTPS and CAT\_TP.  If not defined for a Profile, the default priority order is set as follow: SMS, HTTPS, CAT\_TP | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PROC\_REQ1 | [[2]](#_bookmark7) | 3.1.1 | M | The ISD-P creation process must be compliant with the Figure 10 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ2 | [[2]](#_bookmark7) | 3.1.2 | M | The Key Establishment with Scenario#3-Mutual Authentication process must be compliant with the Figure 11 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ3 | [[2]](#_bookmark7) | 3.1.3 | M | The Download and Installation of the Profile process must be compliant with the Figure 12 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ4 | [[2]](#_bookmark7) | 3.1.4 | M | The Error Management Sub-Routine described in Figure 13 must be called when an error occurs during the key-establishment or the Profile Download and Installation procedures. This process SHALL be compliant with the procedure described in this section. | Procedure Flow |
| PROC\_REQ5 | [[2]](#_bookmark7) | 3.2.1 | M | The profile enabling process must be compliant with the Figure 14 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ5\_1 | [2] | 3.2.1 |  | Profile Enabling process:  If the previously Enabled Profile (now the Disabled) has the Fall-back Attribute, and its POL1 contains the rule “Profile deletion is mandatory when its state is changed to disabled”, this rule SHALL be ignored according to Section 3.6.3.2 in GSMA Remote Provisioning Architecture for the Embedded UICC [1], and the procedure SHALL continue at step 10. | Procedure Flow |
| PROC\_REQ6 | [[2]](#_bookmark7) | 3.2.2 | M | The Connectivity failure case described in Figure 15 must be called when an error occurs during the profile enabling procedure. This process SHALL be compliant with the procedure described in this section. | Procedure Flow |
| PROC\_REQ7 | [[2]](#_bookmark7) | 3.3.1 | M | The Profile Enabling via SM-DP must be compliant with the Figure 16 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ8 | [[2]](#_bookmark7) | 3.3.2 | M | The connectivity failure case described in Figure 17 must be called when an error occurs during the profile enabling via SM-DP procedure. This process SHALL be compliant with the procedure described in this section. | Procedure Flow |
| PROC\_REQ9 | [[2]](#_bookmark7) | 3.4 | M | The Profile Disabling process must be compliant with the Figure 18 and with the procedure described in this section. | Procedure Flow |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PROC\_REQ10 | [[2]](#_bookmark7) | 3.5 | M | The Profile Disabling via SM-DP process must be compliant with the Figure 19 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ11 | [[2]](#_bookmark7) | 3.6 | M | The Profile and ISD-P deletion process must be compliant with the Figure 20 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ12 | [[2]](#_bookmark7) | 3.7 | M | The Profile and ISD-P Deletion via SM-DP must be compliant with the Figure 21 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ13 | [[2]](#_bookmark7) | 3.8 | M | The SM-SR Change process must be compliant with the Figure 23 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ13\_1 | [2] | 3.8 | M | The length of the Random Challenge SHALL be 0x10 or 0x20. | Procedure Flow |
| PROC\_REQ14 | [[2]](#_bookmark7) | 3.9 | M | The eUICC registration process must be compliant with the Figure 23 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ16 | [[2]](#_bookmark7) | 3.11 | M | The POL2 Update via SM-DP process must be compliant with the Figure 25 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ17 | [[2]](#_bookmark7) | 3.12 | M | The POL1Update by MNO process must be compliant with the Figure 26 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ18 | [[2]](#_bookmark7) | 3.13 | M | The Connectivity Parameters Update by MNO must be compliant with the Figure 27 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ19 | [[2]](#_bookmark7) | 3.14 | M | The Connectivity Parameters Update using SCP03 must be compliant with the Figure 28 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ20 | [[2]](#_bookmark7) | 3.15.1 | M | The Default Notification Procedure using SMS must be compliant with the Figure 29 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ21 | [[2]](#_bookmark7) | 3.15.2 | M | The Default Notification Procedure using HTTPS must be compliant with the Figure 30 and with the procedure described in this section. | Procedure Flow |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ3 | [[2]](#_bookmark7) | 4.1.1.1 | M | ES5: CreateISDP  Description:  This function creates an ISD-P on the eUICC.  Parameters:   * ISD-P-AID * Memory quota for the ISD-P (optional)   Command Description:  INSTALL COMMAND  The command is an Install command as defined in GlobalPlatform Card Specification [6] and must be compliant with the Tables defined in section 4.1.1.1.  Privileges granted to the ISD-P, as specified in Annex C, SHALL be at least:   * Security Domain * Trusted Path * Authorized Management   Data Field Returned in the Response Message:  A single byte of '00' SHALL be returned indicating that no additional data is present, as defined in the GlobalPlatform Card Specification [6]. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ4 | [[2]](#_bookmark7) | 4.1.1.2 | M | ES5: EnableProfile  Description:  This function is used to enable a Profile on the eUICC.  The function makes the target Profile Enabled, and disables implicitly the currently Enabled Profile.  Parameters:   * ISD-P-AID   Function flow  Upon reception of the Profile Enabling command, the eUICC shall:   * Verify that the target Profile is in the Disabled state * Verify that POL1 of the currently Enabled Profile allows its disabling * If any of these verifications fail, terminate the command with an error status word * Disable the currently Enabled Profile and Enable the target Profile * Send the REFRESH command in “UICC Reset” mode to the Device according to ETSI TS 102 223 [3] * Send notification   Command Description:  STORE DATA COMMAND  This command is a STORE DATA command, as described in GlobalPlatform Card Specification  [6] and in Tables defined in section 4.1.1.2.  Data Field Returned in the Response Message:  The data field of the response message SHALL NOT be present.  Specific Processing State returned in response Message: ’69 85’: Profile is not in the Disabled state.  ’69 E1’: POL1 of the currently Enabled Profile prevents this action. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ5 | [[2]](#_bookmark7) | 4.1.1.3 | M | ES5: DisableProfile  Description:  This function is used to disable a Profile on the eUICC.  This function makes the target Profile Disabled, and implicitly enables the Profile which has the Fall-back Attribute set.  Parameters:   * ISD-P-AID of the currently Enabled Profile   Function flow  Upon reception of the Profile Disabling command, the eUICC shall:   * Verify that the target Profile is in Enabled state * Verify that POL1 of the currently Enabled Profile allows its disabling * Verify that the target Profile is not the Profile with Fall-back Attribute set * If any of these verifications fail, terminate the command with an error status word * Disable the target Profile and enable the Profile with the Fall-back Attribute set * Send the REFRESH command in “UICC Reset” mode to the Device according to ETSI TS 102 223 [3].   Command Description:  STORE DATA COMMAND  This command is a STORE DATA command, as described in GlobalPlatform Card Specification  [6] and in Tables defined in section 4.1.1.3.  Data Field Returned in the Response Message:  The data field of the response message SHALL NOT be present.  Specific Processing State returned in response Message:  ’69 85’: Profile is not in the Enabled state or Profile has the Fall-back Attribute. ’69 E1’: POL1 of the Profile prevents disabling. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ6 | [[2]](#_bookmark7) | 4.1.1.4 | M | ES5: DeleteProfile  Description:  This function is used to delete a Profile from the eUICC. This function deletes the ISD-P and its associated Profile.  Parameters:   * ISD-P-AID   Function flow  Upon reception of the DELETE command, the eUICC shall:   * Verify that POL1 of the target Profile allows its deletion * Verify that the target Profile is not the Profile with Fall-back Attribute set * Verify that the target Profile is not in the Enabled state * If any of these verifications fail, terminate the command with an error status word * Delete the ISD-P with its Profile   Command Description:  DELETE COMMAND  This function is realized through the GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification Amendment C [9] and in Tables defined in section 4.1.1.4.  Data Field Returned in the Response Message:  A single byte of '00' SHALL be returned indicating that no additional data is present.  Specific Processing State returned in response Message:  ’69 85’: Profile is in Enabled state or Profile has the Fall-back Attribute. ’69 E1’: POL1 of the Profile prevents deletion. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ7 | [[2]](#_bookmark7) | 4.1.1.5 | M | ES5: eUICCCapabilityAudit  Description:  This function is used to query the status of the eUICC.  Parameters:  It MAY be used to ensure the data within the SM-SR’s EIS database is up to date.  This function uses two commands which SHALL be implemented as an extension of the GlobalPlatform functions GET DATA and GET STATUS.  Commands Description:  GET DATA  The GET DATA command is coded according to the Tables defined in section 4.1.1.5. This function can return:   * Number of installed ISD-P and available not allocated memory * ECASD Certificate   Data Field Returned in the Response Message:  The coding of the response message is defined in Tables defined in section 4.1.1.5.  GET STATUS  The GET STATUS command is coded according to Tables defined in section 4.1.1.5. This function can return:   * Each ISD-P-AID * State of the ISD-Ps / Profiles   Data Field Returned in the Response Message:  The coding of the response message is defined in Tables defined in section 4.1.1.5. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ8 | [[2]](#_bookmark7) | 4.1.1.6 | M | ES5: MasterDelete  Description:  This function deletes a target Profile on the target eUICC regardless of POL1 Rules. This function SHALL use the ISD-P token verification key in order to authenticate the source of the command.  Parameter:   * ISD-P-AID * Delete Token as defined by GlobalPlatform Card Specification [6] , provided by the SM-DP   Function flow  Upon reception of the Master Delete command, the eUICC shall:   * Verify that the target Profile is in the Disabled state * Verify that the target Profile is not the Profile with Fall-back Attribute set * Verify the Token (actually performed by the ISD-P). This includes verifying the signature of the Token, and verifying that the values of tags 42, 45, and 5F20 in the Token match the corresponding values in the ISD-P. * If any of these verifications fail, terminate the command with an error status word * Delete the ISD-P with its Profile, regardless of POL1   Command Description:  This function is realized through the GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification Amendment C [9] and in Tables defined in section 4.1.1.6.  Data Field Returned in the Response Message:  A single byte of ‘00’ SHALL be returned indicating that no additional data is present.  Specific Processing State returned in response Message:  ’69 85’: Profile is not in the Disabled state or Profile has the Fall-back Attribute. | Platform Management |
| PF\_REQ8\_1 | [2] | 4.1.1.6 | M | The eUICC SHALL support setting the value of tags 42, 45, and 5F20 by a STORE DATA command defined in GlobalPlatofrm Card Specification [6]. | Platform Management |
| PF\_REQ8\_2 | [2] | 4.1.1.6 | M | If the value of tag 5F20 is not set by the SM-DP, the default value SHALL be the value of the RID of ISD-P defined in section 2.2.3. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ9 | [[2]](#_bookmark7) | 4.1.1.7 | M | ES5: SetFallbackAttribute  Description:  This function sets the Fall-back Attribute for one Profile on the target eUICC.  Parameters:   * ISD-P-AID   Function flow  Upon reception of the STORE DATA command, the eUICC shall:   * Set the Fall-back Attribute for the target Profile * Remove the Fall-back Attribute from the Profile that has the attribute currently assigned Setting of the Fall-back Attribute is done via ISD-R.   Command Description:  STORE DATA Command  This function is realized through the GlobalPlatform STORE DATA command as defined in GlobalPlatform Card Specification [6] and in Tables defined in section 4.1.1.7.  Data Field Returned in the Response Message:  The data field of the response message SHALL NOT be present. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ24 | [[2]](#_bookmark7) | 4.1.1.8 | M | ES5: establishISDRKeySet  Description:  This function is used to perform mutual authentication between the new SM- SR and the eUICC and to establish a shared secret key set between the new SM-SR and the ISD-R.  Parameters:   * Ephemeral public key of the new SM-SR * Certificate for the new SM-SR   Command Description:  This function is realized through GlobalPlatform STORE DATA commands as defined in GlobalPlatform Card Specification [6].  First STORE DATA command Command Message  The STORE DATA command message SHALL be coded according to Tables defined in section 4.1.1.8.  Data Field Returned in the Response Message:  The STORE DATA response SHALL contain the data described in Tables defined in section 4.1.1.8.  Second STORE DATA command Command Message  The STORE DATA command message SHALL be coded according to Tables defined in section 4.1.1.8.  Data Field Returned in the Response Message:  The STORE DATA response SHALL contain the data described in Tables defined in section 4.1.1.8. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ25 | [[2]](#_bookmark7) | 4.1.1.9 | M | ES5: FinaliseISDRhandover  Description:  This function deletes all keys in the ISD-R except for the key ranges indicated by the command parameter(s).  It is intended as a simple clean-up mechanism for the new SM-SR after takeover to get RID of all keys of the previous SM-SR in the ISD-R.  Parameters:   * Key Ranges of keys not to be deleted.   Command Description:  DELETE COMMAND  This function is realized through a GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification [6] with proprietary parameters (see Tables defined in section 4.1.1.9).  Function flow  Upon reception of the DELETE command, the eUICC shall:   * Check that all keys of the key set(s) used for setting up the current secure channel are among the keys not to be deleted. For SCP81, this also includes the key set used for the push SM. If that check fails, the command is terminated without deleting any key. * Delete all keys except those in the key ranges indicated in the command parameters.   Data Field Returned in the Response Message:  The data field of the response message SHALL contain a single byte of ‘00’.  Specific Processing State returned in response Message:  ’69 85’: Key(s) of key set used for the current secure channel is/are among the keys to be deleted. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ26 | [[2]](#_bookmark7) | 4.1.1.10 | M | ES5: UpdateSMSRAddressingParameters  Description:  This function is used to update SM-SR addressing Parameters on the eUICC.  Parameters:   * ISD-R AID * SM-SR addressing Parameters   Function flow  Upon reception of the SM-SR addressing Parameters update command, the eUICC shall: Update the SM-SR addressing Parameters of the targeted ISD-R.  Commands  This command is a STORE DATA command, as described in GlobalPlatform Card Specification  [6] and in Tables defined in section 4.1.1.10.  Data Field Returned in the Response Message:  The data field of the response message SHALL NOT be present. | eUICC Management |
| EUICC\_REQ26\_1 | [[2]](#_bookmark7) | 4.1.1.10 | M | Each of the Tag 'A3', 'A4' and 'A5', SHALL be used to create or update the complete set of addressing parameters for corresponding protocol.  This structure can contain as many TLVs with tag ‘A2’ as there are ISD-Ps.  The SM-SR is responsible to update this list as it sees fit when a new ISD-P is created or after an SM-SR change. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ27 | [[2]](#_bookmark7) | 4.1.1.11 | M | ES5: HandleDefaultNotification  Description:  This function provides a default notification from the eUICC to the SM-SR.  Parameters:   * EID * ISD-P AID * Mobile Equipment Identification (e.g. MEID, IMEI) * Notification Sequence number * Notification type   The eUICC notification is composed of a single BER-TLV tag including several COMPREHENSION-TLV data objects; the COMPREHENSION-TLV format is defined in ETSI TS 102 223 [3].  See Tables defined in section 4.1.1.11.  Secured data structure for eUICC notification over SMS  The data SHALL be sent using definite length coding, and SHALL contain one Command TLV encapsulated in the Command Scripting Template. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ29 | [[2]](#_bookmark7) | 4.1.1.12 | M | ES5: HandleNotificationConfirmation  Description:  This function confirms the notification and triggers potential follow-up activities required by POL1.  Parameters:   * Notification Sequence number   Function flow  Upon reception of the STORE DATA command, the eUICC shall:   * Disable the retry mechanism for the notification * Perform the follow-up activities required by POL1 upon the activity that triggered the original notification * Return the result of any such activity in the response data   Command Description:  This function is realized through the GlobalPlatform STORE DATA command as defined in GlobalPlatform Card Specification [6] and in Tables defined in section 4.1.1.12.  Data Field Returned in the Response Message:  The data field of the response message SHALL either   * not be present, if no follow-up activities had to be performed, or * contain the data structure defined in section 4.1.1.12 if follow-up activities were performed | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ6 | [[2]](#_bookmark7) | 4.1.2.1 | M | ES6: UpdatePOL1byMNO  Description:  This function is used to update POL1 on the eUICC.  Parameters:   * POL1   Function flow  Upon reception of the POL1 update command, the eUICC shall:   * Update POL1 of the ISD-P containing the targeted MNO-SD.   Commands  This function consists of an INSTALL [for personalization] command followed by a STORE DATA command, as described in GlobalPlatform Card Specification [6] and in Tables defined in section 4.1.2.1.  Data Field Returned in the Response Message:  A single byte of '00' SHALL be returned indicating that no additional data is present, as defined in the GlobalPlatform Card Specification [6]. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ7 | [[2]](#_bookmark7) | 4.1.2.2 | M | ES6: UpdateConnectivityParametersByMNO  Description:  This function is used to update Connectivity Parameters on the eUICC.  Parameters:   * Connectivity Parameters   Function flow  Upon reception of the Connectivity Parameters update command, the eUICC shall:   * Update the Connectivity Parameters of the ISD-P containing the targeted MNO-SD.   Commands  This function consists of an INSTALL [for personalization] command followed by a STORE DATA command, as described in GlobalPlatform Card Specification [6].  According to GlobalPlatform Card Specification [6], INSTALL [for personalization] command can only be used on applications Associated with a Security Domain.  As an exception from this rule, the eUICC SHALL allow the MNO-SD to receive this command sequence with data destined to the ISD-P.  INSTALL [for personalization] command: As defined in section 4.1.2.1.  STORE DATA command:  As defined in section 4.1.3.4. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ8 | [[2]](#_bookmark7) | 4.1.3.1 | M | ES8: EstablishISDPKeySet  Description:  This function is used to perform mutual authentication between the SM-DP and the eUICC and to establish a shared secret key set between the SM-DP and the ISD-P.  Parameters:   * ISD-P AID * Ephemeral public key of the SM-DP * Certificate for the SM-DP   Command Description:  This function is realized through GlobalPlatform INSTALL [for personalization] and STORE DATA commands as defined in GlobalPlatform Card Specification [6].  INSTALL [for personalization] command: see Tables defined in section 4.1.3.1. Data Field Returned in the Response Message:  A single byte of '00' SHALL be returned indicating that no additional data is present as defined in the GlobalPlatform **Error! Reference source not found.**.  First STORE DATA command  The STORE DATA command message SHALL be coded according to Tables defined in section 4.1.3.1.  Data Field Returned in the Response Message:  The STORE DATA response SHALL contain the data described in Tables defined in section 4.1.3.1.  Second STORE DATA command  The STORE DATA command message SHALL be coded according to Tables defined in section 4.1.3.1.  Data Field Returned in the Response Message:  The STORE DATA response SHALL contain the data described in Tables defined section 4.1.3.1. | Profile Management |
| EUICC\_REQ30 | [[2]](#_bookmark7) | 4.1.3.2 | M | All ES8 functions in subsequent sections require securing the commands by SCP03.  *(Replaced by the EUICC\_REQ17)* | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ9 | [[2]](#_bookmark7) | 4.1.3.3 | M | ES8: DownloadAndInstallation  Description:  This function is used to load a Profile into an ISD-P on the eUICC. The ISD-P must be already created and also already personalized.  The Profile created by the SM-DP must be compatible with the targeted eUICC.  The Profile SHALL include in particular:   * the setting of POL1, if defined by MNO * the setting of Connectivity Parameters (see section 4.1.3.4) * the setting of ISD-P state from ‘CREATED’ to ‘DISABLED’ when installation is finished   Parameters:   * Profile | Profile Management |
| EUICC\_REQ57 | [[2]](#_bookmark7) | 4.1.3.3 | M | During the downloading process, the Profile SHALL be protected by SCP03t. Description of SCP03t:  This is a secure channel protocol based on GlobalPlatform's SCP03 usable for TLV structures.  The data transported in the command TLVs SHALL consist of the Profile Package specified in the SIMalliance eUICC Profile Package - Interoperable Format Technical Specification [53]; the response TLVs SHALL transport PE responses as provided by the Profile Package processing specified in [53]. The Profile Package consists of a sequence of Profile Element (PE) TLVs.  As the security mechanisms are exactly the same as SCP03, the SCP03 key sets are used for SCP03t. | eUICC Management |
| EUICC\_REQ58 | [[2]](#_bookmark7) | 4.1.3.3 | M | SCP03t does not take that PE structure into account, but treats the whole Profile Package as one block of transparent data. That block of data is split into segments of a maximum size of 1024 bytes (including the tag and length field). The eUICC SHALL support profile command data segments of at least up to this size. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ59 | [[2]](#_bookmark7) | 4.1.3.3 | M | SCP03t initiation uses a TLV equivalent to the INITIALIZE UPDATE APDU.  Secure Channel Initiation: INITIALIZE UPDATE command TLV:  The data used in the command and response TLVs are described in the section 4.1.3.3 and SHALL be encapsulated with the tag '84'.  In case of an error, tag '9F84' is used. The following values are defined:   * '01': error in length or structure of command data * '03': referenced data not found | eUICC Management |
| EUICC\_REQ60 | [[2]](#_bookmark7) | 4.1.3.3 | M | SCP03t initiation uses a TLV equivalent to the EXTERNAL AUTHENTICATE APDU.  Secure Channel Initiation: EXTERNAL AUTHENTICATE command TLV:  The data used in the command and response TLVs are described in the section 4.1.3.3 and SHALL be encapsulated with the tag '85'.  The security level SHALL be set to '33': "C DECRYPTION, R ENCRYPTION, C MAC, and R MAC".  If the message is accepted, a TLV with tag '85' and length zero SHALL be returned. In case of an error, tag '9F85' is used. The following values are defined:   * '01': error in length or structure of command data * '02': security error | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ61 | [[2]](#_bookmark7) | 4.1.3.3 | M | SCP03t Command TLV C-MAC and C-DECRYPTION Generation and Verification: For encapsulating encrypted profile command data in a SCP03t TLV, tag '86' is used.  SCP03t Response R-MAC and R-ENCRYPTION Generation and Verification:  For encapsulating encrypted profile response data in a SCP03t TLV, tag '86' is used. In case of an error, tag '9F86' is used. The following values are defined:   * '01': error in length or structure of command data * '02': security error | eUICC Management |
| EUICC\_REQ31 | [[2]](#_bookmark7) | 4.1.3.4 | M | ES8: UpdateConnectivityParameters SCP03  Description:  This function is used to update Connectivity Parameters on the eUICC.  This function has the following parameter:   * ISD-P AID * Connectivity Parameters   Function flow  Upon reception of the Connectivity Parameters update command, the eUICC shall:   * Update the Connectivity Parameters of the targeted ISD-P   Commands  STORE DATA Command  This command is a STORE DATA command, as described in GlobalPlatform Card Specification  [6] section 11.11.3.2 and in Tables described in section 4.1.3.4.  Data Field Returned in the Response Message:  The data field of the response message SHALL NOT be present. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ32 | [[2]](#_bookmark7) | 5.2.1 | M | ES1: RegisterEIS  Description:  This function allows an eUICC Manufacturer (EUM) to register an eUICC represented by its eUICC Information Set (EIS) within an identified SM-SR information database.  The EIS contains the complete set of data that is applicable for the SM-SR to manage the lifecycle of this eUICC.  This data set is split in two different parts:   * A fixed signed part containing the identification of the eUICC * A variable part containing the keys for the Platform Management plus the list of the different Profile loaded with the identified eUICC   This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the registration function has been successfully executed on the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.2.1. | eUICC Management |
| PM\_REQ10 | [[2]](#_bookmark7) | 5.3.1 | M | ES2: GetEIS  Description:  This function allows the MNO to retrieve up to date the EIS information.  The SM-DP SHALL forward the function request to the SM-SR “ES3.GetEIS” as defined in section 5.4.1.  Input/Output data described in Tables present in section 5.3.1. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ11 | [[2]](#_bookmark7) | 5.3.2 | M | ES2: DownloadProfile  Description:  This function allows the MNO to request that the SM-DP downloads a Profile, identified by its ICCID, via the SM-SR identified by the MNO on the target eUICC, the eUICC being identified by its EID.  Function flow  Upon reception of the function request, the SM-DP SHALL perform the following minimum set of verifications:   * The SM-DP SHALL verify it is responsible for downloading and installation of the Profile SM- DP MAY provide additional verifications   In case one of these conditions is not satisfied, the SM-DP SHALL refuse the function request and return a ‘Function execution status’ indicating ‘Failed’ with the relevant status code (see table below).  The SM-DP SHALL perform/execute the function according to the Profile Download and Installation procedure described in section 3.1.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the function has been successfully executed by the function provider as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.3.2. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ12 | [[2]](#_bookmark7) | 5.3.3 | M | ES2: UpdatePolicyRules  Description:  This function allows the MNO to update POL2 of a Profile, identified by its ICCID, and installed on an eUICC identified by its EID.  The SM-DP SHALL forward this function request to the identified SM-SR by calling the ES3.UpdatePolicyRules function as defined in section 5.4.6.  Input/Ouput data described in Tables present in section 5.3.3. | Profile Management |
| PM\_REQ13 | [[2]](#_bookmark7) | 5.3.4 | M | ES2: UpdateSubscriptionAddress  Description:  This function enables the caller to update the Subscription Address for a Profile in the eUICC Information Set (EIS) of a particular eUICC identified by the EID and ICCID.  The Subscription Address is the identifier, such as MSISDN and/or IMSI, through which the eUICC is accessible from the SM-SR via the mobile network when the Profile is in Enabled state. The function replaces the content of the Subscription Address.  The SM- DP SHALL forward the function request to the SM-SR “ES3.UpdateSubscriptionAddress” as defined in section 5.4.7.  Input/Output data described in Tables present in section 5.3.4. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ12 | [[2]](#_bookmark7) | 5.3.5 | M | ES2: EnableProfile  Description:  This function allows the MNO owner of the Profile to request a SM-DP to enable a Profile in a specified eUICC, eUICC being identified by its EID.  The SM-DP receiving this request SHALL process it according to the “Profile Enabling via SM- DP” procedure described in the section 3.3 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Enabled on the eUICC * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.3.5. | Platform Management |
| PF\_REQ13 | [[2]](#_bookmark7) | 5.3.6 | M | ES2: DisableProfile  Description:  This function allows the MNO to request a Profile Disabling to the SM-DP in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The target Profile is owned by the requesting MNO.  The SM-DP receiving this request SHALL process it according to Profile Disabling via SM-DP procedure described in section 3.5 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Disabled on the eUICC * A ‘Function execution status’ with ‘Executed-WithWarning’, with a status code as defined in section 5.4.9, indicating that the Profile has been disabled on the eUICC, and deleted after application of a POL1 or POL2 rule. * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.4.9   Input/Output data described in Tables present in section 5.3.6. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ14 | [[2]](#_bookmark7) | 5.3.7 | M | ES2: DeleteProfile  Description:  This function allows the MNO to request deletion of the target ISD-P with the Profile to the SM- DP; eUICC being identified by its EID. The SM-DP SHALL forward the function request to the SM-SR “ES3.DeleteISDP” as defined in section 5.4.10.  Input/Output data described in Tables present in section 5.3.7. | Platform Management |
| PF\_REQ15 | [[2]](#_bookmark7) | 5.3.8 | M | ES2: HandleProfileDisabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Disabled on the eUICC identified by its EID.  It is assumed that the ICCID is enough for the SM-DP to retrieve the MNO to notify.  This notification also conveys the date and time specifying when the operation has done.  Input data described in Tables present in section 5.3.8. | Platform Management |
| PF\_REQ16 | [[2]](#_bookmark7) | 5.3.9 | M | ES2: HandleProfileEnabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Enabled on the eUICC identified by its EID.  It is assumed that the ICCID is sufficient for the SM-DP to retrieve the MNO to notify.  This notification also conveys the date and time specifying when the operation has been done. Input data described in Table present in section 5.3.9. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ33 | [[2]](#_bookmark7) | 5.3.10 | M | ES2: HandleSMSRChangeNotification  Description:  This function SHALL be called for notifying each MNO owning a Profile hosted in the eUICC, identified by its EID that the SM-SR has changed. The notification is sent by the new SM-SR to the SM-DP, which route this notification to the MNO.  This notification also conveys the date and time specifying when the operation has been done. Input data described in Tables present in section 5.3.10. | eUICC Management |
| PF\_REQ17 | [[2]](#_bookmark7) | 5.3.11 | M | ES2: HandleProfileDeletedNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been deleted on the eUICC identified by its EID.  This notification also conveys the date and time specifying when the operation has been done. Input data described in Tables present in section 5.3.11. | Platform Management |
| PM\_REQ14 | [[2]](#_bookmark7) | 5.4.1 | M | ES3: GetEIS  Description:  This function allows retrieving the eUICC Information Set (EIS) of a particular eUICC from the SM-SR information database based on the EID.  The retrieved EIS contains only the data that is applicable for that particular SM-DP.  The SM-DP utilises the retrieved EIS, for instance, to verify the eligibility of the eUICC (e.g. type, certificate and memory).  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the download function has been successfully executed on the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.1. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ15 | [[2]](#_bookmark7) | 5.4.2 | M | ES3: AuditEIS  Description:  This function allows the SM-DP to retrieve up to date the EIS information.  The SM-SR SHALL use the relevant functions of the ES5 interface to retrieve the information from the eUICC.  At the end of the successful execution of this function, the SM-SR SHALL update its EIS database upon the basis of this information.  Input/Output data described in Tables present in section 5.4.2. | Profile Management |
| PM\_REQ16 | [[2]](#_bookmark7) | 5.4.3 | M | ES3: CreateISDP  Description:  This function allows the SM-DP to request the creation of an ISD-P to the SM- SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  Function flow  Upon reception of the function request, the SM-SR SHALL perform the following minimum set of verifications:   * The SM-SR is responsible for the management of the targeted eUICC * The Profile identified by its ICCID is not already present within its EIS database (meaning allocated to another ISD-P) * The requested amount of memory can be satisfied SM-SR MAY provide additional verifications   The SM-SR receiving this request SHALL process it according to the “Profile Download and Installation” procedure described in the section 3.1 of this specification.  When the SM-SR ends successfully this function it SHALL update the eUICC EIS by adding a new Profile entry in the EIS.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the ISD-P has been successfully created on the eUICC as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.3. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ17 | [[2]](#_bookmark7) | 5.4.4 | M | ES3: SendData  Description:  This function allows the SM-DP to send securely commands defined in ES8 interface (i.e.: Profile download or establish a key set) to an ISD-P or the ISD-R thru the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  Function flow  Upon reception of the function request, the SM-SR SHALL perform the following minimum set of verifications:   * The SM-SR is responsible for the management of the targeted eUICC * The targeted ISD-P is created on the eUICC. SM-SR MAY provide additional verifications   The data provided by the SM-DP SHALL be a list of C-APDU as defined in ETSI TS 102 226 [5] section 5.2.1.  The SM-SR has the responsibility to build the final Command script, depending on eUICC capabilities and selected protocol:   * by adding the Command scripting template for definite or indefinite length * and, if necessary, by segmenting the provided command script into several pieces and adding the relevant Script chaining TLVs   This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the function has been successfully executed by the function provider as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.4. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ18 | [[2]](#_bookmark7) | 5.4.5 | M | ES3: ProfileDownloadCompleted  Description:  This function allows the SM-DP to indicate to the SM-SR that the Profile download (identified by its ICCID) has been completed on the eUICC; eUICC being identified by its EID.  The Subscription Address is the identifier, such as MSISDN and/or IMSI, through which the eUICC is accessible from the SM-SR via the mobile network when the Profile is in Enabled state. On reception of this function request the SM-SR SHALL immediately update the EIS to set the identified Profile:   * (Conditional) the new Subscription Address. If the Profile is to be Enabled after it is loaded then the Subscription Address becomes mandatory. * (Optional) the provided POL2   At the end of this function call, the Profile state is “Disabled”. This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the function has been   correctly executed   * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.5. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ19 | [[2]](#_bookmark7) | 5.4.6 | M | ES3: UpdatePolicyRules  Description:  This function allows the SM-DP authorized by the MNO to update POL2 of a Profile, identified by its ICCID, and installed on an eUICC identified by its EID.  The function can update a Profile in “Disabled” or “Enabled” state and SHALL return an error for any other Profile state.  The function completely replaces the definition of existing POL2. This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the update Policy Rules function has been successfully executed by the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.6. | Profile Management |
| PM\_REQ20 | [[2]](#_bookmark7) | 5.4.7 | M | ES3: UpdateSubscriptionAddress  Description:  This function enables the caller to update the Subscription Address for a Profile in the eUICC Information Set (EIS) of a particular eUICC identified by the EID and ICCID.  The Subscription Address is the identifier, such as MSISDN and/or IMSI, through which the eUICC is accessible from the SM-SR via the mobile network when the Profile is in Enabled state. The function replaces the content of the Subscription Address.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the UpdateSubscriptionAddress function has been successfully executed by the SM-SR as requested by the function caller * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.7. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ18 | [[2]](#_bookmark7) | 5.4.8 | M | ES3: EnableProfile  Description:  This function allows the SM-DP to request a Profile Enabling to the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The target Profile is managed by the SM-DP authorized by the MNO owner of the Profile.  The SM-SR receiving this request SHALL process it according to “Profile Enabling via SM-DP” procedure described in the section 3.3 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Enabled on the eUICC * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.8. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ19 | [[2]](#_bookmark7) | 5.4.9 | M | ES3: DisableProfile  Description:  This function allows the SM-DP authorized by the MNO to request a Profile Disabling to the SM- SR in charge of the management of the targeted eUICC, eUICC being identified by its EID.  The target Profile SHALL be owned by the requesting MNO.  The SM-SR receiving this request SHALL process it according to Profile Disabling procedure described in section 3.5 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Disabled on the eUICC * A ‘Function execution status’ with ‘Executed-WithWarning’, with a status code as defined below, indicating that the Profile has been disabled on the eUICC, and deleted after application of a POL1 or POL2 rule. * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.9. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ20 | [[2]](#_bookmark7) | 5.4.10 | M | ES3: DeleteISDP  Description:  This function allows the SM-DP to request deletion of the target ISD-P with the Profile to the SM- SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The target Profile can only be a Profile that can be managed by the SM- DP authorized by the MNO.  On reception of the function request, the SM-SR SHALL perform the following minimum set of verifications:   * The SM-SR is responsible for the management of the targeted eUICC * The ISD-P identified by its AID exits on the targeted eUICC * The SM-DP is authorized to delete the target Profile by the MNO owning the target Profile * The POL2 of the target Profile allows the deletion * The target Profile is not the Profile having the Fall-back Attribute   The SM-SR receiving this request SHALL process it according to “Profile and ISD-P deletion via SM-DP” procedure described in section 3.7 of this specification.  In case the target Profile is “Enabled”, the SM-SR SHALL automatically disable it before executing the deletion.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been deleted on the eUICC * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ with ‘Executed-WithWarning’ indicating that the Profile has been deleted on the eUICC with a status code as defined in section **Error! Reference source not found.** (The ISD-P identified by its AID does not exist on the targeted eUICC)   Input/Output data described in Tables present in section 5.4.10. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ21 | [[2]](#_bookmark7) | 5.4.11 | M | ES3: UpdateConnectivityParameters  Description:  This function allows the MNO, or the SM-DP authorized by the MNO to update the Connectivity Parameters store in the ISD-P, identified by its ICCID, and installed on an eUICC identified by its EID.  The function can update a Profile in “Disabled” or “Enabled” state and SHALL return an error for any other Profile state.  The function updates the definition of existing Connectivity Parameters. This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the update of the Connectivity Parameters function has been successfully executed by the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.4.11. | Profile Management |
| PF\_REQ21 | [[2]](#_bookmark7) | 5.4.12 | M | ES3: HandleProfileDisabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Disabled on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to MNO notification endpoint.  This notification also conveys the date and time specifying when the operation has done.  In case of multiply handlers are served SM-SR SHOULD ensure completionTimestamp to be equal for every message.  Input data described in Tables present in section 5.4.12. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ22 | [[2]](#_bookmark7) | 5.4.13 | M | ES3: HandleProfileEnabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Enabled on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to MNO notification endpoint.  This notification also conveys the date and time specifying when the operation has been done. In case of multiply handlers are served SM-SR SHOULD ensure completionTimestamp to be equal for every message.  Input data described in Tables present in section 5.4.13. | Platform Management |
| EUICC\_REQ34 | [[2]](#_bookmark7) | 5.4.14 | M | ES3: HandleSMSRChangeNotification  Description:  This function SHALL be called for notifying each SM-DP authorized by the MNO owning a Profile hosted in the eUICC, identified by its EID that the SM-SR has changed.  The notification is sent by the new SM-SR to the SM-DP, which SHALL route this notification to the MNO.  This notification also conveys the date and time specifying when the operation has been done. Input data described in Tables present in section 5.4.14. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ23 | [[2]](#_bookmark7) | 5.4.15 | M | ES3: HandleProfileDeletedNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been deleted on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to SM-DP notification endpoint.  This notification also conveys the date and time specifying when the operation has been done. In case of multiply handlers are served, SM-SR SHOULD ensure ‘completionTimestamp’ to be equal for every message.  Input data described in Tables present in section 5.4.15. | Platform Management |
| PM\_REQ22 | [[2]](#_bookmark7) | 5.5.1 | M | ES4: GetEIS  Description:  This function allows retrieving the eUICC Information Set (EIS) of a particular eUICC from the SM-SR information database based on the EID.  The retrieved EIS contains only the data that is applicable for that particular MNO.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the download function has been successfully executed on the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.5.1. | Profile Management |
| PM\_REQ23 | [[2]](#_bookmark7) | 5.5.2 | M | ES4: UpdatePolicyRules  Description:  This function allows the MNO to update POL2 of a Profile, identified by its ICCID, and installed on an eUICC identified by its EID.  Input/Output data described in section 5.4.6. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PM\_REQ24 | [[2]](#_bookmark7) | 5.5.3 | M | ES4: UpdateSubscriptionAddress  Description:  This function enables the caller to update the Subscription Address for a Profile in the eUICC Information Set (EIS) of a particular eUICC identified by the EID and ICCID.  The function replaces the content of the Subscription Address.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the UpdateSubscriptionAddress function has been successfully executed by the SM-SR as requested by the function caller * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.5.13. | Profile Management |
| PM\_REQ25 | [[2]](#_bookmark7) | 5.5.4 | M | ES4: AuditEIS  Description:  This function allows the MNO to retrieve the up to date information for the MNO’s Profiles. The SM-SR SHALL only provide information for the Profiles owned by the requesting MNO. The SM-SR SHALL use the relevant functions of the ES5 interface to retrieve the information from the eUICC.  The SM-SR SHALL update its EIS database upon the basis of this information.  Input/Output data described in Tables present in section 5.5.4. | Profile Management |
| PM\_REQ26 | [[2]](#_bookmark7) | 5.5.4 | M | ES4: AuditEIS  If no list of ICCIDs is provided, it is implied that all the EIS data for the Profiles owned by the requesting MNO is required. | Profile Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ24 | [[2]](#_bookmark7) | 5.5.5 | M | ES4: EnableProfile  Description:  This function allows the MNO to request a Profile Enabling to the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The target Profile is managed by the MNO.  On reception of the function request, the SM-SR SHALL perform the following minimum set of verifications:   * The SM-SR is responsible for the management of the targeted eUICC * The Profile identified by its ICCID is loaded on the targeted eUICC * The target Profile is owned by the requesting MNO * The target Profile is in Disabled state * The POL2 of the target Profile and the POL2 of the currently Enabled Profile allow the enabling   The SM-SR receiving this request SHALL process it according to “Profile enabling” procedure described in the section 3.2 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Enabled on the eUICC * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code indicating a Unknown eUICC or with a status code indicating a Unknown ICCID with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.5.5. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ25 | [[2]](#_bookmark7) | 5.5.6 | M | ES4: DisableProfile  Description:  This function allows the MNO to request a Profile Disabling to the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The targeted is owned by the requesting MNO.  The SM-SR receiving this request SHALL process it according to “Profile disabling” procedure described in section 3.4 of this specification.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been Disabled on the eUICC * A ‘Function execution status’ with ‘Executed-WithWarning’, with a status code as defined below, indicating that the Profile has been disabled on the eUICC, and deleted after application of a POL1 or POL2 rule. * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.5.6. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ26 | [[2]](#_bookmark7) | 5.5.7 | M | ES4: DeleteProfile  Description:  This function allows the MNO to request deletion of the target ISD-P with the Profile to the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.  The target Profile can only be a Profile owned by the requesting MNO.  On reception of the function request, the SM-SR SHALL perform the following minimum set of verifications:   * The SM-SR is responsible for the management of the targeted eUICC * The ISD-P identified by its AID exists on the targeted eUICC * The POL2 of the target Profile allows the deletion * The target Profile is not the Profile having the Fall-back Attribute * The target Profile is owned by the requesting MNO and the function request is authorized by the MNO owning the target Profile   The SM-SR receiving this request SHALL process it according to “ISD-P Deletion” procedure described in the section 3.6 of this specification.  In case the target Profile is “Enabled”, the SM-SR SHALL automatically disable it before executing the deletion.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the Profile has been deleted on the eUICC * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ with ‘Executed-WithWarning’ indicating that the Profile has been deleted on the eUICC with a status code as defined in section **Error! Reference source not found.** (The ISD-P identified by its AID does not exist on the targeted eUICC)   Input/Output data described in Tables present in section 5.5.7. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ35 | [[2]](#_bookmark7) | 5.5.8 | M | ES4: PrepareSMSRChange  Description:  This function allows the Initiator to request to a new SM-SR to prepare for a change for an eUICC identified by its EID.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the PrepareSMSRChange function has been successfully executed on the SM-SR as requested by the function caller * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4 Input/Output data described in Tables present in section 5.5.8. | eUICC Management |
| EUICC\_REQ36 | [[2]](#_bookmark7) | 5.5.9 | M | ES4: SMSRChange  Description:  This function allows the initiator to request to the current SM-SR to change for a specific eUICC identified by its EID.  The SM-SR receiving this request SHALL process it according to the “SM-SR Change” procedure described in GSMA Remote Provisioning Architecture for Embedded UICC [1].  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the function has been successfully executed by the function provider as requested by the function caller * A ‘Function execution status’ indicating ‘Expired’ with the status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.5.9. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ27 | [[2]](#_bookmark7) | 5.5.10 | M | ES4: HandleProfileDisabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Disabled on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to MNO notification endpoint.  This notification also conveys the date and time specifying when the operation has done.  In case of multiply handlers are served SM-SR SHOULD ensure completionTimestamp to be equal for every message.  Input data described in Tables present in section 5.5.10. | Platform Management |
| PF\_REQ28 | [[2]](#_bookmark7) | 5.5.11 | M | ES4: HandleProfileEnabledNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been Enabled on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to MNO notification endpoint.  This notification also conveys the date and time specifying when the operation has been done. In case of multiply handlers are served SM-SR SHOULD ensure completionTimestamp to be equal for every message.  Input data described in Tables present in section 5.5.11. | Platform Management |
| EUICC\_REQ37 | [[2]](#_bookmark7) | 5.5.12 | M | ES4: HandleSMSRChangeNotification  Description:  This function SHALL be called for notifying each MNO owning a Profile hosted in the eUICC, identified by its EID, that the SM-SR has changed.  The notification is sent by the new SM-SR.  This notification also conveys the date and time specifying when the operation has been done.  Input data described in Tables present in section 5.5.12. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PF\_REQ29 | [[2]](#_bookmark7) | 5.5.13 | M | ES4: HandleProfileDeletedNotification  Description:  This function SHALL be called to notify that the Profile identified by its ICCID has been deleted on the eUICC identified by its EID.  ICCID MAY be not enough to identify right address of recipient; SM-SR SHOULD map it internally to MNO notification endpoint.  This notification also conveys the date and time specifying when the operation has been done. In case of multiply handlers are served SM-SR SHOULD ensure ‘completionTimestamp’ to be equal for every message.  Input data described in Tables present in section 5.5.13. | Platform Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ38 | [[2]](#_bookmark7) | 5.6.1 | M | ES7: CreateAdditionalKeySet  Description:  This function enables a new SM-SR to request for a new key set to be created in the ISD-R for the eUICC identified by the EID.  The new keyset belongs the new SM-SR and is unknown to the current SM-SR.  The current SM-SR SHALL map this function onto the second STORE DATA command in the ES5.establishISDRKeySet, see section 4.1.1.8, using the following rules:   * The order of TLVs SHALL follow the order denoted in table 45 * The following parameters of this command are not provided by the new SM-SR and it is the current SM-SR’s responsibility to set these parameters as defined below.   + Scenario identifier SHALL be set to ‘03’   + Key Usage Qualifier SHALL be set to '10' (3 secure channel keys)   + Key Access SHALL NOT be present, meaning a default value of ‘00’ (The key MAY be used by the Security Domain and any associated Application)   + Key Type SHALL be set to ‘88’ (AES)   + Key Length SHALL be set to ‘10’ (16 bytes)   + Key Identifier SHALL be set to ‘01’ * Length of Initial value of sequence counter SHALL be 0, meaning default value of sequence counter * The value of other parameters are provided by the new SM-SR.   This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the function has been successfully executed by the function provider as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.6.1. | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ39 | [[2]](#_bookmark7) | 5.6.2 | M | ES7: HandoverEUICC  Description:  This function enables to request for the handover management of an eUICC represented by its eUICC Information Set (EIS).  The EIS contains the complete set of data including information about Profiles, audit trail, which is applicable for the SM-SR to manage the lifecycle of this eUICC  The function provider SHALL execute the function accordingly to the procedure detailed in section 3.8. The handover is only committed at the end of the successfully procedure execution.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the register eUICC function has been successfully executed on the SM-SR as requested by the function caller. * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.6.2. | eUICC Management |
| EUICC\_REQ40 | [[2]](#_bookmark7) | 5.6.3 | M | ES7: AuthenticateSMSR  Description:  This function is used to authenticate the new SM-SR to the eUICC identified by the EID.  The function will return the random challenge generated by the eUICC to be used to create the signature for the second step in the SM-SR key establishment procedure.  This function MAY return:   * A ‘Function execution status’ with ‘Executed-success’ indicating that the AuthenticateSMSR function has been successfully executed by the SM-SR as requested by the function caller * A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 * A ‘Function execution status’ indicating ‘Failed’ with a status code as defined in section 5.1.6.4   Input/Output data described in Tables present in section 5.6.3. | eUICC Management |
| EUICC\_REQ41 | VOID | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| SEC\_REQ23 | [[1]](#_bookmark6) | 2.4 | M | The eUICC SHALL implement the Milenage network authentication algorithm. | Security |
| SEC\_REQ1 | [[1]](#_bookmark6) | 4.4.1 | M | Past or future communications associated with Profile download and installation, between the SM- DP and the eUICC, whenever trappable by third party SHALL NOT be recoverable based upon the compromise of a single long-term key used for message encryption.  *Note: Related to Secure Channel Protocols: this requirement is considered as superseded* | Security |
| SEC\_REQ6 | [[1]](#_bookmark6) | 4.4.2 | M | Communication between the SM-SR and the eUICC SHALL be protected against replay attacks. | Security |
| SEC\_REQ9 | [[1]](#_bookmark6) | 4.4.2 | M | When two security realms are exchanging data, they SHALL at first engage a security negotiation (e.g. EAP, IPSEC, TLS handshake…) resulting in the application of an agreed security level between them.  *Note: Related to TLS: initial states already defined, so this requirement is considered as superseded* | Security |
| SEC\_REQ11 | [[1]](#_bookmark6) | 4.4.2 | M | When negotiating a communication, at least the lowest acceptable common cryptographic suite SHALL apply.  *Note: Related to TLS: initial states already defined, so this requirement is considered as superseded* | Security |
| SEC\_REQ12 | [[1]](#_bookmark6) | 4.4.3 | M | Upon Profile deletion, the eUICC SHALL ensure of the complete wipe of the Profile. | Security |
| SEC\_REQ13 | [[1]](#_bookmark6) | 4.4.3 | M | eUICC SHALL only accept Platform and Profile Management commands sent from an authorized SM-SR or SM-DP.  *Note: In the context of this specification, an authorized SM-SR or SM-DP is a platform that knows the keys that allow communicating with the eUICC. As consequence, initial states and requirements are already defined, so this requirement is considered as superseded* | Security |
| SEC\_REQ14 | [[1]](#_bookmark6) | 4.4.3 | M | eUICC SHALL reject any Platform and Profile Management commands that are in conflict with the Policy Rules of any Profile on the eUICC the only exception being for the master delete command. | Security |
| SEC\_REQ15 | [[1]](#_bookmark6) | 4.4.3 | M | The eUICC SHALL provide a secure way for the SM-DP and SM-SR to check its identity and status in such a way that the entity has a proof of identity and origin. This capability is offered through the Eligibility Verification function. | Security |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| SEC\_REQ19 | [[1]](#_bookmark6) | 4.4.4 | M | The donor SM-SR SHALL NOT be able to access the eUICC once the SM-SR switch procedure has been completed. | Security |
| SEC\_REQ20 | [[1]](#_bookmark6) | 4.4.4 | M | The MNO SHALL be able to update the OTA Keys in its Profile on the eUICC in a secure and confidential way reusing existing OTA Platform mechanisms. | Security |
| SEC\_REQ22 | [[1]](#_bookmark6) | 4.4.6 | M | Policy Rule transport SHALL be treated as per SR2 (SR2=Communication between the SM-SR and the eUICC SHALL be protected against replay attacks).  *Note: Related to Secure Channel Protocols: this requirement is considered as superseded* | Security |
| Requirements related to the conditional requirement EUICC\_REQ14 - HTTPS supported on eUICC | | | | | |
| EUICC\_REQ42 | [[2]](#_bookmark7) | 2.4.3.1 | C | The SM-SR SHALL make use of a special SMS for triggering the opening of an HTTPS session to the eUICC. This SMS SHALL be addressed to the ISD-R. The necessary TAR information SHALL be included in the EIS.  The SMS SHALL comply with the format described in: GlobalPlatform Card Specification Amendment B [8], section “Administration session triggering parameters”. | eUICC Management |
| EUICC\_REQ43 | [[2]](#_bookmark7) | 2.4.4.1.1 | C | The eUICC SHALL support the Transport Layer Security (TLS) protocol v1.2 [15] with at least one of the following Pre-Shared Key Cipher suites as defined in RFC 5487 [17]:   * TLS\_PSK\_WITH\_AES\_128\_GCM\_SHA256 * TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256 | eUICC Management |
| EUICC\_REQ55 | [[2]](#_bookmark7) | 2.4.4.1.1 | C | The eUICC ISD-R SHALL be configured with ‘i’ = ‘04’ to indicate only TLS 1.2 supported as defined in GlobalPlatform Amd B [8]. | eUICC Management |
| EUICC\_REQ56 | [[2]](#_bookmark7) | 2.4.4.1.1 | C | In addition to restrictions to the TLS protocol specified in GP Amendment B [8], the ISD-R and SM-  SR SHALL NOT support TLS Session resumption (RFC 4507 or RFC 5077) nor several parallel TLS sessions. | eUICC Management |
| EUICC\_REQ44 | [[2]](#_bookmark7) | 2.4.4.1.1 | C | The eUICC SHALL support the Transport Layer Security (TLS) protocol v1.2 [15] with the following Pre-Shared Key Cipher suites as defined in RFC 5487 [17]: TLS\_PSK\_WITH\_AES\_128\_CBC\_SHA256  *Note: Replaced by EUICC\_REQ43* | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ45 | [[2]](#_bookmark7) | 2.4.4.1.2 | C | As specified in RFC 4279 [16], the PSK Identity SHALL be first converted to a character string, and then sent encoded in octets using UTF-8 [18] by the eUICC.  In the context of this specification, the PSK Identity before conversion is a sequence of Tag/Length/Value (TLV) objects in hexadecimal string representation. | eUICC Management |
| EUICC\_REQ46 | [[2]](#_bookmark7) | 2.4.4.2 | C | The ISD-R SHALL strictly follow GlobalPlatform Card Specification Amendment B [8] for the format of the POST request | eUICC Management |
| EUICC\_REQ47 | [[2]](#_bookmark7) | 2.4.4.2 | C | The content of the HTTP POST header field X-Admin-From SHALL be filled with the “Agent Id” information standardized in GlobalPlatform Card Specification Amendment B [8], section “Administration Session Triggering Parameters” (the format of this field is not standardized).  “Agent Id” information SHALL include two parts:   * the eUICC identifier (EID) * the identifier of the Security Domain representing the Admin Agent function | eUICC Management |
| EUICC\_REQ48 | [[2]](#_bookmark7) | 2.4.4.2 | C | The eUICC SHALL use the Chunked mode [Transfer-Encoding: chunked CRLF] for the POST request message. | eUICC Management |
| EUICC\_REQ49 | [[2]](#_bookmark7) | 2.4.4.2 | C | The SM-SR SHALL use Chunked mode [Transfer-Encoding: chunked CRLF] for the POST response. | eUICC Management |
| EUICC\_REQ50 | [[2]](#_bookmark7) | 2.4.4.3 | C | POST response sent by the SM-SR containing commands that SHALL be executed by the ISD-R:  HTTP/1.1 200 CRLF  X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF  Content-Type : application/vnd.globalplatform.card-content-mgt;version=1.0 CRLF X-Admin-Next-URI: <uri of the next POST> CRLF  CRLF  [Command script] | eUICC Management |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| EUICC\_REQ51 | [[2]](#_bookmark7) | 2.4.4.3 | C | POST response sent by the SM-SR containing commands that SHALL be executed by the ISD-P:  HTTP/1.1 200 CRLF  X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF  Content-Type : application/vnd.globalplatform.card-content-mgt;version=1.0 CRLF X-Admin-Next-URI: <uri of the next POST> CRLF  X-Admin-Targeted-Application://aid/<rid>/<pix> (of the ISD-P-AID) CRLF CRLF  [Command script] | eUICC Management |
| EUICC\_REQ51\_1 | [2] | 2.4.4.3 | M | Intermediate POST response sent by the SM-SR containing no command to execute but instructing to not close the HTTP session: the eUICC SHALL accordingly send a POST on the next URI provided, with no response body:  HTTP/1.1 204 CRLF  X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF X-Admin-Next-URI: <uri of the next POST> CRLF  CRLF | eUICC Management |
| EUICC\_REQ52 | [[2]](#_bookmark7) | 2.4.4.4 | C | The commands sent to the eUICC within a secure script in HTTP messages SHALL be formatted in an expanded remote command structure with indefinite length coding as defined in ETSI TS 102 226 [5]. As a consequence, the eUICC will provide the answer as an expanded remote response structure with indefinite length coding. | eUICC Management |
| Requirements related to the conditional requirement EUICC\_REQ18 - CAT\_TP supported on eUICC | | | | | |
| EUICC\_REQ53 | [[2]](#_bookmark7) | 2.4.3.2 | C | The SM-SR SHALL make use of a special SMS for triggering the opening of a CAT\_TP session to the eUICC. This SMS SHALL be addressed to the ISD-R. The necessary TAR information SHALL be included in the EIS. The SMS SHALL comply with the format described in: ETSI TS 102 226 [5], using the parameter “Request for BIP channel opening” and “Request for CAT\_TP link establish”. | eUICC Management |

###### Table 25: Requirements in scope

## Out of Scope Requirements

Here are all the requirements’ descriptions that are not covered by this Test Plan. Note that these requirements MAY be implemented in a future version of this Test Plan.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PROC\_REQ15 | [[2]](#_bookmark7) | 3.10 | M | The Master Delete Process must be compliant with the Figure 24 and with the procedure described in this section. | Procedure Flow |
| PROC\_REQ22 | [2] | 3.16 | M | The Fall-back Activation Procedure must be compliant with the Figure 31 and with the procedure described in this section. | Procedure Flow |
| PF\_REQ10 | [[2]](#_bookmark7) | 5.1.2.1 | M | By providing a validity period, the function caller indicates a specific amount of time to the function provider to process the function. As a consequence, during this validity period, the function caller SHALL NOT issue the same request again as it might generate duplicate execution steps within the function provider system. | Platform Management |
| PF\_REQ11 | [[2]](#_bookmark7) | 5.1.2.1 | M | After the end of the validity period, the function provider SHALL no longer continue with new execution steps. It is only mandated to tell the function caller that the function processing has expired. It is then the caller responsibility to either:   * Request the same function again * Or simply abandon the overall process into which the function was called | Platform Management |
| PF\_REQ30 | [2] | 2.4.1 | C | In HTTPS case, the SM-SR and eUICC MAY support DNS resolution to resolve the IP address of the SM-SR | Platform Management |
| SEC\_REQ2 | [[1]](#_bookmark6) | 4.4.1 | M | All cryptographic keys SHALL be kept in secure environment (e.g. HSM, eUICC). | Security |
| SEC\_REQ3 | [[1]](#_bookmark6) | 4.4.1 | M | The keys used by the EUM for eUICC Certificate generation SHALL be stored in a secure environment (i.e. in a Hardware Security Module). | Security |
| SEC\_REQ4 | [[1]](#_bookmark6) | 4.4.1 | M | The MNO SHALL be able to reject to use a non-trusted system for the Embedded UICC management. | Security |
| SEC\_REQ5 | [[1]](#_bookmark6) | 4.4.2 | M | Security realms SHALL be identifiable and mutually authenticated for the purpose of any communication. | Security |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| SEC\_REQ7 | [[1]](#_bookmark6) | 4.4.2 | M | Any end to end data communication between two security realms of the eUICC ecosystem SHALL be origin authenticated, integrity and confidentiality protected, protected against replay attacks and non- repudiable. Non-repudiation MAY NOT apply to communication with the eUICC. | Security |
| SEC\_REQ8 | [[1]](#_bookmark6) | 4.4.2 | M | Network communication links used inside a security realm SHALL be dedicated – i.e. neither public network, neither mutualised. E.g. solutions such as MPLS or GRE are not considered as dedicated links; a solution such as an authenticated and secured VPN is considered as dedicated. | Security |
| SEC\_REQ10 | [[1]](#_bookmark6) | 4.4.2 | M | Security realms SHALL enforce filtering rules, so, that only authorized entities are granted access to allowed services. | Security |
| SEC\_REQ16 | [[1]](#_bookmark6) | 4.4.4 | M | SM-SR SHALL implement an access control mechanism on the request for execution of the SMSR functions only to authorized security realms. | Security |
| SEC\_REQ17 | [[1]](#_bookmark6) | 4.4.4 | M | SM-DP SHALL implement an access control mechanism on the request for execution of the SMDP functions only to authorized security realms. | Security |
| SEC\_REQ18 | [[1]](#_bookmark6) | 4.4.4 | M | Security realm of SM-SR and SM-DP, and eUICC interfaces SHALL have proper counter measures against denial of services attacks. | Security |
| SEC\_REQ21 | [[1]](#_bookmark6) | 4.4.5 | M | The machine to machine Device SHALL NOT be able to access nor modify sensitive Profile data, i.e. credentials, management commands, Policy Rules, authentication algorithm parameters. | Security |
| PROC\_REQ23 | [2] | 2.4.4.5 | C | If supported and if correctly configured by SM-SR and eUICC, the ISD-R MAY request a DNS resolution to retrieve the IP Address of the SM-SR | Procedure Flow |
| PROC\_REQ24 | [2] | 2.4.5 | C | DNS resolution is an optional feature that is triggered only when:   * The eUICC includes a DNS resolver Client configured to initiate the DNS queries to server * The SM-SR relies upon a DNS Resolver Server able to provide the IP address associated to the domain name sent by the client query. * The eUICC determines that it has to resolve the IP address of the SM-SR server | Procedure Flow |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| **ID** | **Source** | **Chapter** | **Support** | **Description** | **Functional group** |
| PROC\_REQ25 | [2] | 2.4.5.2 | C | The DNS resolver of SM-SR and eUICC shall:   * Be compliant to RFC 1035 and RFC 3596 defining the Domain Name System and protocol * Support Query type A (IPv4) and AAAA (IPv6) * Use UDP protocol * Support only Recursive mode: the DNS resolver Server SHALL recursively resolve the given FQDN query, meaning that the answer SHALL contain all the available IP addresses * Send short responses: any response returned by DNS Server must fit in one UDP packet | Procedure Flow |
| PROC\_REQ26 | [2] | 2.4.5.3 | C | The DNS resolution process must be compliant with the Figure 10 and with the procedure described in this section. | Procedure Flow |
| EUICC\_REQ28 | [2] | 2.4.5.1 | C | If:   * the eUICC is requested to open an HTTPS session and * the eUICC supports DNS resolution and * the ISD-R has no IP address configured in the Connection Parameters of its Administration Session Triggering Parameters (as defined by Global Platform Amendment B [8]) and * the ISD-R has a FQDN, and IP addresses of DNS servers, configured in DNS parameters.   the ISD-R has not already resolved the FQDN to an IP address, or has resolved it but has reasons to consider the resolved value is stale | eUICC Management |
| EUICC\_REQ29 | [2] | 2.4.5.1 | C | The eUICC MAY also support other heuristics to determine that DNS resolution is needed and to which DNS servers to send the DNS queries. | eUICC Management |

###### Table 26: Out of Scope Requirem

# Document History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Brief description of change** | **Editor / Company** |
| 1.0 | 13 October 2014 | PSMC approved, first release | Sébastien Kuras, FIME |
| 2.0 | October 2015 | 15ESIMWI311\_01,  15ESIMWI311\_02r1,  15ESIMWI311\_03,  15ESIMWI311\_04,  15ESIMWI311\_05,  15ESIMWI311\_06,  15ESIMWI311\_07,  15ESIMWI311\_08,  15ESIMWI311\_09,  15ESIMWI311\_11,  15ESIMWI311\_12,  15ESIMWI311\_13,  15ESIMWI312\_03r1,  15ESIMWI312\_07r1,  15ESIMWI312\_08r1,  15ESIMWI312\_09r1,  15ESIMWI312\_11r1,  15ESIMWI312\_12r1,  15ESIMWI312\_15r1,  15ESIMWI312\_16r1,  15ESIMWI312\_17r1,  15ESIMWI312\_18r1,  15ESIMWI312\_19,  15ESIMWI312r1\_20,  15ESIMWI312\_21r1,  15ESIMWI313\_01,  15ESIMWI313\_02,  15ESIMWI313\_04,  15ESIMWI313\_05,  15ESIMWI313\_11,  15ESIMWI313\_12,  15ESIMWI313\_13,  15ESIMWI313\_14,  15ESIMWI313\_15,  15ESIMWI313\_16,  15ESIMWI313\_21,  15ESIMWI313\_22r3,  15ESIMWI314\_01,  15ESIMWI314\_02r1,  15ESIMWI314\_03,  15ESIMWI314\_04,  15ESIMWI314\_05, | Sébastien Kuras, FIME |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Brief description of change** | **Editor / Company** |
|  |  | 15ESIMWI315\_01r1,  15ESIMWI315\_02,  15ESIMWI315\_03,  15ESIMWI315\_04,  15ESIMWI315\_05r1,  15ESIMWI315\_06r1,  15ESIMWI316\_01,  15ESIMWI316\_02,  15ESIMWI317\_01,  15ESIMWI317\_02,  15ESIMWI317\_03,  15ESIMWI317\_04,  15ESIMWI317\_05,  15ESIMWI317\_06,  15ESIMWI317\_07,  15ESIMWI317\_08,  15ESIMWI317\_09,  15ESIMWI317\_10,  15ESIMWI317\_11,  15ESIMWI317\_12 |  |
| 3.0 | October 2015 | Third release | Sébastien Kuras, FIME |
| 3.1 | MAY 2016 | 15ESIMWI318\_01,  15ESIMWI318\_02,  15ESIMWI319\_01,  15ESIMWI319\_02,  15ESIMWI319\_03,  15ESIMWI319\_04,  15ESIMWI319\_05r1,  15ESIMWI319\_06,  15ESIMWI319\_07,  15ESIMWI319\_08,  15ESIMWI319\_09r1,  15ESIMWI319\_10,  15ESIMWI319\_11,  15ESIMWI319\_12r1,  15ESIMWI319\_13,  15ESIMWI320\_01r1,  15ESIMWI320\_02r4,  16ESIMWI320\_03r2,  16ESIMWI320\_04,  16ESIMWI320\_05,  16ESIMWI320\_06 | Sébastien Kuras, FIME |
| 3.2 | June 2017 | 16ESIMWI323\_Doc001,  16ESIMWI324\_Doc002,  16ESIMWI324\_Doc003,  16ESIMWI325\_Doc004\_r02, | Thomas Rhodes, Simulity |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Brief description of change** | **Editor / Company** |
|  |  | 16ESIMWI325\_Doc003,  17ESIMWI325\_Doc006\_r03,  17ESIMWI327\_Doc004r01,  17ESIMWI327\_Doc005,  17ESIMWI327\_Doc006r1,  17ESIMWI3281\_Doc\_002,  17ESIMWI3281\_Doc\_003r01,  17ESIMWI3281\_Doc\_004R3,  17ESIMWI3281\_Doc\_005R1,  17ESIMWI3281\_Doc\_006R1,  17ESIMWI3281\_Doc\_007R2,  17ESIMWI3281\_Doc\_008R1,  17ESIMWI3281\_Doc\_009R1,  17ESIMWI3281\_Doc\_010R3,  17ESIMWI3282\_Doc\_002r1,  17ESIMWI3282\_Doc\_003r1,  17ESIMWI3283\_Doc\_002R1,  17ESIMWI3284\_Doc\_002r1,  17ESIMWI3284\_Doc\_003r1,  17ESIMWI3284\_Doc\_004r1,  17ESIMWI3284\_Doc\_005r2,  17ESIMWI3284\_Doc\_007r1,  17ESIMWI3284\_Doc\_008r1,  17ESIMWI3284\_Doc\_009r1,  17ESIMWI3284\_Doc\_010r1,  17ESIMWI329\_Doc\_009r1,  17ESIMWI329\_Doc\_011r1,  17ESIMWI330\_Doc\_004r1,  17ESIMWI330\_Doc\_006r3,  17ESIMWI330\_Doc\_005r1,  17ESIMWI330\_Doc\_012r1,  17ESIMWI330\_Doc\_007r1,  17ESIMWI330\_Doc\_009r1,  17ESIMWI330\_Doc\_003r4,  17ESIMWI330\_Doc\_008r3,  17ESIMWI330\_Doc\_010r1,  17ESIMWI330\_Doc\_013r3,  17ESIMWI331\_Doc\_004R2,  17ESIMWI331\_Doc\_005r1,  17ESIMWI331\_Doc\_006r2,  17ESIMWI331\_Doc\_007r1,  17ESIMWI332\_Doc\_007r1,  17ESIMWI332\_Doc\_008r1,  17ESIMWI333\_Doc\_004r3 |  |

|  |  |  |  |
| --- | --- | --- | --- |
| **Version** | **Date** | **Brief description of change** | **Editor / Company** |
|  |  |  |  |
| 3.3 | August 2018 | 18ESIMWI345\_Doc\_005r1,  18ESIMWI345\_Doc\_006r1,  18ESIMWI345\_Doc\_007r1,  18ESIMWI346\_Doc\_003,  18ESIMWI346\_Doc\_004  18ESIMWI346\_Doc\_005,  18ESIMWI346\_Doc\_006,  18ESIMWI346\_Doc\_007,  18ESIMWI346\_Doc\_011,  18ESIMWI346\_Doc\_012,  18ESIMWI346\_Doc\_013,  18ESIMWI3471\_Doc\_003r1  18ESIMWI3471\_Doc\_004r1,  18ESIMWI348\_Doc\_003r1  18ESIMWI350\_Doc\_002 | Sébastien Kuras, FIME |
| 3.4 | 20 December 2022 | CR3001R02 Correction of ICCID and Required memory condition  CR34002R01 SendData\_InvalidDestination  CR34003R02 KeyEStablishment\_ICCID\_Clarification  CR34004R02 Test adaptation after clarifying validations in WSDL  CR3004R00 Test Spec Reference | Yolanda Sanz, GSMA |

## Document Owner

|  |  |
| --- | --- |
| **Type** | **Description** |
| Document Owner | eSIM Group |
| Editor / Company | Yolanda Sanz, GSMA |

It is our intention to provide a quality product for your use. If you find any errors or omissions, please contact us with your comments. You may notify us at [prd@gsma.com](mailto:prd@gsm.org).

Your comments or suggestions & questions are always welcome.