Remote Provisioning Architecture for Embedded UICC Technical Specification

Version 3.1

27 May 2016

This 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 © 2016 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.

Antitrust Notice

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

Table of Contents

1 Introduction 7
1.1 Overview 7
1.2 Scope 7
1.3 Document Purpose 7
1.4 Intended Audience 7
1.5 Definition of Terms 7
1.6 Abbreviations 10
1.7 References 12
2 General Parts of the Technical Specification 14
2.1 General Architecture 14
2.2 eUICC Architecture 16
2.2.1 Security Domains 16
2.2.2 Identification of eUICC: EID 20
2.2.3 Identification of Security Domains: AID and TAR 21
2.2.4 Profile Structure 22
2.2.5 Secure Channel on Interfaces 23
2.3 Security Overview 24
2.3.1 Certificate Issuer Role 25
2.3.2 Certification Chains 26
2.3.3 General Consideration on Algorithm and Key Length 27
2.4 OTA Communication on ES5 (SM-SR-eUICC) 27
2.4.1 General OTA Requirements 27
2.4.2 Void 28
2.4.3 SMS 28
2.4.4 HTTPS 29
2.5 Communication on ES8 (SM-DP) - eUICC 33
2.6 SM-DP to SM-SR Link Establishment (ES3) 34
2.7 OTA Platform Communication on ES6 (MNO-eUICC) 34
3 Detailed Procedure Specifications 34
3.1 Profile Download and Installation 35
3.1.1 ISD-P Creation 35
3.1.2 Key Establishment with Scenario#3-Mutual Authentication 37
3.1.3 Download and Installation of the Profile 40
3.1.4 Error Management Sub-Routine 43
3.2 Profile Enabling 43
3.2.1 Normal Case 44
3.2.2 Connectivity Failure Case 46
3.3 Profile Enabling Via SM-DP 48
3.3.1 Normal Case 48
3.3.2 Connectivity Failure Case 50
3.4 Profile Disabling 51
3.5 Profile Disabling Via SM-DP 53
3.6 Profile and ISD-P Deletion 55

3.7 Profile and ISD-P Deletion Via SM-DP 56
3.8 SM-SR Change 58
3.9 eUICC Registration at SM-SR: Register a New EIS 62
3.10 Master Delete Procedure 62
3.11 POL2 Update Via SM-DP 64
3.12 POL1Update by MNO 65
3.13 Connectivity Parameters Update by MNO 66
3.14 Connectivity Parameters Update Using SCP03 66
3.15 Default Notification Procedure 67
3.15.1 Notification Using SMS 68
3.15.2 Notification Using HTTPS 70
3.16 Fall-Back Activation Procedure 71
4 eUICC Interface Descriptions 74
4.1 Functions Description 75
4.1.1 ES5 (SM-SR–eUICC) Interface Description 75
4.1.2 ES6 (MNO-eUICC) Interface Description 104
4.1.3 ES8 (SM-DP-eUICC) Interface Description 108
5 Off-Card Interface Descriptions 123
5.1 Function Commonalities 125
5.1.1 Common Data Types 125
5.1.2 Request-Response Function 135
5.1.3 Notification Handler function 137
5.1.4 Functions Input Header 138
5.1.5 Functions Output Header 139
5.1.6 Status Code 140
5.2 ES1 (EUM – SM-SR) Interface Description 144
5.2.1 Register EIS 144
5.3 ES2 (MN0 – SM-DP) Interface Description 145
5.3.1 Getting eUICC Information 145
5.3.2 Download a Profile 146
5.3.3 Updating the Policy Rules of a Profile 148
5.3.4 Updating eUICC Information 149
5.3.5 Profile Enabling 150
5.3.6 Profile Disabling 152
5.3.7 Delete a Profile 153
5.3.8 Notify a Profile is Disabled 154
5.3.9 Notify a Profile Enabling 154
5.3.10 Notify a SM-SR Change 155
5.3.11 Notify a Profile Deletion 156
5.4 ES3 (SM-DP – SM-SR) Interface Description 157
5.4.1 Getting eUICC Information 157
5.4.2 Auditing eUICC Information 158
5.4.3 Create a New ISD-P in an eUICC 159
5.4.4 Download a New Profile 161

5.4.5 Indicating the Profile Download is Completed 163
5.4.6 Updating the Policy Rules of a Profile 165
5.4.7 Updating eUICC Information 166
5.4.8 Profile Enabling 167
5.4.9 Profile Disabling 168
5.4.10 Delete an ISD-P 170
5.4.11 Update Connectivity Parameters 172
5.4.12 Notify a Profile is Disabled 174
5.4.13 Notify a Profile Enabling 174
5.4.14 Notify an SM-SR Change 175
5.4.15 Notify a Profile Deletion 176
5.5 ES4 (MNO – SM-SR) Interface Description 177
5.5.1 Getting eUICC Information 177
5.5.2 Updating the Policy Rules of a Profile 178
5.5.3 Updating eUICC Information 178
5.5.4 Auditing eUICC Information 179
5.5.5 Profile Enabling 180
5.5.6 Profile Disabling 182
5.5.7 Delete a Profile 183
5.5.8 Prepare SM-SR Change 185
- **A ‘Function execution status’ with ‘Expired’ with a status code as defined in section 5.1.6.4 (Note: This Status is under validation by WI4 to be definitively included) 186**
5.5.9 SM-SR Change 186
5.5.10 Notify a Profile is Disabled 187
5.5.11 Notify a Profile Enabling 188
5.5.12 Notify a SM-SR Change 189
5.5.13 Notify a Profile Deletion 189
5.6 ES7 (SM-SR – SM-SR) Interface Description 190
5.6.1 Create Additional Key Set 190
5.6.2 Handover eUICC Information 192
5.6.3 Authenticate SM-SR 193
Annex A Mapping of Functions into Messages (Normative) 194
A.1 Namespaces and Schema References 194
A.2 Message: <rps:RPSMessage> 195
A.2.1 Message Header: <rps:RPSHeader> 196
A.2.2 Message Body: <rps:RPSBody> 198
A.3 Common Types 199
A.3.1 Request Base Type 199
A.3.2 Void 199
A.3.3 Response Base Type 199
A.3.4 Simple Types Mapping 203
A.3.5 Complex Type Mapping 205
A.4 The ES1 Interface Functions 220
A.4.1 The “ES1.RegisterEIS” Function 220

A.5 The ES2 Interface Functions 221
A.5.1 The “ES2.GetEIS” Function 221
A.5.2 The “ES2.DownloadProfile” Function 222
A.5.3 The “ES2.UpdatePolicyRules” Function 223
A.5.4 The “ES2.UpdateSubscriptionAddress” Function 224
A.5.5 The “ES2. EnableProfile” Function 225
A.5.6 The “ES2.DisableProfile” Function 226
A.5.7 The “ES2.DeleteProfile” Function 226
A.5.8 The “ES2.HandleProfileDisabledNotification” Function 227
A.5.9 The “ES2.HandleProfileEnabledNotification” Function 228
A.5.10 The “ES2.HandleSMSRChangedNotification” Function 229
A.5.11 The “ES2.HandleProfileDeletedNotification” Function 229
A.6 The ES3 Interface Functions 230
A.6.1 The “ES3.GetEIS” Function 230
A.6.2 The “ES3.AuditEIS” Function 231
A.6.3 The “ES3.CreateISDP” Function 231
A.6.4 The “ES3.SendData” Function 233
A.6.5 The “ES3.ProfileDownloadCompleted” Function 234
A.6.6 The “ES3.UpdatePolicyRules” Function 235
A.6.7 The “ES3.UpdateSubscriptionAddress” Function 235
A.6.8 The “ES3.EnableProfile” Function 236
A.6.9 The “ES3.DisableProfile” Function 237
A.6.10 The “ES3.DeleteISDP” Function 238
A.6.11 The “ES3. UpdateConnectivityParameters” Function 239
A.6.12 The “ES3.HandleProfileDisabledNotification” Function 240
A.6.13 The “ES3.HandleProfileEnabledNotification” Function 241
A.6.14 The “ES3.HandleSMSRChangeNotification ” Function 241
A.6.15 The “ES3. HandleProfileDeletedNotification” Function 242
A.7 The ES4 Interface Functions 242
A.7.1 The “ES4.GetEIS” Function 242
A.7.2 The “ES4.UpdatePolicyRules” Function 243
A.7.3 The “ES4. UpdateSubscriptionAddress” Function 244
A.7.4 The “ES4.AuditEIS” Function 245
A.7.5 The “ES4.EnableProfile” Function 246
A.7.6 The “ES4.DisableProfile” Function 247
A.7.7 The “ES4.DeleteProfile” Function 247
A.7.8 The “ES4.PrepareSMSRChange ” Function 248
A.7.9 The “ES4.SMSRChange” Function 249
A.7.10 The “ES4.HandleProfileDisabledNotification” Function 250
A.7.11 The “ES4.HandleProfileEnabledNotification” Function 251
A.7.12 The “ES4.HandleSMSRChangedNotification” Function 252
A.7.13 The “ES4. HandleProfileDeletedNotification” Function 252
A.8 The ES7 Interface Functions 253
A.8.1 The “ES7.CreateAdditionalKeySet” Function 253
A.8.2 The “ES7.HandoverEUICC” Function 254

A.8.3 The “ES7.AuthenticateSMSR” Function 255
Annex B Binding to SOA Environment (Normative) 256
B.1 General Recommendations 256
B.2 SOAP Binding 256
B.2.1 Message Binding 257
B.2.2 Security 264
B.2.3 Message Exchange Pattern (MEPs) – HTTPS Binding 265
B.2.4 Binding Examples 269
B.3 Function Binding 272
B.3.1 ES1 272
B.3.2 ES2 272
B.3.3 ES3 274
B.3.4 ES4 276
B.3.5 ES7 278
B.4 Web Service Definition Language (WSDL) 278
Annex C Use of GlobalPlatform Privileges 279
Annex D Data Definitions 285
Annex E EIS Usage in Functions 285
Annex F Key Check Values 287
Annex G Device Requirements 288
Annex H Coding of the PIX for ‘Embedded UICC Remote Provisioning and Management’ (Normative) 289
Annex I List of Identifiers (Normative) 290
Annex J Verification of EID 291
Annex K Document Management 292
K.1 Document History 292
K.2 Other Information 294

1 Introduction

1.1 Overview

This document provides a technical description of the GSMA's 'Remote Provisioning Architecture for Embedded UICC' [1].

1.2 Scope

This specification provides a technical description of:

- The eUICC Architecture
- The interfaces used within the Remote Provisioning Architecture and
- The security functions used within the Remote Provisioning Architecture

1.3 Document Purpose

The aim of this document is to define a technical solution for the remote provisioning and management of the Embedded UICC (eUICC) in machine-to-machine Devices which are not easily reachable. The adoption of this technical solution will provide the basis for ensuring global interoperability between potentially different MNO deployment scenarios, different makes of network equipment (for example SM-DP, SM-SR) and different makes of eUICC platforms.

1.4 Intended Audience

Technical experts working within MNOs, SIM solution providers, machine to machine Device vendors, standards organisations, network infrastructure vendors, Service Providers and other industry bodies.

1.5 Definition of Terms

Term	Description
Actor	Physical entity (person, company or organisation) that can assume a Role in the functional architecture. It is possible for an Actor to assume multiple Roles in the same functional architecture.
Associated (with) / Association	This term refers to a link of an application, an Executable Load File or a security domain to (another) security domain, which provides services to the former as defined in GlobalPlatform Card Specification [6] section 7.2.
Connectivity Parameters	A set of data (for example SMSC address) required by the eUICC to open a communication channel (for example SMS, HTTPS) on a dedicated network.
Customer	A paying party, in particular a legally responsible juridical person or entity.

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 (for example NAA) present in the Profile are not selectable over the eUICC-Terminal interface.
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 (for example, NAA) are selectable over the UICC-Terminal interface.
Executable Load File	An on-card container of one or more application's executable code as defined in GlobalPlatform Card Specification [6].
Executable Module	The on-card executable code of a single application present within an Executable Load File as defined in GlobalPlatform Card Specification [6].
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 (for example firmware and operating system).
EUM Certificate	A certificate issued to a GSMA accredited EUM which can be used to verify eUICC Certificates. This certificate can be verified using the Root Certificate.
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 [6].
Mobile Network Operator	An entity providing access capability and communication services to its Customers through a mobile network infrastructure.
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.
Network Access Application	An application residing on a UICC which provides authorisation to access a network for example a USIM application.
Orphaned Profile	A Profile whose Policy Rules have become unmanageable, for example due to the termination of the Customer's contract with the MNO.
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 on 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.
Profile Component	A Profile Component is an element of the Profile and may be one of the following: An element of the file system like an MF, EF or DF An Application, including NAA and Security Domain POL1 MNO-SD.
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.
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 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.
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 authorised 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, for example SMS, to the eUICC.
Subscription Manager Data Preparation	Role that prepares the Profiles and manages the secure download and installation of these Profiles onto the eUICC.
Subscription Manager Secure Routing	Role that securely performs functions of Platform Management commands and the transport of Profile Management commands.
Telecommunication Service Provider	The organization through which the Subscriber obtains PLMN telecommunication services. This is usually the network operator or possibly a separate body.

1.6 Abbreviations

Abbreviation	Description
AID	Application Identifier
AES	Advanced Encryption Standard
CASD	Controlling Authority Security Domain
CBC	Cipher Block Chaining
CC	Cryptographic Checksum
CERT.DP.ECDSA	Certificate of the SM-DP for its ECDSA key
CERT.SR.ECDSA	Certificate of the SM-SR for its ECDSA key
CERT.ECASD.ECKA	Certificate of the ECASD for its ECKA key
CI	Certificate Issuer
CMAC	Cipher-based Message Authentication Code
ECASD	eUICC Controlling Authority Security Domain
ECDSA	Elliptic Curve cryptography Digital Signature Algorithm
ECKA	Elliptic Curve cryptography Key Agreement algorithm
DAP	Data Authentication Pattern
DR	Derivation Random
EUM	eUICC Manufacturer
EID	eUICC-ID
EIS	eUICC Information Set
ETSI	European Telecommunications Standards Institute
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
eUICC	Embedded UICC
GP	GlobalPlatform
GSMA	GSM Association

ICCID	Integrated Circuit Card ID
IMS	IP Multimedia Subsystem
IMSI	International Mobile Subscriber Identity
ISD	Issuer Security Domain
ISD-P	Issuer Security Domain Profile
ISD-R	Issuer Security Domain Root
ISO	International Standards Organisation
ITU	International Telecommunications Union
LTE	Long Term Evolution
MEP	Message Exchange Pattern
MNO	Mobile Network Operator
MO	Mobile Originated
MOC	Mandatory, Optional or Conditional
MT	Mobile Terminated
NAA	Network Access Application
OTA	Over The Air
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
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
SCP	Secure Channel Protocol
SD	Security Domain
ShS	Shared Secret
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
SK.CI.ECDSA	Private key of the CI for signing certificates
SM	Subscription Manager
SM-DP	Subscription Manager Data Preparation
SM-SR	Subscription Manager Secure Routing
SOA	Service-oriented Architecture
SOAP	Simple Object Access Protocol

TAR	Toolkit Application Reference
TLS	Transport Layer Security
URI	Uniform Resource Identifier
URL	Uniform Resource locator
USIM	Universal Subscriber Identity Module
XML	Extensible Markup Language
W3C	World Wide Web Consortium

1.7 References

Ref	Document Number	Title
[1]	SGP.01	GSMA 'Remote Provisioning Architecture for Embedded UICC' Version 1.1
[2]	ETSI TS 101 220	Smart Cards; ETSI numbering system for telecommunication application providers; Release 9
[3]	ETSI TS 102 223	Smart Cards; Card Application Toolkit (CAT) ; Release 9
[4]	ETSI TS 102 225	Secured packet structure for UICC based applications; Release 12
[5]	ETSI TS 102 226	Remote APDU structure for UICC based applications; Release 9
[6]	GPC_SPE_034	GlobalPlatform Card Specification v.2.2.1
[7]	GPC_GUI_010	GlobalPlatform Card Specification v.2.2.1 UICC Configuration v1.0.1
[8]	GPC_SPE_011	GlobalPlatform Card Specification v.2.2 Amendment B: Remote Application Management over HTTP v1.1.3
[9]	GPC_SPE_025	GlobalPlatform Card Specification v.2.2 Amendment C: Contactless Services v1.1.1
[10]	GPC_SPE_014	GlobalPlatform Card Specification v.2.2 Amendment D: Secure Channel Protocol 03 v1.1.1
[11]	GPC_SPE_042	GlobalPlatform Card Specification v.2.2 Amendment E: Security Upgrade for Card Content Management v1.0.1
[12]	ITU E.212	The international identification plan for public networks and Subscriptions
[13]	3GPP TS 31.115	Secured packet structure for (Universal) Subscriber Identity Module (U)SIM Toolkit applications Release 11
[14]	OMA Smartcard-Web-Server v1.0	OMA-TS-Smartcard_Web_Server-V1_0-20080421-A
[15]	RFC 5246	The TLS Protocol – Version 1.2
[16]	RFC 4279	Pre-Shared Key Cipher suites for Transport Layer Security (TLS)
[17]	RFC 5487	Pre-Shared Key Cipher Suites for TLS with SHA-256/384 and AES Galois Counter Mode
[18]	RFC 3629	Unicode Transformation Format 8-bit
[19]	ISO/IEC 7812	Identification Cards; Identification of issuers

[20]	ETSI TS 124 008	Mobile radio interface Layer 3 specification; Core network protocols; Release 9
[21]	ITU E.118	The international telecommunication charge card
[22]	ITU E.164	International Public Telecommunication Numbering Plan
[23]	GPS_SPE_002	”GlobalPlatform System”: Messaging Specification for Management of Mobile NFC Services, Version 1.1.2
[24]	RFC 4051	Additional XML Security Uniform Resource Identifiers (URIs)
[25]	ETSI TS 102 127	Transport protocol for CAT applications; Release 6
[26]	XML DSIG-CORE	W3C XML Signature Syntax and Processing (Second Edition)
[27]	3GPP TS 31.111	Universal Subscriber Identity Module (USIM) Application Toolkit (USAT) ; Release 9
[28]	3GPP TS 31.116	Remote APDU Structure for (U)SIM Toolkit applications; Release 9
[29]	3GPP TS 24.341	Support of SMS over IP networks; Release 9
[30]	GSMA Security Principles Related to Handset Theft	GSMA Doc Reference: Security Principles Related to Handset Theft 3.0.0 EICTA CCIG Doc Reference: EICTA Doc: 04cc100
[31]	ETSI TS 123 003	Universal Mobile Telecommunications System (UMTS); Numbering, addressing and identification; Release 9
[32]	RFC 768	User Datagram Protocol, Aug 1980.
[33]	RFC 793	Transmission Control Protocol, DARPA Internet Program, Protocol specification, Sept 1981.
[34]	GPC_GUI_049	GlobalPlatform Card, Secure Element Configuration Version 1.0, October 2012
[35]	3GPP TS 27.007	Technical Specification Group Core Network and Terminals; AT command set for User Equipment (UE) ; Release 9
[36]	NIST SP 800-57 Part 1	NIST Special Publication 800-57: Recommendation for Key Management – Part 1: General (Revision 3), July 2012
[37]	BSI TR-02102	BSI – Technische Richtlinie: Kryptographische Verfahren: Empfehlungen und Schlüssellängen, Version: 2013.02
[38]	GSMA PRD IR.92	GSMA PRD IR.92: IMS Profile for Voice and SMS
[39]	3GPP TS 23.040	Technical Specification Group Core Network and Terminals; Technical realisation of the Short Message Service (SMS)
[40]	SOAP	SOAP Version 1.2 Part 1: Messaging Framework (Second Edition) http://www.w3.org/TR/soap12-part1/
[41]	WS-Addressing	Web Services Addressing 1.0, Core , 9th of May 2006 http://www.w3.org/TR/ws-addr-core/
[42]	WS-Addressing-SOAP-Binding	Web Services Addressing 1.0 - SOAP Binding, 9th of May 2006 http://www.w3.org/TR/2006/REC-ws-addr-soap-20060509/
[43]	WS-MakeConnection	Web Services Make Connection (WS-MakeConnection), 2nd February 2009, http://docs.oasis-open.org/ws-rx/wsmc/200702/wsmc-1.1-spec-os.html

[44]	WSS-SOAPMessageSecurity	Web Services Security: SOAP Message Security 1.1 (WS-Security 2004), 1 February 2006, http://docs.oasis-open.org/wss/v1.1/
[45]	WSS-UserNameTokenProfile	Web Services Security UsernameToken Profile 1.1, February 2006, http://docs.oasis-open.org/wss/v1.1/
[46]	WSS-X509TokenProfile	Web Services Security X.509 Certificate Token Profile 1.1, February 2006, http://docs.oasis-open.org/wss/v1.1/
[47]	XML	Extensible Markup Language (XML) 1.0, W3C Recommendation 10-Feb-98, REC-xml-19980210
[48]	XML Signature	XML Signature Syntax and Processing (Second Edition),W3C Recommendation http://www.w3.org/TR/2008/REC-xmldsig-core-20080610/
[49]	BSI TR-03111	BSI Technical Guideline TR-03111: Elliptic Curve Cryptography, Version 2.0
[50]	NIST SP 800-56A	NIST Special Publication SP 800-56A: Recommendation for Pair-Wise Key Establishment Schemes Using Discrete Logarithm Cryptography (Revision 2), May 2013
[51]	ANSSI ECC FRP256V1	Avis relatif aux paramètres de courbes elliptiques définis par l'Etat français. JORF n°0241 du 16 octobre 2011 page 17533. texte n° 30. 2011
[52]	3GPP TS 31.102	Characteristics of the Universal Subscriber Identity Module (USIM) application, Release 9
[53]	SIMAPP	SIMalliance: eUICC Profile Package - Interoperable Format Technical Specification V2.0 http://simalliance.org/euicc/euicc-technical-releases/
[54]	GPC_SPE_007	GlobalPlatform Card Specification v.2.2 Amendment A: Confidential Card Content Management v1.0.1
[55]	NIST SP 800-56C	NIST Special Publication 800-56C Recommendation for Key Derivation through Extraction-then-Expansion

2 General Parts of the Technical Specification

This section contains a technical description and architecture of the Remote Provisioning System for the Embedded UICC. It shall be compliant with the Remote Provisioning Architecture for Embedded UICC [1]. In addition, the statements in this section define the basic characteristics that need to be taken into account when realising this specification.

2.1 General Architecture

This section further specifies the Roles and interfaces associated with the Remote Provisioning and Management of the eUICC, building on GSMA Remote Provisioning Architecture for Embedded UICC [1].

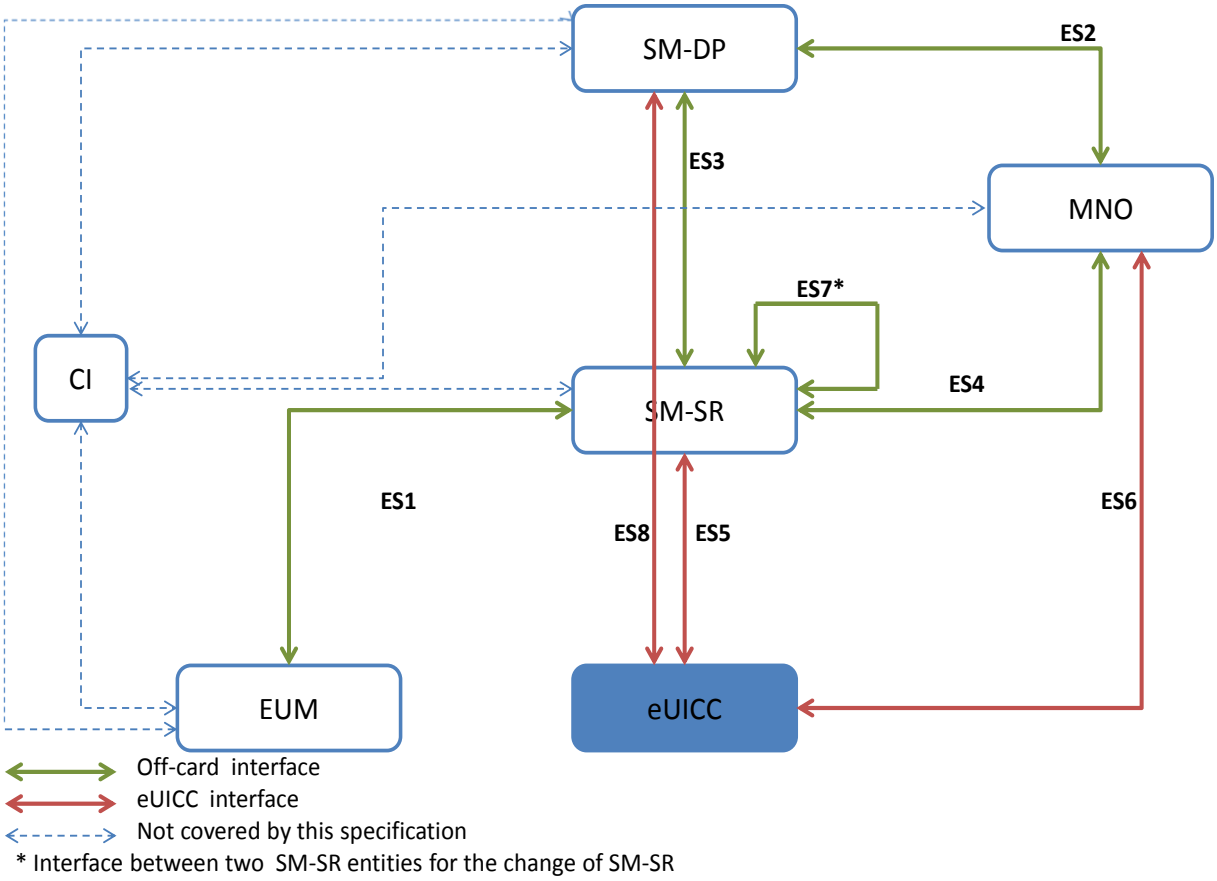


Figure 1: eUICC Remote Provisioning System

The above figure provides the complete description of the eUICC Remote Provisioning and Management system.

The ES5, ES6 and ES8 interfaces are described in section 4.

The ES1, ES2, ES3, ES4 and ES7 interfaces are described in section 5.

NOTE: Functions of the ES2 interface related to Profile ordering and master delete are considered out of the scope of this specification as these functions may be based upon pre-existing MNO processes.

NOTE: The interface between the SM-DP and EUM and the related function for Profile Creation is out of the scope of this specification as this function is based upon proprietary mechanisms.

NOTE: The ES6 interface is based on the RAM and RFM mechanisms described in ETSI TS 102 225 [4] and ETSI TS 102 226 [5].

NOTE: As defined in GSMA Remote Provisioning Architecture for Embedded UICC [1], the Initiator Role is assumed to be played by the MNO and functions related to this Role are specified in the ES4 interface.

2.2 eUICC Architecture

This section focuses on the eUICC architecture which widely leverages current telecommunication standards, as well as GlobalPlatform standards that are especially well adapted to establish Role separation and data isolation. In particular, each entity will have a dedicated Security Domain with different privileges and configuration.

2.2.1 Security Domains

The eUICC architecture comprises the following Security Domains for the purpose of Platform and Profile Management:

- The ISD-R is the representative of the off-card entity SM-SR
- The ECASD is the representative of the off-card entity CI
- An ISD-P is the representative of an off-card entity SM-DP. An eUICC can contain more than one ISD-P

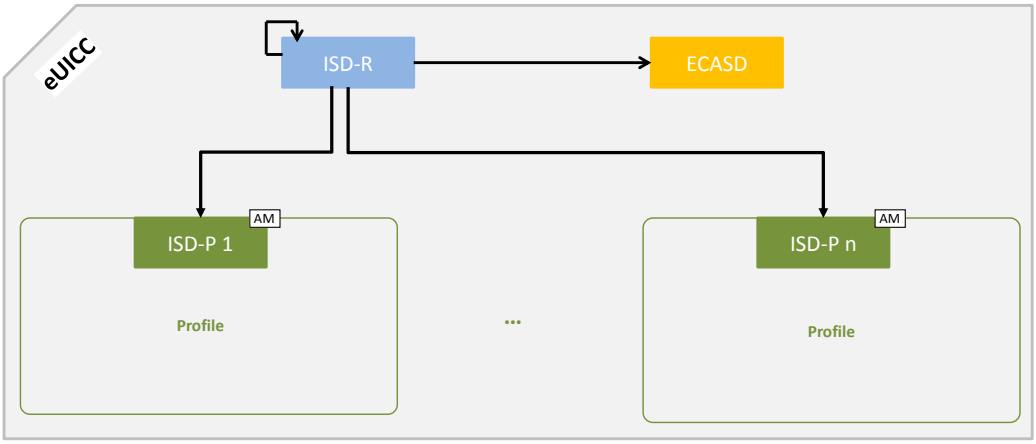


Figure 2: Security Domain Architecture Overview

An ISD, as specified in GlobalPlatform Card Specification [6], does not exist in the architecture of the eUICC.

2.2.1.1 ISD-R

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 LOCKED state shall not be supported by the ISD-R.

The ISD-R privileges shall be granted according to Annex C.

The ISD-R shall only be able to perform Platform Management functions on ISD-Ps.

2.2.1.2 ECASD

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 is involved in the following functions:

- SM-DP key set establishment for Profile Download and Installation
- SM-SR key set establishment for SM-SR Change

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
- EUM key set for key renewal
- EID

The ECASD shall comply with the requirements defined for the CASD in GlobalPlatform Card Specification UICC configuration [7] except:

- AIDs and TAR shall be allocated as defined in section 2.2.3
- Support of SCP 02 is not required
- Only the ISD-R and ISD-Ps shall be able to use the ECASD services

2.2.1.3 ISD-P

An ISD-P hosts a unique Profile.

Only one ISD-P shall be enabled on an eUICC at any point in time.

An ISD-P shall be installed by the ISD-R and then personalized by its related SM-DP (see section 3.1.1). 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.

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.

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.

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.

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.

An ISD-P shall remain associated to the ISD-R during all its life time in order for the ISD-R to be able to perform the Platform Management functions:

- ISD-P Creation: the Association between the ISD-R and an ISD-P shall be created at that time
- ISD-P Deletion and Master Delete
- Profile Enabling and Disabling
- Fall-back Attribute setting
- Transport function: The Association shall allow SCP03/SCP03t establishment between the SM-DP and the ISD-P.

ISD-P shall follow the life-cycle illustrated in the Figure 3, based on the Security Domain life-cycle defined in GlobalPlatform Card Specification [6], section 5.3.

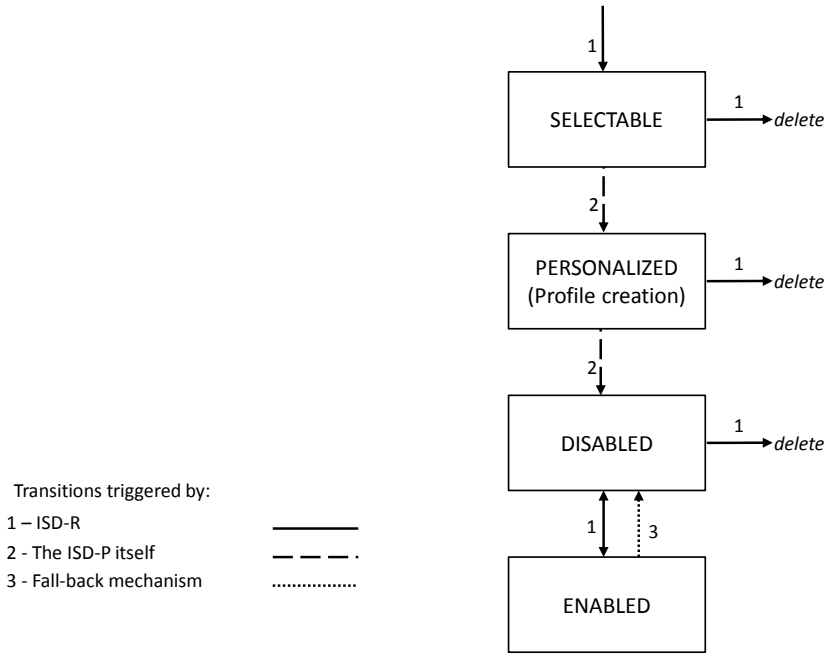


Figure 3: ISD-P Life-Cycle State Transitions

After execution of the procedure described in section 3.1.1, the ISD-P shall be in SELECTABLE state. After execution of the procedure described in section 3.1.2, the ISD-P shall be in PERSONALIZED state.

NOTE: The INSTALLED state for security domains defined in GlobalPlatform Card Specification [6] is skipped by the command for ISD-P creation defined in section 4.1.1.1.

After execution of the procedure described in section 3.1.3 or 3.4, 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.

After execution of the procedure described in section 3.2, 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.

Deletion removes the ISD-P with its Profile from the eUICC.

The LOCKED state shall not be supported by an ISD-P.

For coding the states, table 11-5 of GlobalPlatform Card Specification [6] is modified as follows:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0	0	0	1	1	(INSTALLED)
0	0	0	0	0	1	1	1	SELECTABLE
0	0	0	0	1	1	1	1	PERSONALIZED (Profile creation)
0	0	0	1	1	1	1	1	DISABLED
0	0	1	1	1	1	1	1	ENABLED

Table 1: ISD-P Coding States

These states can be mapped to the architectural states defined in GSMA Remote Provisioning Architecture [1] as shown below:

State (as defined in [1])	State (as defined above)
Created	(INSTALLED)
	SELECTABLE
	PERSONALIZED
Disabled	DISABLED
Enabled	ENABLED
Deleted	No explicit mapping; ISD-P no longer exists on the eUICC

Table 2: ISD-P State Mapping

ISD-P privileges shall be granted according to Annex C.

All Profile Components, in particular the MNO-SD, shall remain linked to the ISD-P in order to enable the following:

- Profile Download and Installation: the Profile Components, which are affiliated with the ISD-P, are created at that time
- ISD-P Deletion and Master Delete: the Profile Components shall be deleted at that time
- Profile Enabling and Disabling: Enable and Disable access to all the Profile Components
- Update of POL1
- Provide read access to POL1 when required for Platform Management functions.

The Application privileges (defined in GlobalPlatform Card Specification [6]) assigned to a Profile Component shall apply according to Annex C.

All Profile Components created by the ISD-P shall always remain affiliated with the ISD-P. In particular it is not possible to change the affiliation of any Profile Component.

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;
- The file system within the Profile cannot be selected by the Device or any application on the eUICC;
- The applications (including NAAs and Security Domains) within the Profile cannot be selected, triggered or deleted.

2.2.2 Identification of eUICC: EID

The EID is the eUICC identifier used in the context of Remote Provisioning and Management of the eUICC.

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 retrieve the EID

The EID shall have the following structure:

- The EID shall always be 32 digits long
- The EID shall always be built of
    - A Major Industry Identifier digit of 8 (1st digit), as defined in ISO/IEC 7812 [19].
    - An additional digit of 9 specifying telecommunications, as defined in ISO/IEC 7812 [19],
    - An additional three digits for country code (3rd to 5th digits).
        - If the country code is one digit long, its value shall be prefixed by two digits of 0,
        - If the country code is two digits long, its value shall be prefixed by one digit of 0.
    - An additional three digits for issuer identifier (6th to 8th digits)
        - If the issuer identifier is one digit long, its value shall be prefixed by two digits of 0,
        - If the issuer identifier is two digits long, its value shall be prefixed by one digit of 0.
    - An additional ten digits for issuer specific information (9th to 18th digits), of which the first five digits (9th to 13th) contain version information about the platform and OS, to be specified by the issuer and the last five digits (14th to 18th) contain additional issuer information,

    - An additional twelve digits for the individual identification number (19th to 30th digits),
    - A last two digits (31st to 32nd digits) containing check digits calculated over all 32 digits as specified below.
- The country code and issuer identifier shall be assigned as specified in ITU E.118 [21]
- The two check digits are calculated as follows:
    - 1. Replace the two check digits by two digits of 0,
    - 2. Using the resulting 32 digits as a decimal integer, compute the remainder of that number on division by 97,
    - 3. Subtract the remainder from 98, and use the decimal result for the two check digits,
        - If the result is one digit long, its value shall be prefixed by one digit of 0.
- When stored as a byte string, the first digit shall be put into the highest four bits of the first byte

Annex J provides a description of how the verification of an EID is performed.

2.2.3 Identification of Security Domains: AID and TAR

The ISD-P AID, the ISD-R AID and the ECASD AID shall follow the structure specified in ETSI TS 101 220 [2], with a RID and a PIX. The ISD-P AID, the ISD-R AID and the ECASD AID shall be 16 bytes long including the TAR.

The RID of the Executable Load File, the Executable Module and the Application of the ISD-R, the ISD-P and the ECASD shall be set to 'A000000559' (as defined in ISO/IEC 7816-5:2004).

The ISD-R application shall be installed by the EUM during eUICC manufacturing. 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 application shall be installed by the EUM during eUICC manufacturing. The ECASD Executable Load File AID and the ECASD Executable Module AID can be freely selected by the EUM.

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.

The ISD-P application shall be installed by SM-SR during the first step of the “Profile Download and Installation” procedure in section 3.1.

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.

NOTE: The choice of having the ISD-P AID allocated by the SM-SR is to avoid conflicts with other ISD-P AIDs used by already installed ISD-Ps; the SM-DP cannot have such visibility.

The MNO-SD application AID and TAR(s) can be freely allocated by the MNO during Profile definition.

2.2.4 Profile Structure

The Profile structure, composed of a set of Profile Components, is specified by, and under the full control of, the MNO. The full Profile structure shall be contained in a unique ISD-P.

The Profile structure shall contain a Profile Component, called MNO-SD, which performs an identical Role as the ISD for a UICC (see GlobalPlatform Card Specification [6]). This MNO-SD is the representative of the MNO owning the Profile, meaning it contains the MNO's OTA Key sets.

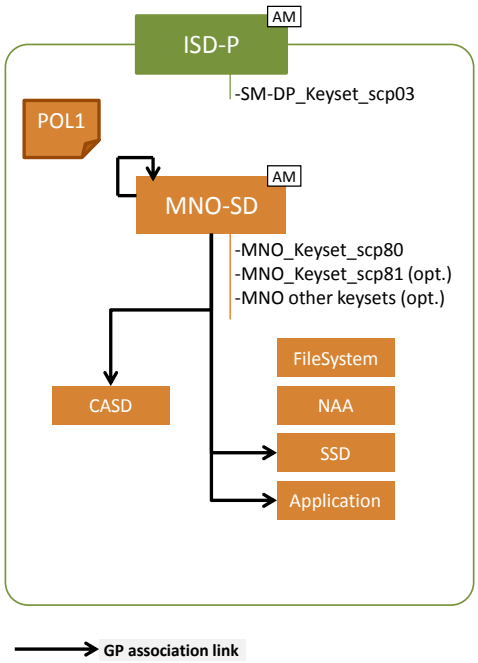


Figure 4: Profile Structure Overview

The Profile in the Figure 4 provides an example of a Profile structure.

The Profile structure shall include:

- The MNO-SD
- At least one NAA
- POL1, even if not used

- The file system

The Profile structure may contain:

- Several Applications (as defined in GlobalPlatform Card Specification [6]) in addition to the MNO-SD
- One CASD (as defined in GlobalPlatform Card Specification UICC Configuration [7])

The privileges that can be allocated to the MNO-SD and to applications shall comply with Annex C.

It shall be possible for the MNO to establish secure channels between the MNO OTA Platform and security domains of the Profile as specified in ETSI TS 102 225 [4] and ETSI TS 102 226 [5].

2.2.5 Secure Channel on Interfaces

2.2.5.1 Secure Channel on ES5 (SM-SR-eUICC)

The ES5 functions are addressed to the eUICC through a secure channel established between the SM-SR and the ISD-R. The eUICC shall support SCP80 and may support SCP81 (defined in ETSI 102 225 [4] and ETSI 102 226 [5]). See also section 2.4.

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].

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].

The key length and algorithm shall comply with section 2.3.3.

The key sets shall be loaded in the ISD-R, and provided to SM-SR, in the EIS, through ES1

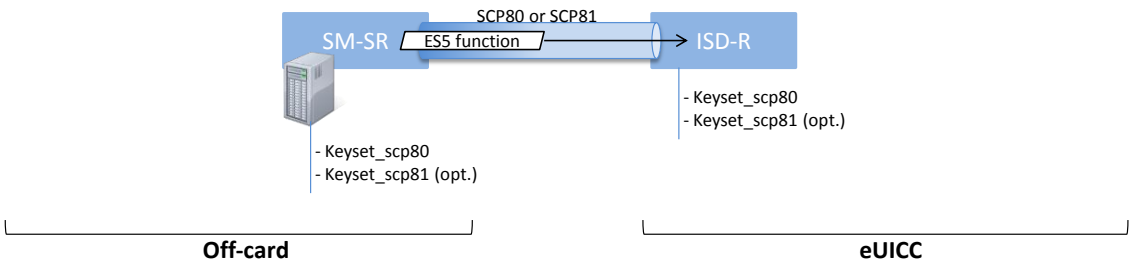


Figure 5: Secure Channel Between SM-SR and ISD-R

2.2.5.2 Secure Channel on ES8 (SM-DP - eUICC)

The ES8 functions are addressed to the eUICC through a secure channel established between the SM-DP and its ISD-P. The eUICC shall support SCP03 for ES8 (as defined in GlobalPlatform Card Specification Amendment D [10], as well as the variant SCP03t defined in this specification (see section 4.1.3.3).

NOTE: SCP03 is the only secure channel defined by GlobalPlatform that complies with requirements of the section 2.3.3.

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]).

The secure channel configuration, key length and algorithm to be used shall comply with section 2.5.

The first SCP03 key set is loaded into the ISD-P by its SM-DP as described in the procedure "Key Establishment with Scenario#3-Mutual Authentication", section 3.1.2.

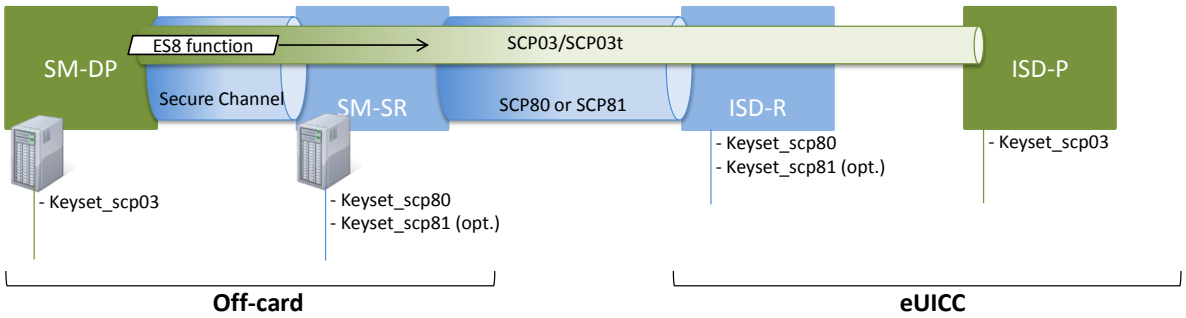


Figure 6: Secure Channel Between SM-DP and ISD-P

2.2.5.3 Secure Channel on ES6 (MNO-eUICC)

The ES6 functions are addressed to the eUICC through a secure channel (as defined in ETSI TS 102 225 [4] and ETSI TS 102 226 [5]) established between the MNO and the MNO-SD (as defined in section 2.2.3).

NOTE: The MNO can also communicate with any other SSD (of the Profile) belonging to the MNO. The Figure 7 only illustrates the secure channel with the MNO-SD.

The initial OTA Key sets are part of the Profile and are loaded by the SM-DP during the "Profile Download and Installation", see section 3.1, or loaded by the EUM before eUICC issuance.

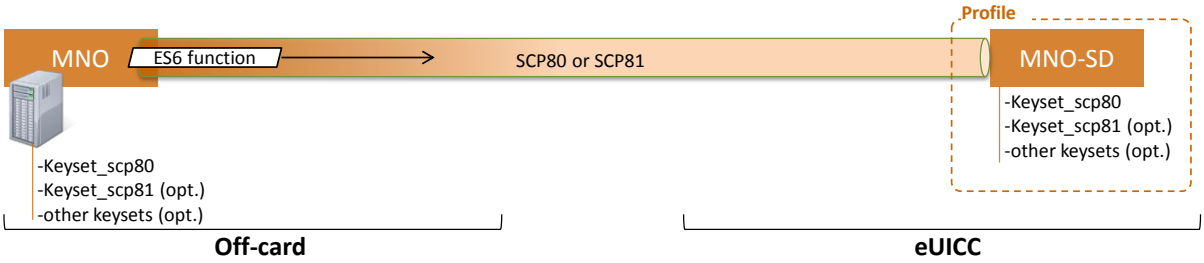


Figure 7: Secure Channel Between MNO and MNO-SD

2.3 Security Overview

This section provides an overview of the overall ecosystem security features.

The expectation of this architecture is to provide a solution offering a security level at least equivalent to the security reached by the current UICC and its management systems.

The security requirements have to be applied to the different Actors and Roles (Customer, MNO, SM-DP, SM-SR, CI, eUICC and eUICC Manufacturer). Each Role is considered as elements which can belong to a security realm and has to fulfil the appropriate certification scheme criteria.

According to section 2.2.3 of the GSMA Remote Provisioning Architecture for Embedded UICC [1]:

- 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.

In addition to the intrinsic security of each security realm, the data exchanged between these entities has to be protected. Any communication between two security realms of the eUICC ecosystem shall be origin authenticated, as well as integrity-Protected and, unless otherwise specified in detailed sections of this specification, confidentiality protected.

For all the procedures described in this specification the security realms are mutually authenticated and they have negotiated a minimal-acceptable common cryptographic suite for further communication.

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. The Profile Management commands (ES8) shall be at least protected by a SCP03 security level as detailed in section 2.5.

Off-card entities shall implement access control mechanisms for all function execution and data access requests. This access shall be authorised and any access shall be traced as defined in the GSMA certification schemes.

2.3.1 Certificate Issuer Role

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 Certification Chains

The Certificate Issuer Role issues certificates for Embedded UICC remote provisioning system entities and acts as a trusted root for the purpose of authentication of the entities of the system.

The following certificates shall be signed and issued by the CI:

- Self-signed Root Certificate
- EUM Certificates
- SM-SR Certificates
- SM-DP Certificates

The following certificates shall be signed and issued by the EUM:

- eUICC Certificates

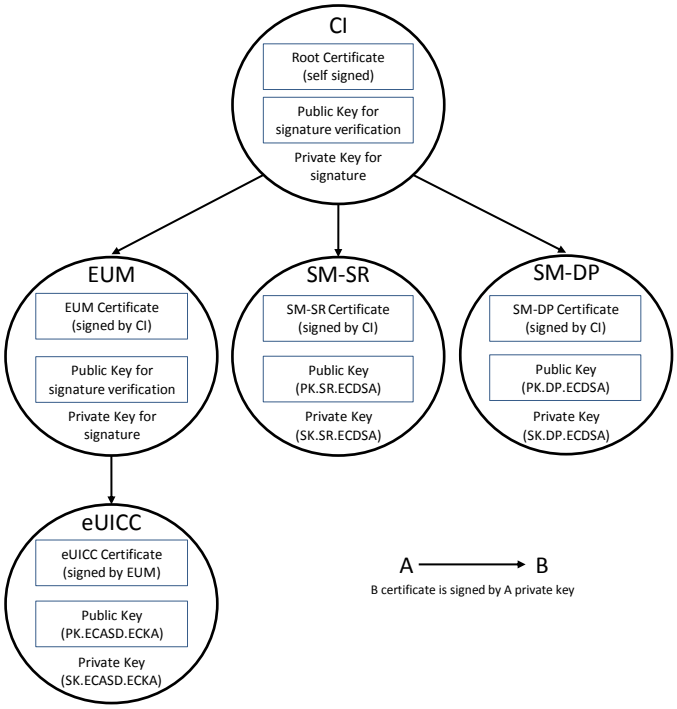


Figure 8: Certificate Chains

The following certificates shall be checked by the eUICC:

- the SM-SR Certificate
- the SM-DP Certificate

The following certificate and key shall be stored in the eUICC:

- the eUICC Certificate
- the Root public key

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 (for example BSI, ANSSI).

2.3.3 General Consideration on Algorithm and Key Length

Following the recommendations of several security agencies (for example NIST: SP 800-57 Part 1: Recommendation for Key Management [36], which was last revised in 2012; BSI: TR-02102: Kryptographische Verfahren: Empfehlungen und Schlüssellängen [37], which was updated in 2013. For an overview see also http://www.keylength.com/en/2/), the following table provides an overview of the key lengths and hashing methods that shall be applied in the context of this specification to ensure a good level of security up to the horizon 2030:

Algorithm	Minimum Key Length
Symmetric (AES)	128 bits, block size of 128 bits
Asymmetric (RSA)	3072 bits
Elliptic curve	256 bits
Hashing for Digital signatures and hash-only applications	SHA-256
Hashing for HMAC, Key Derivation Functions and Random Number Generation	SHA-256

Table 3: Algorithm and Key Length

2.4 OTA Communication on ES5 (SM-SR-eUICC)

2.4.1 General OTA Requirements

In the eUICC Remote Provisioning and Management system the OTA communication is exclusively handled by the SM-SR. The SM-SR can use SMS, CAT_TP and HTTPS for remote OTA communication with the eUICC.

- The eUICC shall support SMS and either CAT_TP or HTTPS or both.
- Device requirements are stated in Annex G.
- The SM-SR shall support SMS, HTTPS and CAT_TP.
- For LTE network deployments the system shall support SMS as defined in GSMA PRD IR.92 [38].
- The SM-SR is free to select the most relevant protocol according to the eUICC and Device capabilities and the platform or Profile Management operation to execute.
- The eUICC shall support the RAM and RFM as defined in ETSI TS 102 226 [5], in particular Expanded Remote Application data format and Script chaining.

2.4.2 Void

2.4.3 SMS

The usage of the SMS protocol may be relevant in several situations:

- SMS for HTTPS session triggering, as defined in ETSI TS 102 226 [5], and also in OMA-Smart Card Web Server [14] (section “Remote Administration Request sent using a MT-SMS”)
- SMS for CAT_TP session triggering as defined in ETSI TS 102 226 [5].
- When a command to be sent to the eUICC can fit into a few SMS; such a solution can be more efficiently handled via SMS, as compared to HTTPS.

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’). Minimum key lengths are defined in section 2.3.3.

- Procedures for the PoR shall follow ETSI TS 102 225 [4] and 3GPP TS 31.115 [13] with the following precisions:In the case that an incoming SMS for the ISD-R does not meet this security level, 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.

SPI2 shall be set to:

- ‘00’: no PoR (this value shall only be used for the notification described in section 3.15.1),
- or to ‘39’: PoR with CC and encryption.

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. Minimun key lengths are defined in section 2.3.3.

All these security requirements shall apply also for the SCP80 secured packets exchanged during a CAT_TP session.

2.4.3.1 SMS for HTTPS Session Triggering

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”.

NOTE: Normally the SM-SR will close the session. However, if needed, the eUICC may close the session.

2.4.3.2 SMS for CAT_TP Session Triggering

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”. These parameters and the corresponding “Data for BIP channel opening” and “Data for CAT_TP link establishment” are separated in two different commands sent in the same push SMS.

NOTE: Normally the SM-SR will close the session. However, if needed, the eUICC may close the session.

2.4.3.3 Command Format in SMS

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.

2.4.4 HTTPS

If HTTPS is used, the following sections shall apply.

2.4.4.1 PSK-TLS

2.4.4.1.1 Cipher Suites

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

The Pre-Shared Keys shall have an entropy of at least 128 bits.

The eUICC ISD-R shall be configured with ‘i’ = ‘04’ to indicate only TLS 1.2 supported as defined in GlobalPlatform Amd B [8].

Session resumption and several parallel sessions shall not be supported.

2.4.4.1.2 PSK-ID Value for TLS Handshake

The PSK-ID specified in this section defines a format suited for PSK random keys. However, a SM-SR may use a dedicated PSK-ID format in particular to be able to manage PSK

derived from master key. In this case the derivation algorithm used shall be robust and follow the NIST recommendation SP800-56C [55], the derived Pre-Shared Keys shall have an entropy of at least 128 bits.

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.

NOTE: As the PSK Identity is expected to be as short as possible, all lengths are coded in one byte. BER-TLV coding is unnecessary in this case.

Description	Length (bytes)	Value
Tag for identifying PSK-ID format	1	‘80’
Length	1	‘01’
Identification of the PSK-ID format.	1	Expected value is ‘02’ indicating a full qualified format for random PSK.
Tag for indicator of EID	1	‘81’
Length of EID	1	‘10’
EID value	16	The EID value. The value shall be coded in hexadecimal string representation.
Tag for security domain AID	1	‘4F’
Length of security domain AID	1	‘10’
Security domain AID value	16	The AID value of the ISD-R. The value shall be coded in hexadecimal string representation.
Tag for key identifier	1	‘82’
Length	1	‘01’
Key identifier	1	The key identifier value. The value shall be coded in hexadecimal representation.
Tag for Key version	1	‘83’
Length	1	‘01’
Key version	1	The key version value. The value shall be coded in hexadecimal representation. Key version number range reserved for SCP81 is '40' to '4F'.

Table 4: PSK-ID Format

Example of PSK-ID before conversion to an UTF-8 string:

‘8001028110010203040506070809010203040506074F10000102030405060708090A0B0C0D0E0F820101830140’

2.4.4.2 HTTP POST Request of ISD-R

The POST request is used by the ISD-R to fetch remote APDU strings and to transmit response strings. The ISD-R shall strictly follow GlobalPlatform Card Specification

Amendment B [8] for the format of the POST request. The content of the HTTP POST header field X-Admin-From shall be filled with the “Agent Id” information standardised in GlobalPlatform Card Specification Amendment B [8], section “Administration Session Triggering Parameters” (the format of this field is not standardised).

“Agent Id” information shall include two parts:

- the eUICC identifier (EID)
- the identifier of the Security Domain representing the Admin Agent function

Each part is built using the following format:

```
//<part-id>/<part-id-type>/<part-id-value>
```

Where:

- `<part-id>` is the tag that specifies which part is defined: “se-id” or “aa-id”
- `<part-id-type>` specifies the type of the identifier that is provided: “eid” or “aid”
- `<part-id-value>` provides the identifier value itself.

Format of the “X-Admin-From” field:

```
//se-id/eid/<EID>;//aa-id/aid/<RID ISD-R AID>/<PIX ISD-R AID>
```

Note that this representation of AID in the format /aid/<RID>/<PIX> is already used in GlobalPlatform for other purposes than the “Agent Id”.

Example of Agent Id field:

```
"//se-id/eid/89001012012341234012345678901224;//aa-
id/aid/0001020304/05060708090A0B0C0D0E0F"
```

The eUICC shall use the Chunked mode [Transfer-Encoding: chunked CRLF] for the POST request message.

The SM-SR shall use Chunked mode [Transfer-Encoding: chunked CRLF] for the POST response.

First request sent by the ISD-R:

POST <initial uri> **HTTP/1.1** CRLF

Host: <SM-SR ip> CRLF

X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF

X-Admin-From://se-id/eid/<value>;//aa-id/aid/ <RID ISDR-AID >/<PIX ISDR-AID> CRLF

CRLF

Return of a command response (no error case) sent by the ISD-R:

POST <uri contained in the previous POST response> **HTTP/1.1** CRLF

Host: <SM-SR ip> CRLF

X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF

X-Admin-From://se-id/eid/<value>;//aa-id/aid/ <RID ISDR-AID value>/<PIX ISDR-AID> CRLF

Content-Type: application/vnd.globalplatform.card-content-mgt-response;version=1.0 CRLF

Transfer-Encoding: chunked CRLF

X-Admin-Script-Status: ok CRLF

CRLF

[response-string]

2.4.4.3 HTTP POST Response of SM-SR

The POST response is used by the SM-SR to transmit the next remote APDU format string to the ISD-R and possibly to provide the next URI that must be used to request the following admin command.

The POST response shall strictly follow the GlobalPlatform Card Specification Amendment B [8].

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]

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]

Last POST response sent by the SM-SR with nothing to do, communication shall be closed:

HTTP/1.1 204 CRLF
X-Admin-Protocol: globalplatform-remote-admin/1.0 CRLF
CRLF

2.4.4.4 Command Format in HTTP Message

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.

2.4.4.5 Sequence for HTTPS Session Triggering

Except if specified differently for a specific procedure, an HTTPS session with the eUICC is always triggered by the SM-SR by sending a MT-SMS as defined in section 2.4.3.1.

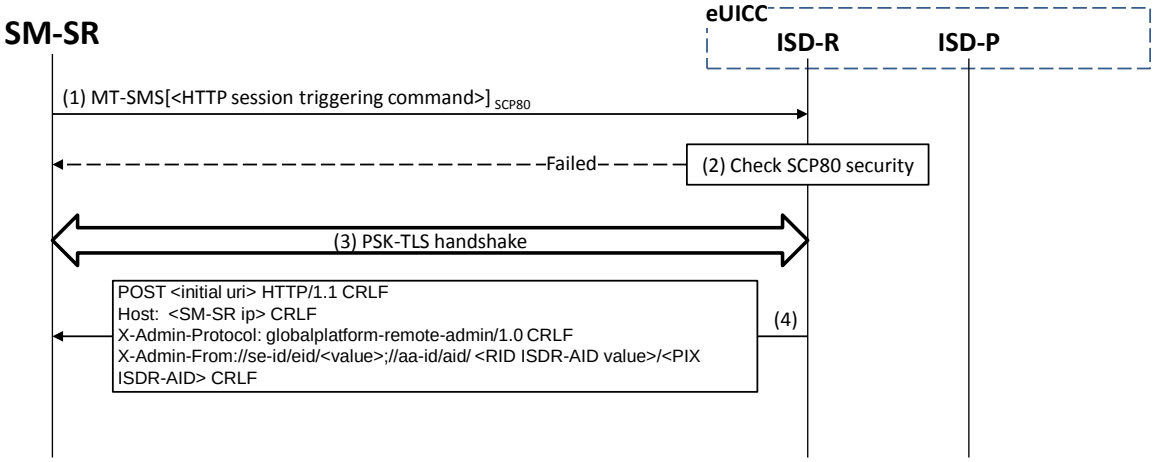


Figure 9: Sequence for HTTPS Session Triggering

1. The SM-SR sends a MT-SMS to the ISD-R for HTTPS session triggering as defined in section 2.4.3.1.
1. The ISD-R checks the security of the MT-SMS. The figure assumes the security is ok as defined in [13], else a PoR shall be returned to the SM-SR to indicate the failure (only in case the received SMS contains an authenticated TP-OA). In case of a temporary or fixable error the SM-SR shall retry or fix the error.
2. The PSK-TLS handshake is performed as defined in [16] and [17]. The figure assumes the security is ok. In case of a temporary or fixable error, the SM-SR shall retry or fix the error.
3. The first POST request is sent to the SM-SR as defined in section 2.4.4.2.

Then the SM-SR can continue with the procedure to execute.

2.5 Communication on ES8 (SM-DP) - eUICC

The ES8 interface is between the SM-DP and its ISD-P and goes through the SM-SR.

The ES8 is realised by a SCP03 or SCP03t secure channel that is tunnelled through the secure channel between the SM-DP and the SM-SR (ES3) and on through into the SCP80

or SCP81 secure channel between the SM-SR and the ISD-R (ES5). It is then provided by the ISD-R to the ISD-P. This is shown in the Figure 6.

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 4.1.3.3), 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.

2.6 SM-DP to SM-SR Link Establishment (ES3)

The link between the SM-DP and the SM-SR (ES3) may have to be established during a procedure. For the "Profile Download and Installation" procedure, the MNO may ask to the SM-DP to contact an SM-SR that may be unknown to the SM-DP. The SM-DP will have to establish a connection with this new SM-SR.

It is assumed in this specification that:

- The MNO, requesting an action of an SM-DP through the ES2 interface, is able to provide the identification of the SM-SR in charge of the management of the eUICC targeted by the function.
- The SM-DP, based on the SM-SR identification provided through the ES2 interface, is able to retrieve the SM-SR address.
- The SM-DP, based on the SM-SR identification and address, is able to establish a new link to the identified SM-SR during any procedure requiring this step.

The procedure describing how the SM-DP establishes a link to the SM-SR (for example: business agreement or technical solution) is not covered by this specification.

2.7 OTA Platform Communication on ES6 (MNO-eUICC)

The ES6 is the interface between the MNO OTA Platform and a Profile inside an eUICC (see also section 2.2.5.3) through a secure channel as defined in ETSI TS 102 225 [4] and ETSI TS 102 226 [5]. This interface is the same as the one used with UICCs.

This specification recommends that OTA Platform communication on ES6 makes use of at least a minimum security settings defined for ES5 in section 2.4.

3 Detailed Procedure Specifications

This section contains the detailed specifications of the procedures that realise the Remote Provisioning and Management system for the eUICC.

3.1 Profile Download and Installation

The Profile Download and Installation procedure is sub-divided into four main steps:

1. ISD-P creation on the eUICC
2. Personalization of the ISD-P with a first key set, called the key establishment procedure
3. Download and installation of the Profile onto the eUICC
4. Optional: Enabling of the newly installed Profile.

3.1.1 ISD-P Creation

The next figure describes the call flow for the first step which is the ISD-P creation. The procedure illustrates the usage of RAM over HTTP as an example of the transport protocol, assuming that the sequence will be followed by a key establishment procedure and the full download of the Profile.

NOTE: CAT_TP could be used as transport protocol and would have an equivalent procedure.

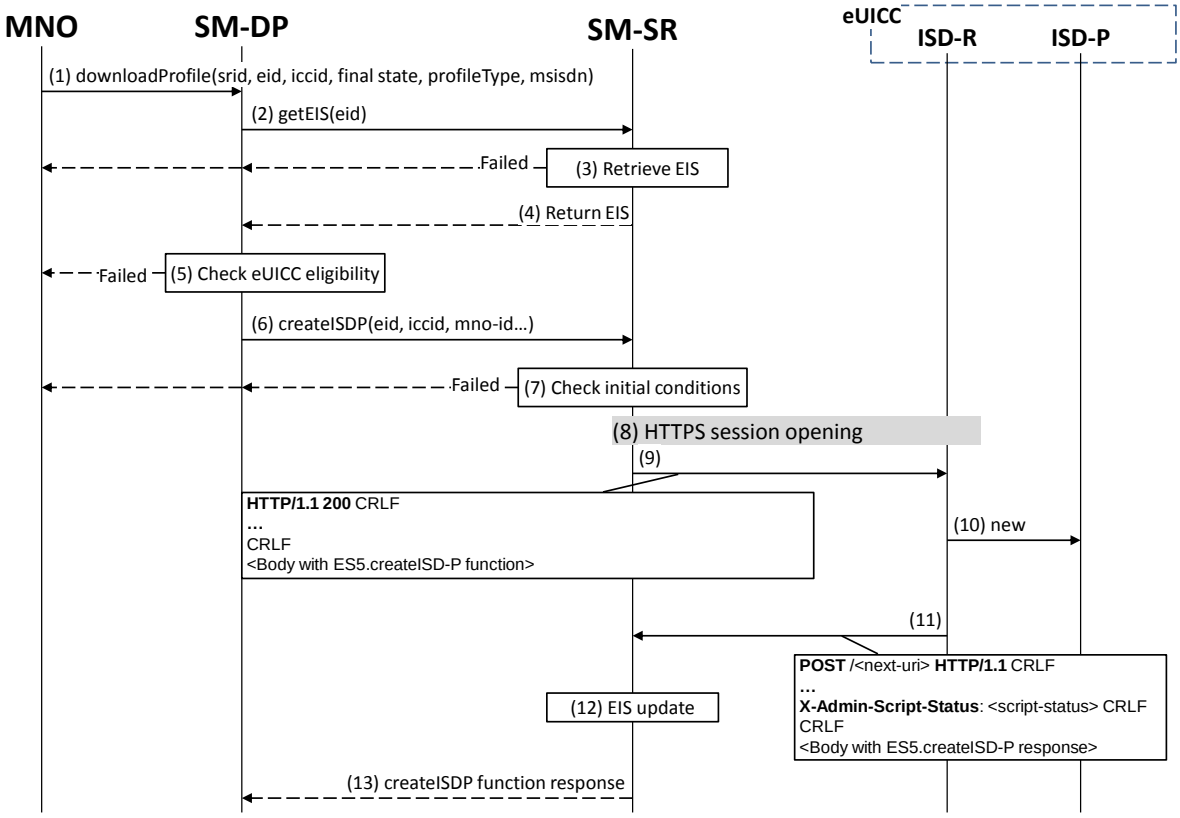


Figure 10: ISD-P creation

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) The MNO owning the Profile to download shall call the "**ES2.DownloadProfile**" function with its relevant input data (the MNO has to provide the SM-SR identification and address). By providing the required final state, the MNO may ask the SM-DP to enable the newly downloaded Profile at the end of the procedure. Else, by default, the Profile will be in the DISABLED state.

(2) The SM-DP on reception of this request shall call the "ES3.GetEIS" function with its relevant input data.

(3) The SM-SR shall retrieve the EIS of the eUICC based on the EID. At this stage the SM-SR may return an error indicating that the eUICC is unknown in its system. The error shall be finally returned to the MNO and the procedure shall end.

(4) The SM-SR shall return the EIS of the eUICC.

(5) The SM-DP shall check the eligibility of the eUICC against the characteristics of the Profile to be downloaded. Although the exact checks performed by the SM-DP are out of scope for this specification, some examples might include:

   a. Is the target Profile compatible with and validated against this type of eUICC? (including the fact that the SM-DP is able to generate the Profile for this type of eUICC).

   b. Is there enough memory? In case of uncertainty of the information contained within the EIS, the SM-DP could request an online audit.

   c. Is the eUICC certified? In case of a non-certified eUICC, the SM-DP may stop the procedure.

   The SM-DP shall verify the ECASD certificate, which was received as part of the EIS, using the EUM Certificate and the CI's Root Certificate and shall extract PK.ECASD.ECKA from the ECASD certificate.

   If any of these conditions is not satisfied or if the certificate verification fails, the SM-DP shall return a response indicating a failure.

(6) The SM-DP shall call the "ES3.CreateISDP" function with its relevant input data.

(7) The SM-SR shall verify that the SM-DP request is acceptable (the verifications that the SM-SR shall perform are described in the section 5.4.3).

   If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating a failure, and the procedure shall stop.

(8) If there is no existing HTTPS session with the eUICC, the SM-SR shall trigger the HTTPS session as defined in section 2.4.4.5.

(9) The SM-SR shall return the HTTP POST response containing the "ES5.CreateISDP" with its relevant input data. The X-Admin-Targeted-Application parameter shall be omitted as the command is targeting the ISD-R.

(10) The ISD-R shall create the ISD-P. In case of an error, the ISD-R shall return the error within the next POST request to the SM-SR. The error shall be finally returned to the SM-DP and the procedure may end depending on the error.

(11) The eUICC shall return the "ES5.CreateISDP" function execution response within the POST request to the SM-SR.

(12) Assuming a successful ISD-P creation, the SM-SR shall update the EIS to reflect the newly created ISD-P.

(13) The SM-SR shall return to the SM-DP the “ES3.CreateISDP” function execution response.

In this sample procedure, it is assumed that the SM-DP has indicated “more to do” in the “ES5.CreateISDP” call. In case the SM-DP did not indicate “more to do”, the SM-SR may end the HTTPS session.

3.1.2 Key Establishment with Scenario#3-Mutual Authentication

The next figure describes the second step in the Profile Download and Installation procedure.

This sequence defines a new scenario called “Scenario#3-Mutual Authentication”. This sequence uses Scenario#3 based on ECKA EG (ElGamal) scheme as defined in GlobalPlatform Card Specification Amendment E [11] complemented by an SM-DP authentication step.

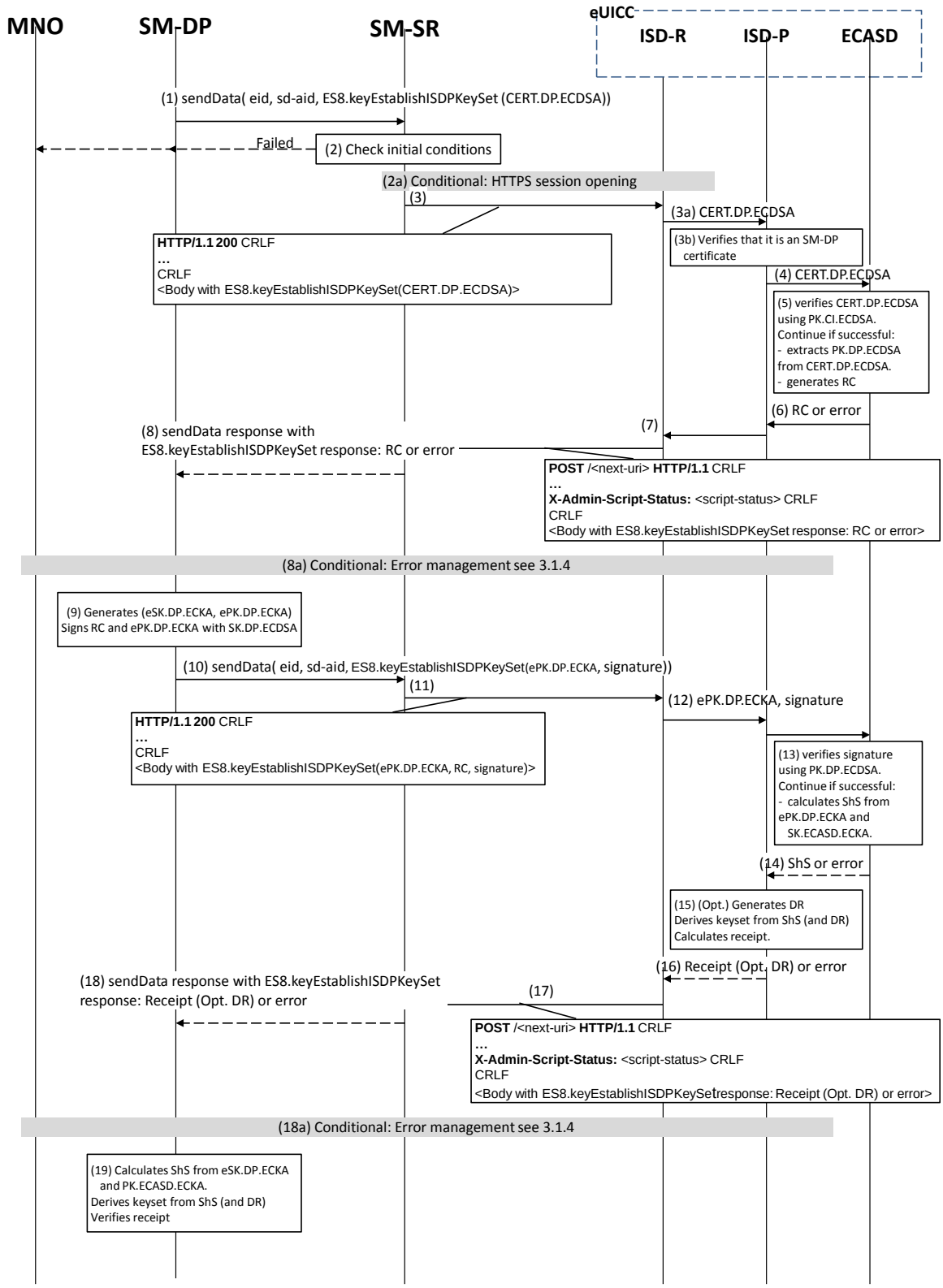


Figure 11: Key Establishment, Scenario #3

Start Conditions:

As a pre-condition, the ISD-P shall be created as defined in section 3.1.1, the eUICC/ECASD shall support the scenario#3-Mutual Authentication and shall be provisioned with the SK.ECASD.ECKA, PK.CI.ECDSA.

Procedure:

(1) The SM-DP shall call the "**ES3.SendData**" function specifying the targeted eUICC, the ISD-P, and the data containing the "**ES8.EstablishISDPKeySet**" function with the certificate identifying the SM-DP. The certificate shall be issued by the SM-DP Certificate Issuer.

(2) The SM-SR shall verify that the SM-DP request is acceptable (the verifications that the SM-SR shall perform are described in the section 5.4.4).

If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating a failure, and the procedure shall stop.

(2a) The SM-SR shall trigger the HTTPS session with the ISD-R if not already opened.

(3) The SM-SR shall return the HTTP POST response with a body containing the "**ES8.EstablishISDPKeySet**" function as provided by the SM-DP in (1).

(3a) The ISD-R shall forward the content of the STORE DATA command contained in the HTTP response to the ISD-P.

(3b) The ISD-P shall verify that it is an SM-DP certificate.

(4) The ISD-P shall forward the CERT.DP.ECDSA to the ECASD for verification.

(5) ECASD shall verify the provided CERT.DP.ECDSA with the PK.CI.ECDSA; if CERT.DP.ECDSA is valid, ECASD shall extract and store the PK.DP.ECDSA and generate a random challenge (RC).

(6) The Random Challenge (or error if any) shall be returned to the ISD-P which forwards it to the ISD-R.

(7) The ISD-R shall return the execution response received from the ISD-P (RC or error) within a new HTTP POST request addressed to the SM-SR.

(8) The SM-SR shall return the content of the received HTTP POST (RC or error) to the SM-DP.

(8a) In case of failure during the key establishment procedure, error management procedure describes in section 3.1.4 shall be executed and the procedure shall stop.

(9) The SM-DP shall generate an ephemeral key pair (related to the targeted ICCID), called ePK.DP.ECKA and eSK.DP.ECKA. The SM-DP signs the received Random Challenge(RC) and the generated ePK.DP.ECKA with the SK.DP.ECDSA.

(10) The SM-DP shall call the "**ES3.SendData**" function specifying the targeted eUICC, the ISD-P and the data containing the "**ES8.EstablishISDPKeySet**" function with the ePK.DP.ECKA and the previously computed signature on Random Challenge (RC) and ePK.DP.ECKA using SK.DP.ECDSA.

(11) The SM-SR shall return the HTTP POST response with a body containing the "**ES8.EstablishISDPKeySet**" function as provided by the SM-DP in (10).

(12) The ISD-P shall forward the ePK.DP.ECKA and signature to the ECASD for verification.

(13) The ECASD shall verify the signature using the previously stored PK.DP.ECDSA. If the signature is not verified, an error shall be returned. Else the ECASD shall calculate the ShS using the ePK.DP.ECKA and the SK.ECASD.ECKA.

(14) The ShS or an error shall be returned to the ISD-P.

(15) The ISD-P:
- May optionally compute a Derivation Random (DR, if requested by the SM-DP in the function call).
- Derives the key set from ShS (and optionally DR).
- Calculates the receipt to be returned to SM-DP.

(16) The ISD-P shall return the calculated receipt (and optionally the DR) or the error to the ISD-R.

(17) The ISD-R shall return the execution response to the ISD-P (receipt (opt. DR) or error) within a new HTTP POST request addressed to the SM-SR.

(18) The SM-SR shall return the content of the received HTTP POST (receipt (opt. DR) or error) to the SM-DP.

(18a) In case of failure during the Download and Installation procedure, the error management procedure describes in section 3.1.4 shall be executed and the procedure shall stop.

(19) The SM-DP symmetrically shall:
- Calculates the ShS using the eSK.DP.ECKA and the PK.ECASD.ECKA,
- Derives the key set from ShS (and optionally DR), and
- Verify the receipt received in the response to ensure that key set derivation is consistent with what has been performed by the ISD-P.

The eUICC shall support key establishment with and without the DR. The SM-DP decides which option to use.

BSI TR-03111 [49] contains recommendations and requirements on the generation and validation of ephemeral keys. In addition, NIST SP 800-56A [50] provides requirements on the destruction of ephemeral keys and other intermediate secret data after their use.

3.1.3 Download and Installation of the Profile

This section describes the third part of the procedure for the Profile Download and Installation step. The procedure illustrates the usage of RAM over HTTP as an example of the transport protocol.

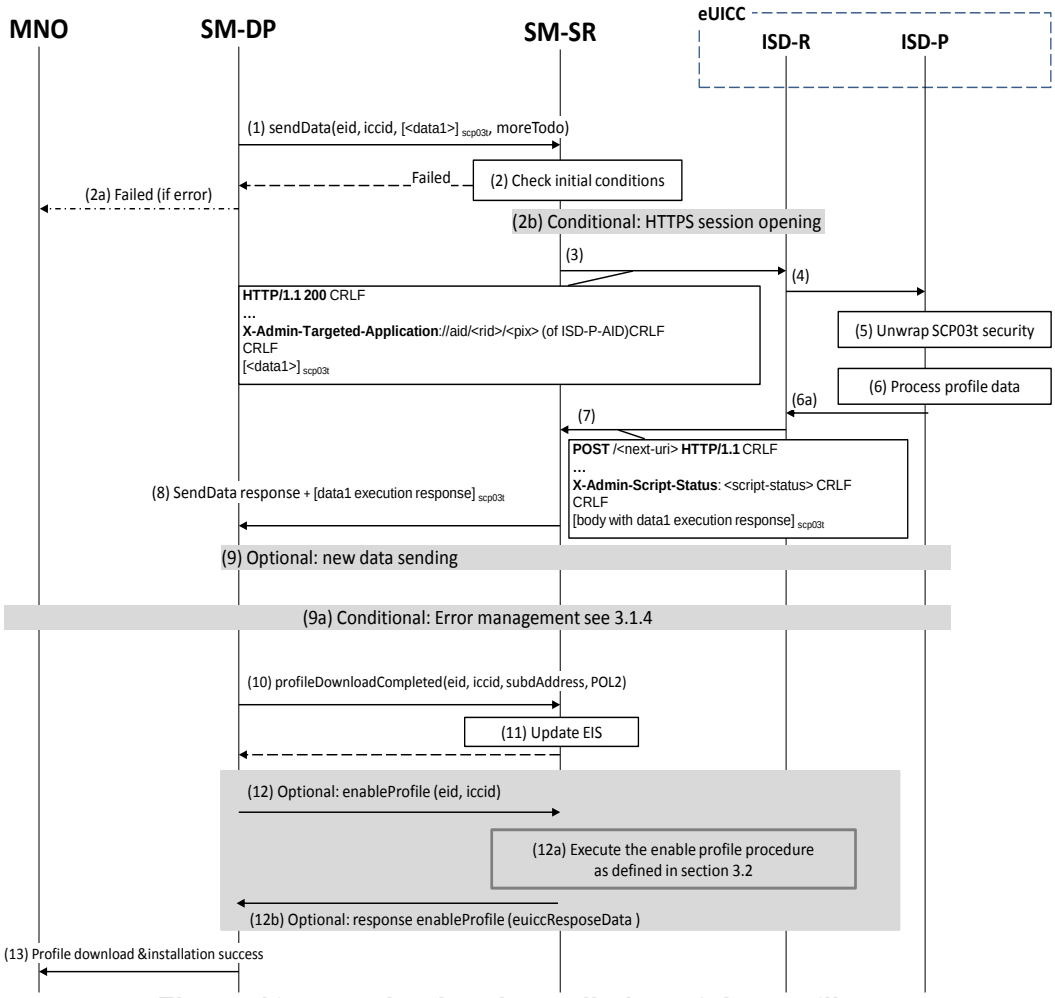


Figure 12: Download and Installation of the Profile

Start Conditions:

As a pre-condition, the ISD-P shall be created and personalized as defined in section 3.1.1 and section 3.1.2.

Procedure:

(1) The SM-DP shall call the "**ES3.SendData**" function providing the Profile data to download as input data. The Profile data has to be given as specified in section 4.1.3.1 and 5.4.4.

(2) The SM-SR shall verify that the SM-DP request is acceptable (the verifications that the SM-SR shall perform are described in section 5.4.4).

- (2a) Depending on the error, the procedure may stop and a global failure message shall be returned to the MNO.
- (2b) The SM-SR shall trigger the HTTPS session opening with the ISD-R if not already opened.

(3) The SM-SR shall return the HTTP POST response containing the secure data as provided by the SM-DP. The X-Admin-Targeted-Application field shall contain the ISD-P-AID.

(4) The ISD-R shall forward the received secure data to the ISD-P identified by the X-Admin-Targeted-Application field.

(5) The ISD-P shall process the security of the received data. The figure illustrates a success case; in case of security failure the error shall be returned within the next POST request to the SM-SR and finally returned to the SM-DP; and the procedure may end depending on the error.

(6) The ISD-P shall process the received command TLV(s).

(6a) The ISD-P shall return the response to the command TLV(s) to the ISD-R.

(7) The ISD-R shall return within the next POST request to the SM-SR.

(8) The SM-SR shall return to the SM-DP the execution status of the "**ES3.SendData**" function.

(9) Optionally the SM-DP may call the same "**ES3.SendData**" function again if the download and installation of the Profile requires several steps. This optional step may be repeated as many times as required.

(9a) In case of failure during the Download and Installation procedure, error management procedure describes in section 3.1.4 shall be executed and the procedure shall stop.

(10) When Profile download is completed the SM-DP shall call the "**ES3.ProfileDownloadCompleted**" function. This basically indicates to the SM-SR that the Profile is downloaded and installed. The SM-DP may take the opportunity to define a POL2 on the Profile. The MNO shall be able to sign the POL2 content even if it is empty.
As requested by the MNO, after Profile installation the SCP03 key set of the ISD-P may:

i) Be retained by the SM-DP. In this case the MNO can instruct the SM-DP to hand over or delete the key set at a later point in time;

ii) Be handed over to the MNO. The keys may be replaced by the MNO;

Be deleted from the eUICC by the SM-DP (using the GlobalPlatform DELETE command).

(11) The SM-SR shall update the EIS reflecting that the Profile is in "DISABLED" state, and POL2 if present.

(12) Optionally, if the MNO has initially requested the Profile to be enabled, the SM-DP shall request the SM-SR to enable the newly installed Profile by calling the "**ES3.EnableProfile**" function.

(12a) The SM-SR handles the "**ES3.EnableProfile**" function request (if called by the SM-DP) as described in section 5.4.8.

(12b) The SM-SR provides the status of the Profile Enabling function to the SM-DP.

(13) Finally, the SM-DP shall return the response to the "**ES2.DownloadProfile**" function call to the MNO.

At the end of this procedure, if the Profile has been enabled, the MNO owning of the Profile is able to perform any remote management operation to the Profile using its own Remote Administration Server.

3.1.4 Error Management Sub-Routine

The next figure describes the flow for error management. This procedure is called when an error occurs during the key-establishment or the Profile Download and Installation procedures. The procedure illustrates the usage of RAM over HTTP as an example of the transport protocol.

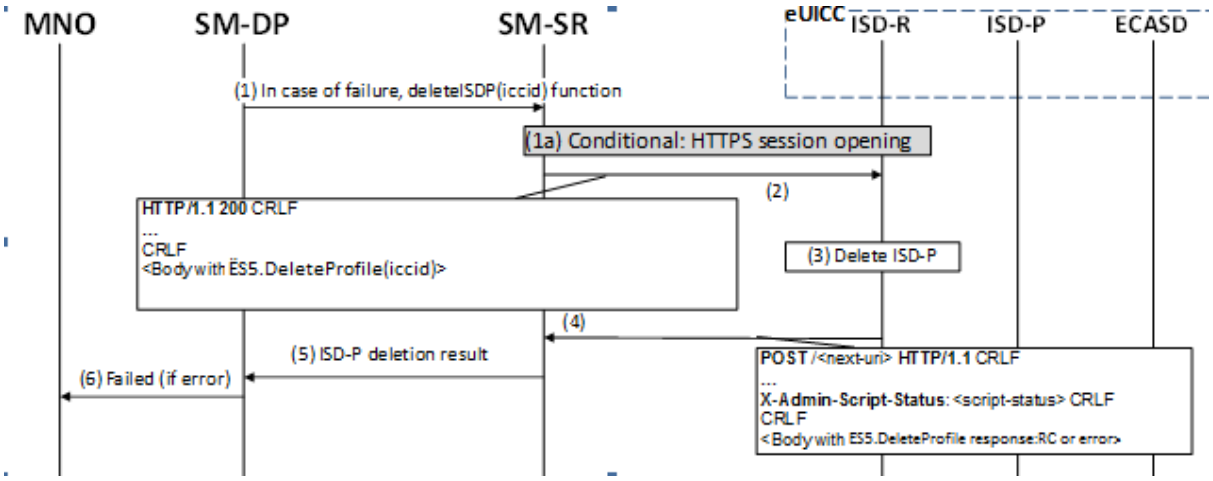


Figure 13: Error Management Sub-Routine

Procedure:

(1) In case of failure during the key establishment procedure or the Download Profile procedure, the SM-DP shall call the **"ES3.DeleteISDP"** function with its relevant input data.

(1a) The SM-SR shall trigger the HTTP session with the ISD-R if not already opened.

(2) The SM-SR shall return the HTTP POST response with a body containing the "**ES5.DeleteProfile"** function with the ICCID.

(3) The ISD-R shall delete the targeted ISD-P.

(4) The ISD-R shall return the execution response to the ISD-P deletion "**ES5.DeleteProfile**" within a new HTTP POST request addressed to the SM-SR.

(5) The SM-SR shall forward the status of the "**ES3.DeleteISDP**"to the SM-DP.

(6) The failure message shall be returned to the MNO.

3.2 Profile Enabling

The Profile Enabling procedure between the MNO and the SM-SR is used to enable a Profile previously downloaded and installed on an eUICC (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 3.5.5). The procedure is initiated by the MNO owning the Profile to be enabled. The procedure illustrates the usage of SMS as a possible transport protocol between SM-SR and eUICC, but can also be performed using other transport protocols.

3.2.1 Normal Case

The sequence flow in the Figure 14 describes the normal case where the target Profile can successfully be enabled.

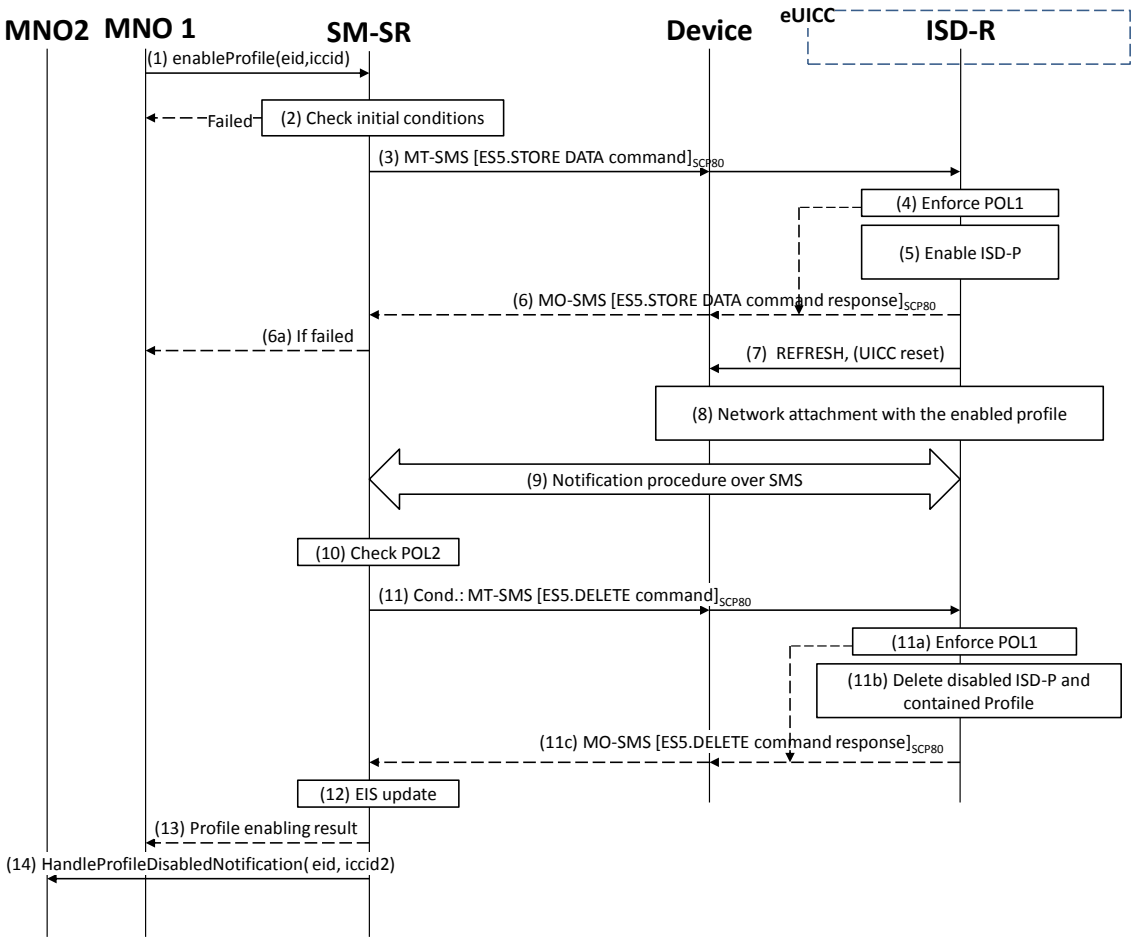


Figure 14: Profile Enabling, Success Case

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) MNO1 of the target Profile shall call the **"ES4.EnableProfile"** function with its relevant input data.

(2) The SM-SR shall verify that the MNO1 request is acceptable (the verifications that the SM-SR shall perform are described in the section 5.5.5), and in particular evaluates POL2 of the currently Enabled Profile. If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating the failure, and the procedure shall end.

(3) The SM-SR shall send an MT-SMS containing the "**ES5.STORE DATA**" command for Profile enabling with its relevant input data (see section 4.1.1.2) to the ISD-R. The SM-SR shall request a PoR to get the execution status of the "**ES5.STORE DATA**" command.

(4) The ISD-R shall enforce POL1 of the currently Enabled Profile. If POL1 rejects enabling of the target Profile, the ISD-R shall return directly the MO-SMS containing the response indicating a failure, and the procedure shall end.

(5) If POL1 allows, the ISD-R shall disable the currently enabled ISD-P and enable the targeted ISD-P.

NOTE: Profile change includes a change of the IMSI that is used to attach to the network. As indicated in 3GPP TS 31.102 [52], such a change requires special caution and should always be accompanied by a REFRESH command to avoid inconsistent information being read by the terminal. So while the targeted ISD-P is marked as enabled in this step, it may actually become effective after the terminal executes the REFRESH command.

(6) The ISD-R shall return the MO-SMS containing the execution status of the "**ES5.STORE DATA**" command to the SM-SR.

(6a) If the response to the "**ES5.STORE DATA**" command indicates a failure, the SM-SR shall return a response indicating the failure to MNO1, and the procedure shall end.

(7) The ISD-R shall send a REFRESH proactive command in UICC reset mode to the Device. This will trigger the execution of a network attach procedure.

NOTE: In case of any error after this steps, indicating that the currently Enabled Profile cannot provide connectivity, the ISD-R shall re-enable the previously Enabled Profile as described in section 3.2.2.

(8) The eUICC and the Device shall perform a network attach procedure with the newly Enabled Profile.

(9) The eUICC shall perform the notification procedure as described in section 4.1.1.11. During this procedure, if the ISD-R doesn't succeed in sending the SMS notification (after having exhausted all possible retries), or doesn't receive the SM-SR notification confirmation, this shall be considered as a fatal error, and the previous note shall apply. On reception of the SM-SR notification confirmation command, if POL1 of the now Disabled Profile contains the rule "Profile deletion is mandatory when it is disabled", the ISD-R shall delete the disabled ISD-P and the contained Profile. The eUICC shall send the response to the notification confirmation indicating whether the disabled ISD-P has been deleted or not.

(10) On reception of the "**ES5.HandleNotificationConfirmation**" response, and if this response indicates that the Disabled Profile has not been deleted, the SM-SR shall evaluate POL2 of the Disabled Profile. If POL2 of the Disabled Profile contains the rule "Profile deletion is mandatory when it is disabled", the SM-SR shall perform step (11), else it shall jump to step (12).

(11) The SM-SR shall send an MT-SMS containing the "**ES5.DELETE**" command with its relevant input data (see section 4.1.1.4) to the ISD-R, targeting the Disabled Profile. The SM-SR shall request a PoR to get the execution status of the "**ES5.DELETE**" command.

(11a) The ISD-R shall enforce POL1 of the target Profile. If POL1 rejects the deletion of the target Profile, the ISD-R shall return the MO-SMS containing the response indicating the corresponding failure, and the procedure shall end.

(11b) If POL1 allows its deletion, the ISD-R shall delete the targeted ISD-P and the contained Profile.

(11c) The ISD-R shall return the MO-SMS to the SM-SR containing the execution status of the "**ES5.DELETE**" command.

(12) According to the executed sequence and the eUICC responses, the SM-SR shall update the EIS to reflect that:

- The target Profile has been enabled
- The previously Enabled Profile has been disabled or deleted.

NOTE: POL1 and POL2 may have different content. As a consequence, both the eUICC and the SM-SR have to ensure the ISD-P deletion based on their respective Policy.

(13) The SM-SR shall return the response to the "**ES4.EnableProfile**" function to MNO1, indicating that the Profile has been enabled.

(14) The SM-SR shall send the "**ES4.HandleProfileDisabledNotification**" or "**ES4.HandleProfileDeletedNotification**" (if deletion was triggered by the evaluation of POL1 and POL2) to MNO2, the owner of the Profile that was enabled at the beginning of the procedure. In case MNO2 has no direct connection with the SM-SR (SM-SR shall be able to detect such a situation based on its own database), the SM-SR shall send this notification to the SM-DP authorised by MNO2 by calling the "**ES3.HandleProfileDisabledNotification**" or the "**ES3.HandleProfileDeleted**Notification". The SM-SR can retrieve the SM-DP identity based on the EIS content. Then the SM-DP, on reception of this notification, shall forward it to MNO2 by calling the "**ES2.HandleProfileDisabledNotification**" or the "**ES2.HandleProfileDeletedNotification**".

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.4.

3.2.2 Connectivity Failure Case

The sequence flow in the Figure 15 describes the case where the target Profile cannot provide connectivity after it is enabled, and when roll-back to the previously Enabled Profile occurs.

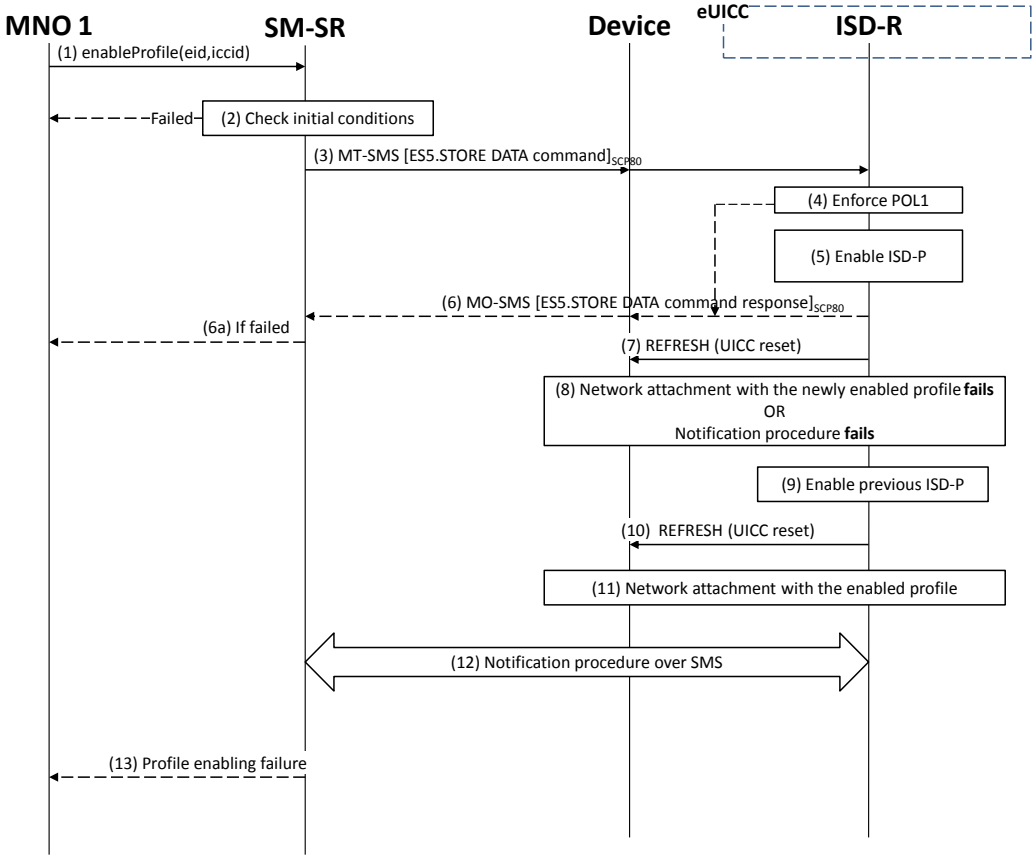


Figure 15: Profile Enabling failure, with roll-back

Start Conditions:

The start conditions are identical to section 3.2.1.

Procedure:

Steps (1), (2), (3), (4), (5), (6), (6a) and (7) are also identical to section 3.2.1.

(8) A network attach failure occurs indicating that the Enabled Profile cannot provide connectivity, or the eUICC doesn't succeed to send the SMS notification (after having exhausted all possible retries), or doesn't receive the SM-SR notification confirmation.

(9) The ISD-R shall enable the Profile that was previously enabled before the reception of the command, to re-establish connectivity.

NOTE: Profile change includes a change of the IMSI that is used to attach to the network. As indicated in 3GPP TS 31.102 [52], such a change requires special caution and should always be accompanied by a REFRESH command to avoid inconsistent information being read by the terminal. So while the targeted ISD-P is marked as enabled in this step, it may actually become effective only after the terminal executes the REFRESH command.

(10) The ISD-R sends a REFRESH proactive command in UICC reset mode to the Device. This will trigger the execution of a new network attach procedure.

(11) The eUICC and the Device shall perform a new network attach procedure with the Profile Enabled before the start of the procedure.

(12) The eUICC shall perform the notification procedure as described in section 4.1.1.11. On reception of the SMS notification, the SM-SR is informed that the target Profile has not been enabled.

(13) Finally, the SM-SR shall return the response to the "**ES4.EnableProfile**" function to MNO1; indicating a failure, the target Profile didn't succeed to provide the connectivity.

3.3 Profile Enabling Via SM-DP

The Profile Enabling procedure between the MNO and the SM-DP is used to enable a Profile previously downloaded and installed on an eUICC (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 3.5.5). The procedure is initiated by the MNO owning the Profile to be enabled. The procedure illustrates the usage of SMS as a possible transport protocol between SM-SR and eUICC, but can be also performed using other transport protocols.

This procedure is similar to the procedure "Enable Profile" described in section 3.2.

3.3.1 Normal Case

The sequence flow in the Figure 16 describes the normal case where the targeted Profile can successfully be enabled.

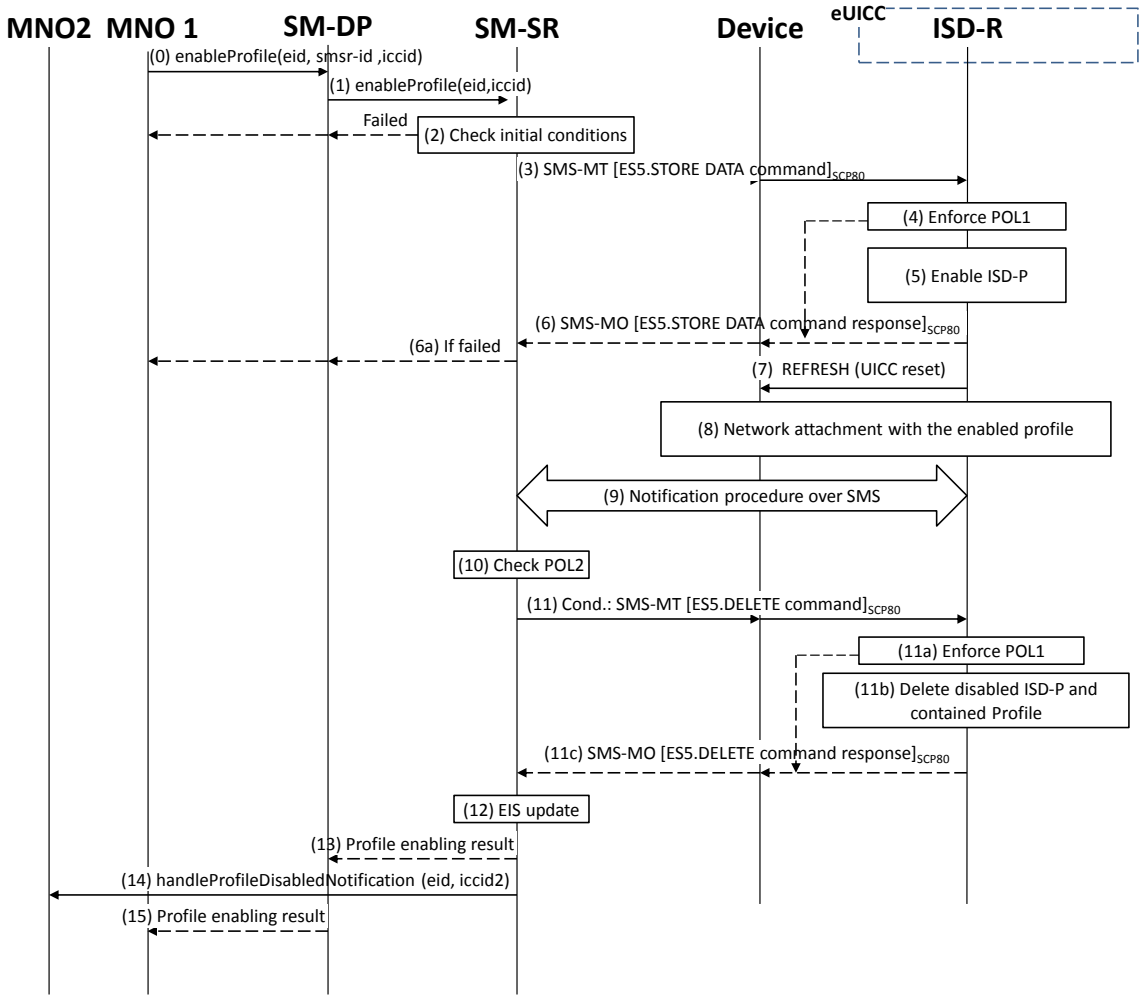


Figure 16: Profile Enabling, Success Case

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(0) MNO1, the owner of the target Profile, shall call the **"ES2.EnableProfile"** function with its relevant input data, see section 5.3.5, in particular the identification of the SM-SR in charge of the management of the target eUICC.

(1) The SM-DP shall forward the request to the SM-SR provided by the MNO and shall call the function "**ES3.EnabledProfile**". During this step the SM-DP may have to establish a link to the SM-SR (see section 2.6).

Steps (2) to (12) are the same as in the procedure "Profile Enabling" described in section 3.2.1.

(13) The SM-SR shall return the response to the "**ES3.EnableProfile**" function to the SM-DP, indicating that the Profile has been enabled.

(14) The SM-SR shall send the "**ES4.HandleProfileDisabledNotification**" or "**ES4.HandleProfileDeletedNotification**" (if deletion was triggered by the evaluation

of POL1 and POL2) to MNO2, the owner of the Profile that was enabled at the beginning of the procedure. In case MNO2 has no direct connection with the SM-SR, the SM-SR shall apply the same process as described in point (14) of section 3.2.1.

(15) Finally, the SM-DP shall return the response to the "**ES2.EnableProfile**" function call to MNO1.

3.3.2 Connectivity Failure Case

The sequence flow in the Figure 17 describes the case where the targeted Profile cannot provide connectivity after it is enabled, and when roll-back to the previously Enabled Profile occurs.

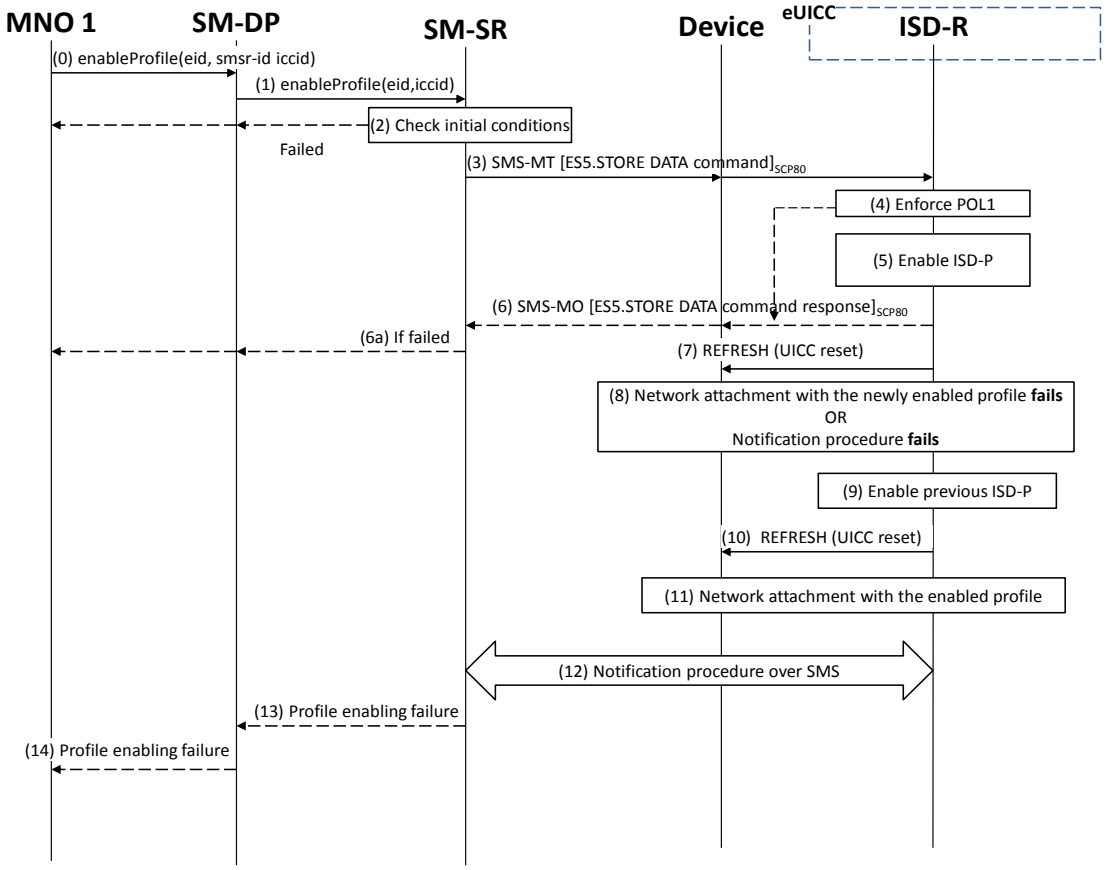


Figure 17: Profile Enabling, With Roll-Back

Start Conditions:

The start conditions are the same as in section 3.3.1.

Procedure:

Steps (0) and (1) are the same as in section 3.3.1.

Steps (2) to (12) are the same as in procedure "Connectivity failure case" as described in section 3.2.2.

(13) The SM-SR shall return the response to the "**ES3.EnableProfile"** function to the SM-DP, indicating a failure, the target Profile didn't succeed to provide the connectivity.

(14) Finally, the SM-DP shall return the response to the "**ES2.EnableProfile**" function to MNO1, indicating a failure, the target Profile didn't succeed to provide the connectivity.

NOTE: In case the previously Enabled Profile can also not provide connectivity, the eUICC shall activate the Fall-back Mechanism.

3.4 Profile Disabling

The Profile Disabling procedure is initiated by the MNO owning the Profile to be disabled. The procedure illustrated using SMS as a possible transport protocol between the SM-SR and the eUICC, but can be also performed using other transport protocols.

The sequence flow in the Figure 18 describes the case where the targeted Profile can successfully be disabled.

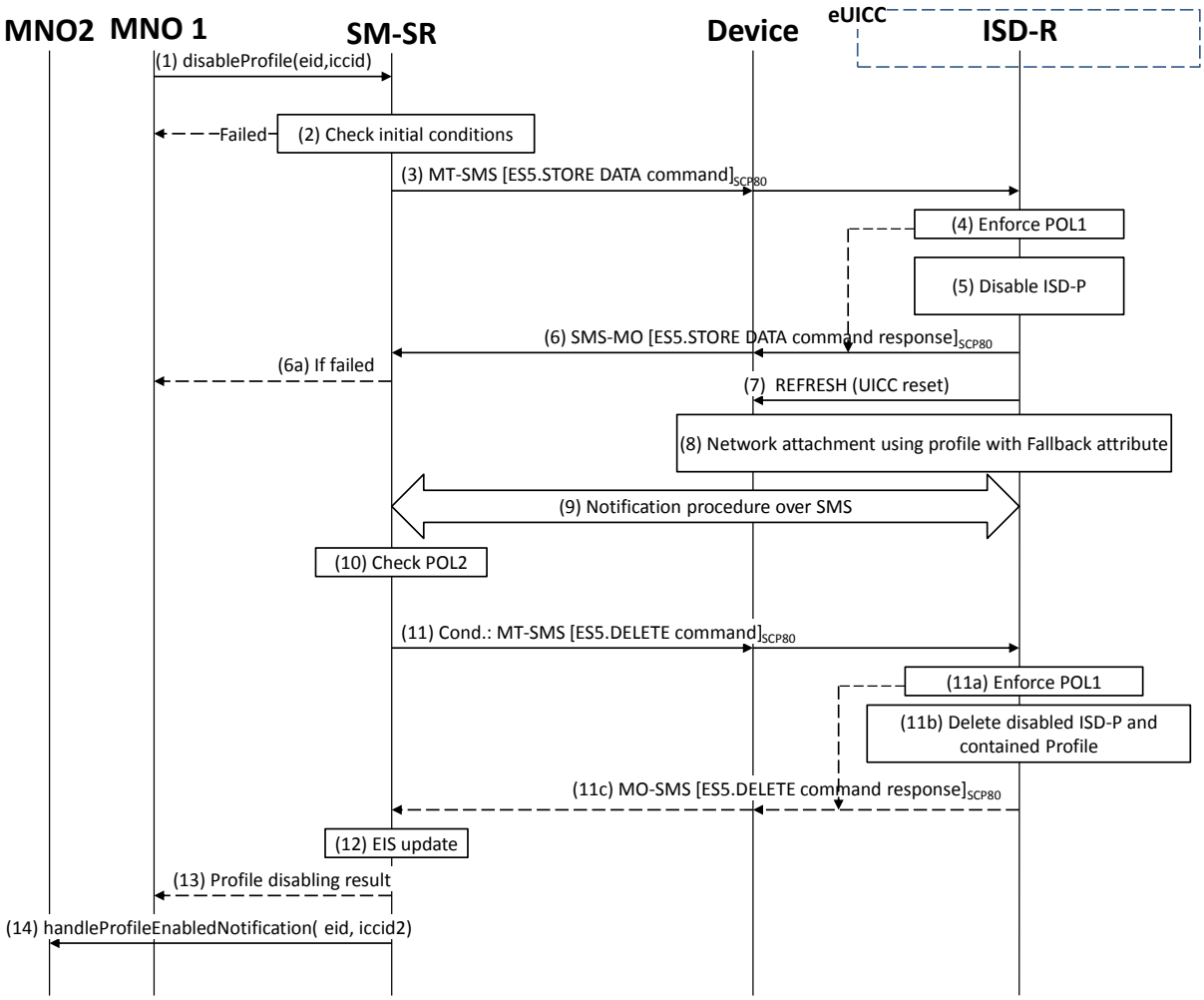


Figure 18: Profile Disabling

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) MNO1, the owner of the target Profile shall call the "**ES4.DisableProfile**" function with its relevant input data.

(2) The SM-SR shall verify that the MNO1 request is acceptable (the verifications that the SM-SR shall perform are described in the section 5.5.6, and in particular checks that

Profile is enabled and Profile disabling is allowed in POL2. If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating the failure, and the procedure shall end.

(3) The SM-SR shall send an MT-SMS containing the "**ES5.STORE DATA**" command for Profile disabling with its relevant input data (see section 4.1.1.3) to the ISD-R. The SM-SR shall request a PoR to get the execution status of the "**ES5.STORE DATA**" command.

(4) The ISD-R shall enforce POL1 of the currently Enabled Profile. In case POL1 rejects disabling, the ISD-R shall return PoR containing the response indicating a failure, and the procedure shall end.

(5) The ISD-R shall disable the targeted ISD-P and the contained Profile and shall enable the Profile with the Fall-back Attribute Set.

NOTE: Profile change includes a change of the IMSI that is used to attach to the network. As indicated in 3GPP TS 31.102 [52], such a change requires special caution and should always be accompanied by a REFRESH command to avoid inconsistent information being read by the terminal. So while the targeted ISD-P is marked as enabled in this step, it may actually become effective only after the terminal executes the REFRESH command.

(6) The ISD-R shall return the MO-SMS containing the execution status of the "**ES5.STORE DATA**" command to the SM-SR.

(6a) If the response to the "**ES5.STORE DATA**" command indicates a failure, the SM-SR shall return a response indicating the failure to MNO1, and the procedure shall end.

(7) The ISD-R sends a REFRESH proactive command in UICC reset mode to the Device. This will trigger the execution of a network attach procedure.

NOTE: In case of any error after this step indicating that the current Enabled Profile cannot provide connectivity, the ISD-R shall re-enable the previously Enabled Profile as described in section 3.2.2.

(8) The eUICC and the Device shall perform a new network attach procedure with the Profile with the Fall-back Attribute Set.

(9) The eUICC shall perform the notification procedure as described in section 4.1.1.11. During this procedure, if ISD-R doesn't succeed to send the SMS notification, or doesn't receive the SM-SR notification confirmation, this shall be considered as an error, and the previous note shall apply.

On reception of the SM-SR notification confirmation command, if POL1 of the Disabled Profile contains the rule "Profile deletion is mandatory when it is disabled", the ISD-R shall delete the disabled ISD-P and the contained Profile. The eUICC shall send the response to the notification confirmation indicating whether the disabled ISD-P has been deleted or not.

(10) On reception of the **ES5.HandleNotificationConfirmation** response, and if the **ES5.HandleNotificationConfirmation** response indicates that the Disabled Profile has not been deleted, the SM-SR shall evaluate POL2 of the Disabled Profile. If POL2 of the Disabled Profile contains the rule "Profile deletion is mandatory when it is disabled", the SM-SR shall perform step (11), else it shall jump to step (12).

(11) The SM-SR shall send an MT-SMS containing the "**ES5.DELETE**" command with its relevant input data (see section 4.1.1.4) to the ISD-R, targeting the Disabled Profile.

The SM-SR shall request a PoR to get the execution status of the "**ES5.DELETE**" command.

(11a) The ISD-R shall enforce POL1 of the target Profile. If POL1 rejects the deletion of the target Profile, the ISD-R shall return the MO-SMS containing the response indicating the corresponding failure, and the procedure shall end.

(11b) If POL1 allows its deletion, the ISD-R shall delete the targeted ISD-P and the contained Profile.

(11c) The ISD-R shall return the MO-SMS to the SM-SR containing the execution status of the "**ES5.DELETE**" command.

(12) According to the executed sequence and the eUICC responses, the SM-SR shall update the EIS to reflect that:
- The Profile having the fall-back attribute has been enabled
- The previously Enabled Profile has been disabled or deleted.

NOTE: POL1 and POL2 may have different content. As a consequence, both the eUICC and the SM-SR have to ensure the ISD-P deletion based on their respective Policy.

(13) The SM-SR shall return the response to the "**ES4.DisableProfile**" function to MNO1, indicating that the Profile has been disabled. In case the Profile has also been deleted because of POL1 or POL2, the function execution response shall include an execution status "Executed-WithWarning" indicating that the Profile has also been deleted.

(14) The SM-SR shall send the "**ES4.HandleProfileEnabledNotification**" to MNO2, the owner of Profile with Fall-back Attribute Set that is now enabled. In case MNO2 has no direct connection with the SM-SR (SM-SR shall be able to detect such situation based on its own database), the SM-SR shall send this notification to the SM-DP authorized by MNO2 by calling the "**ES3.HandleProfileEnabledNotification**". The SM-SR can retrieve the SM-DP identity based on the EIS content. Then the SM-DP, on reception of this notification, shall forward it to MNO2 by calling the "**ES2.HandleProfileEnabledNotification**".

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.4.

3.5 Profile Disabling Via SM-DP

The Profile Disabling procedure is initiated by the MNO owning the Profile to be disabled through the SM-DP. The procedure illustrated using SMS as a possible transport protocol between SM-SR and eUICC, but can be also performed using other transport protocols.

This procedure is similar to the procedure "Disable Profile" described in section 3.4.

The sequence flow in the Figure 19 describes the case where the targeted Profile can successfully be disabled.

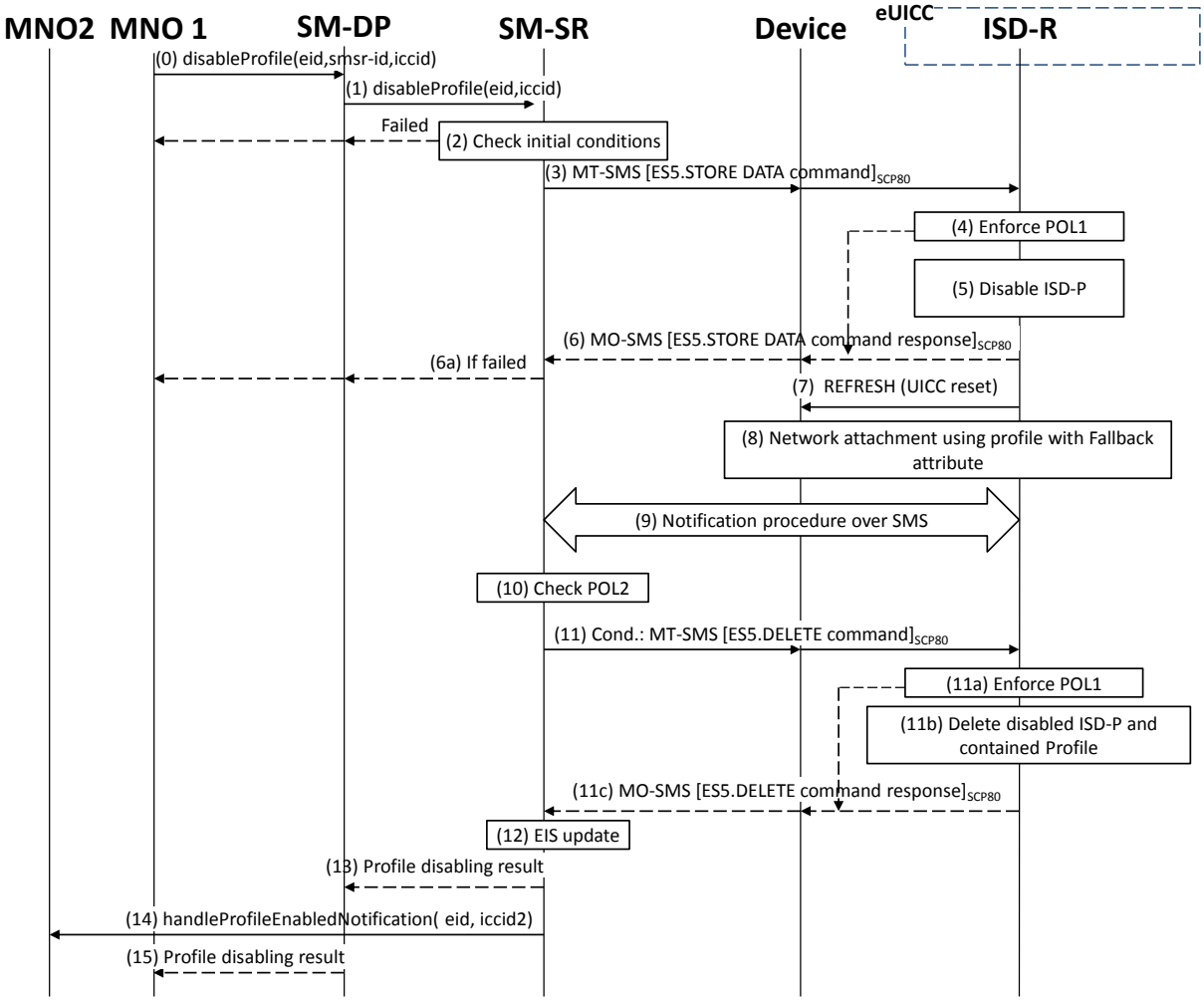


Figure 19: Profile Disabling Via SM-DP

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(0) MNO1, the owner of the target Profile, shall call the "**ES2.DisableProfile**" function with its relevant input data, see section 5.3.6, in particular the identification of the SM-SR in charge of the management of the target eUICC.

(1) The SM-DP shall forward the request to the SM-SR identified by the MNO and shall call the "**ES3.DisableProfile**" function with its relevant input data.

Steps (2) to (12) are the same as in the procedure "Profile Disabling" described in section 3.4.

(13) The SM-SR shall return the response to the "**ES3.DisableProfile"** function to the SM-DP, indicating that the Profile has been disabled. In case the Profile has also been deleted because of POL1 or POL2, the function execution response shall include an execution status "Executed-With Warning" indicating that the Profile has also been deleted.

(14) The SM-SR shall send the "**ES4.HandleProfileEnabledNotification**" to MNO2, the owner of the Profile with Fall-back Attribute Set that is now enabled.

(15) Finally, the SM-DP shall return the response to the "**ES2.DisableProfile**" function call to MNO1.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.4.

3.6 Profile and ISD-P Deletion

The Profile and ISD-P deletion procedure between the MNO and the SM-SR is used to delete the target ISD-P with its Profile on the eUICC (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 3.5.4). The procedure is initiated by the MNO owning the Profile to be deleted. The procedure illustrates the usage of SMS as a possible transport protocol between SM-SR and eUICC, but can be also performed using other transport protocols.

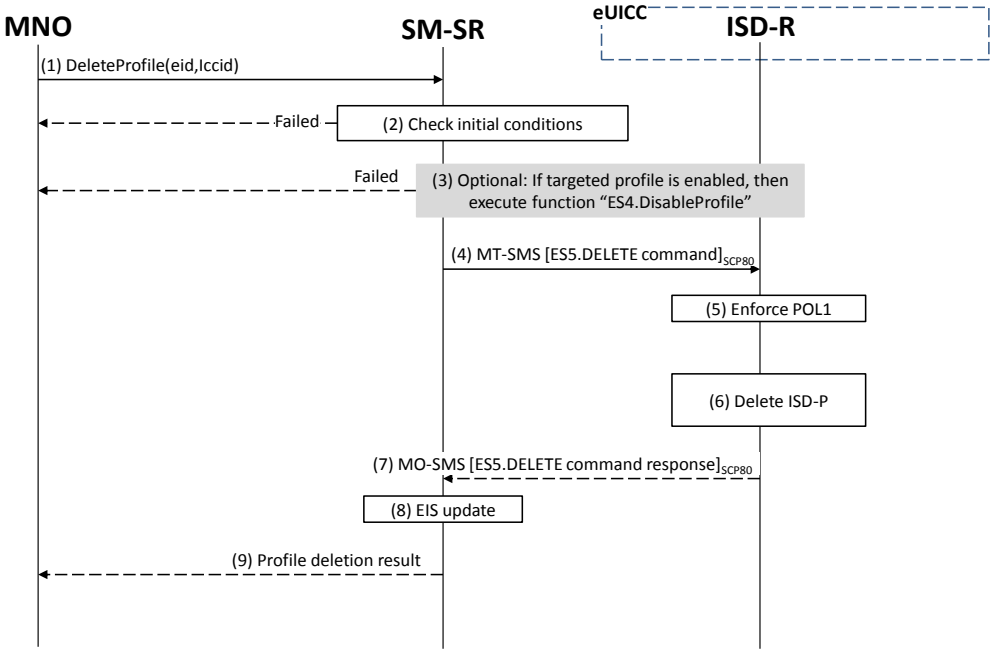


Figure 20: Profile and ISD-P Deletion

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) The MNO owning the target Profile shall call the "**ES4.DeleteProfile**" function with its relevant input data.

(2) The SM-SR shall verify that the MNO request is acceptable (the verifications that the SM-SR shall perform are described in the section 5.4.7), and in particular shall evaluate POL2 of the target Profile. If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating the failure, and the procedure shall end.

(3) The SM-SR shall check the state of the target Profile. If the target Profile is enabled and if POL2 of the target Profile allows it to be disabled, then the SM-SR shall execute the "**ES4.DisableProfile**" function to first disable the target Profile (and thus enable the

Profile having the Fall-back Attribute). In case of error, a response indicating the failure is returned to the MNO, and the procedure shall end.

NOTE: Profile change includes a change of the IMSI that is used to attach to the network. As indicated in 3GPP TS 31.102 [52], such a change requires special caution and should always be accompanied by a REFRESH command to avoid inconsistent information being read by the terminal. So while the targeted ISD-P is marked as enabled in this step, it may actually become effective only after the terminal executes the REFRESH command.

(4) The SM-SR shall send an MT-SMS containing the "**ES5.DELETE**" command with its relevant input data (see section 4.1.1.4) to the ISD-R. The SM-SR shall request a PoR to get the execution status of the "**ES5.DELETE**" command.

(5) The ISD-R, shall enforce POL1. If POL1 rejects deletion of the target Profile, the ISD-R shall return directly the MO-SMS containing the response indicating a failure, and the procedure shall end.

(6) If POL1 allows, the ISD-R shall delete the targeted ISD-P and the contained Profile.

(7) The ISD-R shall return the MO-SMS containing the execution status of the "**ES5.DELETE**" command to the SM-SR.

(8) In case of successful execution, the SM-SR shall update the EIS to reflect the newly deleted Profile.

(9) Finally, the SM-SR shall return the response to the "**ES4.DeleteProfile**" function to the caller MNO.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.4.

3.7 Profile and ISD-P Deletion Via SM-DP

The Profile and ISD-P deletion procedure is used between the MNO and the SM-DP to delete the target ISD-P with its Profile on the eUICC (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 3.5.4). The procedure is initiated by the MNO owning the target Profile and is actually performed by the SM-SR in charge of the eUICC management. The procedure illustrates the usage of SMS as a possible transport protocol between SM-SR and eUICC, but can be also performed using other transport protocols.

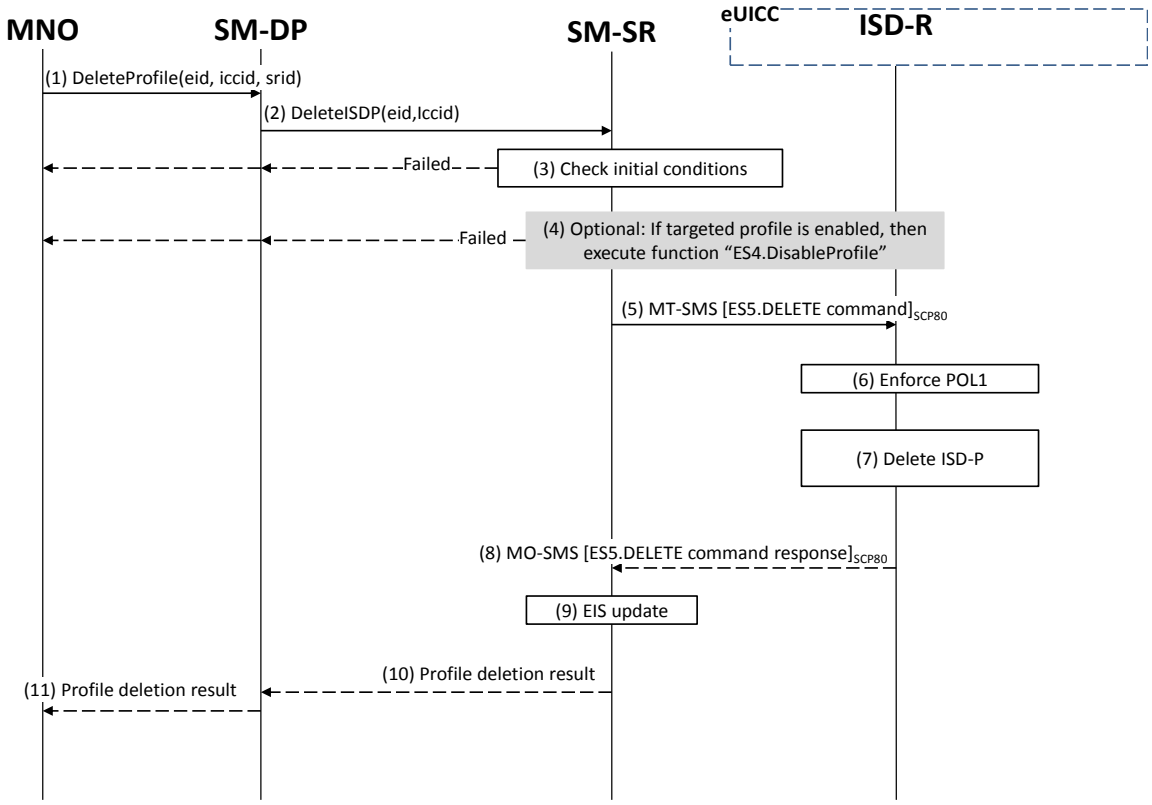


Figure 21: Profile and ISD-P Deletion via SM-DP

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) The MNO owning the Profile which is to be deleted shall call the "**ES2.DeleteProfile**" function with its relevant input data (we assume that the MNO knows the identification and the address of the SM-SR, as the MNO has a Profile on the eUICC managed by this SM-SR). The identification and address of the SM-SR in charge of the management of the eUICC shall be provided at that time to the SM-DP.

(2) The SM-DP shall forward the MNO request to the relevant SM-SR.

(3) The SM-SR shall verify if the SM-DP request is acceptable (the verifications that the SM-SR shall perform are described in section 5.4.9), and, in particular, shall evaluate POL2 of the target Profile. If any of the conditions to be verified are not satisfied, the SM-SR shall return a response indicating a failure, and the procedure shall end.

(4) The SM-SR shall check the state of the target Profile. If the target Profile is enabled and if POL2 of the target Profile allows it to be disabled, then the SM-SR shall execute the "**ES4.DisableProfile**" function to first disable the target Profile (and thus enable the Profile having the Fall-back Attribute). In case of error, a response indicating the failure shall be returned to the SM-DP, who forwards it to the MNO, and the procedure shall end.

NOTE: Profile change includes a change of the IMSI that is used to attach to the network. As indicated in 3GPP TS 31.102 [52], such a change requires special caution and should always be accompanied by a REFRESH

command to avoid inconsistent information being read by the terminal. So while the targeted ISD-P is marked as enabled in this step, it may actually become effective only after the terminal executes the REFRESH command.

(5) The SM-SR shall send an MT-SMS containing the "**ES5.DELETE**" command with its relevant input data (see section 4.1.1.4) to the ISD-R. The SM-SR shall request a PoR to get the execution status of the "**ES5.DELETE**" command.

(6) The ISD-R shall enforce POL1. If POL1 rejects the deletion of the target Profile, the ISD-R shall return the MO-SMS containing the response indicating a failure, and the procedure shall end.

(7) If POL1 allows its deletion, the ISD-R shall delete the targeted ISD-P and the contained Profile.

(8) The ISD-R shall return the MO-SMS to the SM-SR containing the execution status of the "**ES5.DELETE**" command.

(9) In case of successful execution, the SM-SR shall update the EIS to reflect the newly deleted Profile.

(10) The SM-SR shall return the response to the "**ES3.DeleteISDP**" function to the SM-DP.

(11) Finally, the SM-DP shall forward the received response to the MNO.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.4.

3.8 SM-SR Change

The SM-SR Change procedure is used between the Initiator (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 2.3.1) and the SM-SRs to change the SM-SR for the target eUICC (see GSMA Remote Provisioning Architecture for Embedded UICC [1] section 3.5.4) from SM-SR1 to SM-SR2.

This sequence uses the same key establishment mechanism as section 3.1.2.

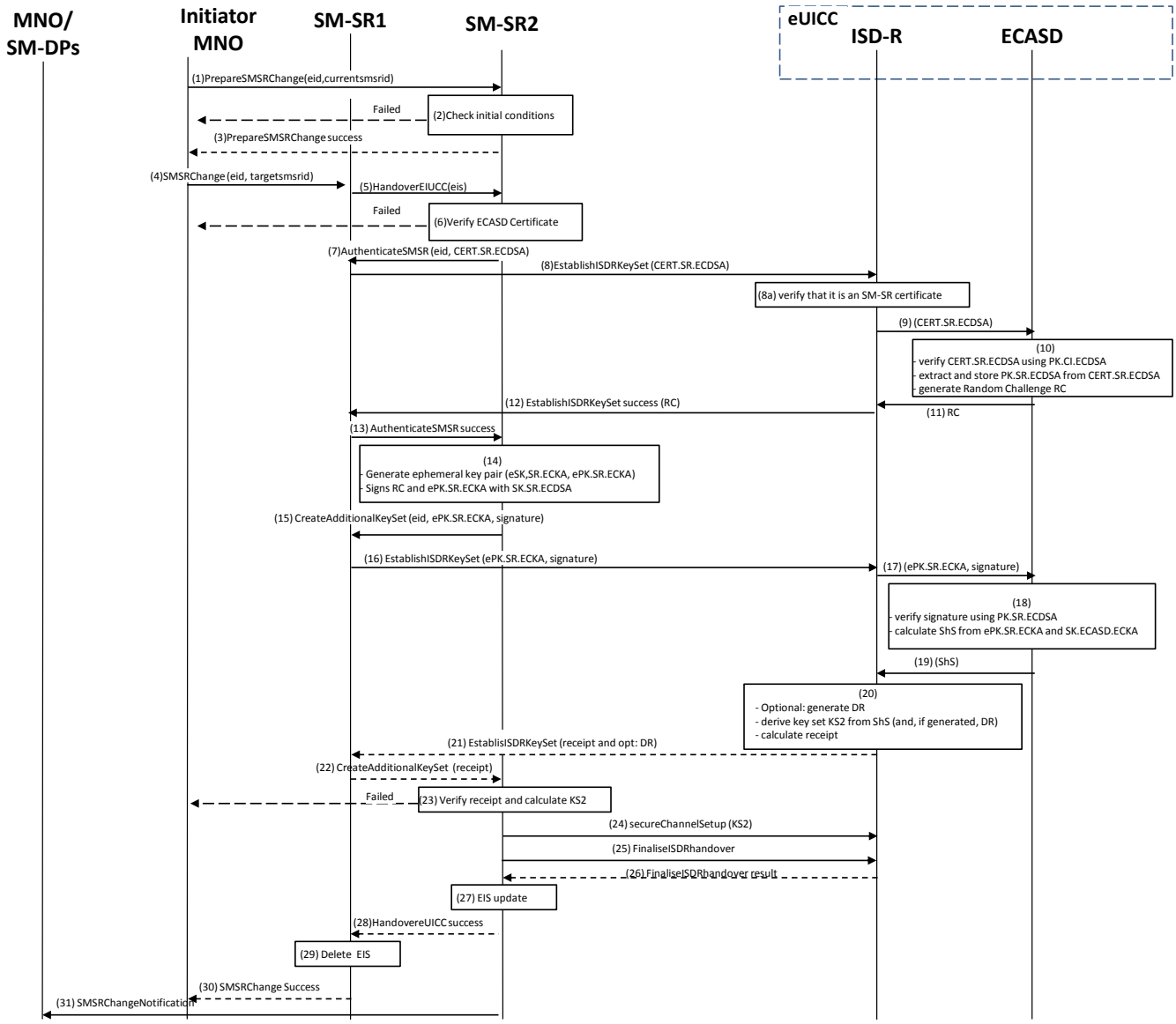


Figure 22: SM-SR Change

NOTE: The same ISD-R is used by both SM-SRs.

NOTE: The actions to perform in relationship to the MNO before the SM-SR change are out of scope of this specification.

NOTE: The settings of the secure connections between the MNOs and the SM-SRs are out of scope of this specification.

NOTE: The interaction between CI and SM-SR2 is out of the scope of this procedure.

Start Conditions:

a) The EID of the eUICC is known to the Initiator MNO.
a) The two SRIDs of the SM-SR1 and SM-SR2 are known to the Initiator MNO.
b) The ISD-R is personalised with the keys of SM-SR1.
c) The change of SM-SR is allowed.

d) SM-SR2 has a certificate signed by the Certificate Issuer (CI).
e) The public key of the CI is stored in the ECASD.

Procedure:

(1) The Initiator MNO shall call the "**ES4.PrepareSMSRChange**" function addressing the new SM-SR with the EID as input data.

(2) SM-SR2 shall verify that it is prepared to manage this eUICC. A failure at this step will stop the procedure and the error information shall be returned to the Initiator MNOs.

(3) SM-SR2 shall return a response indicating a success.

(4) The Initiator MNO shall call the SM-SR1 "**ES4.SMSRChange**" function with its relevant input. SM-SR1 shall verify that there is no pending action for the eUICC. SM-SR1 shall also reject any new management requests for the target eUICC as long as the procedure is on-going. In case of pending action(s), SM-SR1 shall perform all the pending actions and continue the procedure when these actions are completed.

(5) SM-SR1 shall call the SM-SR2 "**ES7.HandoverEUICC**" function with its relevant input data.

(6) SM-SR2 shall verify that:
- It has the capabilities to manage this eUICC.
- The ECASD certificate is valid, using the EUM Certificate and the CI's Root Certificate. The ECASD certificate is part of the EIS and is provided in the HandoverEUICC function. SM-SR2 shall extract PK.ECASD.ECKA from the ECASD certificate.

(7) SM-SR2 shall call the "**ES7.AuthenticateSMSR**" function specifying the targeted eUICC and providing the certificate identifying SM-SR2, CERT.SR.ECDSA.

(8) SM-SR1 shall call the "**ES5.EstablishISDRKeySet**" function with CERT.SR.ECDSA as input data to authenticate SM-SR2.

(8a) The ISD-R shall verify that it is an SM-SR certificate.

(9) The ISD-R shall forward the CERT.SR.ECDSA to ECASD.

(10) The ECASD shall:
- Verify CERT.SR.ECDSA using PK.CI.ECDSA;
- Extract and store PK.SR.ECDSA from CERT.SR.ECDSA;
- Generate a Random Challenge(RC).

(11) The ECASD shall return the Random Challenge (RC) to the ISD-R.

(12) The ISD-R shall return a response indicating a success with the generated Random Challenge RC.

(13) SM-SR1 receives and shall forward the response to SM-SR2.

(14) SM-SR2 shall generate an ephemeral key pair (eSK.SR.ECKA, ePK.SR.ECKA) and sign the received Random Challenge (RC) and ePK.SR.ECKA with SK.SR.ECDSA.

(15) SM-SR2 shall call the "**ES7.CreateAdditionalKeyset**" function specifying the targeted eUICC and providing the ePK.SR.ECKA and the previously generated signature.

(16) SM-SR1 shall call the "**ES5.EstablishISDRKeySet**" function with ePK.SR.ECKA and the signature as input data to request generation of an additional key set KS2.

(17) The ISD-R shall forward ePK.SR.ECKA and the signature to ECASD

(18) The ECASD shall:

- Verify the signature using PK.SR.ECDSA. If unsuccessful an error shall be returned; else
- Calculate ShS from ePK.SR.ECKA and SK.ECASD.ECKA.

(19) The ECASD shall return the ShS to the ISD-R

(20) The ISD-R shall:
- Optional: generates DR;
- Derive key set from ShS (and, if generated, DR);
- Calculate receipt.

(21) The ISD-R shall return a response indicating a success with the calculated receipt and optionally the DR.

(22) SM-SR1 receives and shall forward the response to SM-SR2.

(23) SM-SR2 shall:
- Calculate the ShS from eSK.SR.ECKA and PK.ECASD.ECKA,
- Derive key set KS2 from ShS (and optional DR), and
- Verify the receipt.

(24) SM-SR2 shall open a secure channel using the newly created key set KS2.

(25) SM-SR2 shall call the "**ES5.FinaliseISDRhandover**" to delete the keys of SM-SR.

(26) ISD-R shall return the result of the keys deletion to SM-SR2.

(27) SM-SR2 shall update the EIS to reflect that it now manages the eUICC.

(28) SM-SR2 shall return to SM-SR1 that it has successfully registered the eUICC.

(29) SM-SR1 shall remove the EIS for the target eUICC.

(30) SM-SR1 shall return the result of the SM-SR change function to the Initiator MNO.

(31) SM-SR2 shall call the "**ES4.HandleSMSRChangeNotification**" and "**ES3.HandleSMSRChangeNotification**" functions, with its relevant input data, to notify the MNOs (and the SM-DP s who represent the MNO), owning the Profiles on the eUICC.

The eUICC shall support key establishment with and without the DR. The SM-SR decides which option to use.

BSI TR-03111 [49] contains recommendations and requirements on the generation and validation of ephemeral keys. In addition, NIST SP 800-56A [50] provides requirements on the destruction of ephemeral keys and other intermediate secret data after their use.

End Conditions:

b) The ISD-R is personalised with only the key set of SM-SR2
f) The eUICC is registered within SM-SR2
g) The EIS and EID reside within SM-SR2
h) SM-SR1 is no longer related to the eUICC and its EIS record has been erased
i) The MNO(s) owning the Profile(s) are aware of the change
j) The Initiator MNO is aware of the SM-SR change

NOTE: Optionally, SM-SR2 may:

- Personalise other key sets to have the capability to use other secure channels

- Update the HTTPS parameters of the admin agent in the ISD-R, as specified in GlobalPlatform Card Specification Amendment B [8]

3.9 eUICC Registration at SM-SR: Register a New EIS

This procedure is used by the EUM to register an EIS representing an eUICC at the SM-SR. The EIS is required by the SM-SR to perform Platform Management functions and enable Profile Management functions on the eUICC.

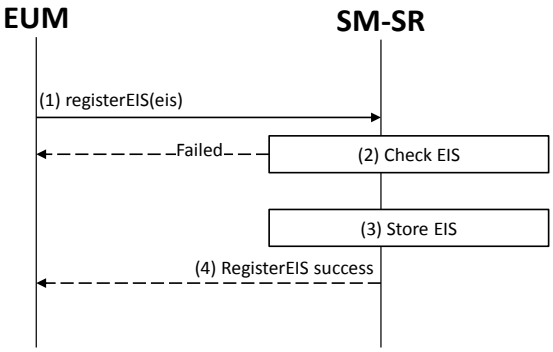


Figure 23: EIS Registration at SM-SR

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

It is assumed that the EUM has been given the SM-SR identity and address by the entity that has ordered the eUICC.

Procedure:

(1) The EUM that has manufactured the eUICC shall call the "**ES1.RegisterEIS**" function with the EIS data. The EIS shall include the data according to Annex E. The EIS shall be signed by the EUM.

(2) The SM-SR shall verify that the EUM request is acceptable (the verifications that the SM-SR shall perform are described in the section5.2.1). If any of the conditions to be verified is not satisfied, the SM-SR shall return a response indicating the failure, and the procedure shall end.

(3) The SM-SR shall store the new EIS in its database.

(4) The SM-SR shall return the successful response to the "**ES1.RegisterEIS**" function to the caller EUM.

3.10 Master Delete Procedure

This procedure deletes an Orphaned Profile regardless of the Profile's Policy Rules. The procedure illustrates the usage of SMS as a possible transport protocol between SM-SR and eUICC, but can be also performed using other transport protocols.

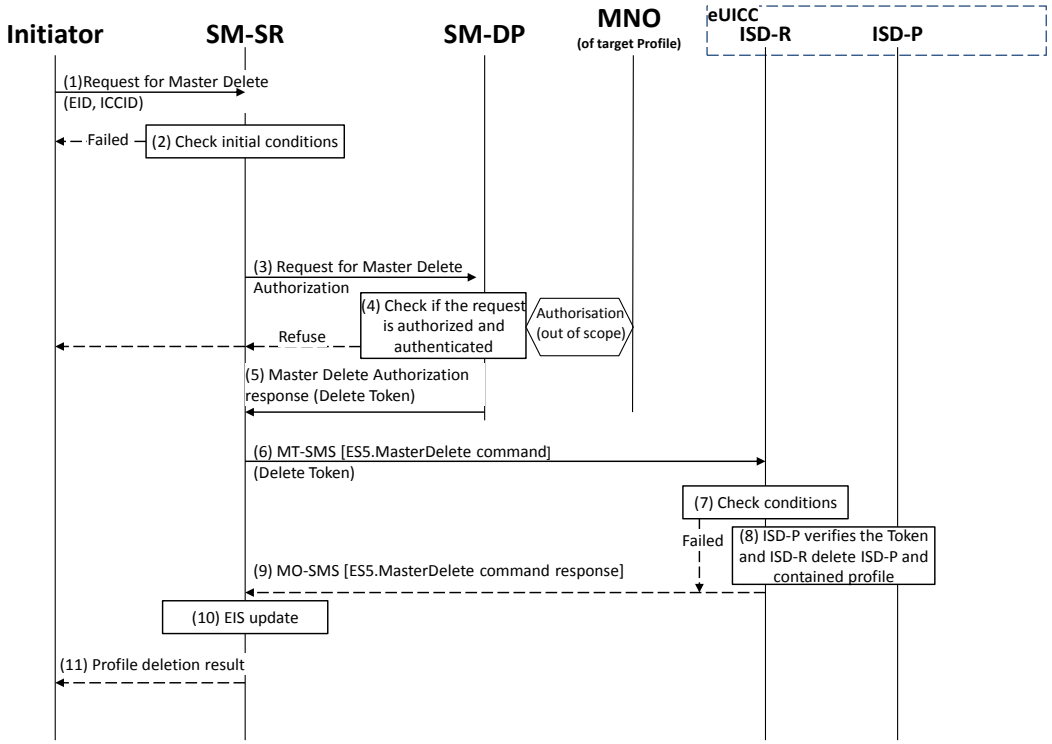


Figure 24: Master Delete

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1], plus:

c) The target Profile has been verified not to be the Profile which has the Fall-back Attribute Set.

Procedure:

(1) The Initiator shall send a Master Delete request to the SM-SR containing at least ICCID and EID (the function used in this step is not covered in this specification).

(2) The SM-SR shall verify that the request is acceptable (at least the preconditions are satisfied). If any of the verifications fails, the SM-SR shall return a response indicating the failure, and the procedure shall end.

(3) SM-SR shall send a request for Master Delete authorisation to the SM-DP, which is associated with the target Profile (the function used in this step is not covered in this specification).

(4) SM-DP shall verify that the request is authenticated and authorized.
The SM-DP also requests authorisation from the MNO owner of the target Profile.

NOTE: The definition of this interface is out of the scope of this document.

If the verification of the request from the SM-SR fails, or if the MNO does not give its authorisation, the SM-DP shall return that the deletion of target Profile is not allowed, and the procedure shall end.

(5) If deletion of the Profile is allowed, a delete token as defined in section 4.1.1.6, shall be returned to the SM-SR.

(6) The SM-SR shall send an MT-SMS containing the "**ES5.MasterDelete**" command with its relevant input data (see section 4.1.1.6) to the ISD-R. The SM-SR shall request a PoR to get the execution status of the "**ES5.MasterDelete**" command.

(7) The ISD-R shall execute the function as described in section 4.1.1.6. In case of an error, a response indicating the failure shall be returned (step 9) to the SM-SR and the procedure shall end.

(8) The ISD-P shall verify the token and if successful, the ISD-R shall delete the targeted ISD-P and the contained Profile.

(9) The ISD-R shall return the MO-SMS containing the execution status of the "**ES5.MasterDelete**" command to the SM-SR.

(10) In case of successful execution, the SM-SR shall update the EIS to reflect the deleted Profile.

(11) Finally, the SM-SR shall return the response to the Master Delete request to the Initiator (the function used in this step is not covered in this specification).

NOTE 1: The MT SMS and MO-SMS shall be secured according to section 2.4.

NOTE 2: The token shall be usable only once.

3.11 POL2 Update Via SM-DP

This procedure is used by the MNO to update POL2 via the SM-DP.

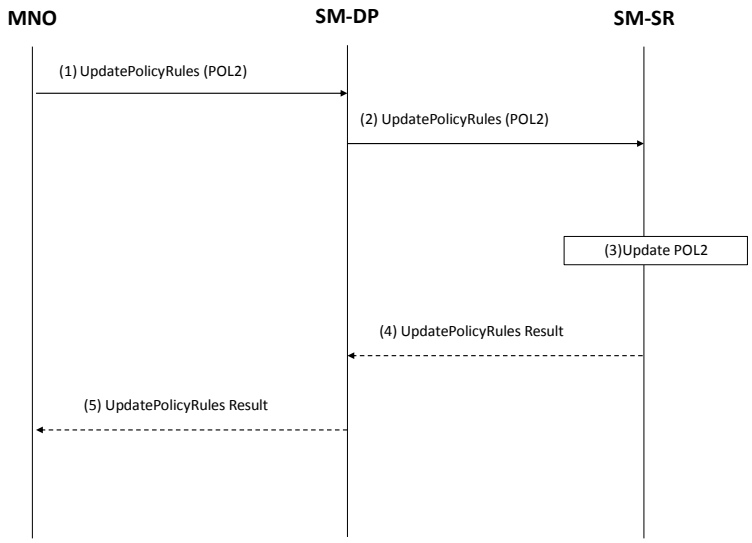


Figure 25: POL2 Update Via SM-DP

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) The MNO owner of the target Profile shall call the "**ES2.UpdatePolicyRules**" function with its relevant input data, as described in section 5.3.3, in particular the identification of the SM-SR in charge of the management of the target eUICC.

(2) The SM-DP shall forward the request to the SM-SR identified by the MNO and shall call the "**ES3.UpdatePolicyRules**" function with its relevant input data, as described in section 5.4.6

(3) The SM-SR shall update the POL2 of the targeted eUICC's EIS.

(4) The SM-SR shall return the execution status of the "**ES3.UpdatePolicyRules**" to the SM-DP.

(5) Finally, the SM-DP shall return the execution status of the "**ES2.UpdatePolicyRules**" command to the MNO.

3.12 POL1Update by MNO

This procedure allows the Update of POL1 by the MNO via the ES6 interface. For updating the POL1, the MNO shall use its OTA Keys hosted in the MNO-SD.

The procedure illustrates the usage of SMS as a possible transport protocol between the MNO and eUICC, but can be also performed using other transport protocols.

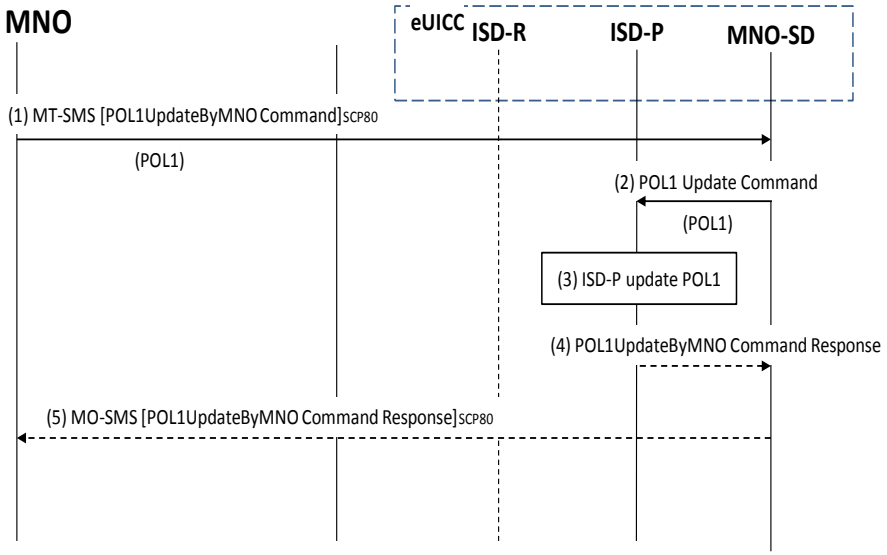


Figure 26: POL1 Update Via MNO

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1].

Procedure:

(1) The MNO owning the target Profile shall send a MT-SMS containing the "**ES6.UpdatePOL1byMNO**" function with its relevant input data (as described in section 4.1.2.1).

(2) The MNO-SD receives this request and shall transfer it to the ISD-P with POL1 as input data.

(3) The ISD-P shall process POL1 update of the target profile.

(4) The ISD-P shall return the execution status of the "**ES6.UpdatePOL1byMNO**" to MNO-SD.

(5) Finally, the MNO-SD shall return the MO-SMS containing the execution status of the "**ES6.UpdatePOL1byMNO**" command to the MNO.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.7.

3.13 Connectivity Parameters Update by MNO

This procedure allows the update of the Connectivity Parameters by the MNO on the ES6 interface.

The procedure illustrates the usage of SMS as a possible transport protocol between the MNO and eUICC, but can be also performed using other transport protocols.

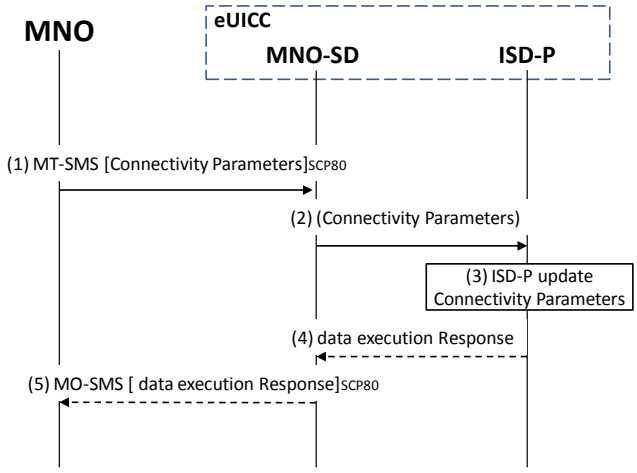


Figure 27: Connectivity Parameters Update by MNO

Start condition:

The MNO wants to update the Connectivity Parameters in their Profile

Procedure:

(1) The MNO owning the target Profile shall send a MT-SMS containing the Connectivity Parameters to the MNO-SD.

(2) The MNO-SD shall transfer the Connectivity Parameters to the ISD-P.

(3) The ISD-P shall update the Connectivity Parameters.

(4) The ISD-P shall return the execution status to the MNO-SD.

(5) The MNO-SD shall send the MO-SMS containing the execution status to the MNO.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.7.

3.14 Connectivity Parameters Update Using SCP03

This procedure allows the update of the Connectivity Parameters using SCP03by the SM-DP on the ES8 interface.

The procedure illustrates the usage of SMS as a possible transport protocol between the SM-SR and eUICC, but can be also performed using other transport protocols.

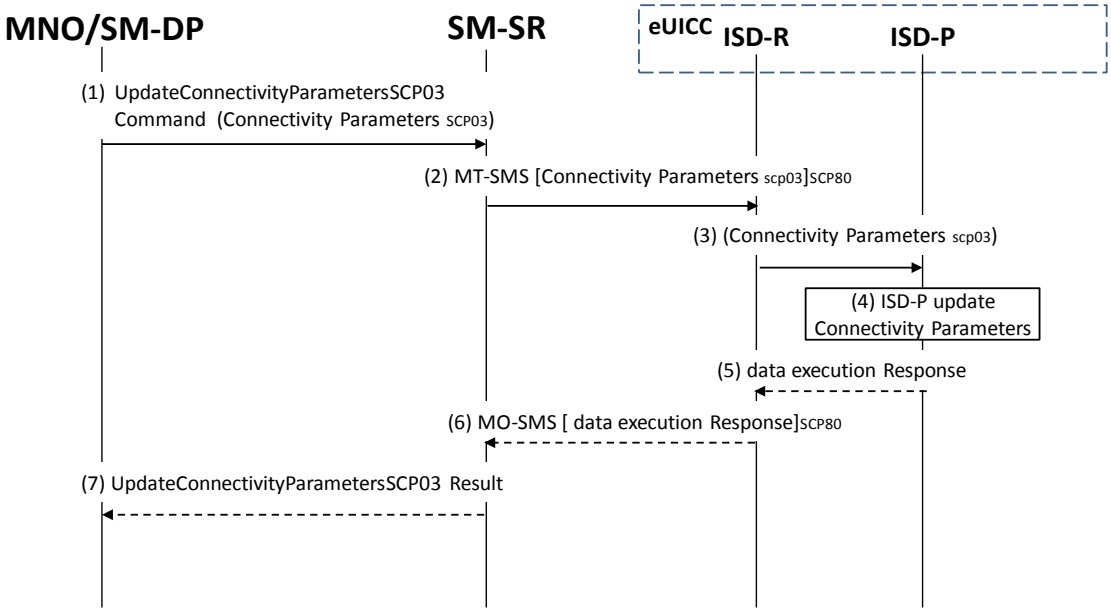


Figure 28: Connectivity Parameters Update Using SCP03

The start conditions are described in [1].

(1) The MNO, or the SM-DP, on behalf of the MNO owning the target Profile, shall send a request containing "**ES3.UpdateConnectivityParameters**" function with its relevant input data (as described in section 5.4.6). The <data> parameter shall contain an SCP03 script as defined in section 4.1.3.2 including the command "**ES8.UpdateConnectivityParametersSCP03**"

(2) The SM-SR shall send a ciphered MT-SMS containing the ciphered data provided by the SM-DP.

(3) The ISD-R shall transfer the <data> to the ISD-P.

(4) The ISD-P shall update the Connectivity Parameters.

(5) The ISD-P shall return the execution status of the "**ES3.UpdateConnectivityParameters**" to ISD-R.

(6) The ISD-R shall send the ciphered MO-SMS containing the execution status of the "**ES3.UpdateConnectivityParameters**" command to the SM-SR.

(7) Finally, the SM-SR shall return the execution status of the "**ES3.UpdateConnectivityParameters**" command to the SM-DP.

NOTE: The MT-SMS and MO-SMS shall be secured according to section 2.7

3.15 Default Notification Procedure

This section provides a default notification procedure from the eUICC to the SM-SR. This default notification carries information about the eUICC and the Device.

This notification is initiated by the eUICC in some conditions:

- First network attachment of the Device: this indicates to the SM-SR in charge of managing of the eUICC that the eUICC has been deployed on the field. The notification of “First network attachment” happens only once in the eUICC’s lifetime. It is triggered when the eUICC is network attached the very first time. Nevertheless, note that this notification will be retried until the effective reception by the SM-SR, including further network attachments if not succeeded during the first network attachment session.
- After an explicit new Profile Enabling request: this indicates to the SM-SR which is the Profile which is currently enabled. This notification happens right after the network attachment:
  - With the newly Enabled Profile in case of successful attachment
  - Or with the previously Enabled Profile or with the Profile having the Fall-back Attribute, after the attachment with the requested Profile has failed.
- After activation of the Fall-back Mechanism: this indicates to the SM-SR that the Profile with the Fall-back Attribute has been enabled. This notification happens right after the network attachment.

The notification may happen either on SMS, CAT_TP or HTTPS. The content of the notification message is the same whatever protocol is used. The eUICC is free to select the most relevant protocol according to the Device’s capabilities.

The notification has to be confirmed by the SM-SR. The confirmation will depend on the protocol used for notification.

On reception of the SM-SR notification confirmation, the eUICC may perform any operation as specified in one of the procedures including the notification sequence (like for instance deletion of an ISD-P after its disabling, see section 3.6). After the eUICC has performed the follow-up activities, the eUICC shall respond to the SM-SR notification confirmation function, including the identification of the operation performed if any.

3.15.1 Notification Using SMS

This figure describes the notification sequence over SMS. It is applicable either for first “power on” of the Device, or the enabling of a Profile (after explicit request or Fall-back Mechanism).

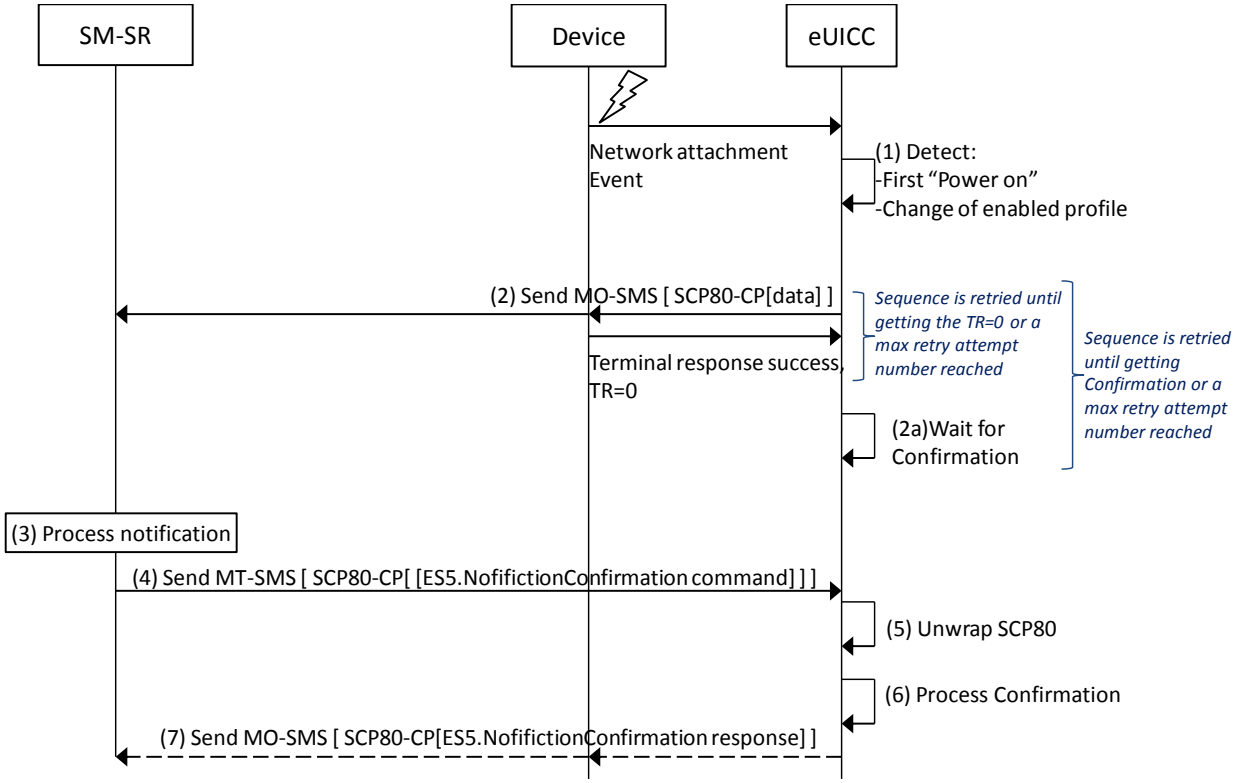


Figure 29: Sequence Flow of Notification Over SMS

(1) At the end of the start-up sequence, the eUICC detects a first "power on" or a situation where the Enabled Profile has changed compared to the previous eUICC reset.

(2) The eUICC sends an MO-SMS envelop. The SMS contains a secure SCP 80 Command Packet (MO-SMS shall be formatted as defined in section 2.4.3 with security set to cryptographic checksum and no ciphering, the counter value of the Command Packet shall be set to '0000000000' and SPI set to "No counter available") using the SCP80 keys of the ISD-R, and containing the notification data structure described in section 4.1.1.11. The secured data shall be coded as described in section 4.1.1.11.

The eUICC shall use the network information of the Enabled Profile.

NOTE: This deviates from the typical secured packets generation defined in ETSI TS 102 225 [4].

The eUICC shall retry sending until getting a successful response of the Device ('0X'). Note, that the eUICC shall implement a mechanism to avoid attempting an infinite number of retries. Finally the eUICC shall use another protocol in case of final failure for sending the notification using SMS.

(2a) The eUICC shall wait for the SM-SR confirmation. If no confirmation is received by the eUICC after a certain amount of time (dependent on the configuration), eUICC shall restart from step (2) (with the same sequence number).

(3) The SM-SR processes the notification.

(4) The SM-SR sends an MT-SMS containing the "**ES5. HandleNotificationConfirmation**" command defined in section 4.1.1.12 in a SCP80 command packet. This MT-SMS shall target the entity on the eUICC that has sent the notification.

(5) The ISD-R un-wraps the SCP80 security layer

(6) The eUICC processes the notification confirmation data; this may include follow-up activities as required by the procedure where this sequence is used.

(7) The eUICC shall return the MO-SMS containing the response of the "**ES5.HandleNotificationConfirmation**" response. The MO-SMS shall be secured according to section 2.4.3. The eUICC shall retry sending until getting a successful response of the Device ('0X'). Note, that the eUICC shall implement a mechanism to avoid attempting an infinite number of retries.

3.15.2 Notification Using HTTPS

This figure describes the notification sequence over HTTPS. It is applicable either for first "power on" of the Device, or the enabling of a Profile (after explicit request or Fall-back Mechanism).

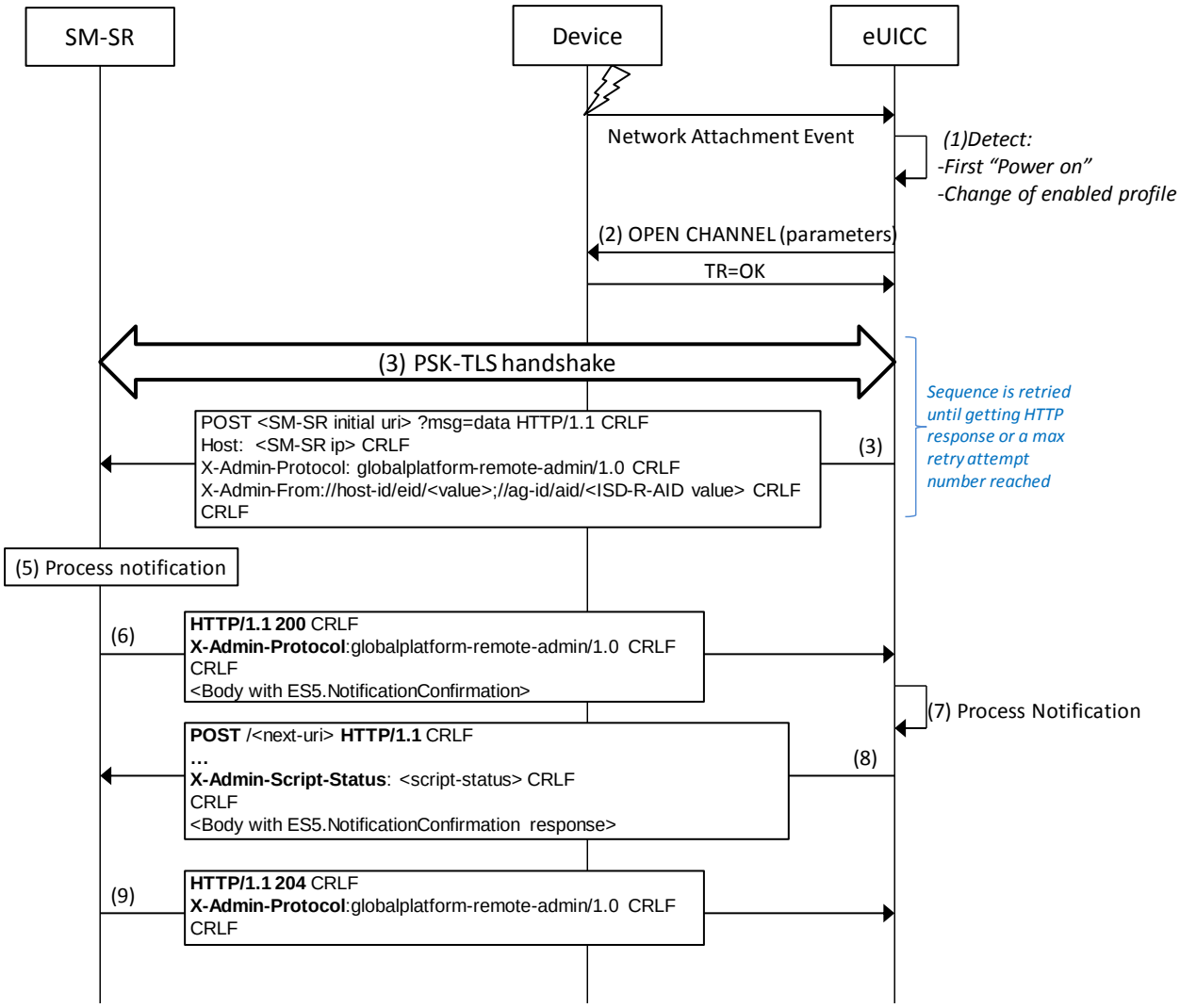


Figure 30: Sequence Flow of Notification Over HTTPS

(1) At the end of the start-up sequence, the eUICC detects a first "power on" or a situation where the Enabled Profile has changed compared to the last eUICC reset.

(2) The eUICC opens a BIP channel with the relevant parameters to address the SM-SR. This includes having access to the Network Access Name, User Login and User Password of the Enabled Profile.

(3) The ISD-R of the eUICC negotiates the PSK-TLS handshake with the SM-SR. The TLS session shall be opened as defined in section 2.4.3. The ISD-R shall apply the retry Policy as defined in GlobalPlatform Card Specification Amendment B [8].

(4) The eUICC sends the first HTTP POST. The notification contains the SM-SR URL with the special query parameter "?msg" containing the data for eUICC notification defined in section 4.1.1.11. The data of the notification shall be coded as hexadecimal string (see section 5.1.1.1) with no spaces.

(5) ISD-R shall apply the retry Policy as defined in GlobalPlatform Card Specification Amendment B [8].

(6) The SM-SR processes the notification

(7) The SM-SR shall return an HTTP response with a body containing the "**ES5.ConfirmationNotification**" command acknowledging the reception of the notification.

(8) The eUICC processes the notification confirmation; this may include follow-up activities as required by the procedure where this sequence is used.

(9) The eUICC shall return the execution response of the "**ES5.ConfirmationNotification**" command within a new HTTP POST request addressed to the SM-SR.

(10) The SM-SR shall return an HTTP response "204 No content".

3.16 Fall-Back Activation Procedure

The Fall-back Mechanism shall be activated in case of loss of network connectivity by the current Enabled Profile. The eUICC shall disable the current Enabled Profile and enable the Profile with Fall-back Attribute set.

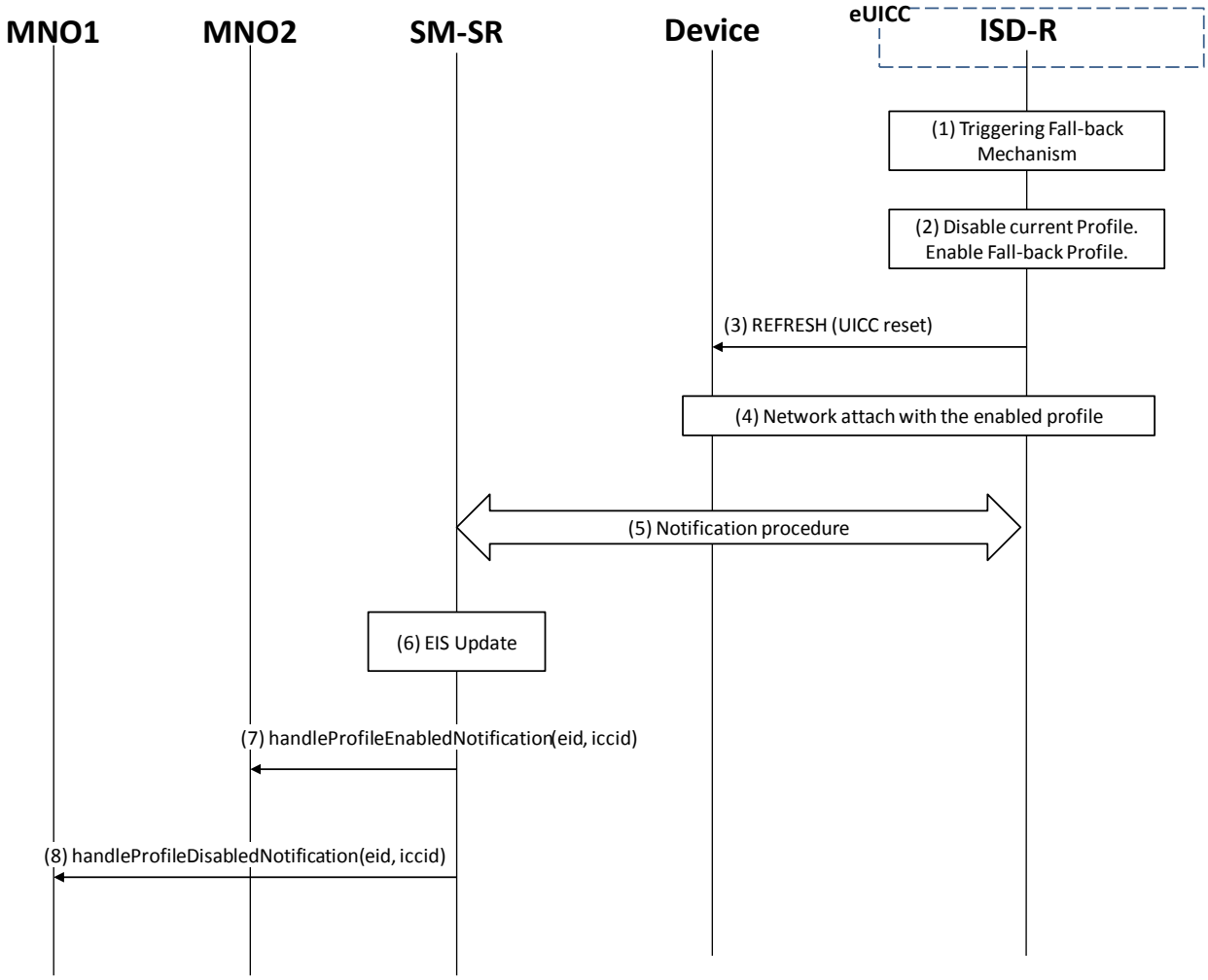


Figure 31: Fall-Back Activation Procedure

Start Conditions:

The start conditions are described in GSMA Remote Provisioning Architecture for the Embedded UICC [1]

The profile with Fall-back Attribute set has already been installed and is in disabled state.

Procedure:

(1) The Fall-back Mechanism is triggered in accordance with the Start Conditions.

(2) Ignoring POL1 of the Enabled Profile, the ISD-R shall disable the currently Enabled Profile and shall enable the Profile with the Fall-back Attribute set.

(3) The ISD-R shall request the Device to perform the toolkit REFRESH command in UICC Reset mode. This will trigger the execution of the network attach procedure.

(4) The eUICC and the Device shall perform a new network attach procedure.

(5) The eUICC shall perform the notification procedure as described in section 4.1.1.11. The ISD-R shall ensure that all supported Default Notification mechanism will be used to inform SM-SR about the Fall-back occurrence. After having exhausted all possible retries to inform the SM-SR, the eUICC shall stay in this state and continue trying to notify the SM-SR. On reception of the SMS notification, the SM-SR is informed that the Fall-back mechanism was triggered and the last Enabled Profile has been disabled.

(6) The SM-SR shall update the EIS to reflect that:
- The Profile having the Fall-back Attribute has been enabled
- The previously Enabled Profile has been disabled

(7) The SM-SR shall send the "**ES4.HandleProfileEnabledNotification**" to MNO2, the owner of Profile with Fall-back Attribute Set that is now enabled. In case MNO2 has no direct connection with the SM-SR (SM-SR shall be able to detect such situation based on its own database), the SM-SR shall send this notification to the SM-DP that acts on behalf of MNO2 by calling the "**ES3.HandleProfileEnabledNotification**". The SM-SR can retrieve the SM-DP identity based on the EIS content. Then the SM-DP, on reception of this notification, shall forward it to MNO2 by calling the "**ES2.HandleProfileEnabledNotification**".

(8) The SM-SR shall send the "**ES4.HandleProfileDisabledNotification**" to MNO1, the owner of the Profile that was enabled at the beginning of the procedure. In case MNO1 has no direct connection with the SM-SR (SM-SR shall be able to detect such a situation based on its own database), the SM-SR shall send this notification to the SM-DP that acts on behalf of MNO1 by calling the "**ES3.HandleProfileDisabledNotification**". The SM-SR can retrieve the SM-DP identity based on the EIS content. Then the SM-DP, on reception of this notification, shall forward it to MNO1 by calling the "**ES2.HandleProfileDisabledNotification**".

If the previously Enabled Profile has the POL1 rule "disable not allowed" set, then the eUICCcan only switch back to this Profile until the POL1 of this Profile is changed. As long as POL1 is not changed, this Profile can only be deleted by Master Delete function.

NOTE: A mechanism may permit to request to the eUICC to switch back to the previously enabled profile once the network connectivity has been restored. The technical solution of the mechanism mentioned above is out of scope.

4 eUICC Interface Descriptions

This section contains the technical descriptions of those interfaces within the Remote Provisioning and Management system involving the eUICC directly, including the following:

- ES5, interface between SM-SR and the eUICC.
- ES6, interface between the MNO and the eUICC
- ES8, interface between SM-DP and the eUICC.

The following table presents the normative list of all the functions that are defined in this section.

Request-response functions:

Interface	Function group	Functions	Function provider entity
ES5	Platform Management	CreateISDP	ISD-R
		EnableProfile	
		DisableProfile	
		DeleteProfile	
		eUICCCapabilityAudit	
		MasterDelete	
		SetFallbackAttribute	
	eUICC Management	EstablishISDRKeySet	
		FinaliseISDRhandover	
		UpdateSMSRAddressingParameters	
ES6	Profile Management	UpdatePOL1byMNO	MNO-SD
		UpdateConnectivityParametersByMNO	
ES8	Profile Management	DownloadAndInstallation	ISD-P
		EstablishISDPKeySet	
		UpdateConnectivityParameters SCP03	

Table 5:: Request Response Functions

Notification handler functions:

Interface	Function group	Notification handler functions	Function provider Role
ES5	Platform Management	`HandleDefaultNotification` `HandleNotificationConfirmation`	SM-SR

Table 6: Notification Handler Functions

4.1 Functions Description

NOTE: If any command, such as Profile Enabling, Profile Disabling, and Profile Download and Installation does not complete successfully, the eUICC shall maintain the state it was in before it received the command.

4.1.1 ES5 (SM-SR–eUICC) Interface Description

4.1.1.1 ISD-P Creation

Function name: CreateISDP

Related Procedures: ISD-P creation

Function group: Platform Management

Function Provider entity: ISD-R

Description: This function creates an ISD-P on the eUICC.

Parameters:

- ISD-P-AID
- Memory quota for the ISD-P (optional)

Prerequisite:

- The SM-SR has assigned an ISD-P-AID.

Command Description:

INSTALL COMMAND

The command is an Install command as defined in GlobalPlatform Card Specification [6].

The following tables describe the installation command and the specific parameters within the data field:

Code	Value	Meaning
CLA	‘80’	See GlobalPlatform Card Specification section 11.1
INS	‘E6’	INSTALL
P1	‘0C’	See GlobalPlatform Card Specification [6] section 11.5.2.1
P2	‘00’	See GlobalPlatform Card Specification [6] section 11.5.2.2
Lc	‘xx’	Data Field Length

Code	Value	Meaning
Data	'xxxx…'	See GlobalPlatform Card Specification [6] section 11.5.2.3
Le	'00'	

Table 7: INSTALL Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	0	0	1	0	0	0	For make selectable
0	0	0	0	0	1	0	0	For Install

Table 8: INSTALL Reference Control Parameter P1

Reference Control Parameter P2 – ISD-P State Coding

P2 is set to '00'; according to GlobalPlatform Card Specification [6] section 11.5, this means no information provided.

Data Field

Name	Length	Value Description	MOC
Length of Executable Load File AID	'1'	'05' - '10'	M
Executable Load File AID	'05' – '10'	'xxxx'	M
Length of Executable Module AID	'1'	'05' - '10'	M
Executable Module AID	'05' – '10'	'xxxx'	M
Length of Application AID	'1'	'05' - '10'	M
Application AID (ISD-P-AID)	'05' – '10'	'xxxx'	M
Length of Privileges	'1'	'01' - '03'	M
Privileges	'1'or '3'	See [6] section 11.1.2	M
Length of Install Parameters field	'2'-'n'		M
Install Parameters field	'0-n'	See [6] section 11.5.2.3.7	M
Length of Install Token	'1'	00	M

Table 9: INSTALL Command Data Field

Privileges

Privileges granted to the ISD-P, as specified in Annex C, shall be at least:

- Security Domain
- Trusted Path
- Authorized Management

Install Parameters

Tag	Length	Value Description	MOC
'C9'	'1-n'	Application Specific Parameters: see GlobalPlatform Card Specification [6] section 11.5.3.2.	M

		Tag	Length	Value Description	MOC
		'81'	2	Secure Channel Protocol Identifier and Implementation Option "i"	M (n occurrences)
'EF'	'1-n'	System Specific Parameters			O
		Tag	Length	Value Description	MOC
		'83'	'2' or '4'	Cumulative Granted Non Volatile Memory	O

Table 10: INSTALL Parameters

Data *Returned*

None

***Response* Message**

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] section 11.5.3.1.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.5.3.2.

4.1.1.2 Profile Enabling

Function name: EnableProfile

Related Procedures: Profile Enabling

Function group: Platform Management

Function Provider entity: ISD-R

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

Prerequisites:

- SM-SR shall check that POL2 of both the currently Enabled Profile and the target Profile allow this action.

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].

Code	Value	Meaning
CLA	‘80’	
INS	‘E2’	STORE DATA
P1	‘88’	Reference Control Parameter P1 0
P2	‘00’	Block Number (Not used for Enable command)
Lc	‘XX’	Length of data field
Data	‘XX’	Application Data and MAC (if present)
Le	Not present	

Table 11: STORE DATA COMMAND Message

Parameter P1 is coded according to the following table:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 12: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A03'	Var	Enable Profile			M
		Tag	Length	Value Description	MOC
		'4F'	5-16	ISD-P-AID	M

Table 13: Enable Attribute Data Field

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.11.3.2.

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.

4.1.1.3 Profile Disabling

Function name: DisableProfile

Related Procedures: Profile Disabling

Function group: Platform Management

Function Provider entity: ISD-R

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

Prerequisites:

- SM-SR has checked that POL2 allows this action

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].

Code	Value	Meaning
CLA	‘80’	
INS	‘E2’	STORE DATA
P1	'88'	Reference control parameter P1
P2	‘00’	Block Number (Not used for Enable command)
Lc	‘xx’	Length of data field
Data	‘xx’	Application Data and MAC (if present)
Le	Not present	

Table 14: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 15: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A04'	Var	Disable Profile			M
		Tag	**Length**	**Value Description**	**MOC**
		'4F'	5-16	ISD-P-AID	M

Table 16: Disable Attribute Data Field

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.11.3.2.

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.

4.1.1.4 Profile Deletion

Function name: DeleteProfile

Related Procedures: Profile and ISD-P deletion, Profile and ISD-P deletion via SM-DP

Function group: Platform Management

Function Provider entity: ISD-R

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

Prerequisites:

- SM-SR shall check that POL2 allows this action
- The target Profile shall not be the Profile with the Fall-back Attribute set

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 realised through the GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification Amendment C [9].

Command Message

The DELETE command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' - '8F', 'C0' - 'CF' or 'E0' - 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E4'	DELETE
P1	'00'	Reference control parameter P1
P2	'40'	Reference control parameter P2
Lc	'xx'	Length of data field
Data	'xxxx…'	TLV coded objects (and MAC if present)
Le	'00'	

Table 17: DELETE Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	-	-	-	-	-	-	-	Last (or only) command
-	X	X	X	X	X	X	X	RFU

Table 18: DELETE Reference Control Parameter P1

Reference Control Parameter P2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	1	–	–	–	–	–	–	Delete a root security domain and all associated Applications
–	–	X	X	X	X	X	X	RFU

Table 19: DELETE Command Reference Control Parameter P2

Data Field Sent in the Command Message

The data field of the DELETE command message shall contain the TLV coded name(s) of the object to be deleted.

Tag	Length	Value Description	MOC
'4F'	5-16	ISD-P-AID	M

Table 20: DELETE [card content] Command Data Field

Response Message

Data Field Returned in the Response Message:

A single byte of '00' shall be returned indicating that no additional data is present.

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.2.3.2.

Specific Processing State returned in response Message:

’69 85’: Profile is in Enable State or Profile has the Fall-back Attribute.

’69 E1’: POL1 of the Profile prevents deletion.

4.1.1.5 eUICC Capability Audit

Function name: eUICCCapabilityAudit

Related Procedures: -

NOTE: This function is not present in any procedure, however, may be used and requested at any point of time by the Profile owner or SM-SR.

Function group: Platform Management

Function Provider entity: ISD-R

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.

GET DATA

This function can return:

- Number of installed ISD-P and available not allocated memory
- ECASD Certificate

GET STATUS

This function can return:

- Each ISD-P-AID
- State of the ISD-Ps / Profiles

Prerequisites:

- None

Commands Description:

GET DATA

The GET DATA command is coded according to the following table:

Code	Value	Meaning
CLA	'80'	See GlobalPlatform Card Specification section 11.1.4.1
INS	'CA'	GET DATA
P1	'xx'	See below
P2	'xx'	See below
Lc	'xx'	Not present if no command data, otherwise length of data field
Data	'xxxx...'	Not present, or command data
Le	'00'	

Table 21: GET DATA Command Message

Parameter P1 and P2

The P1 and P2 parameters define the tag of the data object to be read.

Tag ‘FF 21’: Extended Card Resources Information available for Card Content Management, as defined in ETSI TS 102 226 [5].

Tag ‘BF 30’: Forwarded CASD Data mechanism as defined in GlobalPlatform Card Specification Amendment C [9].

This mechanism allows to retrieves ECASD data through the ISD-R.

Data field

If the P1 and P2 parameters are set to ‘BF 30’, the data field shall include one (and only one) of the following requests:

- ECASD recognition data: ‘5C 01 66’
- ECASD Certificate Store (containing ECASD Public Key Certificates): ‘5C 02 7F 21’

Response Message

If certificate data is requested, the certificate shall be returned TLV-coded as follows:

Name	Length	Value Description	MOC
Forwarded CASD Data tag	2	‘BF 30’	C
Length of the response	1, 2 or 3	'00' - '7F', or '81 80' - '81 FF' or '82 01 00' - '82 FF FF'	C
Certificate store tag	2	‘7F 21‘	M
Length of the certificate	1, 2 or 3	'00' - '7F', or '81 80' - '81 FF' or '82 01 00' - '82 FF FF'	M
Certificate data	n	‘xxxx…’	M

Table 22: GET DATA Command Data Field

Certificate Data

The following table describes the certificate data which will be returned by the eUICC Capability Audit command.

Tag	Length	Value Description			MOC
'7F21'	Var.	Certificate			M
		Tag	**Length**	**Value Description**	**MOC**
		'93'	1-16	Certificate Serial Number	M
		'42'	1-16	CA Identifier	M
		'5F20'	1-16	Subject Identifier (eUICC Identifier)	M

		'95'	2	Key Usage '0080': Key Agreement	M
		'5F25'	4	Effective Date (YYYYMMDD, BCD format) – optional	M
		'5F24'	4	Expiration Date (YYYYMMDD, BCD format)	M
		'45'	1-16	ECASD Image Number	M
		'73'	Var.	Discretionary Data: 'C0' var eUICC Supplier Identifier 'C1' var eUICC Product Line Identifier 'C2' var eUICC Extended GSMA SAS Accreditation Serial Number	M
		'7F49'	Var.	Public Key	M
		'5F37'	Var.	Signature (to be computed as described in GlobalPlatform Card Specification Amendment E [11] and the signature shall include all the field starting from tag '93' to tag '7F49')	M

Table 23: Certificate Data Field

Public Key Data Object

The public key data object contains an elliptic curves (EC) public key and the corresponding domain parameters.

Tag	Length	Value Description			MOC
'7F49'	Var.	Public Key Data Object			M
		Tag	**Length**	**Value Description**	**MOC**
		'B0'	Var	Public key – Q	M
		'F0'	'01'	Key Parameter Reference '00': NIST P-256 '03': brainpoolP256r1 '40': FRP256V1 [51]	M

Table 24: Public Key Data Object Data Field

An ECASD shall have at least one set of elliptic curve parameters preloaded (see GlobalPlatform Card Specification Amendment E [11]) as defined in the table above.

GET STATUS

The GET STATUS command is coded according to the following table:

Code	Value	Meaning
CLA	'80'	See GlobalPlatform Card Specification [6] section 11.1.4.1
INS	'F2'	GET STATUS
P1	'40'	See GlobalPlatform Card Specification [6] section 11.4.2.1.
P2	'xx'	See GlobalPlatform Card Specification [6] section 11.4.2..2.

Code	Value	Meaning
Lc	'xx'	Length of the data field
Data	'4F xx …'	See GlobalPlatform Card Specification [6] section 11.4.2.3.
Le	'00'	

Table 25: GET STATUS Command Message

Parameter P1

The following value will be used for P1:

'40' – Applications and Supplementary Security Domains only

Parameter P2

The parameter P2 controls the number of consecutive GET STATUS commands.

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
X	X	X	X	X	X	-	-	RFU
-	-	-	-	-	-	1	-	Response Data Structure according to table 11-36 of GlobalPlatform Card Specification [6].
-	-	-	-	-	-	-	0	Get first or all occurrence(s)
-	-	-	-	-	-	-	1	Get next occurrence(s)

Table 26: GET STATUS Command Reference Control Parameter P2

Data field sent in the Command Message

The GET STATUS command message data field shall contain at least one TLV coded search qualifier: the AID (tag '4F'). It shall be possible to search for all the occurrences that match the selection criteria according to the reference control parameter P1 using a search criteria of '4F 00'.

The search is limited to the ISD-P instances.

The following other search criteria shall be supported: Life Cycle State (tag '9F70') and ISD-P Attributes (tag '53').

The tag list (tag '5C') indicates to the UICC how to construct the response data for each eUICC entity matching the search criteria.

The data field is structured as follows:

Tag	Length	Value Description	MOC
'4F'	0-16	Application AID	M
'xx' or 'xxxx'	0-n	Other search criteria	O
…	…	…	…
'5C'	1-n	Tag list	M

Table 27: GET STATUS Command Data Field

Response Message

Data Field Returned in the Response Message:

The tag list (tag '5C') identifies the extended information for ISD-P. The coding of the response message is defined as followed:

Tag	Length	Name			MOC
'E3'	Variable	GlobalPlatform Registry related data			M
		Tag	Length	Name	MOC
		'4F'	5-16	AID	M
		'9F70'	1	Life Cycle State	C
		'53'	1	ISD-P Attributes (see Table 29)	C

Table 28: GET STATUS Command Data Field Return

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
X	X	X	X	X	X	X	-	RFU
-	-	-	-	-	-	-	0	Fall-back Attribute not set
-	-	-	-	-	-	-	1	Fall-back Attribute set

Table 29: ISD-P Attributes

ISD-P State Coding

The life cycle of the ISD-P is coded as define in section 2.2.1.3.

Processing State returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.3.3.2.

4.1.1.6 Master Delete

Function name: MasterDelete

Related Procedures: Master Delete Procedure

Function group: Platform Management

Function Provider entity: ISD-R

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(AES key with key version number '70' and key identifier '01') in order to authenticate the source of the command.

Parameter:

- ISD-P-AID
- Delete Token, calculated as defined in GlobalPlatform Card Specification Amendment D [10] , provided by the SM-DP

Prerequisites:

- The target Profile shall not be the Profile which has the Fall-back Attribute set.
- The target Profile shall be in the Disabled state.

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).
- If any of these verifications fail, terminate the command with an error status word.
- Delete the ISD-P with its Profile, regardless of POL1.

As token protection is only used by this command, this token shall be processed by the ISD-P even though the ISD-P does not have the token verification privilege. No receipt shall be generated by the command.

NOTE: This deviates from the typical handling of tokens by SDs.

Command Description:

This function is realised through the GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification Amendment C [9].

Command Message

DELETE COMMAND

The DELETE command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' – '8F', 'C0' – 'CF' or 'E0' – 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E4'	DELETE
P1	'00'	Reference control parameter P1
P2	'40'	Reference control parameter P2
Lc	'xx'	Length of data field
Data	'xxxx…'	TLV coded objects (and MAC if present)
Le	'00'	

Table 30: DELETE Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	-	-	-	-	-	-	-	Last (or only) command
-	X	X	X	X	X	X	X	RFU

Table 31: DELETE Reference Control Parameter P1

Reference Control Parameter P2

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	1	–	–	–	–	–	–	Delete a root security domain and all associated Applications
–	–	X	X	X	X	X	X	RFU

Table 32: DELETE Command Reference Control Parameter P2

The Delete [card content] Data Field shall contain the following parameters:

Tag	Length	Value Description			MOC
'4F'	5-16	ISD-P-AID to be deleted			M
'B6'	Var.	Control Reference Template for Digital Signature			M
		Tag	**Length**	**Value Description**	**MOC**
		'42'	1-n	Identification Number of the ISD-P	M
		'45'	1-n	Image Number of the ISD-P	M
		'5F20'	1-n	Application Provider identifier	M
		'93'	1-n	Token identifier number	M
'9E'	1-n	Delete Token			M

Table 33: DELETE [card content] Command Data Field

Response Message

Data Field Returned in the Response Message:

A single byte of '00' shall be returned indicating that no additional data is present.

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.2.3.2.

Specific Processing State returned in response Message:

'69 85': Profile is not in the Disabled state or Profile has the Fall-back Attribute.

4.1.1.7 Set Fall-back Attribute

Function name: SetFallbackAttribute

Related Procedures: -

Function group: Platform Management

Function Provider entity: ISD-R

Description: This function sets the Fall-back Attribute for one Profile on the target eUICC.

Parameters:

- ISD-P-AID

Prerequisites:

- The Profile to be assigned the Fall-back Attribute must have Provisioning capability.

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 realised through the GlobalPlatform STORE DATA command as defined in GlobalPlatform Card Specification [6].

Command Message

The STORE DATA command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' – '8F', 'C0' – 'CF' or 'E0' – 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E2'	STORE DATA
P1	'88'	Reference control parameter P1
P2	'00'	Block number
Lc	'xx'	Length of data field
Data	'xxxxx…'	Application data and MAC (if present)
Le		Not present

Table 34: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 35: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A05'	Var	Set Fall-back Attribute			M
		Tag	Length	Value Description	MOC
		'4F'	5-16	ISD-P-AID	M

Table 36: Set Fall-back Attribute Data Field

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.11.3.2.

4.1.1.8 ISD-R Key Set Establishment

Function name: establishISDRKeySet

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider entity: ISD-R

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.

This function is based on Scenario 3 as defined in “GlobalPlatform Card Specification Amendment E [11]. Scenario 3 is modified by adding the additional step of authentication of the new SM-SR to the eUICC.

Adding this step to Scenario 3 requires an additional STORE DATA command to precede the command defined for Scenario 3. This new command provides the eUICC with the certificate of the new SM-SR and retrieves a random challenge from the eUICC. This random challenge then has to be signed by the new SM-SR and sent to the eUICC in the second command to prove to the eUICC that the new SM-SR is in possession of the private key related to the certificate presented. The sequence is pictured in Figure 22 of section 3.8.

Parameters:

- Ephemeral public key of the new SM-SR
- Certificate for the new SM-SR

Prerequisites:

- The ECASD certificate was provided to and verified by the new SM-SR
- The new SM-SR has generated an ephemeral key pair
- The new SM-SR has a signature from the CI.

Command Description:

This function is realised 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 the following table:

Code	Value	Meaning
CLA	‘80’ – ‘8F’, ‘C0’ – ‘CF’ or ‘E0’ – ‘EF’	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	‘E2’	STORE DATA
P1	‘09’	Reference control parameter P1
P2	‘00’	Block number
Lc	‘xx’	Length of data field
Data	‘xxxxx…’	Application data and MAC (if present)
Le	‘00’	

Table 37: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	-	-	-	-	-	-	-	More blocks
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-			1	Case 4 command as defined in GlobalPlatform Amendment A [54]
-	-	-	-	-	X	X	-	RFU

Table 38: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description					MOC
‘3A01’	Var	Certificate of off-card entity					M
		Tag	Length	Value Description			MOC
		‘7F21’	Var.	Certificate			M
				Tag	Length	Value Description	MOC
				‘93’	1-16	Certificate Serial Number	M
				‘42’	1-16	CA Identifier	M
				‘5F20’	1-16	Subject Identifier	M
				‘95’	1	Key Usage, Signature Verification	M
				‘5F25’	4	Effective Date (YYYYMMDD, BCD format)	O
				‘5F24’	4	Expiration Date (YYYYMMDD, BCD format)	M
				‘73’	3-127	Discretionary Data 'C8 01 02': SM-SR certificate other TLVs may follow	M
				‘7F49’	Var.	Public Key – details see tables below	M
				‘5F37’	Var.	Signature	M

Table 39: Data Field for Key Establishment

The following TLV-encoded data are signed off-card with SK.CI.ECDSA to generate the content of tag '5F37' (signature), as described in GlobalPlatform Card Specification Amendment E [11]:

Tag	Length	Value Description	MOC
'93'	1-16	Certificate Serial Number	M
'42'	1-16	CA Identifier	M
'5F20'	1-16	Subject Identifier	M
'95'	1	Key Usage, Signature Verification	M
'5F25'	4	Effective Date (YYYYMMDD, BCD format) – if present	C
'5F24'	4	Expiration Date (YYYYMMDD, BCD format)	M
'73'	3-127	Discretionary Data	M
'7F49'	Var.	Public Key	M

Table 40: Data Signed to Generate the SM-SR Certificate

Key format is defined in section 4.1.1.5.

Response Message

Data Field Returned in the Response Message:

The STORE DATA response shall contain the following data:

Tag	Length	Data Element
'85'	Variable	Random Challenge

Table 41: Response Data for Send SM-SR Certificate

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.11.3.2.

Second STORE DATA commandCommand Message

The STORE DATA command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' - '8F', 'C0' - 'CF' or 'E0' - 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E2'	STORE DATA
P1	'89'	Reference control parameter P1
P2	'01'	Block number
Lc	'xx'	Length of data field
Data	'xxxxx…'	Application data and MAC (if present)
Le	'00'	

Table 42: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	-	-	1	Case 4 command as defined in GlobalPlatform Amendment A [54]
-	-	-	-	-	X	X	-	RFU

Table 43: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description					MOC
'3A02'	Var	Key Establishment					M
		Tag	Length	Value Description			MOC
		'A6'	Var	CRT tag (KAT)			
				Tag	Length	Value Description	MOC
				'90'	2	Scenario identifier '03' (see GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1) Scenario Parameters '0X' (See Table 45)	M
				'95'	'01'	Key Usage Qualifier '5C' (1 secure channel base key) or '10' (3 secure channel keys) (see GlobalPlatform Card Specification [6] Table 11-17)	M
				'96'	'01'	Key Access according to GlobalPlatform Card Specification [6] Table 11-18	O
				'80'	'01'	Key Type according to GlobalPlatform Card Specification [6] Table 11-16	M
				'81'	'01'	Key Length (in bytes)	M
				'82'	'01'	Key Identifier = '00' - '7F'	M
				'83'	'01'	Key Version Number = '01' - '7F'	M
				'91'	'00', '02', '03', '05' or '08'	Initial value of sequence counter	M
				'45'	1-n	Security Domain Image Number (SDIN)	O
				'84'	1-n	HostID (shall only be present if scenario parameter b3 is set)	C
'7F49'	Var	ePK.SR.ECKA					M
'5F37'	Var.	Signature					M

Table 44: Data Field for Key Establishment

b8	b7	b6	b5	b4	b3	b2	b1	Description

0	**0**	**0**	**0**	**–**	**X**	**X**	**X**	As defined in GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1
-	**-**	**-**	**-**	**1**	**–**	**–**	**–**	Certificate Verification precedes Key Establishment

Table 45: Scenario Parameters

In case the scenario parameter specifies usage of HostID+CardID (bit b3=1), then the SM-SR and the eUICC shall use the SIN-LV and SDIN-LV of ISD-R, in lieu of the IIN-LV and CIN-LV of the card; this deviates from GP Amendment E [11].

The SM-SR knows the SIN and SDIN of ISD-R as per the EIS.

The following TLV-encoded data are signed off-card with SK.SR. ECDSA to generate the content of DGI '5F37' (signature), as described in GlobalPlatform Card Specification Amendment E [11]:

DGI	Length	Value Description					MOC
'3A02'	Var	Key set Establishment					M
		Tag	**Length**	**Value Description**			**MOC**
		'A6'	Var	CRT tag (KAT)			M
				Tag	**Length**	**Value Description**	**MOC**
				'90'	2	Scenario identifier '03' (see GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1) Scenario Parameters '0X' (See Table 45)	M
				'95'	'01'	Key Usage Qualifier '5C' (1 secure channel base key) or '10' (3 secure channel keys) (see GlobalPlatform Card Specification [6] Table 11-17)	M
				'96'	'01'	Key Access according to GlobalPlatform Card Specification [6] Table 11-18– if present	C
				'80'	'01'	Key Type according to GlobalPlatform Card Specification [6] Table 11-16	M
				'81'	'01'	Key Length (in bytes)	M
				'82'	'01'	Key Identifier = '00' - '7F'	M
				'83'	'01'	Key Version Number = '01' - '7F'	M
				'91'	'00', '02', '05' or '08'	Initial value of sequence counter	M
				'45'	1-n	Security Domain Image Number (SDIN) – if present	C
				'84'	1-n	HostID (shall only be present if scenario parameter b3 is set)	C
'7F49'	Var	ePK.SR.ECKA					M

DGI	Length	Value Description	MOC
'0085'	Var	Random Challenge	M

Table 46: Data Signed to Generate the Signature

Response Message

Data Field Returned in the Response Message:

The STORE DATA response shall contain the following data:

Tag	Length	Data Element	MOC
'85'	Variable	DR	C
'86'	Variable	receipt	M

Table 47: Response Data for Scenario #3

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6]

4.1.1.9 Finalisation of the ISD-R Handover

Function name: FinaliseISDRhandover

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider entity: ISD-R

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.

Prerequisites:

- None.

Command Description:

DELETE COMMAND

This function is realised through a GlobalPlatform DELETE command as defined in GlobalPlatform Card Specification [6] with proprietary parameters. This command is sent to the ISD-R.

The DELETE command shall have the following parameters:

Code	Value	Meaning
CLA	'80'	See GlobalPlatform Card Specification [6] section 11.1.4.1
INS	'E4'	DELETE

Code	Value	Meaning
P1	'00'	See GlobalPlatform Card Specification [6] section 11.2.2.1
P2	'00'	See GlobalPlatform Card Specification [6] section 11.2.2.2
Lc	'xx'	Length of data field
Data	'xxxx..'	TLV coded objects: Delete [card content] Data Field (See below)
Le	'00'	

Table 48: DELETE Command Message

The Delete [card content] Data Field shall contain one or two instances of following TLV:

Tag	Length	Value Description	MOC
'F2'	3	Range of keys NOT to be deleted. The 3 bytes are coded as follows: byte 1: Key Version Number of the key range byte 2: Key Identifier of first key of the key range byte 3: Key Identifier of last key of the key range	M

Table 49: Delete [card content] Command Data Field

NOTE: Two TLVs allow for one SCP80 and one SCP81 key set to "survive" key clean-up.

Example:

'F2 03 06 01 03 F2 03 43 01 02' will delete all keys except those with Key Version Number – Key identifier: '06' – '01', '06' – '02', '06' – '03', '43' – '01' and '43' – '02'.

<u>Function flow</u>

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.

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall contain a single byte of '00'.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.2.3.2.

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.

4.1.1.10 SM-SR Addressing Parameters Update

Function name: UpdateSMSRAddressingParameters

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider entity: ISD-R

Description:

This function is used to update SM-SR addressing Parameters on the eUICC.

This function may be used by the SM-SR outside the SM-SR Change procedure in case some parameters have changed.

This function has the following parameter:

- ISD-R AID
- SM-SR addressing Parameters

NOTE: The HTTPS Connectivity Parameters can be updated by the function defined in GlobalPlatform Card Specification Amendment B [8].

Prerequisites

- None

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] section 11.11.2.

Code	Value	Meaning
CLA	‘80’	
INS	‘E2’	STORE DATA
P1	‘88’	Reference Control Parameter P1 0
P2	‘00’	Block Number (Not used for Enable
Lc	‘XX’	Length of data field
Data	‘XX’	Application Data and MAC (if present)
Le	Not present	

Table 50: STORE DATA Command Message

Parameter P1 is coded according to the following table:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 51: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A07'	Var	Connectivity Parameters			M
		Tag	Length	Value Description	MOC
		'A3'	n	SMS parameters	C
		'A4'	n	CAT_TP and BIP parameters	C

Table 52: Send SM-SR Certificate Data Field

SMS parameters value Description coding

Tag	Lenght	Value Description
'81'	2-12	SM-SR Platform Destination Address*

Table 53: SMS Coding

*SM-SR Platform Destination Address is coded as specified for the TP-Destination-Address in 3GPP TS 23.040 [39].

CAT_TP link and BIP open channel parameters

Description
UICC/terminal interface transport level*
Data Destination Address comprehension TLV **

Table 54: CAT_TP Link and BIP Open Channel Parameters

*As defined in ETSI TS 102 226 [5] in the section “Data for CAT_TP link establishment” and “Data for BIP channel opening”.

**As defined in ETSI TS 102 223 [3].

The CR bit of the tags shall be set to zero.

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.11.2.

4.1.1.11 Handle Default Notification

Function name: HandleDefaultNotification

Related Procedures: Profile Enabling, Profile Enabling via SM-DP, Profile Disabling, Fall-back

Function group: eUICC Management

Function Provider entity: ISD-R

Description: This function provides a default notification from the eUICC to the SM-SR.

Parameters:

- EID
- ISD-P AID
- Mobile Equipment Identification (for example MEID, IMEI)
- Notification Sequence number
- Notification type

Prerequisites:

The eUICC has received a notification of network attachment. ***Notification Message***

> Note: There is no single method implemented by all devices to notify the eUICC of network attachment. The eUICC may rely on various heuristics to determine that network attachement is effective. As a worst-case safeguard, the eUICC shall attempt to send profile change notifications within a time interval of 10 STATUS events after card reset.

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].

Description	Length (bytes)	Value	MOC
eUICC notification tag	1	'E1'. To avoid conflicts with the values defined in ETSI TS 101 220 [2], a tag from the proprietary class is used.	M
Length (A+B+C+D+E+F)	1 or 2	BER-TLV coding length	M

EID	A	See below	M
Notification type	B	See below	M
Notification sequence number	C	See below	M
ISD-P- AID	D	See ETSI TS 102 223 [3], clause 8.60	M
IMEI	E	See ETSI TS 102 223 [3], clause 8.20	C
MEID	F	See ETSI TS 102 223 [3], clause 8.81	C

Table 55: Data Format for Notification

IMEI and MEID are optional. In case the eUICC encounters any issue while getting the Mobile Equipment Identification of the Device, no value is provided. If both IMEI and MEID are retrieved, only one could be sent to limit overall message length.

COMPREHENSION-TLV for EID

Byte(s)	Description	Length
1	EID tag, '4C'.	1
2	Length (X) of the EID (shall not exceed 16 bytes)	1
3 to X+2	EID value, as defined in section 2.2.2.	X

Table 56: COMPREHENSION-TLV for EID

COMPREHENSION-TLV for Notification type

Byte(s)	Description	Length
1	Notification type tag, '4D'.	1
2	Length= '01'	1
3	Notification type	1

Table 57: COMPREHENSION-TLV for Notification type

Notification type:

Coding:

- '01': eUICC declaration – First network attachment
- '02': Profile change succeeded
- '03': Profile change failed and Roll-back
- '04': Void
- '05': Profile change after Fall-back
- '06' to 'FF': RFU

COMPREHENSION-TLV for Notification sequence number

Byte(s)	Description	Length
1	Notification sequence number tag, '4E'.	1

2	Length='02'	1
3 to 4	Notification sequence number value	2

Table 58: COMPREHENSION-TLV for Notification Sequence Number

The notification sequence number identifies the notification message, and allows the SM-SR to distinguish a new notification from a retry. In case of a retry, the eUICC shall use the same notification sequence number. When a Notification Confirmation has been successfully received by the SM-SR, the eUICC shall increment the sequence number for the next notification.

NOTE : Depending on the eUICC implementation, the notification may also contain additional TLVs using EUM-specific tags.

An SM-SR is not required to record or process such specific tags, and can simply ignore them

In any case, the size of the complete notification shall fit into one SMS-MO if the notification is sent by SMS, and shall not exceed the size of 240 bytes if sent by HTTP or CAT-TP.

Secured data structure for eUICC notification over SCP80

The secured data containing the eUICC notification is the COMPREHENSION-TLV structure specified above, with no added layer of Expanded format.
This deviates from the requirement described in section 2.4.3.3.

Default Notification Protocol Priority

A protocol priority order for default notification may be defined for every Profile, using SMS, HTTPS and CAT_TP.

If not defined for a Profile, the default priority order is set as follow:

Priority	Protocol
1	SMS
2	HTTPS
3	CAT_TP

Table 59: Default Notification Protocol Priority

4.1.1.12 Notification Confirmation

Function name: HandleNotificationConfirmation

Related Procedures: Handle Default Notification

Function group: eUICC Management

Function Provider entity: ISD-R

Description: This function confirms the notification and triggers potential follow-up activities required by POL1.

Parameters:

- Notification Sequence number

Prerequisites:

- The SM-SR has received a notification from the eUICC.

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:

STORE DATA Command

This function is realised through the GlobalPlatform STORE DATA command as defined in GlobalPlatform Card Specification [6].

Command Message

The STORE DATA command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' – '8F', 'C0' – 'CF' or 'E0' – 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E2'	STORE DATA
P1	'89'	Reference control parameter P1
P2	'00'	Block number
Lc	'xx'	Length of data field
Data	'xxxxx…'	Application data and MAC (if present)
Le	'00'	

Table 60: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	-	-	1	Case 4 command
-	-	-	-	-	X	X	-	RFU

Table 61: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A08'	Var	Notification Confirmation			M
		Tag	**Length**	**Value Description**	**MOC**
		'4E'	2	Notification Sequence number	M

Table 62: Notification Confirmation Data Field

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall contain the data structure below.

- Not be present, if no follow-up activities had to be performed, or
- Contain the data structure below if follow-up activities were performed.

Tag	Length	Value Description			MOC
'80'	Var	List of Deleted ISD-Ps			M
		Tag	Length	Value Description	MOC
		'4F'	16	AID of ISD-P	M

Table 63: Notification Confirmation Response Data Field

NOTE: In the current version, the response will carry only one AID. However, the structure is defined in a generic way so that results of other follow-up activities can be added when required.

NOTE: If no follow-up activity has been performed at all, the data field shall contain tag 80 followed by a length of zero, and no value.

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.11.3.2.

4.1.2 ES6 (MNO-eUICC) Interface Description

4.1.2.1 Policy Rules Update by MNO

Function name: UpdatePOL1byMNO

Related Procedures: Pol1 Update by MNO

Function group: Profile Management

Function Provider entity: MNO-SD

Description: This function is used to update POL1 on the eUICC.

This function has the following parameter:

- POL1

Prerequisites

- The Profile is enabled

Function flow

Upon reception of the POL1 update command, the eUICC shall:

- Update POL1 of the ISD-P containing the targeted MNO-SD.

Commands

INSTALL [for personalization] command

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:

Code	Value	Meaning
CLA	'80'	See GlobalPlatform Card Specification section 11.1
INS	'E6'	INSTALL
P1	'20'	See GlobalPlatform Card Specification [6] section 11.5.2.1
P2	'00'	See GlobalPlatform Card Specification [6] section 11.5.2.2
Lc	'xx'	Data Field Length
Data	'xxxx…'	See GlobalPlatform Card Specification [6] section 11.5.2.3
Le	'00'	

Table 64: INSTALL [for Personalization] Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	1	0	0	0	0	0	For personalization

Table 65: INSTALL [for Personalization] Reference Control Parameter P1

Reference Control Parameter P2 – ISD-P State Coding

P2 is set to '00': according to GlobalPlatform Card Specification [6] section 11.5, this means no information is provided.

Data Field

Name	Length	Value	MOC
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M
Length of Application AID	'1'	'05' – '10'	M

Name	Length	Value	MOC
Application AID (Reserved value for Profile's ISD-P)	'05 – 10'	'xxxx'	M
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M

Table 66: INSTALL Command Data Field

The reserved value for Profile's ISD-P indicates that the Security Domain targeted by the INSTALL [for personalization] command is the ISD-P of the Profile containing the MNO-SD.

NOTE: This mechanism avoids the MNO having to know and keep track of the ISD-P AID assigned by the SM-SR.

Response Message

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] section 11.5.3.1.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.5.3.2.

Specific Processing State returned in response Message:

None

STORE DATA command:

This command is a STORE DATA command, as described in GlobalPlatform Card Specification [6].

Code	Value	Meaning
CLA	'80'	
INS	'E2'	STORE DATA
P1	'88'	Reference Control Parameter P1 0
P2	'00'	Block Number (Not used for Enable
Lc	'XX'	Length of data field
Data	'XX'	Application Data and MAC (if present)
Le	Not present	

Table 67: STORE DATA Command Message

Parameter P1 is coded according to the following table:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data

-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 68: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A06'	Var	POL1 Policy Rules			M
		Tag	Length	Value Description	MOC
		'81'	'01'	New POL1	M

Table 69: POL1 Update Data Field

POL1 coding

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	0	0	0	No POL1 rule active
-	-	-	-	-	0	-	1	Disabling of this Profile not allowed
-	-	-	-	-	0	1	-	Deletion of this Profile not allowed
-	-	-	-	-	1	0	0	Profile deletion is mandatory when its state is changed to disabled
-	-	-	-	-	X	X	X	Other combinations are forbidden
X	X	X	X	X	-	-	-	RFU

Table 70: POL1 Coding

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.
Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.11.3.2

4.1.2.2 Connectivity Parameters Update by MNO

Function name: UpdateConnectivityParametersByMNO

Related Procedures: Connectivity Parameters Update by MNO

Function group: Profile Management

Function Provider entity: MNO-SD

Description: This function is used to update Connectivity Parameters on the eUICC.

This function has the following parameter:

- Connectivity Parameters

Prerequisites

- The Profile is enabled

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.

4.1.3 ES8 (SM-DP-eUICC) Interface Description

4.1.3.1 ISD-P Key Set Establishment

Function name: EstablishISDPKeySet

Related Procedures: Key Establishment

Function group: Profile Management

Function Provider entity: ISD-P

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.

This function is based on Scenario 3 as defined in GlobalPlatform Card Specification Amendment E [11]. Scenario 3 is modified by adding the additional step of authentication of the SM-DP to the eUICC.

Adding this step to Scenario 3 requires an additional STORE DATA command to precede the command defined for Scenario 3. This new command provides the eUICC with the certificate of the SM-DP and retrieves a random challenge from the eUICC. This random challenge then has to be signed by the SM-DP and sent to the eUICC in the second command to prove to the eUICC that the SM-DP is in possession of the private key related to the certificate presented. The sequence is pictured in Figure 11 of section 3.1.2.

NOTE: A complementary scenario based on RSA is FFS.

Parameters:

- ISD-P AID
- Ephemeral public key of the SM-DP
- Certificate for the SM-DP

Prerequisites:

- The ECASD certificate was provided to and verified by the SM-DP
- SM-DP has generated an ephemeral key pair
- SM-DP has a signature from the CI
- ISD-P was created

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

Command Message

Code	Value	Meaning
CLA	‘80’	See GlobalPlatform Card Specification section 11.1
INS	‘E6’	INSTALL
P1	‘20’	See GlobalPlatform Card Specification [8] section 11.5.2.1
P2	‘00’	See GlobalPlatform Card Specification [8] section 11.5.2.2
Lc	‘xx’	Data Field Length
Data	‘xxxx…’	See GlobalPlatform Card Specification [8] section 11.5.2.3.6
Le	‘00’	

Table 71: INSTALL [for Personalization] Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	0	1	0	0	0	0	0	For personalization

Table 72: INSTALL [for personalization] Reference Control Parameter P1

Reference Control Parameter P2 – ISD-P State Coding

P2 is set to '00': according to GlobalPlatform Card Specification [8] section 11.5, this means no information is provided.

Data Field

Name	Length	Value	MOC
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M
Length of Application AID	'1'	'05' – '10'	M
Application AID of ISD-P	'05 – 10'	'xxxx'	M
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M
Length of data	'1'	'00'	M

Table 73: INSTALL Command Data Field

Response Message

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 [6] section 11.5.3.1.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [8] section 11.5.3.2.

Specific Processing State returned in response Message:

None

First STORE DATA command

Command Message

The STORE DATA command message shall be coded according to the following table:

Code	Value	Meaning
CLA	‘80’ – ‘8F’, ‘C0’ – ‘CF’ or ‘E0’ – ‘EF’	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	‘E2’	STORE DATA
P1	‘09’	Reference control parameter P1
P2	‘00’	Block number
Lc	‘xx’	Length of data field
Data	‘xxxxx…’	Application data and MAC (if present)
Le	‘00’	

Table 74: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
0	-	-	-	-	-	-	-	More blocks
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	-	-	1	Case 4 command as defined in GlobalPlatform Amendment A [54]
-	-	-	-	-	X	X	-	RFU

Table 75: STORE DATA Reference Control Parameter P

Data Field Sent in the Command Message

DGI	Length	Value Description					MOC
‘3A01’	Var	Certificate of off-card entity					M
		Tag	Length	Value Description			MOC
		‘7F21’	Var.	Certificate			M
				Tag	Length	Value Description	MOC
				‘93’	1-16	Certificate Serial Number	M
				‘42’	1-16	CA Identifier	M
				‘5F20’	1-16	Subject Identifier	M
				‘95’	1	Key Usage, Signature Verification	M
				‘5F25’	4	Effective Date (YYYYMMDD, BCD format)	O
				‘5F24’	4	Expiration Date (YYYYMMDD, BCD format)	M
				‘73’	3-127	Discretionary Data 'C8 01 01': SM-DP certificate other TLVs may follow	M
				‘7F49’	Var.	Public Key – details see tables below	M
				‘5F37’	Var.	Signature	M

Table 76: Send SM-DP Certificate Data Field

The following TLV-encoded data are signed off-card with SK.CI. ECDSA to generate the content of tag '5F37' (signature), as described in GlobalPlatform Card Specification Amendment E [11]:

Tag	Length	Value Description	MOC
'93'	1-16	Certificate Serial Number	M
'42'	1-16	CA Identifier	M
'5F20'	1-16	Subject Identifier	M
'95'	1	Key Usage, Signature Verification	M
'5F25'	4	Effective Date (YYYYMMDD, BCD format) – if present	C
'5F24'	4	Expiration Date (YYYYMMDD, BCD format)	M
'73'	3-127	Discretionary Data	M
'7F49'	Var.	Public Key	M

Table 77: Data Signed to Generate the SM-DP Certificate

Key format is defined in section 4.1.1.5.

Response Message

Data Field Returned in the Response Message:

The STORE DATA response shall contain the following data:

Tag	Length	Data Element
'85'	Variable	Random Challenge

Table 78: Response Data for Send SM-DP Certificate

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.11.3.2.[6]

Second STORE DATA command

Command Message

The STORE DATA command message shall be coded according to the following table:

Code	Value	Meaning
CLA	'80' - '8F', 'C0' - 'CF' or 'E0' - 'EF'	See section 11.1.4 of GlobalPlatform Card Specification [6]
INS	'E2'	STORE DATA
P1	'89'	Reference control parameter P1
P2	'01'	Block number
Lc	'xx'	Length of data field
Data	'xxxxx…'	Application data and MAC (if present)
Le	'00'	

Table 79: STORE DATA Command Message

Reference Control Parameter P1

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	-	-	1	Case 4 command as defined in GlobalPlatform Amendment A [54]
-	-	-	-	-	X	X	-	RFU

Table 80: STORE DATA Reference Control Parameter

Data Field Sent in the Command Message

DGI	Length	Value Description					MOC
'3A02'	Var	Key set Establish					M
		Tag	Length	Value Description			MOC
		'A6'	Var	CRT tag (KAT)			
				Tag	Length	Value Description	MOC
				'90'	2	Scenario identifier '03' (see GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1) Scenario Parameters '0X' (See Table 82)	M
				'95'	'01'	Key Usage Qualifier '5C' (1 secure channel base key) or '10' (3 secure channel keys) (see GlobalPlatform Card Specification [6] Table 11-17)	M
				'96'	'01'	Key Access according to GlobalPlatform Card Specification [[6] Table 11-18	O
				'80'	'01'	Key Type according to GlobalPlatform Card Specification [6] Table 11-16	M
				'81'	'01'	Key Length (in bytes)	M
				'82'	'01'	Key Identifier = '00' - '7F'	M
				'83'	'01'	Key Version Number = '01' - '7F'	M
				'91'	'00', '02', '05' or '08'	Initial value of sequence counter	M
				'45'	1-n	Security Domain Image Number (SDIN)	O
				'84'	1-n	HostID (shall only be present if scenario parameter b3 is set)	C
'7F49'	Var	ePK.DP.ECKA					M
'5F37'	Var.	Signature					M

Table 81: Send SM-DP Certificate Data Field

The following TLV-encoded data are signed off-card with SK.DP. ECDSA to generate the content of DGI '5F37' (signature), as described in GlobalPlatform Card Specification Amendment E [11]:

b8	b7	b6	b5	b4	b3	b2	b1	Description
0	0	0	0	–	X	X	X	As defined in GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1
-	-	-	-	1	–	–	–	Certificate Verification precedes Key Establishment

Table 82: Scenario Parameters

In case the scenario parameter specifies usage of HostID+CardID (bit b3=1), then the SM-DP and the eUICC shall use the SIN-LV and SDIN-LV of ISD-R, in lieu of the IIN-LV and CIN-LV of the card; this deviates from GP Amendment E [11], and ISD-P has no SIN/SDIN yet.

The SM-DP can know the SIN and SDIN of ISD-R by inspecting the EIS retrieved on ES3.

DGI	Length	Value Description					MOC
'3A02'	Var	Key set Establishment					M
		Tag	Length	Value Description			MOC
		'A6'	Var	CRT tag (KAT)			M
				Tag	Length	Value Description	MOC
				'90'	2	Scenario identifier '03' (see GlobalPlatform Card Specification Amendment E: Security Upgrade for Card Content Management [11] section 4.8.1) Scenario Parameters '0X' (See Table 82)	M
				'95'	'01'	Key Usage Qualifier '5C' (1 secure channel base key) or '10' (3 secure channel keys) (see GlobalPlatform Card Specification [6] Table 11-17)	M
				'96'	'01'	Key Access according to GlobalPlatform Card Specification [6] Table 11-18 – if present	C
				'80'	'01'	Key Type according to GlobalPlatform Card Specification [6] Table 11-16	M
				'81'	'01'	Key Length (in bytes)	M
				'82'	'01'	Key Identifier = '00' - '7F'	M
				'83'	'01'	Key Version Number = '01' - '7F'	M
				'91'	'00', '02', '05' or '08'	Initial value of sequence counter	M
				'45'	1-n	Security Domain Image Number (SDIN) – if present	C
				'84'	1-n	HostID (shall only be present if scenario parameter b3 is set)	C

DGI	Length	Value Description	MOC
'7F49'	Var	ePK.DP.ECKA	M
'0085'	Var	Random Challenge	M

Table 83: Data Signed to Generate the Signature

Response Message

Data Field Returned in the Response Message:

The STORE DATA response shall contain the following data:

Tag	Length	Value Description	MOC
'85'	Variable	DR	C
'86'	Variable	Receipt	M

Table 84: Response Data for Scenario #3

Processing State Returned in the Response Message:

As defined in GlobalPlatform Card Specification [6] section 11.11.3.2.

4.1.3.2 Command coding for SCP03

All ES8 functions in subsequent sections require securing the commands by SCP03.

NOTE: The profile package itself is protected by SCP03t as defined in the next section.

Opening an SCP03 secure channel requires the following two commands:
INITIALIZE UPDATE

Code	Value	Description
CLA	See to GlobalPlatform Card Specification Amendment D [10]	Section 7.1.1
INS	See to GlobalPlatform Card Specification Amendment D [10]	INITIALIZE UPDATE
P1	'XX'	indicates the version of the target ISD-P key set used
P2	'XX'	indicates the key identifier of the target ISD-P
Lc	'08'	Length of the host challenge
Data	'XX…XX'	For Host Challenge see GlobalPlatform Card Specification Amendment D [10] for computation of the host challenge
Le	'00'	

Table 85: Initialize Update Command

EXTERNAL AUTHENTICATE

Code	Value	Description
CLA	See GlobalPlatform Card Specification Amendment D [10]	Section 7.1.2.
INS	See GlobalPlatform Card Specification Amendment D [10]	EXTERNAL AUTHENTICATE
P1	'33'	indicates the security level and shall be set to request C-DECRYPTION, R-DECRIPTION, R-MAC and C-MAC (see to GlobalPlatform Card Specification Amendment D [10])
P2	'00'	
Lc	'10'	Length of the host cryptogram and MAC
Data	'XX…XX'	Host cryptogram and MAC (see to GlobalPlatform Card Specification Amendment D [10] for computation of host cryptogram and MAC)
Le		Not Present

Table 86: External Authenticate Command

These two commands shall be following by any ES8 command as defined in subsequent sections depending on the procedure to be performed.

Those ES8 commands and their responses are modified by encrypting the data part and adding a MAC as defined in GlobalPlatform Card Specification Amendment D [10].

4.1.3.3 Download and Installation of a Profile

Function name: DownloadAndInstallation

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider entity: ISD-P

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.

NOTE: The ISD-P identification is provided within the ES5 transport protocol.

The Profile shall be protected by SCP03t. 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

Prerequisites:

- ISD-P must be created
- ISD-P must be PERSONALIZED (as defined in GlobalPlatform Card Specification[6])

- Connection is secured via SCP03t

Check Figure 12 for further details.

Description of the SCP03t security protocol:

This section defines a secure channel protocol based on GlobalPlatform's SCP03 usable for TLV structures, named SCP03t hereafter.

Tag values are defined so that they can be used without conflict within the expanded remote management format which is used to transport data inside SCP80 or SCP81 of ES5.

As no SWs are used, errors are indicated by a special response tag with tag number +'80' (resulting in a 2 byte tag).

The data transported in the command TLVs specified below shall consist of the Profile Package specified in [53]; the response TLVs shall transport Profile Element (PE) responses as provided by the Profile Package processing specified in [53]. The Profile Package consists of a sequence of PE TLVs. However, 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.

The options allowed in SCP03 are limited as follows in SCP03t:

- Response security is always the same as command security (if no error).
- BEGIN/END R-MAC SESSION is not supported.
- Only one option is defined: MAC + encryption.

The following sections describe the changes required to move from SCP03 to SCP03t. Everything else is inherited from SCP03.

As the security mechanisms are exactly the same as SCP03, the SCP03 key sets are used for SCP03t.

Secure channel initiation uses 2 TLVs equivalent to the INITIALIZE UPDATE and the EXTERNAL AUTHENTICATE APDUs.

Thereafter, command and response TLVs are protected in the same way as SCP03 APDUs.

Each command TLV triggers one response TLV. A response may be empty or carry response data from the application layer.

Secure Channel Initiation: INITIALIZE UPDATE command TLV

The following data shall be encapsulated in a TLV with tag '84':

Name	Length	Presence
Key version number	1 byte	Mandatory

Name	Length	Presence
Key identifier	1 byte	Mandatory (and value must be '00', consistently with SCP03)
Host challenge	8 bytes	Mandatory

Table 87: Initialize Update Command TLV

The data field of the response TLV shall contain the concatenation without delimiters of the following data elements, encapsulated in a TLV with tag '84'.

Name	Length	Presence
Key diversification data	10 bytes	Mandatory
Key information	3 bytes	Mandatory
Card challenge	8 bytes	Mandatory
Card cryptogram	8 bytes	Mandatory
Sequence Counter	3 bytes	Mandatory

Table 88: Response TLV Data Elements

In case of an error, tag '9F44' is used. The following values are defined:

- '01': error in length or structure of command data
- '03': referenced data not found

Secure Channel Initiation: EXTERNAL AUTHENTICATE command TLV

The following data shall be encapsulated in a TLV with tag '85':

Name	Length	Presence
Security level	1 byte	Mandatory
Host cryptogram	8 bytes	Mandatory
MAC	8 bytes	Mandatory

Table 89: External Authenticate Command TLV

The security level shall be set to '33': "C DECRYPTION, R ENCRYPTION, C MAC, and R MAC".

The MAC shall be computed based on the values present in the TLV, as follows:

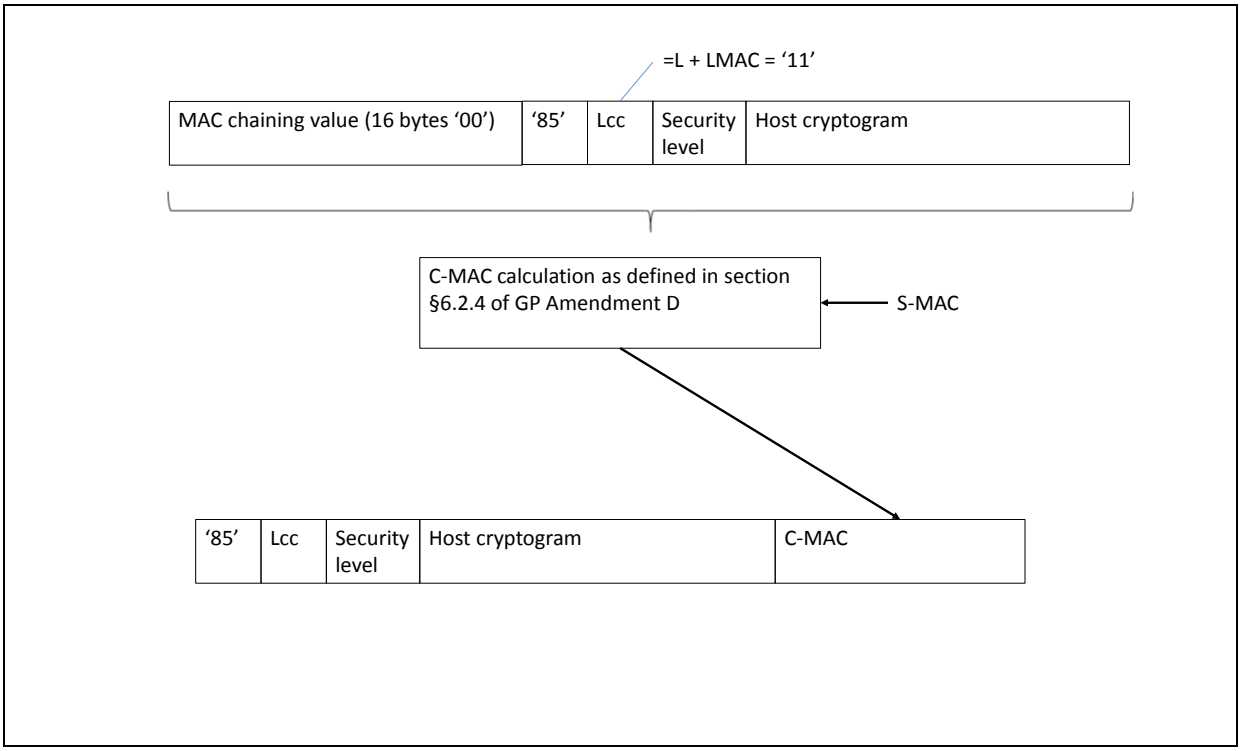


Figure 32: TLV Command Data Field Encryption

If the message is accepted, a Response TLV with tag '85' and length zero shall be returned.

This TLV does not return an R-MAC.

In case of an error, tag '9F45' is used. The following values are defined:

- '01': error in length or structure of command data
- '02': security error

Command TLV C-MAC and C-DECRYPTION Generation and Verification

For encapsulating encrypted profile command data in a SCP03t TLV, tag '86' is used.

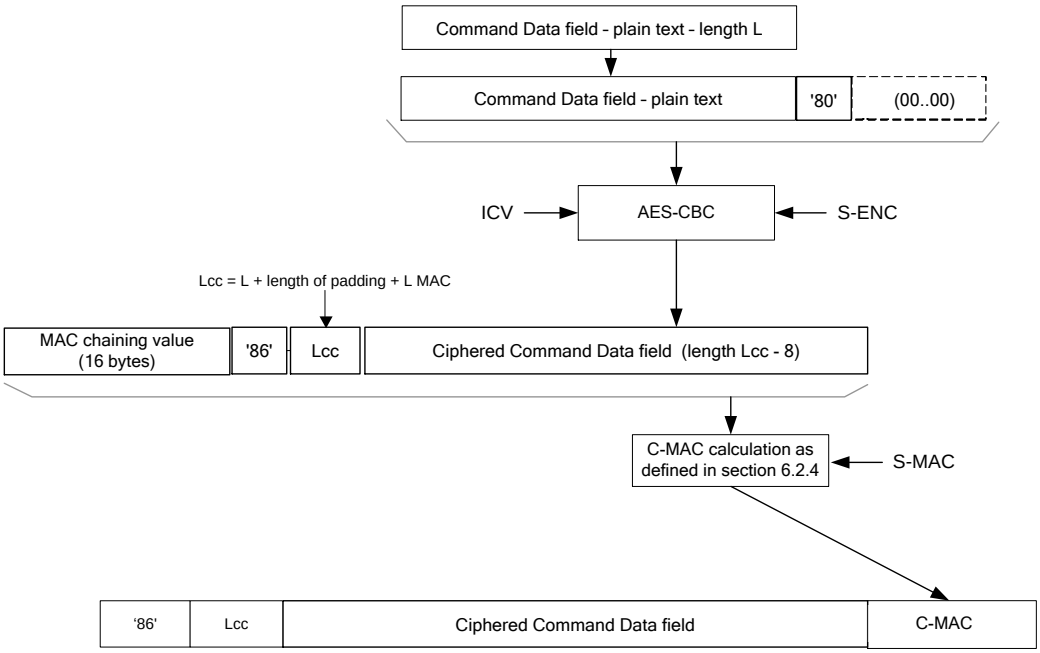


Figure 33: **TLV Command Data Field Encryption**

Response R-MAC and R-ENCRYPTION Generation and Verification

For encapsulating encrypted profile response data in a SCP03t TLV, tag '86' is used.

If there is no response data field, the length is ‘00’, and no R-MAC shall be generated, so the response TLV shall be ’86 00’.In case of an error, tag '9F46' is used, and no R-MAC nor R-ENCRYPTION shall be generated. The response data field contains only one byte, the following values are defined:

- '01': error in length or structure of command data
- '02': security error

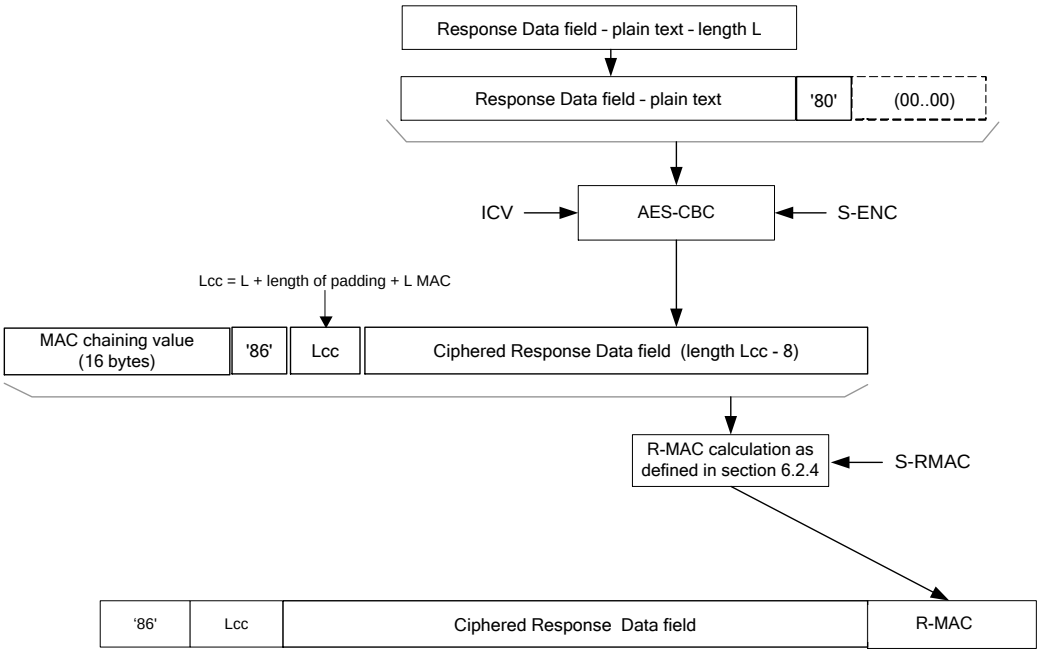


Figure 34: TLV Response Data Field Encryption

4.1.3.4 Connectivity Parameters Update using SCP03

Function name: UpdateConnectivityParameters SCP03

Related Procedures: Connectivity Parameters Update using SCP03

Function group: eUICC Management

Function Provider entity: ISD-P

Description: This function is used to update Connectivity Parameters on the eUICC.

This function has the following parameter:

- ISD-P AID
- Connectivity Parameters

Prerequisites

- None

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.

Code	Value	Meaning
CLA	‘80’	
INS	‘E2’	STORE DATA
P1	‘88’	Reference Control Parameter P1 0
P2	‘00’	Block Number (Not used for Enable
Lc	‘XX’	Length of data field
Data	‘XX’	Application Data and MAC (if present)
Le	Not present	

Table 90: STORE DATA Command Message

Parameter P1 is coded according to the following table:

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
1	-	-	-	-	-	-	-	Last block
-	0	0	-	-	-	-	-	No general encryption information or non-encrypted data
-	-	-	0	1	-	-	-	DGI format of the command data field
-	-	-	-	-	X	X	X	RFU

Table 91: STORE DATA Reference Control Parameter P1

Data Field Sent in the Command Message

DGI	Length	Value Description			MOC
'3A07'	Var	Connectivity Parameters			M
		Tag	Length	Value Description	MOC
		'A0'	n	SMS parameters	C
		'A1'	n	HTTP parameters	C
		'A2'	n	CAT_TP parameters	C

Table 92: Connectivity Parameters Data Field

NOTE 1: The order of the TLVs in the Connectivity parameter DGI defines the priority.

NOTE 2: Multiple occurrences of each Connectivity Parameters TLV are possible.

SMS parameters coding

Description
SMSC Address*

Table 93: SMS Parameters Coding

* Comprehension TLVs as defined in ETSI TS 102 223 [3]. The CR bit of the tags shall be set to zero.

HTTP and CAT_TP parameters coding

Description
Bearer description*

Network Access Name (NAN) *
User Login*
User Password*

Table 94: HTTP and CAT_TP parameters coding

* Comprehension TLVs as defined in ETSI TS 102 223 [3]. The CR bit of the tags shall be set to zero.

Response Message

Data Field Returned in the Response Message:

The data field of the response message shall not be present.

Processing State returned in the Response Message:

See GlobalPlatform Card Specification [6] section 11.11.3.2.

5 Off-Card Interface Descriptions

This section provides the description of the interfaces and functions within the Remote Provisioning and Management system outside the eUICC, including the following:

- ES1, interface between the two entities fulfilling the Role EUM and the Role SM-SR.
- ES2, interface between the two entities fulfilling the Role MNO and the Role SM-DP.
- ES3, interface between the two entities fulfilling the Role SM-DP and the Role SM-SR.
- ES4, interface between the two entities fulfilling the Role MNO and the Role SM-SR.
- ES7, interface between the two entities fulfilling the Role SM-SR and the Role SM-SR.

The functions in this section are grouped into function groups. Each function group is provided by a unique Role and corresponds to an autonomous and consistent functionality.

When a function group is implemented by a Role, all the functions associated to this function group shall be implemented by that Role. In other words, function groups cannot be partially implemented; if a special function is requested, then all the functions of the corresponding function group shall be implemented.

The following table presents the normative list of all the functions that are defined in this section.

Request-response functions:

Interface	Function group	Functions	Function provider Role
ES1	eUICC Management	RegisterEIS	SM-SR

ES2	Profile Management	GetEIS DownloadProfile UpdatePolicyRules UpdateSubscriptionAddress	SM-DP
	Platform Management	EnableProfile DisableProfile DeleteProfile	SM-DP
ES3	Profile Management	GetEIS AuditEIS CreateISDP SendData ProfileDownloadCompleted UpdatePolicyRules UpdateSubscriptionAddress UpdateConnectivityParameters	SM-SR
	Platform Management	EnableProfile DisableProfile DeleteISDP	SM-SR
ES4	Profile Management	GetEIS UpdatePolicyRules UpdateSubscriptionAddress AuditEIS	SM-SR
	Platform Management	EnableProfile DisableProfile DeleteProfile	SM-SR
	eUICC Management	PrepareSMSRChange SMSRchange	SM-SR
ES7	eUICC Management	CreateAdditionalKeySet HandoverEUICC AuthenticateSMSR	SM-SR

Table 95: Request-Response Functions

Notification handler functions:

Interface	Function group	Notification handler functions	Function handler/recipient

ES2	Platform Management	`HandleProfileDisabledNotification`
		`HandleProfileEnabledNotification`
		`HandleProfileDeletedNotification`
			MNO

	eUICC Management	`HandleSMSRChangeNotification`	MNO

ES3	Platform Management	`HandleProfileDisabledNotification`
		`HandleProfileEnabledNotification`
		`HandleProfileDeletedNotification`
			SM-DP

	eUICC Management	`HandleSMSRChangeNotification`	SM-DP

ES4	Platform Management	`HandleProfileDisabledNotification`
		`HandleProfileEnabledNotification`
		`HandleProfileDeletedNotification`
			MNO

	eUICC Management	`HandleSMSRChangeNotification`	MNO

Table 96: Notification Handler Functions

5.1 Function Commonalities

Each functions represents an entry points that is provided by a Role (function provider), and that can be called by other Roles (function requester).

5.1.1 Common Data Types

The functions provided in this section deal with management of eUICC and Profile, so that the common data defined in this section need to be used in most of the functions.

5.1.1.1 Simple Types

Type name	Description	Type definition
Hexadecimal String	String of even length composed of characters between '0' to '9' and 'A' to 'F' or 'a' to 'f'.	
AID	The AID (Application IDentifier) of an Executable Load File, an Executable Module, a security domain, or an Application.	Hexadecimal string representation of 5 to 16 bytes.
DATETIME	Any date and time used within any interface of this specification	String format as specified by W3C: YYYY-MM-DDThh:mm:ssTZD Where: YYYY = four-digit year MM = two-digit month (01=Jan, etc.) DD = two-digit day of month (01-31) hh = two digits of hour (00 -23) mm = two digits of minute (00 - 59) ss = two digits of second (00 - 59) TZD = time zone designator (Z, +hh:mm or -hh:mm) Ex: 2001-12-17T09:30:47Z
EID	The EID type is for representing an eUICC-ID. An eUICC-ID is primarily used in the "Embedded UICC Remote Provisioning and Management System" to identify an eUICC. See section 2.2.2 for EID description.	Hexadecimal string

ICCID	The ICCID type is for representing an ICCID (Integrated Circuit Card IDentifier). The ICCID is primarily used to identify a Profile. ICCID is defined according to ITU-T recommendation E.118 [21] , with the derivation that up to 20 digits are allowed.	String representation of up to 20 decimal digits, non-swapped. Ex: 8947010008, 8947010000123456784. 89470100001234567846
KCV	The KCV stands for "Key Check Value". It provides the material for receiving entity to ensure that it uses the same key value as the sending entity. See Annex F for detail of KCV computation.	Hexadecimal string
MSISDN	The Mobile Station ISDN (Integrated Services Digital Network) Number	String representation of up to 15 decimal digits, as defined in [22]
IMSI	The IMSI (International Mobile Subscriber Identity) used to identify the Subscriber of a Mobile Subscription.	String representation of up to 15 decimal digits including MCC (3 digits) and MNC (2 or 3 digits), as defined in ITU E.212 [12]
OID	An Object IDentifier	String representation of an OID, i.e. of integers separated with dots (for example: '1.2', '3.4.5')
TAR	The TAR (Toolkit Application Reference) of a security domain or an Application.	String - Hexadecimal string representation of exactly 3 bytes
VERSION	The Version type is for indicating a version of any entity used within this specification. A version is defined by its major, minor and revision number	String representation of three integers separated with dots (for example: '1.15.3').

Table 97: Simple Types

5.1.1.2 Complex Types

5.1.1.3.1 SUBSCRIPTION ADDRESS

The **SUBSCRIPTION-ADDRESS** type is defined by:

Data name	Description	Type	No.	MOC
msisdn	The MSISDN of the Subscription associated to this Profile.	MSISDN	1	C
imsi	The IMSI of the Subscription associated to this Profile.	IMSI	1	C

Table 98: Subscription Address

Either the MSISDN, the IMSI, or both, shall be present.

NOTE: Additional address types could be added depending of the deployment mode (for example: SIP-URI).

5.1.1.3.2 POL2-RULE

The POL2-RULE type is defined by the following data structure:

Data name	Description	Type	No.	MOC
subject	Identifies the subject on which the rule has to be applied. In the current version of this release, the possible subject is restricted to "PROFILE".	Enumeration{PROFILE}	1	M

action	Identifies the action/function on which the rule has to be applied.	Enumeration{ENABLE, DISABLE, DELETE}	1	M
qualification	Indicates the final result of the rule that has to be applied.	Enumeration{ Not allowed, Auto-delete}	1	M

Table 99: POL2 Rule

The Policy Rules defined in Stage 2 are translated as follows:

4. “Disabling of this Profile not allowed”
Subject=”PROFILE”, action=”DISABLE”, qualification=”Not allowed”

5. “Deletion of this Profile not allowed”
Subject=”PROFILE”, action=”DELETE”, qualification=”Not allowed”

6. “Profile deletion is mandatory when it is disabled”
Subject=”PROFILE”, action=”DISABLE”, qualification=”Auto-delete”

Any other combination shall be treated as **not valid** regarding this specification release.

5.1.1.3.3 POL2

The POL2 type is defined by the following data structure:

Data name	Description	Type	No.	MOC
Rules	List of Policy Rules defined for a given Profile.	POL2-RULE	1..N	O

Table 100: POL2 Type

An empty POL2 shall be represented as a POL2 data structure having no rules inside

5.1.1.3.4 PROFILE INFO

The **PROFILE INFO** type is defined by:

Data name	Description	Type	No.	MOC
iccid	Identification of the Profile.	ICCID	1	M
isd-p-aid	The ISD-P-AID of the ISD-P containing the Profile. This is the AID that has been allocated at ISD-P creation time by the SM-SR. The TAR of the ISD-P is included in the ISD-P-AID. See section 2.2.1.3.	AID	1	M
mno-id	Identification of the MNO owner of the Profile. Once this information is associated to the Profile, it remains unchanged during the Profile's life-time.	OID	1	M
fallbackAttribute	Boolean value to indicate the Profile having the Fall-back Attribute set.	Boolean	1	M
subscriptionAddress	The address of the Subscription associated to this Profile.	SUBSCRIPTION-ADDRESS	1	M
state	The current state of the ISD-P containing the Profile as per defined in GSMA Remote Provisioning Architecture for Embedded UICC [1]. The 'Deleted' state is not defined as a possible state; a 'Deleted' ISD-P will simply not appear in the list of eUICC Profiles.	Enumeration{ Created, Enabled, Disabled}	1	M
smdp-id	Identification of the SM-DP that has initially downloaded and installed the Profile. This value can be empty in case the Profile has been loaded during issuance of the eUICC, else the value is mandatory. Once this information is associated to the Profile, it remains unchanged during the Profile's life-time.	OID	1	C
ProfileType	Indicates, through an SM-DP reference, the type of Profile generated by the SM-DP (for example 3G_16K)	String	1	O
allocatedMemory	Indicates the amount of memory allocated to the ISD-P to contain the Profile. Note that the allocated memory is different from the real required memory space for Profile installation; most of the time the allocated memory will be greater than the strict required memory space value. The value is expressed in bytes.	Integer	1	M
freeMemory	Indicates the amount of memory free within Profile allocated space. This information is provided in case of using the quota management mechanism. The value is expressed in bytes.	Integer	1	C
pol2	Contains the POL2 rules defined for this Profile.	POL2	1	M

Table 101: Profile Info

5.1.1.3.5 KEY-COMPONENT

The **KEY-COMPONENT** type is defined by:

Data name	Description	Type	No.	MOC
type	Definition of the key type coding. This defines the algorithm associated with the key, coded on 1 byte. Meaning of the byte value follows GlobalPlatform Card Specifications [6], section 11.1.8. i.e. '88' AES (16, 24, or 32 long keys)	Hexadecimal string representation of exactly 1 byte	1	M
value	The value as a binary data. This data shall be encrypted with a transport key agreed between the provider and the requester.	Hexadecimal string	1	M

Table 102: Key Component

5.1.1.3.6 KEY

The **KEY** type is defined by:

Data name	Description	Type	No.	MOC
index	The Key index as an integer value between 0 and 127 (as defined in GlobalPlatform Card Specifications)	integer	1	M
kcv	The Key Check Value of the key	KCV	1	M
KeyComponents	A simple key is defined using only one Key Component, but it is also possible to define keys with multiple key components (like RSA keys)	KEY-COMPONENT	1..N	M

Table 103: Key Type

NOTE: A key may be:

- a symmetric key. In this case the key will be composed of a single key component. The key value being the same in SM-SR and eUICC SD.
- an asymmetric key. In this case, the key will be most probably be composed of multiple key components. The key value in SM-SR being the counter part of the key value in the eUICC (i.e.: the public key at the SM-SR and the private key in the eUICC or vice-versa)

5.1.1.3.7 CERTIFICATE

The **CERTIFICATE** type is defined by:

Data name	Description	Type	No.	MOC
index	Indicates the index of the private key, being the private counterpart of the certificate. Index is an integer value between 0 and 127 (as defined in GlobalPlatform Card Specifications)	Integer	1	M
ca-id	Identifier of the CA that has issued (and signed) the certificate. This shall match the CA Identifier included in the certificate itself.	OID	1	M
value	Value of the certificate. The certificate shall be coded according to GlobalPlatform Card Specification UICC Configuration [7], section 9.2.1	Hexadecimal string	1	M

Table 104: Certificate Type

5.1.1.3.8 KEYSET

The **KEY SET** type is defined by:

Data name	Description	Type	No.	MOC
version	The version of the key set (as an integer value). The version value of a key set shall be unique within SD definition. Possible values are from 1 to 127. Example: '48' stands for a SCP03 version '30'	Integer	1	M
type	Generally key set usage (SCP03...) can be fully deduced from the key set version. If version information should not be used, this element shall be present to indicate the real usage of this key set.	Enumeration{ SCP03, SCP80, SCP81, TokenGeneration, ReceiptVerification, CA}	1	O
cntr	The counter value linked to the key set. This element is optional: value '0' as to be considered if missing.	Integer	1	O
keys	List of keys contained in the key set	KEY	1..128	C(1)
certificates	A certificate (as defined in GlobalPlatform context) as a counter part of a secret key loaded in a key set.	CERTIFICATE	1..128	C(1)

Table 105: KeySet Type

NOTE: A key set provisioned at SM-SR level may be composed of a set of keys or certificates.

A key set shall include at least one key or certificate. But for a given index, it may exist only one key or one certificate.

5.1.1.3.9 SECURITY-DOMAIN

The **SECURITY-DOMAIN** type is defined by:

Data name	Description	Type	No.	MOC
aid	The AID of the security domain	AID	1	M
tars	The list of TARs allocated to security domain, as an SD may have several TARs. If this list is empty, the implicit TAR is defined by the byte 13, 14, 15 of the AID.	TAR	1..N	O
sin	The security domain Provider Identification Number as defined in GlobalPlatform Card Specification [6]. The owner of the security domain endorsing the Role defined in the 'role' data	Hexadecimal string	1	M
sdin	The security domain Identification Number as defined in GlobalPlatform Card Specification [6]	Hexadecimal string	1	M
role	Identification of the Role of the security domain.	numeration{ISD-R, ECASD}	1	M
keysets	The list of key sets defined within the security domain	KEYSET	1..127	M

Table 106: Security Domain Type

5.1.1.3.10 EUICC-CAPABILITIES

The **EUICC-CAPABILITIES** type allows listing the capabilities supported by the eUICC.

The **EUICC-CAPABILITIES** type is defined by:

Data name	Description	Type	No.	MOC
CAT_TP-Support	If CAT_TP according to ETSI TS 102 127 [25] is supported by the eUICC.	Boolean	1	M
CAT_TP-Version	Shall contain the highest supported release number of ETSI TS 102 127 (defining CAT_TP) that is implemented by the eUICC. Conditional to the support of the CAT_TP-Support. In case of support, the supported version shall be at least the minimum version mandated by the present specification.	Version	1	C
HTTP-Support	If RAM over HTTP according to GlobalPlatform Card Specification Amendment B [8] is supported by the eUICC.	Boolean	1	M
HTTP-Version	Shall contain the highest supported release number of GlobalPlatform Amendment B (defining RAM over HTTP) that is implemented by the eUICC. Conditional to the support of the HTTP-Support. In case of support, the supported version shall be at least the minimum version mandated by the present specification.	Version	1	C
secure-packet-version	Shall contain the highest supported release number of ETSI TS 102 225 (defining secure packet) that is implemented by the eUICC. The support of this feature as defined in ETSI TS 102 225 is not optional. The supported version shall be at least the minimum version mandated by the present specification.	Version	1	M
Remote-provisioning-version	Shall contain the highest supported release number of this specification SGP.02 that is implemented by the eUICC. The support of this feature is obviously not optional. As a consequence, the eUICC shall be compliant will all relevant specifications referenced in this specification. Note 1: If the document version has less than three digits, one or more extra ".0" shall be appended to obtain a value that conforms to the type "Version" defined in Table 97. For example a value of "3.1.0" denotes that the eUICC is compliant with version 3.1 of SGP.02 Note 2: For backwards-compatibility reasons, if an EIS contains a value "1.1.0" for this field, the off-card entity shall assume that the eUICC is compliant with version 3.0 of SGP.02 or earlier.	Version	1	M

Table 107: eUICC Capabilities Type

5.1.1.3.11 AUDIT TRAIL RECORD

The **AUDIT-TRAIL RECORD** type contains the description of a Platform or a Profile Management operation performed by SM-SR or a notification received by SM-SR from the given eUICC.

The **AUDIT-TRAIL-RECORD** type is defined by:

Data name	Description	Type	No.	MOC
EID	The EID type is for representing an eUICC-ID. An eUICC-ID is primarily used in the "Embedded UICC Remote Provisioning and Subscription Management System" to identify an eUICC. See section 2.2.2 for EID description.	EID	1	M
SMSRid	SMSRid defined SM-SR storing given eUICC	OID	1	M
operationDate	Date and time of logged operation	DATETIME	1	M
operationType	Notification Type as defined in section 4.1.1.11 or Command Type as defined below.	Integer	1	M
requesterId	Identification of the entity that has requested the operation to be performed on the eUICC	OID	1	C
status	For command type Function Execution Status as defined in section 5.1.5 is stored	ExecutionStatus	1	C
ISD-P-AID	The ISD-P-AID of the ISD-P containing the Profile. Empty in case it is not applicable for given operation type	AID	1	C
ICCID	The ICCID type is for representing an ICCID (Integrated Circuit Card IDentifier). The ICCID is primarily used to identify a Profile. ICCID is defined according to ITU-T recommendation E.118 [21].	String	1	C
IMEI	See ETSI TS 102 223 [3], clause 8.20	Hexadecimal String	1	C
MEID	See ETSI TS 102 223 [3], clause 8.81	Hexadecimal String	1	C

Table 108: Audit Trail Record Type

NOTE: Requester Id OID is empty in case of notification.

5.1.1.3.12 Command Type

Command type coding:

- '0100': CreateISDP
- '0200': EnableProfile
- '0300': DisableProfile
- '0400': DeleteProfile
- '0500': eUICCCapabilityAudit
- '0600': MasterDelete
- '0700': SetFallbackAttribute
- '0800': EstablishISDRkeyset
- '0900':FinaliseISDRhandover
- '0A00' to 'FF00' RFU

NOTE: 1st byte is reserved for Notification Type as defined in section 4.1.1.11

5.1.1.3.13 EIS

The **EIS** type is for representing eUICC Information Set.

Data name	Description	Type	No.	MOC
eid	Identification of the eUICC. See section 5.1.1.1 for type description. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime.	EID	1	M
eum-id	Identification of the eUICC Manufacturer (i.e. Card Vendor) that has manufactured the eUICC. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime. The 'eum-id' indication, jointly with the 'platformType' and 'version, may especially be useful for the SM-DP to perform the Profile generation and packaging.	OID	1	M
productionDate	The date/time where the eUICC has been manufactured by the card vendor. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime.	DATETIME	1	M
platformType	Indication of the eUICC platform/OS type. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime. The content of this field is not enforced by this specification; the EUM can use any convenient string value.	String	1	M
platformVersion	Indication of the version of the eUICC platform/OS type. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime.	VERSION	1	M
remainingMemory	Indicates the current total available memory (whatever the underlying technology, flash or eeprom) for Profile download and installation. This value may be either:- • a value cached by the SM-SR based on the initial total memory and memory required by all Profiles currently loaded on the eUICC. • a value retrieved from the eUICC The value is expressed in Bytes. This information is initially provided by the EUM at registration time, but may change according to the eUICC usage.	Integer	1	M
availablememoryforprofiles	Indicates the free memory (whatever the underlying technology, flash, eeprom) without any Profile, available for Profile(s) loading and installation. This value is initially provided by the EUM at the registration time. It is calculated when the ISD-R and the ECASD are created, instantiated and personalized. This value can evolve during the card life cycle when a patch or a filter is applied or when the ECASD or the ISD-R configuration is modified. This value shall be updated each time the ISD-R, ECASD or the OS is modified.	Integer	1	M

lastAuditDate	Some information part of the EIS can be refreshed by requesting directly the information to the eUICC to have the list of information that can be retrieved. This indicates the last date where such operation has been performed, and so indicating the freshness of the information stored at SM-SR level. This information is optional. If not present, it means that no audit has been performed on the eUICC.	DATE	1	O
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M
isd-p-loadfile-aid	AID of the Executable Load File to be used for instantiation of an ISD-P. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime.	AID	1	M
isd-p-module-aid	AID of the Executable Module to be used for instantiation of an ISD-P. This information is initially provided by the EUM at registration time, and remains unchanged during all the eUICC's lifetime.	AID	1	M
profiles	List of Profiles currently installed on the eUICC. This information is initially provided by the EUM at registration time, and may change during the eUICC's lifetime.	PROFILE	1..N	M
ISD-R	Contains the information related to the ISD-R	SECURITY-DOMAIN	1	M
ECASD	Contains the information related to the ECASD	SECURITY-DOMAIN	1	M
eUICC-Capabilities	Contains the capabilities supported by the eUICC. This information is initially provided by the EUM at registration time.	EUICC-CAPABILITIES	1	M
audit trail	History of all the platform and Profile Management operations or eUICC notifications related to the eUICC	AUDIT-TRAIL-RECORD	0..N	M
eumCertificateId	Indicates the EUM Certificate that has been used to perform the signature. This data contains the "Serial Number" of the certificate.	String	1	M

signatureAlgorithm	Indicates the signature algorithm used by the EUM to sign the relevant part of the EIS. See Annex E to have details of the data that shall be included in the signature.
	The algorithm naming follows RFC 4051 [24]
		Enumeration{
		rsa-sha256, rsa-sha384,
		rsa-sha512,
		ecdsa-sha256,
		ecdsa-sha384,
		ecdsa-sha512
		}
			1	M

Signature	Signature value of the EUM. See Annex E to have details of the data included in the computation of the signature	Byte[]	1	M

Table 109: EIS Type

NOTE: The ISD-P(s) are not represented in the EIS as a pure SECURITY-DOMAIN data type; ISD-P information is directly included in the Profile representation without distinction as the SM-SR doesn't have access to ISD-P credentials.

5.1.2 Request-Response Function

A request-response function functionally corresponds to the case where a requestor Role sends a request message to a replier Role which receives and processes the request, ultimately returning a message in response. A function may take input data and may provide output data. A function may also deliver no output data.

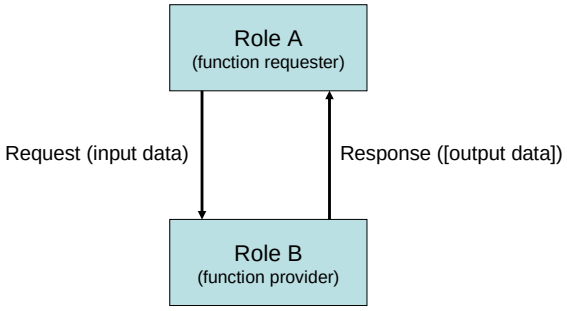


Figure 35: **Functions as a Request-Response Data Exchange**

At function definition level nothing is said about if the function is synchronous or asynchronous.

5.1.2.1 Validity Period

When a function is called, the function provider takes the responsibility to execute all the individual execution steps that are required to complete the function. Such processing may require some time to complete, but the function caller might want this processing duration to not exceed a specific amount of time called the "function validity period", as detailed in the following use cases:

- The function processing might no longer be valuable if it ends after the validity period. For example, a function is only valuable if it is executed within a minute. If more than a minute has elapsed, then it is no longer required to continue the function execution.
- Processing might not want to wait for an external event that might not occur before a very long time or an event that might even never occur at all. For example, it is possible when performing an OTA dialog that the Device is unreachable (switched off, lost…), or that an acknowledgement message coming from the Device is lost on the network (for example the loss of a PoR coming from an eUICC). If so, it might not be acceptable to wait several days or weeks for the Device to be switched on again, or even to wait forever for an acknowledgement message that will never come.
- It is desirable that the function provider system is not overloaded with requests that will be pending for a long period. The function caller would like to be notified as soon as possible that the function cannot be processed within a specific amount of time, and may then implement a calling side retry Policy.

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.

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.

Input data name	Description	Type	No.	MOC
Function Requester Identifier	Identification of the function requester.	String	1	M
Function Call Identifier	Identification of the function call. This identifier enables to manage function call retry policies. When requesting for the execution of a function, the function caller shall provide a unique Function Call Identifier. Uniqueness is to be ensured in its own perimeter. In case the function caller wants to retry the same function, then it shall perform the same function call, providing the same Function Call Identifier. On function provider side, when receiving this retry attempt, if a call to a function if performed with an Function Call Identifier of a function already in process in its system, then the function provider shall refuse the new call If the function provider does not want to implement any retry Policy, then it might ignore this field. The Function Call identifier is only mandatory for request-response functions. It shall not be present for notification functions.	String	1	C
Validity Period	This field defines the length of the period (provided as a **number of seconds**) during which the request is valid. The period starts at the time the function call was received by the function provider and ends a number of seconds later. During this period of time, the function provider has the responsibility to execute the function. The function provider, on reception of the function call, may: • **Accept** the function call: in that case the function provider accepts the provided validity period • **Reject** the function call: if the function provider immediately considers that the validity period is invalid (for example too long or too short) or cannot fulfil the requirements (i.e. cannot start the sequence of operations so that all of the operations are completed within the validity period), it shall not process the function and shall immediately return a Function Execution Status output parameter with a Status field set to 'Failed' and a Subject code and Reason code of the Status Code Data field set to 'Validity Period not accepted'. The function provider shall also indicate to the function caller an acceptable amount of time into which the request could be fulfilled, by setting the Acceptable Validity Period field in the output header. The Validity Period is only present (but optional) for request-response functions. If not specified, the function caller doesn't require any specific validity period. Nevertheless the function provider is free to apply any internal rule to restrict the validity period (it could be the case to ensure that a function request will never stay stacked in the system). In that case the function provider shall indicate to the function caller the applied validity period value in the Acceptable Validity Period field in the output header. The Validity Period shall not be present for notification functions.	Integer	1	O

Table 110: Validity Period Functions Identifier

5.1.2.2 Exceptions

During the processing of a function, an unexpected behaviour may happen. This event, called an exception in this specification, may cause the function to be ended before the functional work to be completed (the exception is then considered as an error), or may let the functional work continue, but under specific conditions (the exception is then called a warning).

This is the function provider's responsibility to give information on any exception encountered during the processing of a function; however the behaviour of the function caller when receiving this exception may depend on its own context (for example stop its current processing, or perform a retry attempt, or try a workaround processing, etc.)

5.1.3 Notification Handler function

In some cases, functions are considered as notifications as they functionally correspond to events sent from one Role to another. If so, the Role that generates the notification is called the notification source or the notifier, and the Role that receives the notification is called the notification destination or the notification recipient or notification handler.

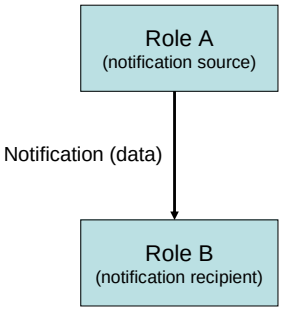


Figure 36: **Notification as One Way Events**

By definition, no validity period is applied for a notification, and no data can be returned back by the notification recipient to the notification source.

Similarly, no exception is expected in the context of a notification.

5.1.4 Functions Input Header

All functions (request-response and notification handler) shall include the following header as part of the input data:

Input data name	Description	Type	No.	MOC
Function Requester Identifier	Identification of the function requester.	String	1	M
Function Call Identifier	Identification of the function call. This identifier enables to manage function call retry policies. When requesting for the execution of a function, the function caller shall provide a unique Function Call Identifier. Uniqueness is to be ensured in its own perimeter. In case the function caller wants to retry the same function, then it shall perform the same function call, providing the same Function Call Identifier. On function provider side, when receiving this retry attempt, if a call to a function if performed with an Function Call Identifier of a function already in process in its system, then the function provider shall refuse the new call If the function provider does not want to implement any retry Policy, then it might ignore this field. The Function Call identifier is only mandatory for request-response functions. It shall not be present for notification functions.	String	1	C
Validity Period	This field defines the length of the period (provided as a **number of seconds**) during which the request is valid. The period starts at the time the function call was received by the function provider and ends a number of seconds later. During this period of time, the function provider has the responsibility to execute the function. The function provider, on reception of the function call, may: • **Accept** the function call: in that case the function provider accepts the provided validity period • **Reject** the function call: if the function provider immediately considers that the validity period is invalid (for example too long or too short) or cannot fulfil the requirements (i.e. cannot start the sequence of operations so that all of the operations are completed within the validity period), it shall not process the function and shall immediately return a Function Execution Status output parameter with a Status field set to 'Failed' and a Subject code and Reason code of the Status Code Data field set to 'Validity Period not accepted'. The function provider shall also indicate to the function caller an acceptable amount of time into which the request could be fulfilled, by setting the Acceptable Validity Period field in the output header. The Validity Period is only present (but optional) for request-response functions. If not specified, the function caller doesn't require any specific validity period. Nevertheless the function provider is free to apply any internal rule to restrict the validity period (it could be the case to ensure that a function request will never stay stacked in the system). In that case the function provider shall indicate to the function caller the applied validity period value in the Acceptable Validity Period field in the output header. The Validity Period shall not be present for notification functions.	Integer	1	O

Table 111: Functions Input Headers

Additionally to this common header, each function may define its own set of additional input data.

5.1.5 Functions Output Header

All functions (request-response) shall include the following header as part of the output data. Notifications don't have any output data.

Output data name	Description	Type	No.	MOC
`Processing Start`	The start time and date of the real processing of the function by the function provider (and not the time and date of reception of the request).	DATETIME	1	O
`Processing End`	The function processing end time and date.	DATETIME	1	O
`Acceptable Validity Period`	In case the validity period provided as input parameter is not acceptable, then the function provider shall return an acceptable value to the function caller (see section 5.1.4) as a **number of seconds**. In case the function call has been rejected because of a non acceptable validity period the function caller might then call again the same function with a validity period that is more convenient (but that may however differ from the exact value of the Acceptable Validity Period field sent in response to the previous function call).	Integer	1	C
`Function Execution Status`	Indicates whether the processing has been completed correctly or not. If required, provides information to give details on the processing result (status code, status code reason, status message…). The Execution Status type is described below.	ExecutionStatus	1	M

Table 112: Functions Output Headers

Where an Execution Status is:

Data name	Description	Type	No.	MOC
Status	It indicates whether the processing has been completed correctly or not. Value 'Executed-Success' means that the function has been processed correctly. Application output data MAY optionally be part of the function response. Value 'Executed-WithWarning' means that the function has been processed correctly, but that warnings have been generated during this execution. Application output data MAY optionally be part of the function response in order to provide details on the warnings. Value 'Failed' means that the function execution has encountered errors during its processing. The Status Code Data output structure shall give the reason of error in the processing (values depend on the function and may be implementation dependant). Value 'Expired' means that the validity period of the request has expired before the completion of the function processing. The Status Code Data output structure MAY give the reason of expiration of the function.	Enumeration {Executed-Success, Executed-WithWarning, Failed, Expired}	1	M
Status code data	It provides the reason of the Status. Present only if the Status is 'Execute-WithWarning', 'Failed', or 'Expired'.	Status code data	1	O

	The Status Code Data type is described below.			

Table 113: Execution Status

Where a Status code data is:

Data name	Description	Type	No.	MOC
Subject code	Represents the system element concerned by the exception. A normative list of subjects is given in section 5.1.6.1	OID	1	M
Reason code	Represents the reason of the exception. A normative list of reasons is given in section 5.1.6.2	OID	1	M
Subject identifier	The identifier of the subject or any identification data of the subject that caused the exception (for example ICCID of the Profile when the Subject is a "Profile"). The possible values of the Subject Identifier depend on the function.	String	1	O
Message	It provides a textual and human readable explanation of the exception. The Message value is implementation dependant	String	1	O

Table 114: Status Code

5.1.6 Status Code

Status codes are used in a function call to indicate that an exception occurred during the processing of the function.

The “status code” is part of the Function output header (as defined in section 5.5.5). In this specification, the status codes are representing any exception from a simple warning to an error.

- When an error is raised (function output header status is ‘Failed’), it means that the expected functional behaviour has not been completed.
- When a warning is raised (function output header status is ‘Executed-WithWarning’’), it means that the expected functional behaviour has been completed, but under specific conditions that should be pointed out by the function provider.

Both Subject code and Reason code fields of the Status code data of the function output header are represented by an OID (Object IDentifier). These identifiers refer to a list of pre-defined elements and reasons (see below for details).

5.1.6.1 Subject Code

The Subject code represents, from the function provider perspective, the entity on which the exception occurred. The subject code can either be its own system (for example: an internal error), a part of the system (for example: eUICC, Profile …) or even the function caller itself (for example: Identification issue).

GlobalPlatform System, Messaging Specification for Management of Mobile-NFC Services [23] already defines some subject codes that are organised as a tree

representation. This specification proposes to reuse the category “1. Generic” as defined in [23].

The subjects codes linked with the “Remote Provisioning Architecture for Embedded UICC”, are regrouped under a dedicated category, which has the identifier value “8. eUICC Remote Provisioning” to avoid any conflict with the categories already defined in [23].

The possible values for the Subject code used in the context of this specification are defined as follow:

1. Generic
    - 1.1. Function Requester
    - 1.2. Function Provider
        - 1.2.1. Validity Period
    - 1.3. Protocol
        - 1.3.1. Protocol Format
        - 1.3.2. Protocol Version
    - 1.4. External Resource
    - 1.5 Extension Resource
    - 1.6 Function
    - 8. eUICC Remote Provisioning
    - 8.1 eUICC
        - 8.1.1 EID
    - 8.2 Profile
        - 8.2.1 Profile ICCID
        - 8.2.2 POL1
        - 8.2.3 POL2
        - 8.2.4 Void
        - 8.2.5 Profile Type
        - 8.2.6 Subscription Address
    - 8.3 ISD-P
        - 8.3.1 ISD-P-AID
    - 8.4 ISD-R
    - 8.5 ECASD
        - 8.5.1 Certification Request
        - 8.5.2 Embedded UICC Certificate Authority
    - 8.6 EIS
    - 8.7 SM-SR

5.1.6.2 Reason Code

The Reason code represents, from the function provider perspective, the reason why the exception occurred.

As for Subject Code, GlobalPlatform System, Messaging Specification for Management of Mobile-NFC Services [23] already defines some reason codes that are organised as a tree representation. This specification proposes to reuse the following categories coming from [23]:

5. Access error
6. Format error
7. Conditions of use not satisfied
8. Processing error
9. Transport error
10. Security error

The possible values for the Reason code are defined as follow:

1. Access Error
   - 1.1. Unknown (Identification or Authentication)
   - 1.2. Not Allowed (Authorisation)
2. Format Error
   - 2.1. Invalid
   - 2.2. Mandatory Element Missing
   - 2.3. Conditional Element Missing
3. Conditions of Use Not Satisfied
   - 3.1. Unsupported
   - 3.2. Maximum Size Exceeded
   - 3.3. Already in Use (Uniqueness)
   - 3.4. Invalid Destination
   - 3.5. Invalid Transition
   - 3.6. Related Objects Exists
   - 3.7. Unavailable
   - 3.8. Refused
   - 3.9 Unknown
4. Processing Error
   - 4.1. Function Already in Progress
   - 4.2. Execution Error
   - 4.3. Stopped on Warning
   - 4.4. Busy
   - 4.5. Operation Already Processed
   - 4.6. Not Present / Missing
   - 4.7. Generation Not Possible

4.8. Insufficient Memory

4.9. Unassigned

5. Transport Error

5.1. Inaccessible

5.2. Timeout

5.3. Time to Live Expired

5.4 Delivered With No Response

5.5 Connection Lost

6. Security Error

6.1. Verification Failed

6.2. Decipher Failed

5.1.6.3 Status Code Example

Identification issue example:

State: The function requester tries to access a function, but its credentials are not known to the function provider

Function processing: The function provider raises an internal exception, as the function requester couldn't be identified

Returned Status Code:

- Subject code: **1.1** – Function requester
- Reason code: **1.1** – Unknown

Platform Management issue:

State: The function requester tries to create a new ISD-P, but with an ICCID already in used for another Profile

Function processing: The function provider raises an internal exception, as there is a conflicting AID.

Returned Status Code:

- Subject code: **8.2.1** – Profile ICCID
- Reason code: **3.3** – Already in use

5.1.6.4 Common Function Status Code

The following table provides the normative list of status codes that may be raised by any function defined in this specification. These statuses shall be implemented.

In addition each function may raise additional specific status codes. In that case, it is defined explicitly in the function description.

As an implementer's choice, it is also possible that a function may return additional status codes not described in this specification. The function caller shall be ready to handle such situation.

Common status code when 'Function execution status' is 'failed'

Subject code	Subject	Reason code	Reason	Description
1.1	Function requester	1.1	Unknown (Identification or Authentication)	The function caller is unknown to the function provider.
1.1	Function requester	1.2	Not allowed (authorisation)	The function caller is not allowed to use this function.
1.2	Function provider	4.2	Execution error	Internal processing error (this status code shall be returned only when no more accurate status code can be returned)
1.2	Function provider	4.4	Busy	Busy: not possible to process the function for the moment
1.2.1	Validity period	3.8	Refused	The requested validity period is not accepted by the function provider.
1.6	Function	2.1	Invalid	An input parameter of the function is invalid (wrong format, not acceptable value…). The contextual message conveyed with the status code data shall indicate the name of the concerned parameter.
1.6	Function	2.2	Mandatory Element Missing	A mandatory input parameter of the function is missing. The contextual message conveyed with the status code data shall indicate the name of the concerned parameter.
1.6	Function	2.3	Conditional Element Missing	A conditional input parameter of the function is missing. The contextual message conveyed with the status code data shall indicate the name of the concerned parameter.

Table 115: Function Execution Status 'Failed' Codes

Common status code when 'Function execution status' is 'Expired'

Subject code	Subject	Reason Code	Reason	Description
1.6	Function	5.3	Time to live expired	The function execution request has expired (end of validity period has been reached). This may be because the server had no time to execute the function or because the function was requesting a remote communication with the eUICC which was not present on the network during all the validity period.

Table 116: Function Execution Status 'Expired' Codes

5.2 ES1 (EUM – SM-SR) Interface Description

5.2.1 Register EIS

Function name: RegisterEIS

Related Procedures: eUICC registration at SM-SR: register a new EIS

Function group: eUICC Management

Function Provider: SM-SR

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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
EIS	This is the eUICC Information Set of the eUICC. See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1-N	M

Table 117: Register EIS Additional Input Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.6	EIS	1.3	Already registered	Indicates that the EIS identified by this EID, is already register within the EIS database of the SM-SR
8.6	EIS	1.4	Error in signature calculation	During the verification of the EIS signature, an error occurred.
8.6	EIS	1.5	Data inconsistency	During the consistency review of the EIS data, an error was found (for example free memory is bigger than full memory)

Table 118: Register EIS Specific Status Codes

5.3 ES2 (MN0 – SM-DP) Interface Description

5.3.1 Getting eUICC Information

Function name: GetEIS

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-DP

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC to be audited. See section 5.1.1.1 for type description.	EID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M

Table 119: Get EIS Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
eis	The relevant eUICC Information Set of the eUICC See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M

Table 120: Get EIS Additional Output Data

Specific status codes

In addition to those returned by **ES3.UpdatePolicyRules**, this function may return:

Subject code	Subject	Reason code	Reason	Description
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be solved by the SM-DP

Table 121: Get EIS Specific Status Codes

5.3.2 Download a Profile

Function name: DownloadProfile

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-DP

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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the target eUICC. See section 5.1.1.1 for type description.	EID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M
profileType	Identification of the Profile type to download and install in the eUICC.	String	1	C
iccid	Identification of the Profile to download and install. See section 5.1.1.1 for type description.	ICCID	1	C
enableProfile	Indicates if the Profile shall be enabled after downloading and installation.	BOOLEAN	1	M

Table 122: Download Profile Additional Input Data

NOTE: MNO can either provide ICCID or the Profile type. In case the Profile type is provided, the SM-DP is free to select one of the Profiles that matches the Profile type.

Additional output data:

Output data name	Description	Type	No.	MOC
iccid	Indicates the Profile ICCID that has been downloaded and installed	ICCID	1	C
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES5 depending of the requested function.	Hex binary	1	O

Table 123: Download Profile Additional Output Data

Specific status codes

Subject Code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the SM-SR.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile, identified by this iccid is unknown to the SM-DP.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.5	Profile Type	3.9	Unknown	Indicates that the Profile type identified by this profileType is unknown to the SM-DP.
8.2.5	Profile Type	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the ProfileType.
8.4	ISD-R	4.2	Execution error	Error during execution of the enabling command on the eUICC. The output data “euiccResponseData contains the exact response coming from the eUICC except in case of fallback.
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be resolved by the SM-SR

Table 124: Download Profile Additional Output Data

5.3.3 Updating the Policy Rules of a Profile

Function name: UpdatePolicyRules

Related Procedures: -

Function group: Profile Management

Function Provider: SM-DP

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile See section 5.1.1.1 for type description.	ICCID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M
pol2	The POL2 to associate with the identified Profile. See section 5.1.1.1 for type description.	POL2	1	M

Table 125: Update Policy Rules Additional Input Data

Additional output data:

None

Specific status codes

In addition to those returned by **ES3.UpdatePolicyRules**, this function may return:

Subject code	Subject	Reason code	Reason	Description
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be solved by the SM-DP

Table 126: Update Policy Specific Status Codes

5.3.4 Updating eUICC Information

Function name: UpdateSubscriptionAddress

Related Procedures: Profile Download and Installation, Profile Enabling, Profile Enabling via SM-DP

Function group: Profile Management

Function Provider: SM-DP

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. For consistency within

the system, it is the responsibility of the caller to ensure that all data is provided. The SM-DP shall forward the function request to the SM-SR "**ES3.UpdateSubscriptionAddress**" as defined in section 5.4.7.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC to be audited. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the target Profile. See section 5.1.1.1 for type description.	ICCID	1	M
newSubscriptionAddress	The new Subscription Address	Subscription Address	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M

Table 127: Update Subscription Address Additional Input Data

This function has no additional output data.

Specific status codes

In addition to those returned by **ES3.UpdateSubscriptionAddress**, this function may return:

Subject code	Subject	Reason code	Reason	Description
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be resolved by the SM-DP

Table 128: Update Subscription Address Specific Status Codes

5.3.5 Profile Enabling

Function name: EnableProfile

Related Procedures: Profile Enabling via SM-DP

Function group: Platform Management

Function Provider: SM-DP

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 or a specific status code as defined in the table here after.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M
iccid	Identification of the Profile to enable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 129: Enable Profile Additional Input Data

Additional output data:

- None

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the SM-SR.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the SM-SR.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the SM-SR but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the impacted Profiles doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the impacted Profiles doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the enabling command on the eUICC.
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be resolved by the function provider.

Table 130: Enable Profile Specific Status Codes

5.3.6 Profile Disabling

Function name: DisableProfile

Related Procedures: Profile Disabling via SM-DP

Function group: Platform Management

Function Provider: SM-DP

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 '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 or a specific status code as defined in the table here after

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M
iccid	Identification of the Profile to disable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 131: Disable Profile Additional Input Data

Additional output data:

None

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1	eUICC	3.8	Refused	Indicates that the target Profile can't be disabled. (for example the Profile is the only Profile in the eUICC)
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the SM-SR.

8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the SM-SR.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the SM-SR but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the target Profile doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the target Profile doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the disabling command on the eUICC.
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be solved by the SM-SR

Table 132: Disable Profile Specific Status Codes

5.3.7 Delete a Profile

Function name: DeleteProfile

Related Procedures: Profile and ISD-P Deletion

Function group: Platform Management

Function Provider: SM-DP

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile to disable. See section 5.1.1.1 for type description.	ICCID	1	M
smsr-id	Identification of the SM-SR currently in charge of eUICC management. This information may change during the eUICC's lifetime.	OID	1	M

Table 133: Delete Profile Additional Input Data

Additional output data:

None

Specific status codes

In addition to those returned by **ES3.DeleteISDP**, this function may return:

Subject code	Subject	Reason code	Reason	Description
8.7	SM-SR	3.9	Unknown	Indicates that the SM-SR, identified by this smsr-id, is unknown to or whose address cannot be resolved by the SM-DP

Table 134: Delete Profile Specific Status Codes

5.3.8 Notify a Profile is Disabled

Function name: HandleProfileDisabledNotification

Related Procedures: Profile Download and Installation, Profile Enabling via SM-DP, Profile Enabling, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: MNO

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been disabled. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 135: Handle Profile Disabled Notification Additional Input Data

5.3.9 Notify a Profile Enabling

Function name: HandleProfileEnabledNotification

Related Procedures: Profile Disabling and Profile Disabling via SM-DP, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: MNO

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. What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been enabled. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 136: Handle Profile Enabled Notification Additional Input Data

5.3.10 Notify a SM-SR Change

Function name: HandleSMSRChangeNotification

Related Procedures: SM-SR Change

Function group: eUICC Management

Notification handler/recipient: MNO

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.

This notification is not related to a particular Profile. It is up to the notification recipient to perform any action related to each Profile that is deployed on this eUICC.

Additional input data:

Input data name	Description	Type	No.	MOC
eis	The relevant part of the eUICC Information Set linked with the MNO owning the Profile hosted in the eUICC. See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 137: Handle SM-SR Change Notification Additional Input Data

No output data is expected in response to this notification.

5.3.11 Notify a Profile Deletion

Function name: HandleProfileDeletedNotification

Related Procedures: Profile Enabling, Profile Enabling via SM-DP

Function group: Platform Management

Notification handler/recipient: MNO

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. What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been deleted. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 138: Handle Profile Deleted Notification Additional Input Data

5.4 ES3 (SM-DP – SM-SR) Interface Description

5.4.1 Getting eUICC Information

Function name: GetEIS

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-SR

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 (for example 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M

Table 139: Get EIS Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
eis	The relevant eUICC Information Set of the eUICC. See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	C

Table 140: Get EIS Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EID is unknown to the function provider
8.6	EIS	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage this EIS, identified by this EID.

Table 141: Get EIS Specific Status Codes

5.4.2 Auditing eUICC Information

Function name: AuditEIS

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function provider: SM-SR

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC to be audited. See section 5.1.1.1 for type description.	EID	1	M

Table 142: Audit EIS Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
eis	The relevant eUICC Information Set of the eUICC See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M

Table 143: Audit EIS Additional Output Data

Specific status codes

Subject Code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.6	EIS	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage this EIS, identified by this EID.
1.6	Function	5.4	Delivered With No Response	The function execution request has been delivered to remote entity but no response is received.

Table 144: Audit EIS Specific Status Codes

5.4.3 Create a New ISD-P in an eUICC

Function name: CreateISDP

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-SR

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.

In case one of these conditions is not satisfied, the SM-SR shall refuse the function request and return a ‘Function execution status’ indicating ‘Failed’ with the relevant status code (see table below).

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 with following values:

- The iccid value received as parameter
- The isd-p-aid value as allocated by the SM-SR

- The mno-id value received as parameter
- The state value as 'Created'
- The smdp-id retrieved from the authentication context of the caller
- The allocated memory value received as parameter

NOTE: The initial Subscription Address and the initial POL2 can be set after the Profile is completely downloaded using the "**ES3.ProfileDownloadCompleted**" function.

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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
Eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
Iccid	Identification of the Profile to download and install. See section 5.1.1.1 for type description.	ICCID	1	M
mno-id	The identification of the MNO owning the Profile	OID	1	M
RequiredMemory	Indicates the amount of memory allocated to the ISD-P to contain the Profile. The value is expected in Bytes.	Integer	1	M
moreToDo	Indicates to the function provider that the function caller has something else to do with the targeted eUICC right after this function execution. This indication may be used by the function provider to decide if it has to keep the remote communication channel with the eUICC open (this may be relevant or not, depending on the remote communication channel. This is the case for instance for Remote Administration over HTTPS as defined in section 2.4.4. The only purpose is to optimise resource management and save execution time of the overall procedure. It is up to the function provider to support this feature or not. This input data is optional; if missing the function provider shall consider that the function caller has nothing else to do.	Boolean	1	O

Table 145: Create ISD-P Additional Input Data

NOTE: In ES5.CreateISDP function, if the "RequiredMemory" parameter is equal to '00', no "Cumulative Granted Non Volatile Memory" parameter shall be used in INSTALL command.

Additional output data:

Output data name	Description	Type	No.	MOC
isd-p-aid	The AID, allocated by the SM-SR, of the ISD-P containing the Profile. The Tar value is included in the AID. See Annex G "Coding of the PIX for 'Embedded UICC Remote Provisioning and Management' (Normative)".	AID	1	M
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES8 depending of the requested function.	Hexadecimal string	1	O

Table 146: Create ISD-P Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.2.1	Profile ICCID	3.3	Already in use	Indicates that the ICCID is already allocated to another Profile managed by the function provider.
8.4	ISD-R	4.2	Execution error	Error during execution of the creation command on the eUICC. In that case, the output data "euiccResponseData contains the exact response coming from the eUICC.
8.1	eUICC	4.8	Insufficient memory	The eUICC has not enough free memory to execute the creation of the new ISD-P with this required amount of memory.

Table 147: Create ISD-P Specific Status Codes

5.4.4 Download a New Profile

Function name: SendData

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-SR

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 through the SM-SR in charge of the management of the targeted eUICC; eUICC being identified by its EID.

<u>Function flow</u>

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.

In case one of these conditions is not satisfied, the SM-SR shall refuse the function request and return a 'Function execution status' indicating 'Failed' with the relevant status code (see table below).

This function allows sending commands defined in the ES8 interface in several steps. This may be necessary in case of the data is too big compared to eUICC capabilities. It is up to the function caller to determine if it has to handle this situation based on the eUICC capabilities described in EIS.

The SM-SR is free to select the most relevant OTA protocol to communicate up to the eUICC. As a consequence, the data format provided by the function caller shall not depend of the selected OTA protocol capabilities (for example SM-DP can consider there is no limit on data length). 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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
sd-aid	Identification of the SD which shall process the APDUs contained in the data argument. sd-aid could identificatify the ISD-P or the ISD-R.	AID	1	M
data	The data to send into the targeted ISD-P and eUICC. The data shall contain a list a C-APDU as defined in ETSI TS 102 226 [5], section 5.2.1. C-APDU can contain any of the commands defined in ES8 interface. The commands shall be secured according to section 2.5.	Hexadecimal string	1	M
moreToDo	See section 5.4.3 for description of this input data	Boolean	1	O

Table 148: Send Data Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the Random Challenge (RC) in case of the key establishment procedure or the detailed error returned by the eUICC in case of one command execution failed on eUICC. The response data is defined in ES8 depending of the requested function.	Hexadecimal string	1	C

Table 149: Send Data Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.3.1	SD-AID	3.9	Unknown	Indicates that the ISD-P or the ISD-R identified by this SD-AID is unknown to the function provider.
8.3.1	SD-AID	3.4	Invalid destination	Indicates that the ISD-P or the ISD-R identified by this SD-AID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.3	ISD-P	4.2	Execution error	Error during execution of one command, when error occurs at ISD-P level.

Table 150: Send Data Specific Status Codes

5.4.5 Indicating the Profile Download is Completed

Function name: ProfileDownloadCompleted

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-SR

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.

This function allows optionally to set a first Subscription Address, typically the MSISDN, and saves it in the EIS, and optionally a first POL2 associated to the newly download Profile. In case no POL2 is provided at that time, it means that the Profile won't be protected by any POL2 at SM-SR side. But the POL2 may be set or updated at any time later using the "UpdatePolicyRules" function defined in section 5.4.6.

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 Subscription Address may be set or updated at any time later using the "**UpdateSubscriptionAddress**" function defined in section 5.4.7.

On reception of this function request the SM-SR shall immediately update the EIS to set the identified Profile:

- (Optional) the provided ProfileType as defined in section 5.1.1.3.4
- (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". The SM-DP may call the function "**ES3.EnableProfile**" (see section 5.4.8) to enable the Profile if required by the MNO.

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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile See section 5.1.1.1 for type description.	ICCID	1	M
ProfileType	Indicates, through an SM-DP reference, the type of Profile generated by the SM-DP (for example 3G_16K)	String	1	O
subscriptionAddress	The Subscription Address related to the identified Profile	SUBSCRIPTION-ADDRESS	1	O
pol2	The POL2 to associate with the identified Profile.	POL2	1	O

Table 151: Profile Download Completed Additional Input Data

Additional output data:

No additional data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider

8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.3	POL2	2.1	Invalid	Indicates that the POL2 is invalid

Table 152: Profile Download Completed Specific Status Codes

5.4.6 Updating the Policy Rules of a Profile

Function name: UpdatePolicyRules

Related Procedures: -

Function group: Profile Management

Function Provider: SM-SR

Description: This function allows the SM-DP authorised 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. It means that it is the responsibility of the caller to provide the complete definition of 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile See section 5.1.1.1 for type description.	ICCID	1	M

pol2	The POL2 to associate with the identified Profile. See section 5.1.1.1 for type description.	POL2	1	M

Table 153: Update Policy Rules Additional Input Data

Table 154: Void

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.3	POL2	2.1	Invalid	Indicates that the POL2 is invalid.

Table 155: Update Policy Rules Specific Status Codes

5.4.7 Updating eUICC Information

Function name: UpdateSubscriptionAddress

Related Procedures: Profile Download and Installation, Profile Enabling, Profile Enabling via SM-DP

Function group: Profile Management

Function Provider: SM-SR

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. For consistency within the system, it is the responsibility of the caller to ensure that all data is provided. 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the target Profile. See section 5.1.1.1 for type description.	ICCID	1	M
newSubscriptionAddress	The new Subscription Address	Subscription Address	1	M

Table 156: Update Subscription Address Additional Input Data

Additional output data:

This function has no additional output data:

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EIS identified by this EID, is unknown to the function provider
8.2.1	ICCID	1.1	Unknown	Indicates that the Profile identified by the ICCID, is unknown to the function provider
8.2.6	Subscription Address	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage the Subscription Address.

Table 157: Update Subscription Address Specific Status Codes

5.4.8 Profile Enabling

Function name: EnableProfile

Related Procedures: Profile Enabling via SM-DP

Function group: Platform Management

Function Provider: SM-SR

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 authorised 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 of a specific status code as defined in the table here after.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile to enable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 158: Enable Profile Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES8 depending of the requested function.	Hexadecimal String	1	O

Table 159: Enable Profile Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the impacted Profiles doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the impacted Profiles doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the enabling command on the eUICC. In that case, the output data "euiccResponseData" contains the exact response coming from the eUICC.

Table 160: Enable Profile Specific Status Codes

5.4.9 Profile Disabling

Function name: DisableProfile

Related Procedures: Profile Disabling via SM-DP

Function group: Platform Management

Function Provider: SM-SR

Description: This function allows the SM-DP authorised 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 '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 or a specific status code as defined in the table here after

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile to disable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 161: Disable Profile Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case of the function execution failed at the eUICC.	Hexadecimal String	1	O

Table 162: Disable Profile Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1	eUICC	3.8	Refused	Indicates that the target Profile can't be disabled. (for example the Profile is the only Profile in the eUICC)
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the SM-SR.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to SM-SR.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the SM-SR but installed on another eUICC than the one identified by the function

				caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the target Profile doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the target Profile doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the disabling command on the eUICC. In that case, the output data "euiccResponseData contains the exact response coming from the eUICC.

Table 163: Disable Profile Specific Status Codes

5.4.10 Delete an ISD-P

Function name: DeleteISDP

Related Procedures: Profile and ISD-P Deletion via SM-DP

Function group: Platform Management

Function Provider: SM-SR

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 authorised 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 exists on the targeted eUICC
- The SM-DP is authorised 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 may provide additional verifications.

In case one of these conditions is not satisfied, the SM-SR shall refuse the function request and return a 'Function execution status' indicating 'Failed' with the relevant status code (see table below).

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 is described in section 4.1.1.3 of this specification.

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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile to delete. See section 5.1.1.1 for type description.	ICCID	1	M

Table 164: Delete ISD-P Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES8 depending of the requested function.	Byte[]	1	O

Table 165: Delete ISD-P Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.1	Profile ICCID	3.8	Refused	Indicates that the Profile cannot be deleted because it is the last Profile of the eUICC or the Fall-back Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the Profile doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the Profile doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the deletion (or disabling) command on the eUICC. In that case, the output data "euiccResponseData contains the exact response coming from the eUICC.

Table 166: Delete ISD-P Specific Status Codes

NOTE: Stating that in case of disable function is performed as automatic operation before deletion, this function may raises any status code coming of the execution of the function defined in section 5.1.6.4.

5.4.11 Update Connectivity Parameters

Function name: UpdateConnectivityParameters

Related Procedures: -

Function group: Profile Management

Function Provider: SM-SR

Description: This function allows the MNO, or the SM-DP authorised 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile See section 5.1.1.1 for type description.	ICCID	1	M
connectivityParameters	The connectivityParameters to associate with the identified Profile as describe in section 4.1.3.4 "Connectivity Parameters Update using SCP03".	Hexadecimal String	1	M

Table 167: Update Connectivity Parameters Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case of update of the Connectivity Parameters in the ISD-P on the targeted eUICC.	Hexadecimal String	1	O

Table 168: Update Connectivity Parameters Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.3	ISD-P	4.2	Execution error	Error during execution of Connectivity Parameters update. In that case, the output data "euiccResponseData" contains the exact response coming from the eUICC.

Table 169: Update Connectivity Parameters Specific Status Codes

5.4.12 Notify a Profile is Disabled

Function name: HandleProfileDisabledNotification

Related Procedures: Profile Download and Installation, Profile Enabling via SM-DP, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: SM-DP

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
Iccid	Identification of the Profile that has been disabled. See section 5.1.1.1 for type description.	ICCID	1	M
mno-id	Identification of the MNO owner of the Profile that has been disabled. See section 5.1.1.1 for type description.	OID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 170: Handle Profile Disabled Notification Additional Input Data

5.4.13 Notify a Profile Enabling

Function name: HandleProfileEnabledNotification

Related Procedures: Profile Disabling, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: SM-DP

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been disabled. See section 5.1.1.1 for type description.	ICCID	1	M
mno-id	Identification of the MNO owner of the Profile that has been enabled. See section 5.1.1.1 for type description.	OID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 171: Handle Profile Enabled Notification Additional Input Data

5.4.14 Notify an SM-SR Change

Function name: HandleSMSRChangeNotification

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: SM-DP

Description: This function shall be called for notifying each SM-DP authorised 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.

This notification is not related to a particular Profile. It is up to the notification recipient to perform any action related to each Profile that is deployed on this eUICC

Additional input data:

Input data name	Description	Type	No.	MOC
eis	The relevant part of the eUICC Information Set linked with the MNO owning the Profile hosted in the eUICC. See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M
mno-id	Identification of the MNO concerned by the SM-SR change. See 5.1.1.1 for type description.	OID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 172: Handle SM-SR Change Notification Additional Input Data

Additional output data:

No output data is expected in response to this notification.

5.4.15 Notify a Profile Deletion

Function name: HandleProfileDeletedNotification

Related Procedures: Profile Enabling, Profile Enabling via SM-DP

Function group: Platform Management

Notification handler/recipient: SM-DP

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been deleted. See section 5.1.1.1 for type description.	ICCID	1	M
mno-id	Identification of the MNO owner of the Profile that has been deleted.	OID	1	M

	See section 5.1.1.1 for type description.			
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 173: Handle Profile Deleted Notification Additional Input Data

5.5 ES4 (MNO – SM-SR) Interface Description

5.5.1 Getting eUICC Information

Function name: GetEIS

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function Provider: SM-SR

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. The MNO utilises the retrieved EIS, for instance, to verify the eligibility of the eUICC (for example 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M

Table 174: Get EIS Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
eis	The relevant eUICC Information Set of the eUICC See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	C

Table 175: Get EIS Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EID, is unknown to the function provider
8.6	EIS	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage this EIS, identified by this EID.

Table 176: Get EIS Specific Status Code

5.5.2 Updating the Policy Rules of a Profile

Function name: UpdatePolicyRules

Related Procedures: -

Function group: Profile Management

Function Provider: SM-SR

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 general description of this function is detailed in section 5.4.6 of this specification.

5.5.3 Updating eUICC Information

Function name: UpdateSubscriptionAddress

Related Procedures: Profile Enabling

Function group: Profile Management

Function Provider: SM-SR

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. For consistency within the system, it is the responsibility of the caller to ensure that all data is provided. 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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the target Profile. See section 5.1.1.1 for type description.	ICCID	1	M
newSubscriptionAddress	The new Subscription Address	Subscription Address	1	M

Table 177: Update Subscription Address Additional Input Data

Additional output data:

This function has no additional output data.

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EIS identified by this EID, is unknown to the function provider
8.2.1	ICCID	1.1	Unknown	Indicates that the Profile identified by the ICCID, is unknown to the function provider
8.2.6	Subscription Address	1.2	Not Allowed (Authorisation)	Function caller is not allowed to manage the Subscription Address.

Table 178: Update Subscription Address Status Codes

5.5.4 Auditing eUICC Information

Function name: AuditEIS

Related Procedures: Profile Download and Installation

Function group: Profile Management

Function provider: SM-SR

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.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC.to be audited See section 5.1.1.1 for type description.	EID	1	M
iccid-list	List of "iccid" identifying Profiles to be audited	ICCID	1..N	C

Table 179: AuditEIS Additional Input Data

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.

Additional output data:

Output data name	Description	Type	No.	MOC
Eis	For the relevant eUICC Information Set see section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.Only data for the requested Profiles is returned within EIS. The Profiles that do not belong to the requestor are not included in the response. This access control function is realised within the SM-SR, there is no need to limit the data on the eUICC side.	EIS	1	M

Table 180: AuditEIS Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider
8.2	Profile	1.2	Not Allowed (Authorisation)	One or more Profiles identified by ICCIDs in the list do not belong to function requester
8.6	EIS	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage this EIS, identified by this EID.
1.6	Function	5.4	Delivered With No Response	The function execution request has been delivered to the remote entity but no response is received.

Table 181: AuditEIS Additional Specific Status Codes

5.5.5 Profile Enabling

Function name: EnableProfile

Related Procedures: Profile Enabling

Function group: Platform Management

Function Provider: SM-SR

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 may provide additional verifications.

In case one of these conditions is not satisfied, the SM-SR shall refuse the function request and return a 'Function execution status' indicating 'Failed' with the relevant status code (see table below).

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
  with a status code indicating a Unknown ICCID
- With a status code as defined in section 5.1.6.4 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
Eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
Iccid	Identification of the Profile to enable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 182: Enable Profile Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES5 depending of the requested	Hex binary	1	O

	function.			

Table 183: Enable Profile Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of one the impacted Profiles don't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of one the impacted Profiles don't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the enabling command on the eUICC. In that case, the output data "euiccResponseData contains the exact response coming from the eUICC.

Table 184: Enable Profile Specific Status Codes

5.5.6 Profile Disabling

Function name: DisableProfile

Related Procedures: Profile Disabling

Function group: Platform Management

Function Provider: SM-SR

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 '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 or a specific status code as defined in the table below

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile to disable. See section 5.1.1.1 for type description.	ICCID	1	M

Table 185: Disable Profile Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC. The response data is defined in ES5 depending of the requested function.	Hex binary	1	O

Table 186: Disable Profile Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1	UICC	3.8	Refused	Indicates that the target Profile can't be disabled. (for example the Profile is the only Profile in the eUICC)
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the target Profile doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the target Profile doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the disabling command on the eUICC. In that case, the output data "euiccResponseData" contains the exact response coming from the eUICC.

Table 187: Disable Profile Specific Status Codes

5.5.7 Delete a Profile

Function name: DeleteProfile

Related Procedures: Profile and ISD-P Deletion

Function group: Platform Management

Function Provider: SM-SR

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 authorised by the MNO owning the target Profile.

The SM-SR may provide additional verifications.

In case one of these conditions is not satisfied, the SM-SR shall refuse the function request and return a ‘Function execution status’ indicating ‘Failed’ with the relevant status code (see table below).

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 is described in section 4.1.1.3.

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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M

iccid	Identification of the Profile to delete. See section 5.1.1.1 for type description.	ICCID	1	M

Table 188: Delete Profile Additional InputDdata

Additional output data:

Output data name	Description	Type	No.	MOC
euiccResponseData	Contains the detailed error returned by the eUICC in case the function execution failed on eUICC	Byte[]	1	O

Table 189: Delete Profile Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	3.9	Unknown	Indicates that the eUICC, identified by this EID, is unknown to the function provider.
8.2.1	Profile ICCID	3.9	Unknown	Indicates that the Profile identified by this ICCID is unknown to the function provider.
8.2.1	Profile ICCID	3.4	Invalid destination	Indicates that the Profile identified by this ICCID is known to the function provider but installed on another eUICC than the one identified by the function caller.
8.2.1	Profile ICCID	1.2	Not Allowed (Authorisation)	Indicates that the function caller is not allowed to perform this function on the target Profile.
8.2.1	Profile ICCID	3.8	Refused	Indicates that the Profile cannot be deleted because it is the last Profile of the eUICC or the Fall-back Profile.
8.2.2	POL1	3.8	Refused	The POL1 of the Profile doesn't allow this operation.
8.2.3	POL2	3.8	Refused	The POL2 of the Profile doesn't allow this operation.
8.4	ISD-R	4.2	Execution error	Error during execution of the deletion (or disabling) command on the eUICC. In that case, the output data "euiccResponseData" contains the exact response coming from the eUICC.

Table 190: Delete Profile Specific Status Codes

NOTE: Stating that in case of disable function is performed as automatic operation before deletion, this function may raises any status code coming of the execution of the function section 5.1.6.4.

5.5.8 Prepare SM-SR Change

Function name: PrepareSMSRChange

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: SM-SR

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.

The check is used to give the opportunity to the new SM-SR to ensure that any necessary business agreement is in place.

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' with 'Expired' with a status code as defined in section 5.1.6.4 (Note: This Status is under validation by WI4 to be definitively included)
- A 'Function execution status' indicating 'Failed' with a status code as defined in section 5.1.6.4 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	provide the EID of the eUICC See section 5.1.1.1 for type description.	EID	1	M
currentSMSRid	Identification of the current SM-SR. See section 5.1.1.1 for type description.	OID	1	M

Table 191: Prepare SM-SR Change Additional Input Data

Table 192: Void

Table 193: Void

Specific status codes

Subject Code	Subject	Reason code	Reason	Description
1.2	Function Provider	3	Condition Of Use Not satisfied	Indicates that function provider is not capable of managing the eUICC identified by this EID,

Table 194: Prepare SM-SR Change Specific Status Codes

5.5.9 SM-SR Change

Function name: SMSRChange

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: SM-SR

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 of a specific status code as defined in the Specific status code table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
targetSMSRid	Identification of the new SM-SR. See section 5.1.1.1 for type description.	OID	1	M

Table 195: SM-SR Change Additional Input Data

Specific status codes

Subject Code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EID , is unknown to the function provider
8.1	eUICC	1.2	Not Allowed (Authorisation)	Function requester is not allowed to manage the eUICC
8.4	ISD-R	4.2	Execution error	Error during the creation of a new key set at the ISD-R level
8.5.2	eUICC Certificate Authority Certification	6.3	Certificate Expired	ECASD Certificate Expired

Table 196: SM-SR Change Specific Status Codes

5.5.10 Notify a Profile is Disabled

Function name: HandleProfileDisabledNotification

Related Procedures: Profile Download and Installation, Profile Enabling, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: MNO

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been disabled. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 197: Handle Profile Disabled Notification Additional Input Data

5.5.11 Notify a Profile Enabling

Function name: HandleProfileEnabledNotification

Related Procedures: Profile Disabling, Fall-back Activation Procedure

Function group: Platform Management

Notification handler/recipient: MNO

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been disabled. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 198: Handle Profile Enabled Notification Additional Input Data

5.5.12 Notify a SM-SR Change

Function name: HandleSMSRChangeNotification

Related Procedures: SM-SR Change

Function group: eUICC Management

Notification handler/recipient: MNO

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.

This notification is not related to a particular Profile. It is up to the notification recipient to perform any action related to each Profile that is deployed on this eUICC.

Additional input data:

Input data name	Description	Type	No.	MOC
eis	The relevant part of the eUICC Information Set linked with the MNO owning the Profile hosted in the eUICC. See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 199: Handle SM-SR Change Notification Additional Input Data

Additional output data:

No output data is expected in response to this notification.

5.5.13 Notify a Profile Deletion

Function name: HandleProfileDeletedNotification

Related Procedures: Profile enabling, Profile Enabling via SM-DP

Function group: Platform Management

Notification handler/recipient: MNO

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.

What is performed by the MNO receiving this notification is out of scope of this specification.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
iccid	Identification of the Profile that has been deleted. See section 5.1.1.1 for type description.	ICCID	1	M
completionTimestamp	Indication of the date/time when the operation has been performed. See section 5.1.1.1 for type description.	DATETIME	1	M

Table 200: Handle Profile Deleted Notification Additional Input Data

5.6 ES7 (SM-SR – SM-SR) Interface Description

5.6.1 Create Additional Key Set

Function name: CreateAdditionalKeySet

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: current SM-SR

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 key set 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. The following parameters used within this command as defined in Table 42 are not provided by the new SM-SR and it is the current SM-SR's responsibility to set these parameters as defined below.

- Key Usage Qualifier shall be set to '10' (3 secure channel keys)
- Key Access shall be set to '00' (The key may be used by the Security Domain and any associated Application)
- Key Type shall be set to '88' (AES)

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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See 5.1.1.1 for type description.	EID	1	M
keyVersionNumber	The Key Version Number of the to-be-created keyset.	Integer	1	O
initialSequenceCounter	The initial value of the Sequence Counter of the keyset	Integer	1	O
eccKeyLength	The length of the Elliptic Curve Cryptography keys.	Enumeration {ECC-256, ECC-354, ECC-512, ECC-521 }	1	M
scenarioParameter	Scenario parameter as defined in Table 4-17 of the Amendment E of GlobalPlatform 2.2 Card Specification [11]	Hexadecimal String representation of 1 Byte	1	M
hostId	Host ID as defined in Table 4-17 of the Amendment E of GlobalPlatform 2.2 Card Specification [11]	Hexadecimal String	1	C
ephemeralPublicKey	The ephemeral public key calculated by new SM-SR	Byte[]	1	M
signature	The signature associated to the authenticate SM-SR function. The signature is computed off-card by the new SM-SR SK.SR. ECDSA. See section 4.1.1.8	Hexadecimal String	1	M

Table 201: Create Additional Key Set Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
derivationRandom	A random number generated in the SE for additional entropy in the key derivation process	Hexadecimal String	1	C
receipt	A Message Authentication Code (MAC)	Hexadecimal String	1	M

Table 202: Create Additional Key Set Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EID, is unknown to the function provider
8.4	ISD-R	4.2	Execution error	Error during the creation of the key set at the ISD-R level. In that case, the output data “euiccResponseData contains the exact response coming from the eUICC.

Table 203: Create Additional Key Set Specific Status Codes

5.6.2 Handover eUICC Information

Function name: HandoverEUICC

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: SM-SR

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 of a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eis	The eUICC Information Set of the eUICC See section 5.1.1.1 for type description. The list of EIS data fields that shall be included is defined in Annex E.	EIS	1	M

Table 204: Handover EUICC Additional Input Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
1.2	Function Provider	3	Condition Of Use Not satisfied	Indicates that function provider is not capable of managing the eUICC identified by this EID.
8.1.1	EID	1.1	Unknown	Indicates that the preparation step hasn't been performed for the eUICC
8.4	ISD-R	4.2	Execution error	Error during the creation of the key set at the ISD-R level. In that case, the output data "euiccResponseData contains the exact response coming from the eUICC.
8.5.2	eUICC Certificate Authority Certificate	6.3	Certificate Expired	ECASD Certificate expired

Table 205: Handover eUICC Specific Status Codes

5.6.3 Authenticate SM-SR

Function name: AuthenticateSMSR

Related Procedures: SM-SR Change

Function group: eUICC Management

Function Provider: SM-SR

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 or a specific status code as defined in the table below.

Additional input data:

Input data name	Description	Type	No.	MOC
eid	Identification of the targeted eUICC. See section 5.1.1.1 for type description.	EID	1	M
smsrCertificate	SM-SR Certificate. The format of this field is a byte array which content corresponds to the full content of tag '7F21' (including the two '7F21' bytes) defined in Table 39	Byte[]	1	M

Table 206: Authenticate SM-SR Additional Input Data

Additional output data:

Output data name	Description	Type	No.	MOC
randomChallenge	The random challenge	Byte[]	1	M

Table 207: Authenticate SM-SR Additional Output Data

Specific status codes

Subject code	Subject	Reason code	Reason	Description
8.1.1	EID	1.1	Unknown	Indicates that the EID, is unknown to the function provider
8.4	ISD-R	4.2	Execution error	Error during the creation of the Random Challenge at the ISD-R level
8.5.3	SM-SR Certificate	6.3	Certificate Expired	SM-SR certificate expired

Table 208: Authenticate SM-SR Specific Status Codes

Table 209: Void

Annex A Mapping of Functions into Messages (Normative)

This Annex provides the mapping of the functions defined in section 0 into messages to be exchanged between the Roles.

Any technology can be used to transport those messages (mail, file, Web Services…) as soon as it is agreed between the sender and the receiver.

However, for interoperability purpose, Annex B of this specification specifies the particular binding to the Web Service technology, following the OASIS and W3C WS-* standard.

All along this Annex we can indifferently use either “function caller” or “sender entity” wording to designate the entity that has issued the function execution request. It is also the case regarding “function provider” and “receiver entity” to designate the entity that executes the function.

A.1 Namespaces and Schema References

In the context of this specification, a specific namespace is used:

- rps: http://namespaces.gsma.org/esim-messaging/1

The “1” at the end of the URI indicates the major version (for example 1) of this specification.

The XML schema defined in this specification refers to the following namespaces:

- xs: Extensible Markup Language (XML) 1.0, W3C Recommendation as defined in [47].
- ds: XML Signature Syntax and Processing (Second Edition), W3C Recommendation as defined in [48].

A.2 Message: <rps:RPSMessage>

A message in the context of GSMA Embedded UICC Remote Provisioning and Management is composed of a mandatory header and a mandatory body.

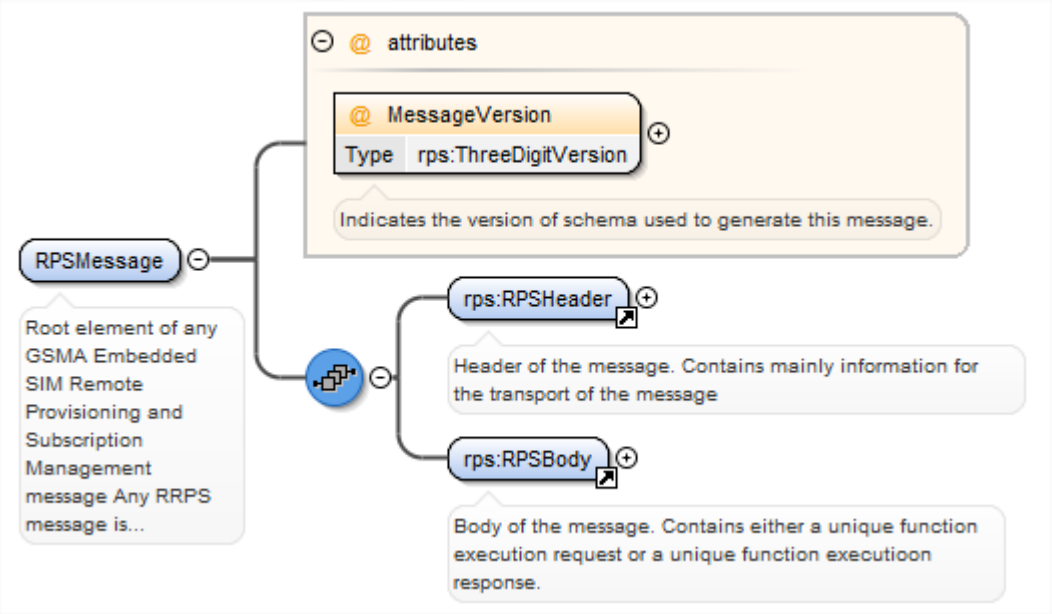


Figure 37: RPSMessage

The <rps:RPSMessage> is the root element defining a message.

The <rps:RPSMessage> can convey either a function execution request or a function execution response.

The attribute @MessageVersion in the instance document indicates the version of the schema used to generate the message. This attribute makes reference to the <xs:schema>@version attribute that indicates the version of the schema. This information may be used by the receiving entity to determine the schema to use for validation of the incoming message.

A.2.1 Message Header: <rps:RPSHeader>

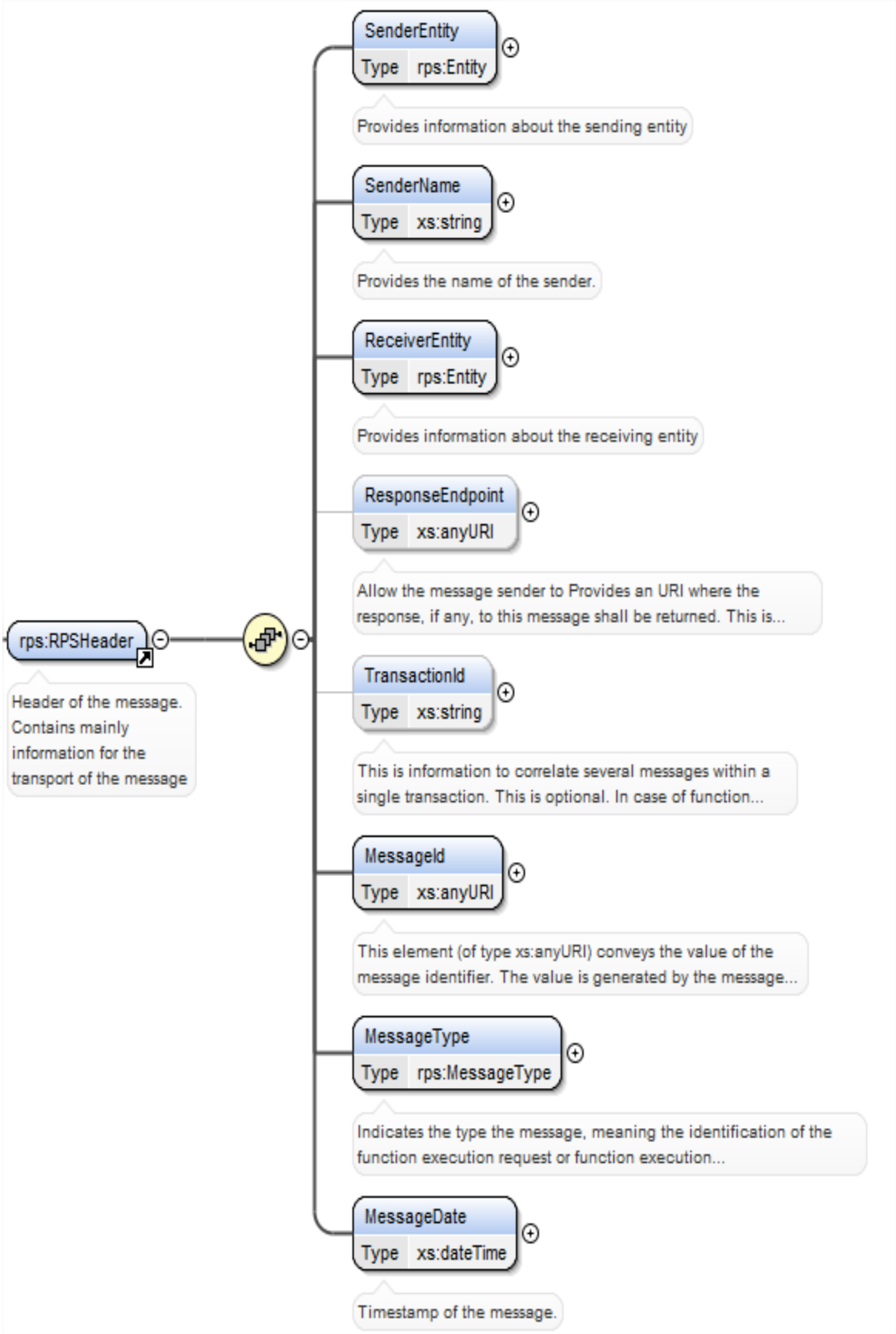


Figure 38: <rps:RPSHeader>

The <rps:RPSHeader> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:SenderEntity>	The <rps:SenderEntity>/<rps:EntityId> element is mapped to the Function Requester Identifier data of the function input header (See section 5.1.4). Provides identification about the sending entity.
<rps:SenderName>	This element has no particular mapping with any function input data specified in section 0. Provides the name of the sender. It represents the human user behind the system of the <rps:SenderEntity>. This specification doesn't mandate anything regarding its usage by the receiver entity.
<rps:ReceiverEntity>	This element has no particular mapping with any function input data specified in section 0. Provides identification of the receiver entity. Only the <rps:ReceiverEntity>/<rps:EntityID> shall be set. The <rps:ReceiverEntity>/<rps:EntityName> shall not be set.
<rps:ResponseEndpoint>	This element has no particular mapping with any function input data specified in section 0. Allow the message sender to provide an URI where the response to this message, if any, shall be returned. This is optional; if missing the receiver entity shall consider the originating point of the message as the response endpoint.
<rps:ContextId>	This element has no particular mapping with any function input data specified in section 0. This identifier may be used to provide end-to-end logging management between the different web services. This is optional. If present, this parameter shall be included if the function provider entity generates a new request to another function provider entity.
<rps:TransactionId>	This element has no particular mapping with any function input data specified in section 0. This information allows the sender entity to correlate several messages within a single transaction. It is the sender entity responsibility to ensure uniqueness of this information. This is optional. If present, the receiver entity shall provide the same information in the transactionId of the function execution response message, if any.
<rps:MessageId>	This element has no particular mapping with any function input data specified in section 0. This element (of type xs:anyURI) conveys the value of the message identifier. The value is generated by the sender entity and must be UNIQUE. To make the MessageID unique between different senders it must be prefixed with the domain portion of the sender. Then the suffix part of the message id is freely chosen by the sender, it could a simple integer value, a date; nothing is mandated except the uniqueness. For example: "http://MySenderEntityId/1234"
<rps:MessageType>	This element has no particular mapping with any function input data specified in section 0. Indicates the type the message, meaning the identification of the function execution request or function execution response. The Message type for a function execution request shall include the 'Request' qualifier at the end; example: "ES3-GetEISRequest" The Message type for a function execution response shall include the 'Response' qualifier at the end; example: "ES3-GetEISResponse". The Message type for a notification function shall include the 'Notification' qualifier at the end; example: "ES3-HandleProfileDisabledNotification"
<rps: MessageDate>	This element has no particular mapping with any function input data specified in section 0. Timestamp of the message.
<rps:RelatesTo>	This element (of type xs:anyURI) conveys the value of the message identifier of the initial message request. This element shall be present only in case of response.

Table 210: <rps:RPSHeader> Attributes

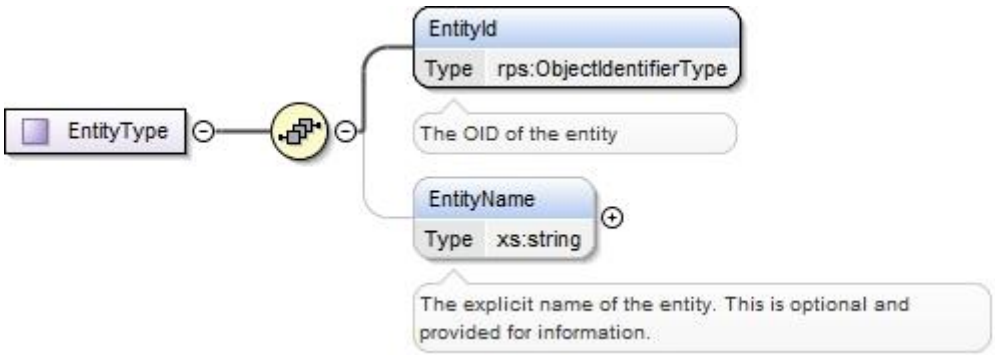


Figure 39: <rps:Entity>

The <rps:EntityType> type is for representing a message entity; it is used for example to represent a sending entity and a receiver entity in the <rps:RPSHeader> element.

Attribute/Element	Mapping/Description
<rps:EntityId>	The OID of the entity.
<rps:EntityName>	The explicit name of the entity. This is optional and provided for information.

Table 211: <rps:EntityType> Attributes

A.2.2 Message Body: <rps:RPSBody>

The <rps:RPSBody> is the element that contains the core of the message. In the context of this specification, it shall be composed of one single element defined within one of the interfaces ES1, ES2, ES3, ES4 and ES7.

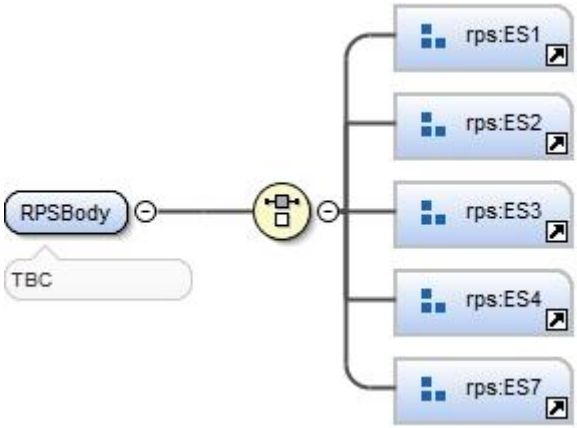


Figure 40: <rps:RPSBody>

Each element under the <rps:RPSBody> matches either a function execution request, a function execution response or a notification.

A.3 Common Types

A.3.1 Request Base Type

The function input header defined in section 5.1.4 shall be mapped to the <rps:BaseRequestType> type described in the following figure:

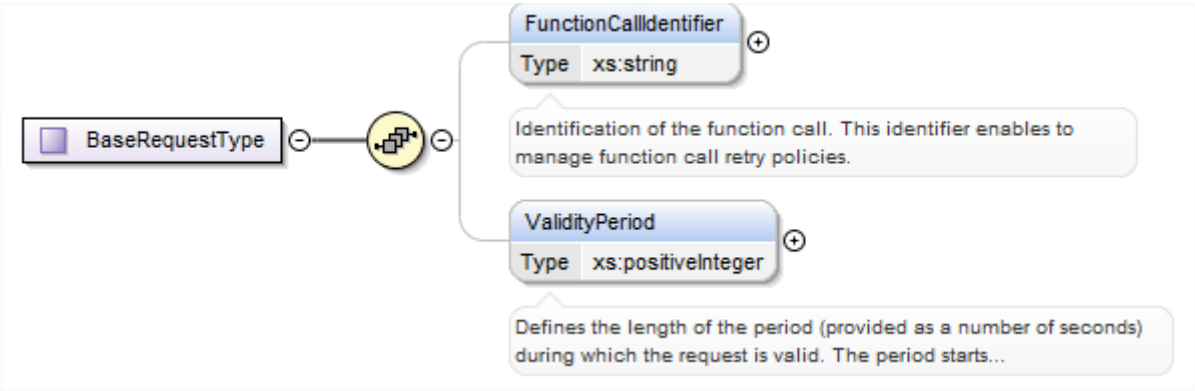


Figure 41: <rps:BaseRequestType>

The <rps:BasicRequestType> acts as a base type that each Request shall extend.

As the Function Requester Identifier is already mapped to the <rps:RPSHeader>.<rps:SenderEntity>.<rps:EntityId> element, it is not put in the <rps:BasicRequestType>.

The <rps:BasicRequestType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:FunctionCallIdentifier>	This element is mapped to the "Function Call Identifier" data of the function input header. Refer section 5.1.4 for description of this data.
<rps:ValidityPeriod>	This element is mapped to the "Validity period" data of the function input header. Refer section 5.1.4 for description of this data.

Table 212: <rps:BasicRequestType> Attributes

A.3.2 Void

A.3.3 Response Base Type

The function output header defined in section 5.1.5 shall be mapped to the <rps:BaseResponseType> type described in the following figure:

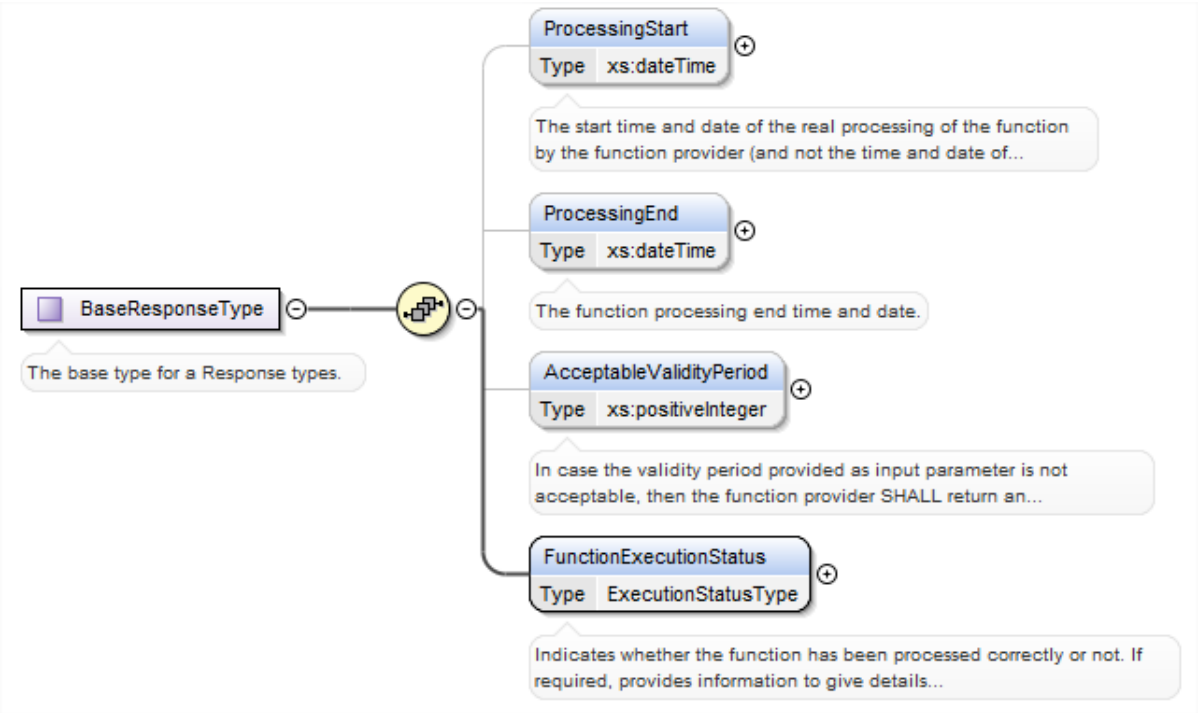


Figure 42: <rps:BaseResponseType>

The <rps:BaseResponseType> acts as a base type that each Response shall extend.

The <rps:BaseResponseType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:ProcessingStart>	This element is mapped to the "Processing start" data of the function output header. Refer section 5.1.5 for description of this data.
<rps:ProcessingEnd>	This element is mapped to the "Processing end" data of the function output header. Refer section 5.1.5 for description of this data.
<rps:AcceptableValidityPeriod>	This element is mapped to the "Acceptable validity Period" data of the function output header. Refer section 5.1.5 for description of this data.
<rps:FunctionExecutionStatus>	This element is mapped to the "Function Execution Status" data of the function output header. Refer section 5.1.5 for description of this data.

Table 213: <rps:BaseResponseType> Attributes

The <rps:FunctionExecutionStatus> is of type <rps:FunctionExecutionStatusType> described below:

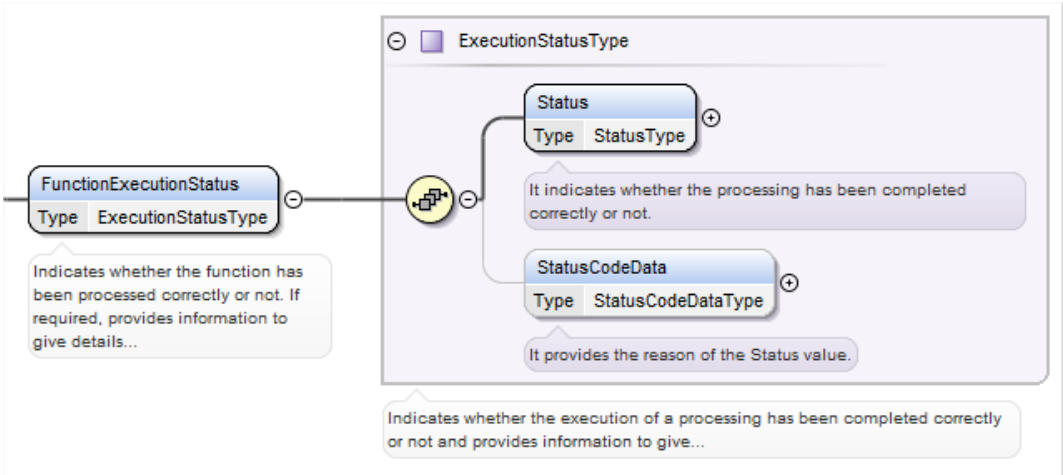


Figure 43: <rps:FunctionExecutionStatusType>

The <rps:FunctionExecutionStatusType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Status>	This element is mapped to the "Status" data of the function output header. Refer section 5.1.5 for description of this data.
<rps:StatusCodeData>	This element is mapped to the "Status code data" data of the function output header. Refer section 5.1.5 for description of this data.

Table 214: <rps:FunctionExecutionStatusType> Attributes

The <rps:StatusCodeData> is of type <rps:StatusCodeDataType> described below:

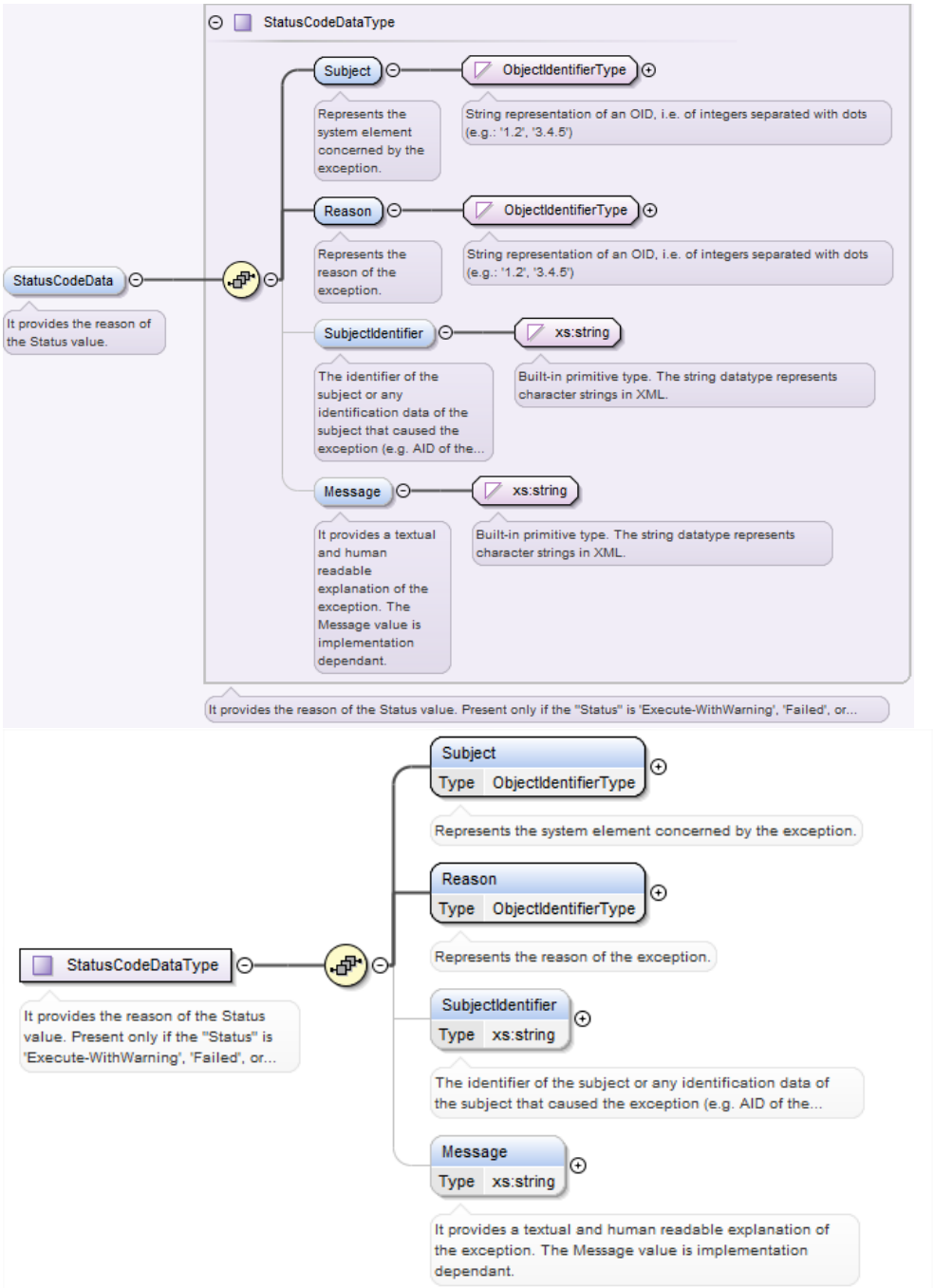


Figure 44: <rps:StatusCodeDataType>

The <rps:StatusCodeDataType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Subject>	This element is mapped to the “Subject” data of the function output header. Refer section 5.1.5 for description of this data.
<rps:Reason>	This element is mapped to the “Reason” data of the function output header. Refer section 5.1.5 for description of this data.
<rps:SubjectIdentifier>	This element is mapped to the “Subject identifier” data of the function output header. Refer section 5.1.5 for description of this data.

<rps:Message>	This element is mapped to the “Message” data of the function output header. Refer section 5.1.5 for description of this data.

Table 215: <rps:StatusCodeDataType> Attributes

A.3.4 Simple Types Mapping

A.3.4.1 AID

The AID (Application IDentifier) type defined in section 5.1.1.1 shall be mapped to the <rps:AIDType>. The type is defined as a hexadecimal string representation of 5 to 16 bytes.

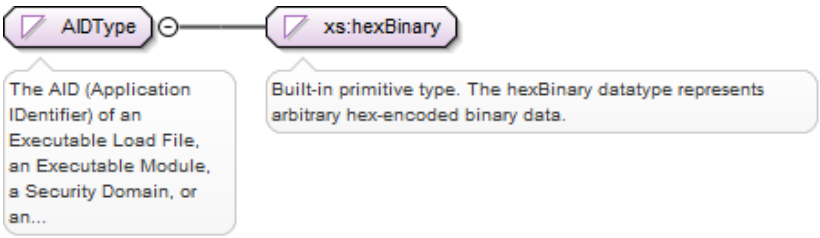


Figure 45: <rps:AIDType>

A.3.4.2 Datetime

The Datetime type defined in section 5.1.1.1 shall be mapped to the simple built-in XML time <xs:datetime>.

A.3.4.3 EID

The EID (eUICC IDentifier) type defined in section 5.1.1.1 shall be mapped to the <rps:EIDType>.

The type is defined as a hexadecimal string representation of 16 bytes.

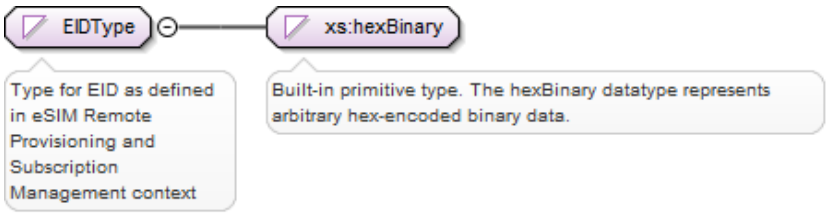


Figure 46: <rps:EIDType>

A.3.4.4 ICCID

The ICCID (Integrated Circuit Card IDentifier) type defined in section 5.1.1.1 shall be mapped to the <rps:ICCIDType>. The type is defined as a string representation (up to 20 digits), non-swapped as per ITU E.118 representation. Example: 893301000000000011

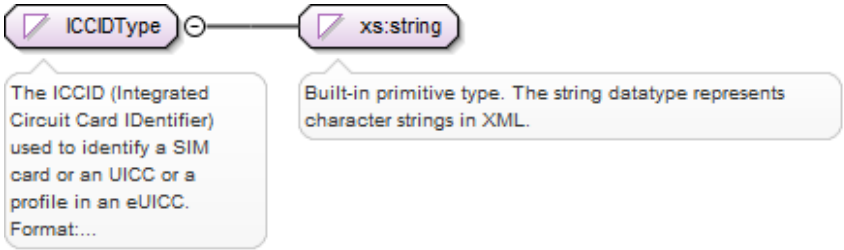


Figure 47: <rps:ICCIDType>

A.3.4.5 MSISDN

The MSIDN (Mobile Station ISDN Number) type defined in section 5.1.1.1 shall be mapped to the <rps:MSISDNType>. The type is defined as a string representation of up to 15 decimal digits as defined in ITU E.164, including Country code, National Destination Code (optional) and Subscriber Number. Example: 380561234567

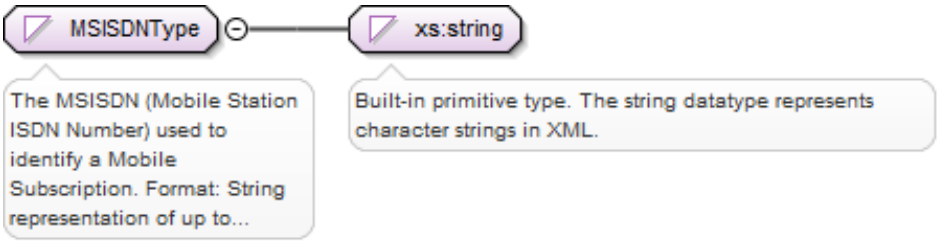


Figure 48: <rps:MSISDNType>

A.3.4.6 IMSI

The IMSI (International Mobile Subscriber Identity) type defined in section 5.1.1.1 shall be mapped to the <rps:IMSIType>. The type is defined as a string representation of up to 15 decimal digits including MCC (3 digits) and MNC (2 or 3 digits), as defined in ITU E.212. Example: 242011234567890

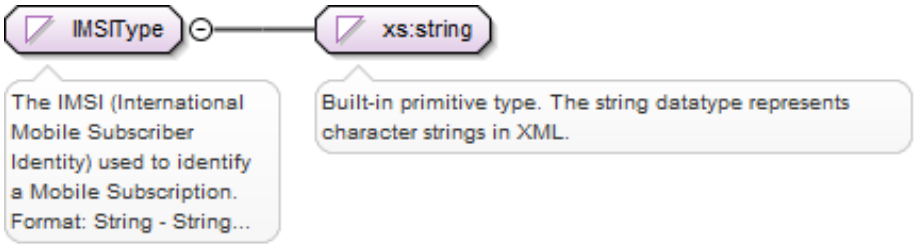


Figure 49: <rps:IMSIType>

A.3.4.7 OID

The OID (Object Identifier) type defined in section 5.1.1.1 shall be mapped to the <rps:ObjectIdentifierType>. The type is defined as a string representation of an OID, i.e. of integers separated with dots (for example: '1.2', '3.4.5').

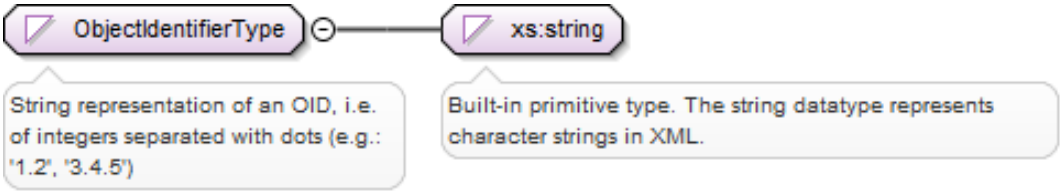


Figure 50: <rps:ObjectIdentifierType>

A.3.4.8 TAR

The TAR (Toolkit Application reference) type defined in section 5.1.1.1 shall be mapped to the <rps:TARType>. The type is defined as a hexadecimal string representation of exactly 3 bytes. Example: 363443.

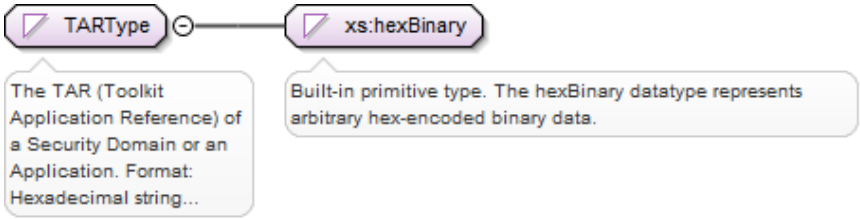


Figure 51: <rps:TARType>

A.3.4.9 Version

The Version type defined in section 5.1.1.1 shall be mapped to the <rps: ThreeDigitVersion>. The type is defined as a string representation of exactly 3 integers separated by a '.'.
Example: 1.15.9

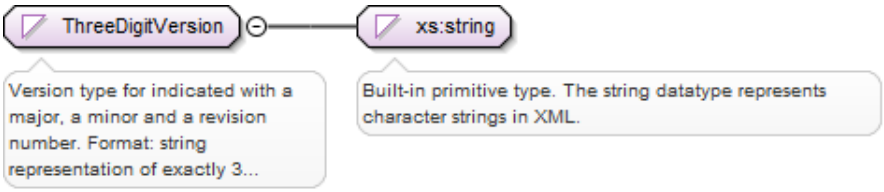


Figure 52: <rps: ThreeDigitVersion >

A.3.5 Complex Type Mapping

A.3.5.1 EIS

The EIS type defined in section 5.1.1.3.13 shall be mapped to the <rps: EISType>. This type contains the whole information defined for EIS, but depending on the function where it is used, it may be filled with only a partial content.

All information requiring to be signed by the EUM at registration time is regrouped under the element <rps:EumSignedInfo>. The signature is provided within the <ds:Signature> element (see section A.3.5.2 of this Annex).

The <rps:EumSignedInfo> shall contain the definition of the ECASD security domain including the certificate value.

The other elements <rps: remainingMemory>, <rps:AvailableMemoryForProfiles>, <rps:LastAuditDate >, <rps:Smsr-id>, <rps:ProfileInfo>, <rps:Isd-r>, <rps:AuditTrail>, <rps:AdditionalProperties> are not included in the signature.

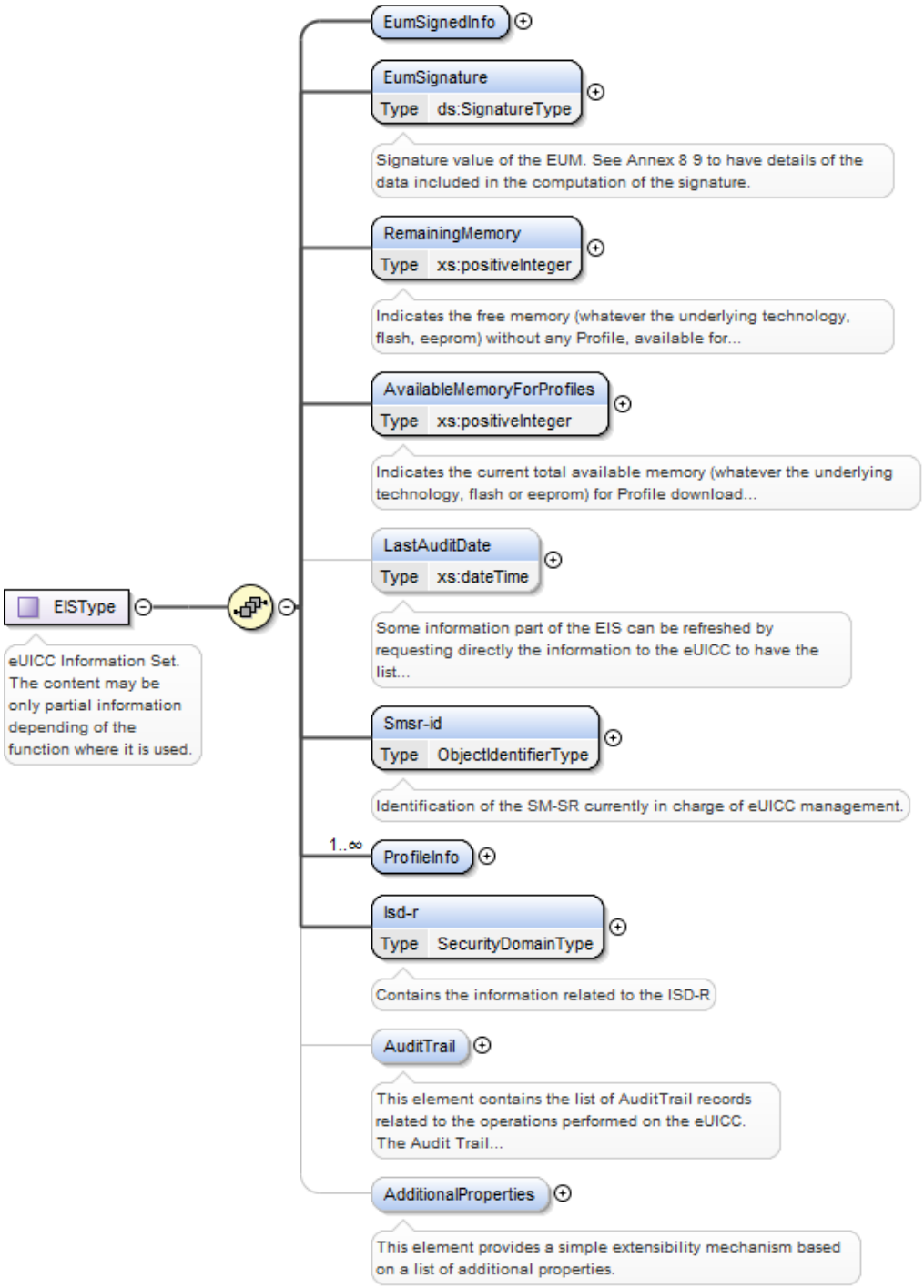


Figure 53: <rps: EISType >

The <rps:EISType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:EumSignedInfo>	This element regroups the information signed by the EUM. See section A.3.5.2 of this Annex for description of this element.
<rps:EumSignature>	This element contains the signature of the EUM. It maps the "eumCertificateId", "signatureAlgorithm" and "signature" data of the EIS described in section 5.1.1.3.13. See section A.3.5.3 of this Annex for description of the <rps:EumSignature>.
<rps:RemainingMemory>	This element maps the "remainingMemory" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type.
<rps:AvailableMemoryForProfiles>	This element maps the "availableMemoryforProfiles" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type.
<rps:LastAuditDate>	This element maps the "lastAuditDate" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type.
<rps:SmSr-Id>	This element maps the "smsr-id" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type.
<rps:ProfileInfo>	This element maps the "profiles" data of the EIS type. The <rps:EISType> may contains several <rps:Profile> elements. Refer section 5.1.1.3.13 for description of this data type. See section A.3.5.4 of this Annex for description of the <rps:ProfileInfo> element.
<rps:Isd-r>	This element maps the "ISD-R" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:Isd-r> is of type <rps:SecurityDomainType>. Refer section A.3.5.5 of this Annex for description of this type.
<rps:AuditTrail>	This element maps the "audit trail" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. This element can be missing, when <rps:EIS> is used in the context of the **ES1.registerEIS** function. After this step, the EIS shall content at least one record. Refer section A.3.5.6 of this Annex for description of the <rps:AuditTrail> element.
<rps:AdditionalProperties>	This element is not mapped to any data of the EIS type. It provides a simple extensibility mechanism that can be used to provide additional information about the eUICC without breaking the XML validation process. See hereunder for definition of the <rps:AdditionalProperties>. This element is optional.

Table 216: <rps:EISType> Attributes

The <rps:AdditionalProperties> allows a neutral represent of any data based on a "Key:Value pair" representation. Such representation can be used without breaking the XML validation process.

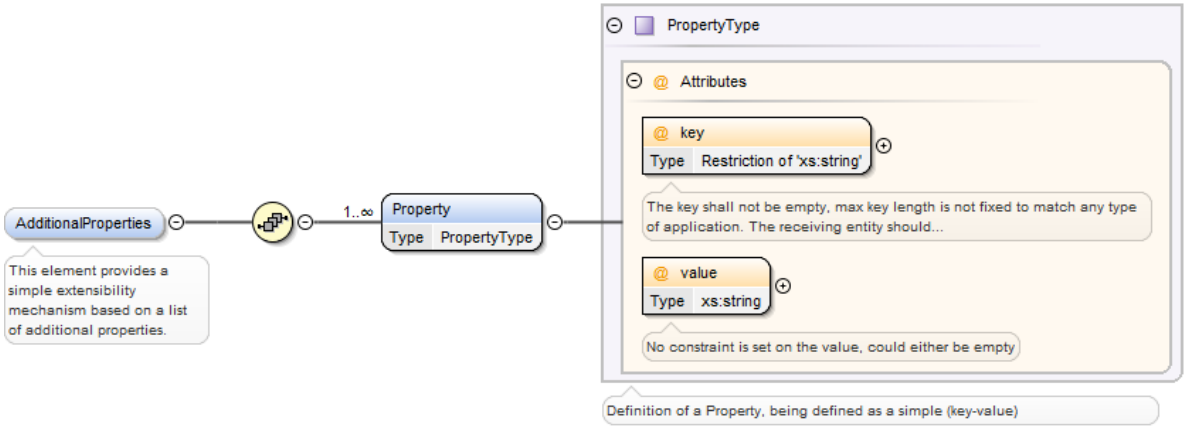


Figure 54: <rps: EISType >

The <rps: AdditionalProperties> is composed of a list of <rps:Property> with the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:key>	The key of the property. The key shall be unique within the property list. The key shall not be empty, max key length is not fixed to match any type of application.
<rps:value>	Contains a simple string value. No constraint is set on the value, could either be empty.

Table 217: <rps: AdditionalProperties> Attributes

A.3.5.2 EUM Signed Info

The <rps:EumSignedInfo> element contains all data included in the signature performed by the EUM.

Figure 55: <rps:EumSignedInfo>

The <rps:EumSignedInfo> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Eid>	This element maps the "eid" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:EIDType> is described in section A.3.4.3 of this Annex.
<rps:Eum-Id>	This element maps the "eum-id" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:ObjectIdentifierType> is described in section A.3.4.7 of this Annex.
<rps:ProductionDate>	This element maps the "productionDate" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type.
<rps:PlatformType>	This element maps the "platformType" data of the EIS. Refer section 5.1.1.3.13 for description of this data.

<rps:PlatformVersion>	This element maps the "platformVersion" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:ThreeDigitversion> is described in section A.3.4.9 of this Annex.
<rps:Isd-p-loadfile-aid>	This element maps the "isd-p-loadfile-aid" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:AID> is described in section A.3.4.1 of this Annex.
<rps:Isd-p-module-aid>	This element maps the "isd-p-module-aid" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:AID> is described in section A.3.4.1 of this Annex.
<rps:Ecasd>	This element maps the "ECASD" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. The <rps:EIDType> is described in section A.3.5.5 of this Annex.
<rps:EuiccCapabilities>	This element maps the "eUICC-Capabilities" data of the EIS type. Refer section 5.1.1.3.13 for description of this data type. Refer section **Error! Reference source not found.** of this Annex for description of the <rps: EuiccCapabilities>.

Table 218: <rps:EumSignedInfo> Attributes

A.3.5.3 EUM Signature

The EUM signature over some information of the EIS is provided within the <rps:EumSignature> element of type <ds:SignatureType> as defined in XML Signature Syntax and Processing (Second Edition) [26].

The <rps:EumSignature> shall include:

- A <ds:SignedInfo> element specifying:
  - a canonicalization method,
    This specification mandates the support of at least the following method
    'http://www.w3.org/2001/10/xml-exc-c14n#'
    Others canonicalization methods may be optionally supported
  - a signature method; this specification mandates usage of one of the following signature method to have a compliant level of security (RSA and EC key length following recommendation given in section 2.3.3)
    http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
    http://www.w3.org/2001/04/xmldsig-more#rsa-sha384
    http://www.w3.org/2001/04/xmldsig-more#rsa-sha512
    http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256
    http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha384
    http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha512
  - a unique reference element, with no URI attribute as the signed info applies always only on the whole <rps:EumSignedInfo> element (so no need to specify it in the instance document); and with a digesting method as one of:
    http://www.w3.org/2001/04/xmlenc#sha256
    http://www.w3.org/2001/04/xmldsig-more#sha384
    http://www.w3.org/2001/04/xmlenc#sha512

- A <ds:KeyInfo> containing a reference to the certificate used to generate the signature. This is achieved by including a <ds:X509Data> element containing either:
  - a <ds:X509SubjectName>, providing the subject value of a certificate that the receiving entity is supposed to have.
  - Or a <ds:X509Certificate>, containing the full certificate definition (including the public key)
- <ds:SignatureValue> element providing the signature value applied on whole <ds:SignedInfo> element, as specified by the W3C, after application of the specified canonicalization, transform and digesting methods as specified within the <ds:SignedInfo> element.

Example of <ds:Signature>:

```
<EumSignature xmlns:ds="http://www.w3.org/2000/09/xmldsig#">
    <ds:SignedInfo>
        <ds:CanonicalizationMethod Algorithm=http://www.w3.org/2001/10/xml-exc-c14n#/>
        <ds:SignatureMethod Algorithm=http://www.w3.org/2001/04/xmldsig-more#rsa-sha256/>
        <ds:Reference>
            <ds:DigestMethod Algorithm=http://www.w3.org/2001/04/xmlenc#sha256/>
            <ds:DigestValue>dHLkPm5pcyBub3QgYSBzaWduYXR1cmGB</ds:DigestValue>
        </ds:Reference>
    </ds:SignedInfo>
    <ds:SignatureValue>dHLkPm5pcyBub3QgYSBzaWduYXR1cmGB</ds:SignatureValue>
    <ds:KeyInfo>
        <ds:X509Data>
            <ds:X509SubjectName>CN=gsma, O=GSMA, C=UK</ds:X509SubjectName>
        </ds:X509Data>
    </ds:KeyInfo>
</ds:Signature>
```

A.3.5.4 Profile Info

The <rps:ProfileInfo> element contains the description of one Profile loaded on the eUICC (the <rps:Eis> may contain several Profile <rps:ProfileInfo>). The <rps:ProfileInfo> maps the PROFILE INFO type defined in section 5.1.1.3.4.

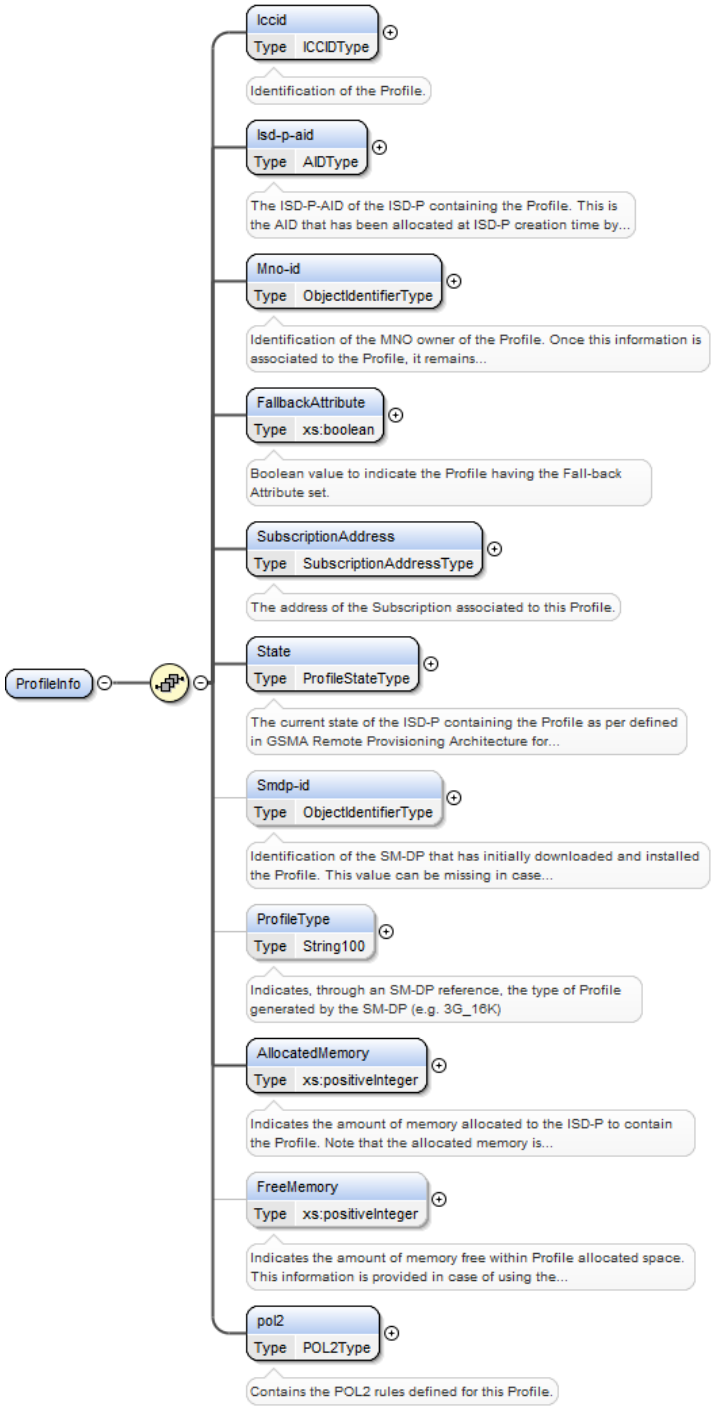


Figure 56: <rps:ProfileInfo>

The <rps:ProfileInfo> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Iccid>	This element maps the "iccid" data of the PROFILE INFO type. Refer section 5.1.1.3.4 for description of this data type. The <rps:ICCICType> is described in section A.3.4.4 of this Annex.
<rps:Isd-p-aid>	This element maps the "isd-p-aid" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:AIDType> is described in section A.3.4.1 of this Annex.

<rps:Mno-id>	This element maps the "mno-id" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:ObjectIdentifierType> is described in section A.3.4.7 of this Annex.
<rps:FallbackAttribute>	This element maps the "fallbackAttribute" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type.
<rps:SubscriptionAddress>	This element maps the "SubscriptionAddress" data of PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:SubscriptionAddressType> is described in section A.3.5.8 of this Annex.
<rps:State>	This element maps the "state" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:ProfileStateType> is an enumeration with possible values "Created", "Disabled" or "Enabled".
<rps:Smdp-id>	This element maps the "smdp-id" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:ObjectIdentifierType> is described in section A.3.4.7 of this Annex.
<rps:ProfileType>	This element maps the "profileType" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type.
<rps:AllocatedMemory>	This element maps the "allocatedMemory" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type.
<rps:FreeMemory>	This element maps the "FreeMemory" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type.
<rps:Pol2>	This element maps the "pol2" data of the PROFILE INFO type. See section 5.1.1.3.4 for description of this data type. The <rps:Pol2Type> is described in section 0 of this Annex.

Table 219: <rps:ProfileInfo> Attributes

A.3.5.5 Security Domain

The <rps:SecurityDomainType> provides description of a Security Domain and maps the SECURITY-DOMAIN type defined in section 5.1.1.3.9. It is used to contain the information of the ISD-R and ECASD.

Figure 57: <rps:SecurityDomainType>

The <rps:SecurityDomainType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Aid>	This element maps the "aid" data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type. The <rps:AIDType> is described in section A.3.4.1 of this Annex.
<rps:Tar>	This element maps the "tars" data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type. A Security Domain may have several tars. The <rps:TARType> is described in section A.3.4.8 of this Annex.

<rps:Sin>	This element maps the “sin” data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type.
<rps:Sdin>	This element maps the “sidn” data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type.
<rps:Role>	This element maps the “role” data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type. The <rps:SDRoleType> is an enumeration with possible values “ISD-R” or “ECASD”.
<rps:Keyset>	This element maps the “key sets” data of the SECURITY-DOMAIN type. See section 5.1.1.3.9 for description of this data type. A Security Domain may have up to 127 key sets. The <rps:Keyset> element is described hereunder.

Table 220: <rps:SecurityDomainType> Attributes

The <rps:Keyset> element contains the description of a key set and maps the KEYSET type defined in section 5.1.1.3.8.

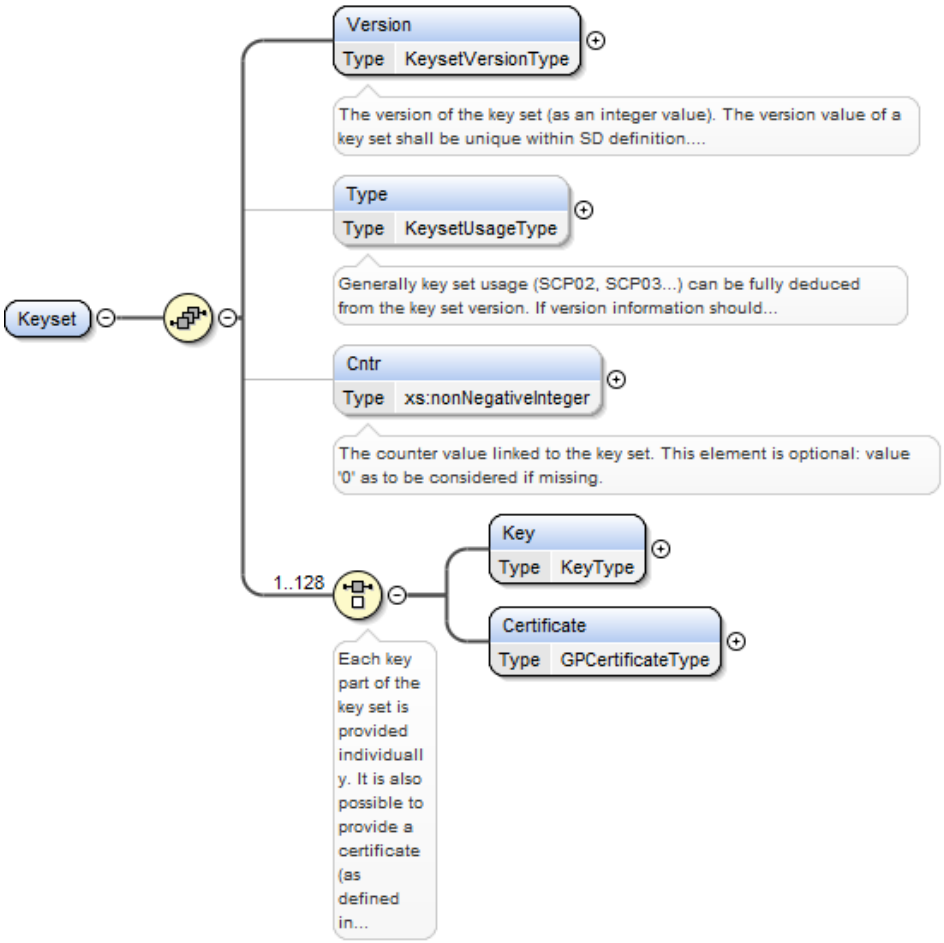


Figure 58: <rps:Keyset>

The <rps:Keyset> element is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Version>	This element maps the “version” data of KEYSET type. See section 5.1.1.3.8 for description of this data type.

<rps:Type>	This element maps the “type” data of KEYSET type. See section 5.1.1.3.8 for description of this data type. The <rps:KesetUsageType> is an enumeration with possible values “SCP03”, “SCP80”, “SCP81”, “TokenGeneration”, “ReceiptVerification”, “CA”.
<rps:Cntr>	This element maps the “cntr” data of KEYSET type. See section 5.1.1.3.8 for description of this data type.
<rps:Key>	This element maps the “keys” data of KEY type. See section 5.1.1.3.8 for description of this data type. A key set may have up to 128 keys or certificates. The <rps:KeyType> element is described hereunder in this section.
<rps:Certificate>	This element maps the “certificates” data of CERTIFICATE type. See section 5.1.1.3.8 for description of this data type. A key set may have up to 128 keys or certificates. The <rps:GPCertificateType> is described hereunder in this section.

Table 221: <rps:Keyset> Attributes

The <rps:KeyType> contains the description of a key and maps the KEY type defined in section 5.1.1.3.6.

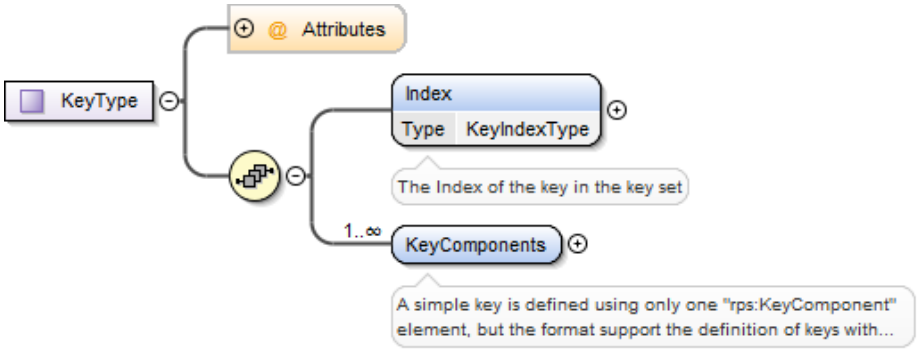


Figure 59: <rps:KeyType>

The <rps:KeyType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
@kcv	This attribute maps the “kcv” data of KEY type. See section 5.1.1.3.6 for description of this data type.
<rps:Index>	This element maps the “index” data of KEY type. See section 5.1.1.3.6 for description of this data type.
<rps:keyComponents>	This element maps the “keyComponents” data of KEY type. See section 5.1.1.3.6 for description of this data type. A key may have several components. The <rps:KeyComponent> element is described hereunder in this section.

Table 222: <rps:KeyType> Attributes

The <rps:KeyComponent> element contains the description of a key component and maps the KEY-COMPONENT type defined in section 5.1.1.3.5.

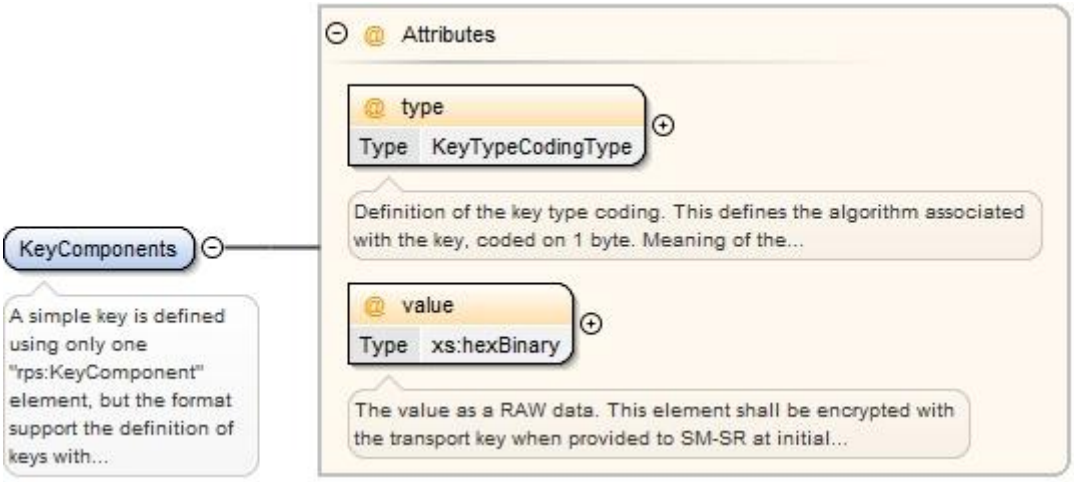


Figure 60: <rps:KeyComponent>

The <rps:KeyComponent> element is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
@type	This attribute maps the "type" data of KEY-COMPONENT type. See section 5.1.1.3.5 for description of this data type.
@value	This element maps the "value" data of KEY- COMPONENT type. See section 5.1.1.3.5 for description of this data type.

Table 223: <rps:KeyComponent> Attributes

The <rps:GPCertificateType> contains the description of a key and maps the CERTIFICATE type. defined in section 5.1.1.3.7.

Figure 61: <rps:GPCertificateType>

The <rps:GPCertificateType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Index>	This element maps the "index" data of CERTIFICATE type. See section 5.1.1.3.7 for description of this data type.
<rps:CaId>	This element maps the "ca-id" data of CERTIFICATE type. See section 5.1.1.3.7 for description of this data type.
<rps:Value>	This element maps the "value" data of CERTIFICATE type. See section 5.1.1.3.7 for description of this data type.

Table 224: <rps:GPCertificateType> Attributes

A.3.5.6 Audit Trail

The <rps:AuditTrail> contains a list of <rps:Record> of type <rps:AuditTrailRecordType>.

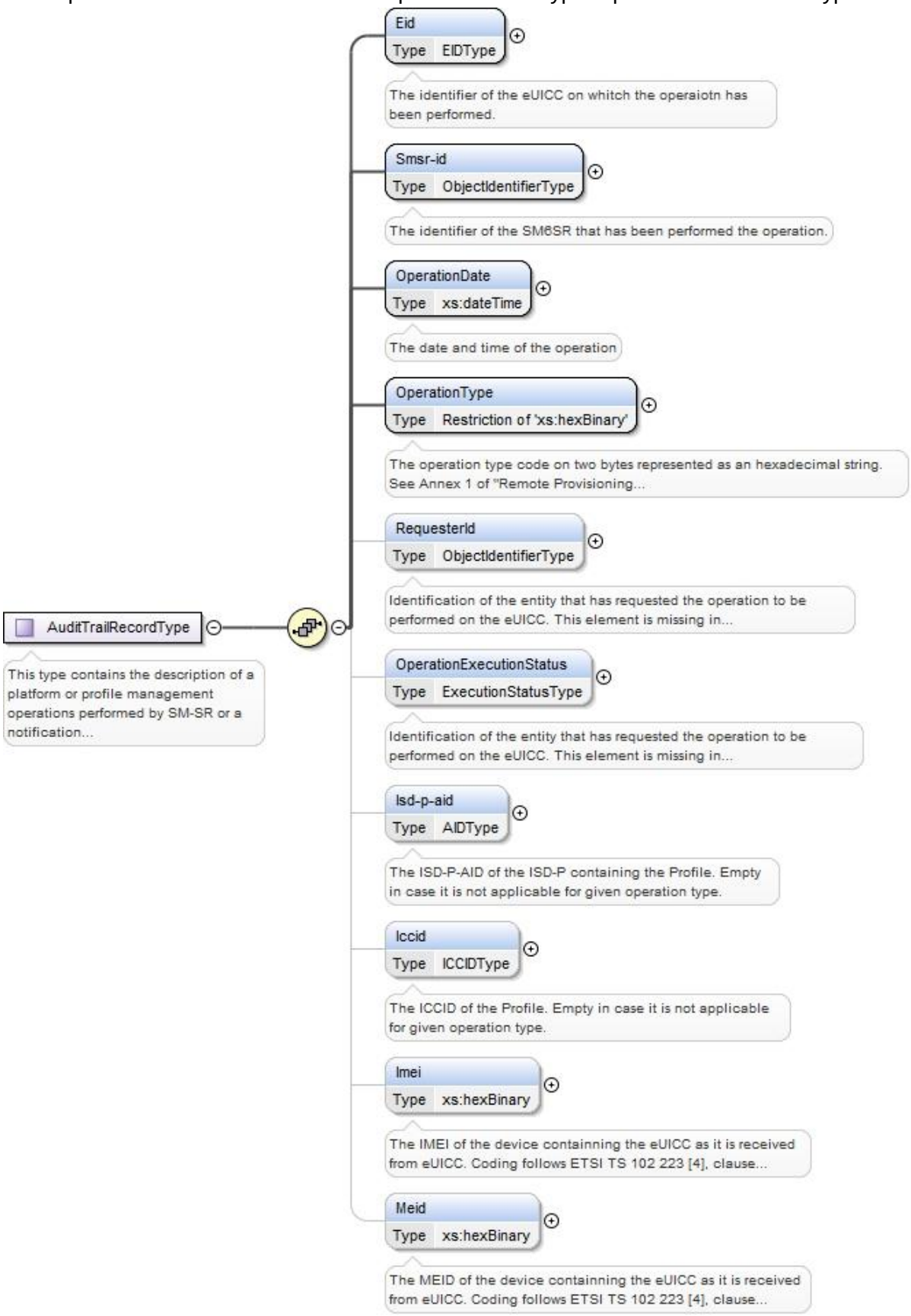


Figure 62: <rps:AuditTrail> Element

The <rps:AuditTrailRecordType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Eid>	This element maps the "eid" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:EIDType> is described in section A.3.4.3 of this Annex.
<rps:Smsr-id>	This element maps the "SMSRId" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:ObjectIdentifierType> is described in section A.3.4.7 of this Annex.
<rps:OperationDate>	This element maps the "operationDate" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type.
<rps:OperationType>	This element maps the "operationType" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The operation type is coded on two bytes represented as a hexadecimal string, with possible following values: '0001': Notification: eUICC declaration – First network attachment '0002': Notification: Profile change succeeded '0003': Notification: Profile change failed and Roll-back '0004': to '00FF': RFU for other Notification type '0005' Notification: Profile change after Fall-back '0100': CreateISDP '0200': EnableProfile '0300': DisableProfile '0400': DeleteProfile '0500': eUICCCapabilityAudit '0600': MasterDelete '0700': SetFallbackAttribute '0800': EstablishISDRkeyset '0900':FinaliseISDRhandover '0A00' to 'FF00' RFU for other commands type
<rps:RequesterId>	This element maps the "requesterId" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:ObjectIdentifierType> is described in section A.3.4.7 of this Annex.
<rps:OperationExecutionStatus>	This element maps the "status" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:ExecutionStatusType> is described in section A.3.3 of this Annex.
<rps:Isd-p-aid>	This element maps the "isd-p-aid" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:AIDType> is described in section A.3.4.1 of this Annex.
<rps:Iccid >	This element maps the "iccid" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The <rps:ICCIDType> is described in section A.3.4.4 of this Annex.
<rps:Imei>	This element maps the "IMEI" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The value is the hexadecimal value as received from the eUICC.

<rps:Meid>	This element maps the "MEID" data of the AUDIT-TRAIL-RECORD type. See section 5.1.1.3.11 for description of this data type. The value is the hexadecimal value as received from the eUICC.

eUICC CapabilitiesThe <rps:EuiccCapabilities> element maps the EUICC-CAPABILITIES data type defined in section 5.1.1.3.10.

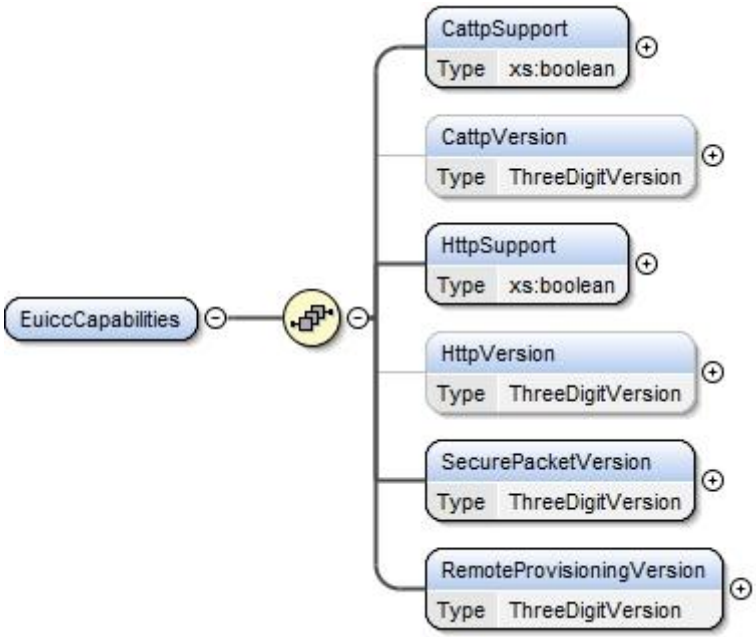


Figure 63: <rps:EuiccCapabilities>

The <rps:EuiccCapabilities> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:CattpSupport>	This element maps the "CAT_TP-Support" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.
<rps:CattpVersion>	This element maps the "CAT_TP-Version" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.
<rps:HttpSupport>	This element maps the "HTTP-Support" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.
<rps:HttpVersion>	This element maps the "HTTP-Version" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.
<rps:SecurePacketVersion>	This element maps the "secure-packet-version" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.
<rps:RemoteProvisioningVersion>	This element maps the "remote-provisioning-version" data of EUICC-CAPABILITIES type. See section 5.1.1.3.10 for description of this data type.

A.3.5.7 POL2 and POL2 rules

The <rps:POL2Type> maps the POL2 data type defined in section 5.1.1.3.3.

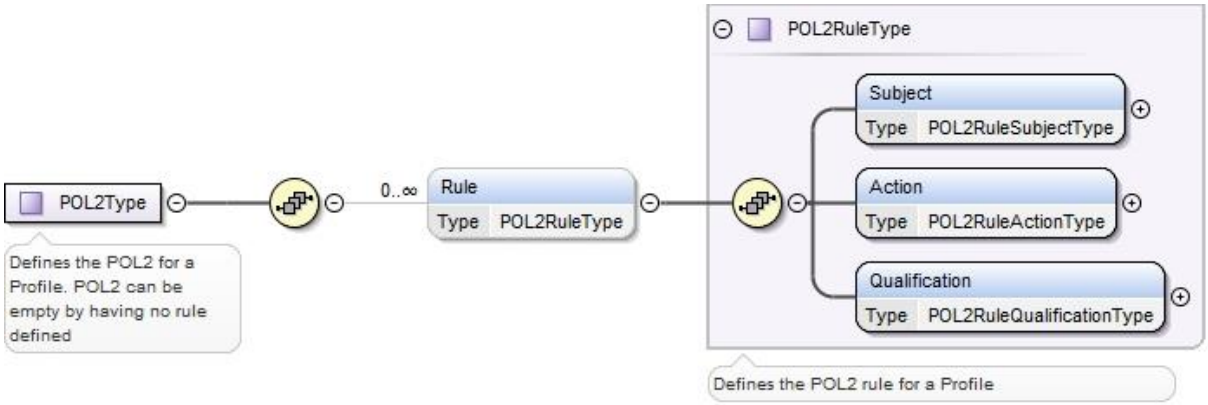


Figure 64: <rps:POL2Type>

The <rps:POL2Type> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Rule>	This element maps the "Rules" data of POL2 type. See section 5.1.1.3.3 for description of this data type. A POL2 may have several <rps:Rule>. The <rps:POL2RuleType> is defined hereunder in this section.

The <rps:POL2RuleType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Subject>	This element maps the "subject" data of POL2-RULE type. See section 5.1.1.3.2 for description of this data type. The <rps:POL2RuleSubjectType> is defined as an enumeration with possible value "PROFILE".
<rps:Action>	This element maps the "action" data of POL2-RULE type. See section 5.1.1.3.2 for description of this data type. The <rps:POL2RuleActionType> is defined as an enumeration with possible values "ENABLE", "DISABLE" or "DELETE".
<rps:Qualification>	This element maps the "qualification" data of POL2-RULE type. See section 5.1.1.3.2 for description of this data type. The <rps:POL2RuleQualificationType> is defined as an enumeration with possible values "Not-Allowed", "Auto-Delete".

A.3.5.8 Subscription Address

The <rps:SubscriptionAddressType> maps the SUBSCRIPTION-ADDRESS data type defined in section 5.1.1.3.1.

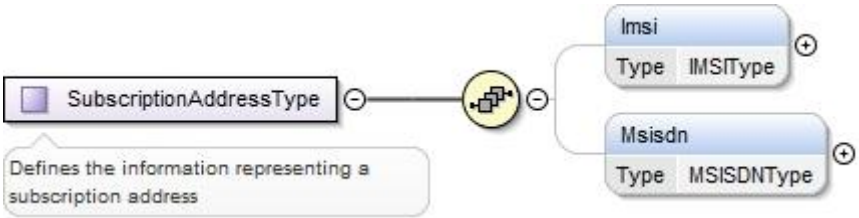


Figure 65: <rps:SubscriptionAddressType>

The <rps:SubscriptionAddressType> is composed of the following elements and attributes:

Attribute/Element	Mapping/Description
<rps:Imsi>	This element maps the "imsi" data of SUBSCRIPTION-ADDRESS type. See section 5.1.1.3.1 for description of this data type. The <rps:IMSIType> is described in section A.3.4.6 of this Annex.
<rps:Msisdn>	This element maps the "msisdn" data of SUBSCRIPTION-ADDRESS type. See section 5.1.1.3.1 for description of this data type. The <rps:MSISDNType> is described in section A.3.4.5 of this Annex.

A.4 The ES1 Interface Functions

A.4.1 The "ES1.RegisterEIS" Function

The input data of the "**ES1.RegisterEIS**" function defined in section 5.2.1 shall be mapped to the <rps:ES1-RegisterEISRequest> element described in the following figure:

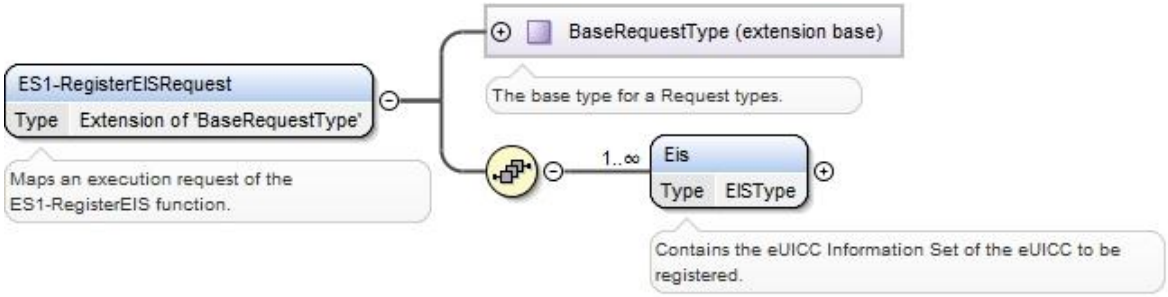


Figure 66: <rps:RegisterEISRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES1-RegisterEISRequest".

The output data of the "**ES1.RegisterEIS**" function defined in section 5.2.1 shall be mapped to the <rps:ES1-RegisterEISResponse> element described in the following figure:

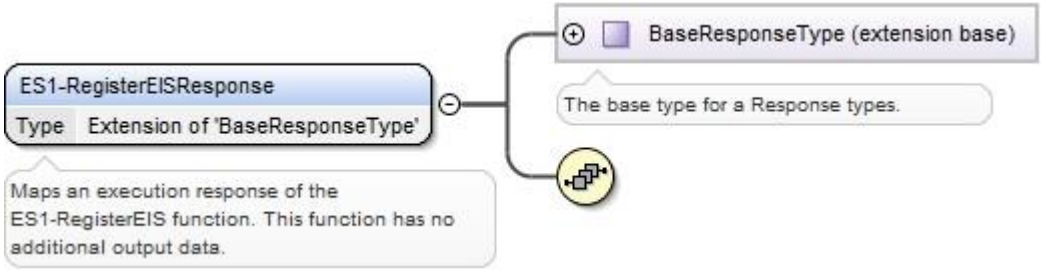


Figure 67: <rps:RegisterEISResponse>

This response doesn't carry any additional output data.

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "RegisterEISResponse".

A.5 The ES2 Interface Functions

A.5.1 The “ES2.GetEIS” Function

The input data of the “**ES2.GetEIS**” function defined in section 5.3.1 shall be mapped to the <rps:ES2-GetEISRequest> element described in the following figure:

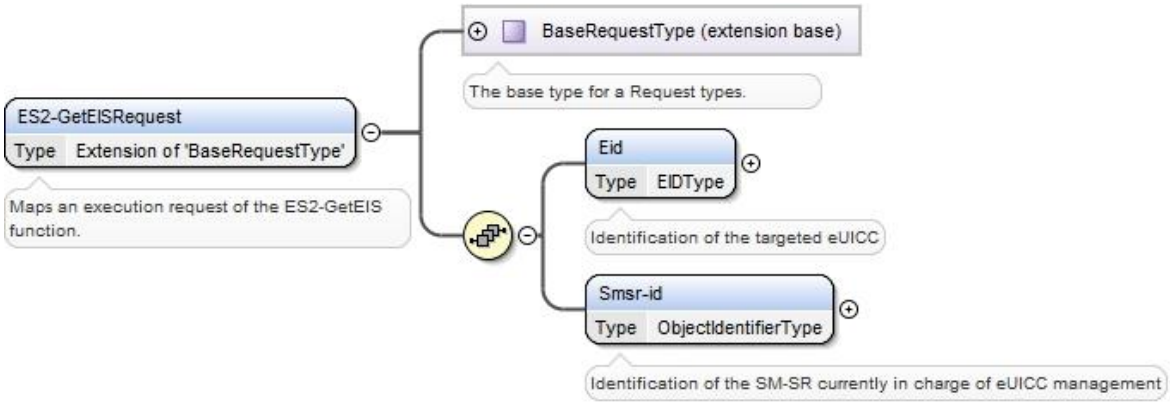


Figure 68: <rps:ES2-GetEISRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-GetEISRequest".

The output data of the “**ES2.GetEIS**” function defined in section 5.3.1 shall be mapped to the <rps:ES2-GetEISResponse> element described in the following figure:

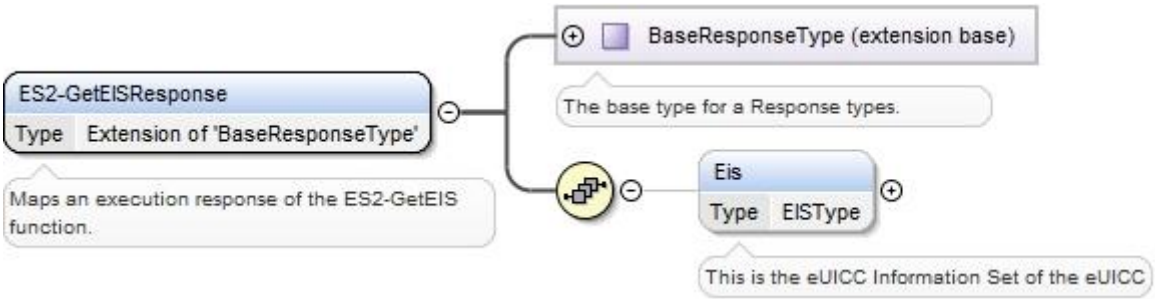


Figure 69: <rps:ES2-GetEISResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES2-GetEISResponse".

In case of function execution success or success with warning, the returned <rps:Eis> shall be filled with EIS as described in Annex E.

In case of function execution failure or expiration, no EIS shall be returned.

A.5.2 The "ES2.DownloadProfile" Function

The input data of the "**ES2.DownloadProfile**" function defined in section 5.3.2 shall be mapped to the <rps:ES2-DownloadProfileRequest> element described in the following figure:

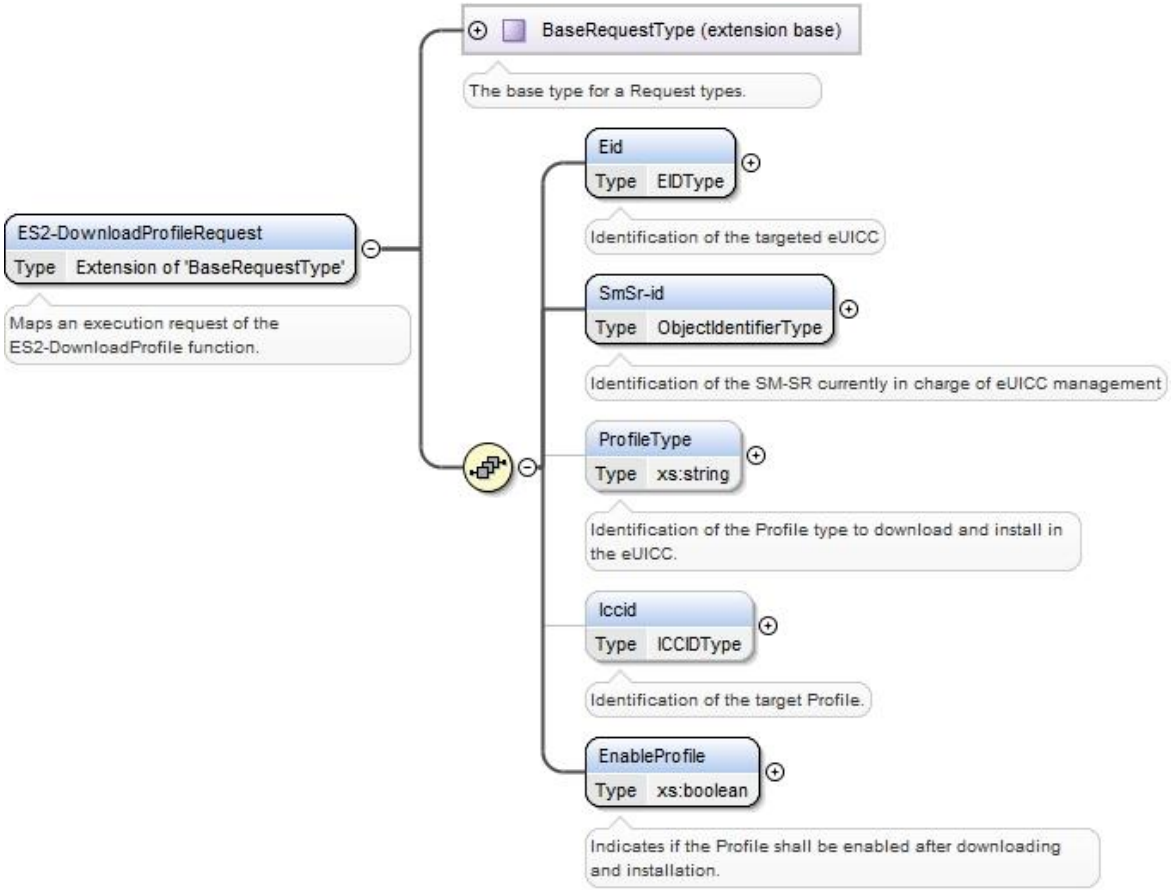


Figure 70: <rps:ES2-DownloadProfileRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-DownloadProfileRequest".

The output data of the "**ES2.DownloadProfile**" function defined in section 5.3.2 shall be mapped to the <rps:DownloadProfileResponse> element described in the following figure:

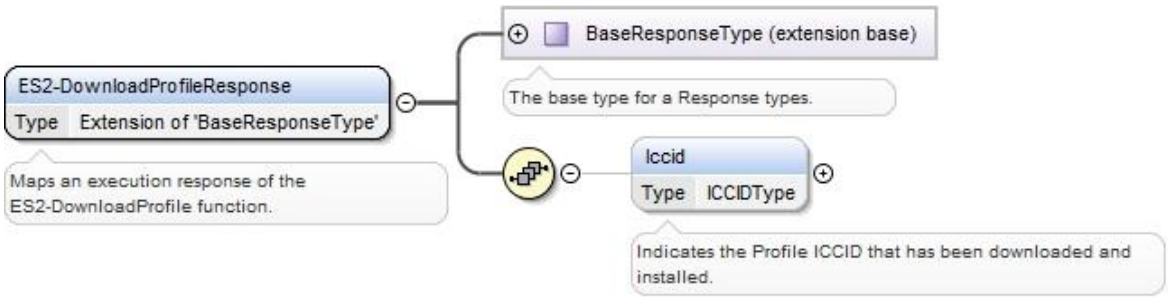


Figure 71: <rps:ES2-DownloadProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-DownloadProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution.

A.5.3 The "ES2.UpdatePolicyRules" Function

The input data of the "**ES2.UpdatePolicyRules**" function defined in section 5.3.3 shall be mapped to the <rps:ES2-UpdatePolicyRulesRequest> element described in the following figure:

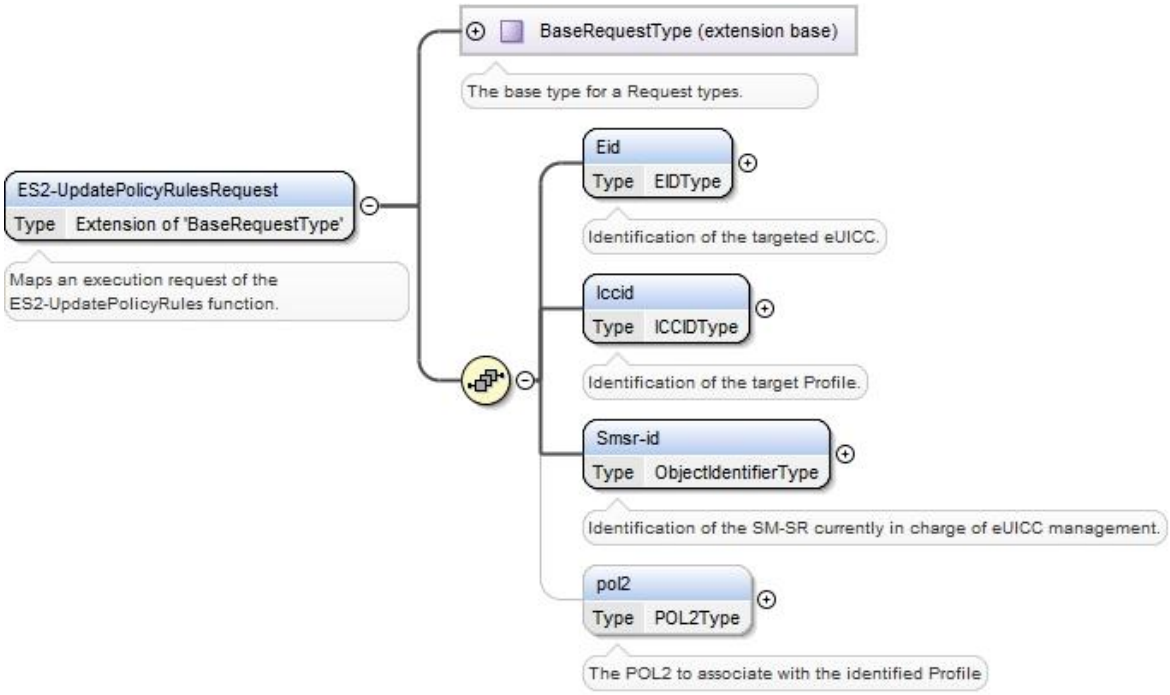


Figure 72: <rps:ES2-UpdatePolicyRulesRequest >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-UpdatePolicyRules".

The output data of the "**ES2.UpdatePolicyRules**" function defined in section 5.3.3 shall be mapped to the <rps:ES2.UpdatePolicyRulesResponse> element described in the following figure:

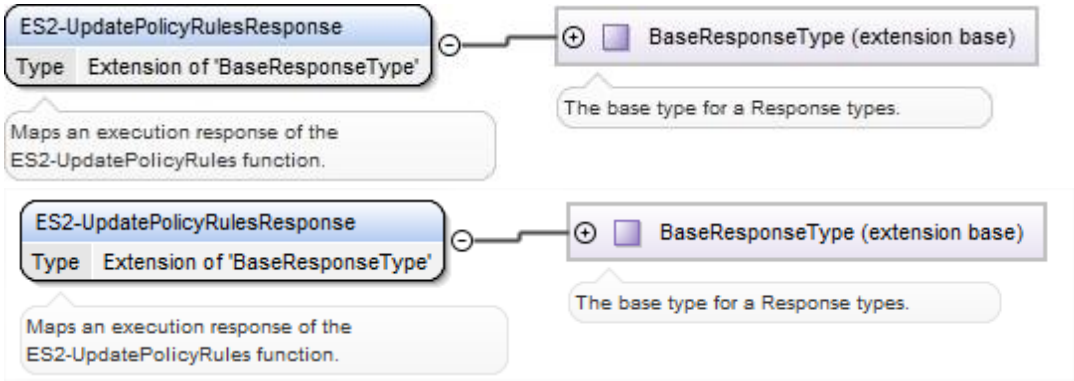


Figure 73: <rps:ES2-UpdatePolicyRulesResponse >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES2-UpdatePolicyRulesResponse".

In case of function execution success or success with warning, the returned function may not return any eUICC response in the <rps:EuiccResponseData> element.

In case of function execution failure or expiration the eUICC response may be provided.

A.5.4 The “ES2.UpdateSubscriptionAddress” Function

The input data of the “**ES2.UpdateSubscriptionAddress**” function defined in section 5.3.4 shall be mapped to the <rps:ES2-UpdateSubscriptionAddressRequest> element described in the following figure:

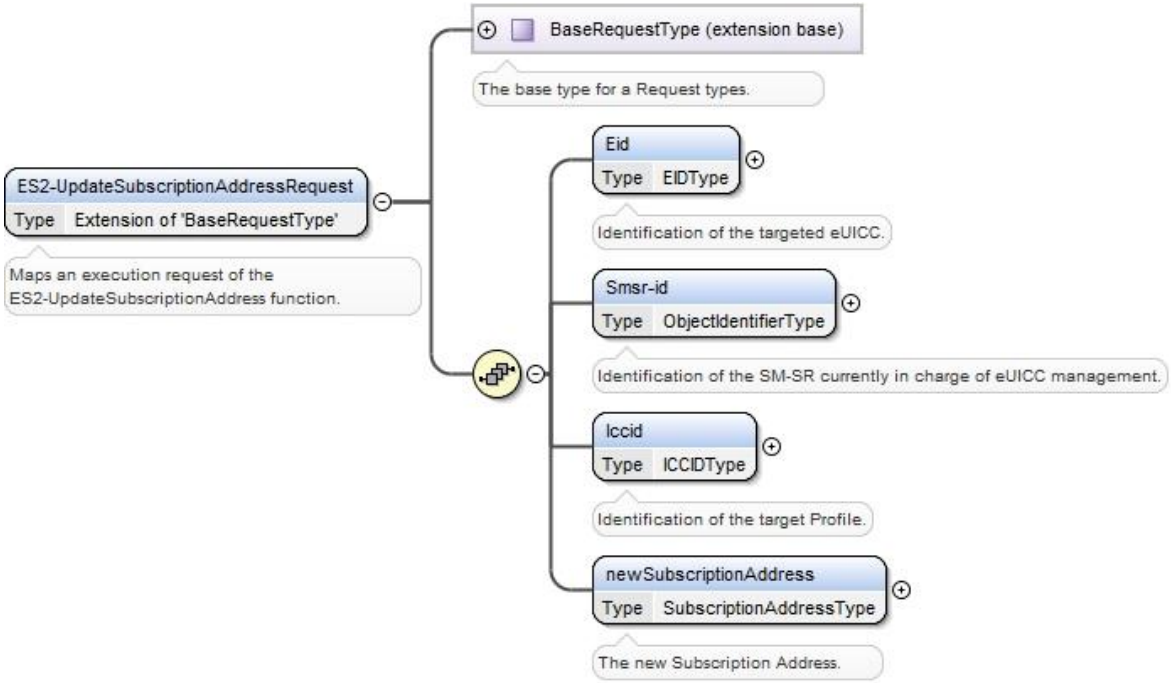


Figure 74: <rps:ES2-UpdateSubscriptionAddressRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-UpdateSubscriptionAddressRequest".

The output data of the “**ES2.UpdateSubscriptionAddress**” function defined in section 5.3.4 shall be mapped to the <rps:ES2-UpdateSubscriptionAddressResponse> element described in the following figure:

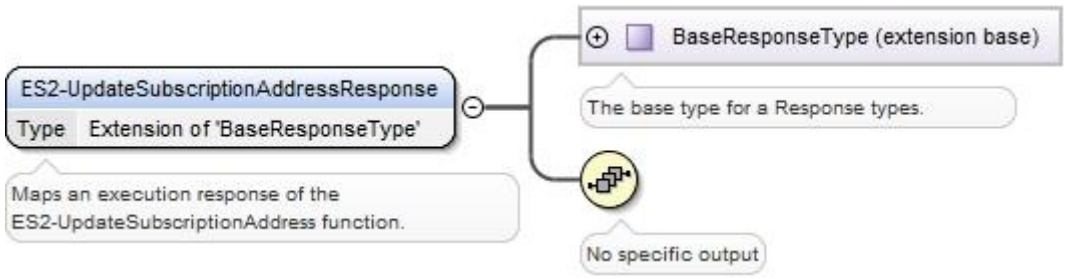


Figure 75: <rps:ES2-UpdateSubscriptionAddressResponse>

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-UpdateSubscriptionAddressResponse".

A.5.5 The "ES2. EnableProfile" Function

The input data of the "**ES2.EnableProfile**" function defined in section 5.3.5 shall be mapped to the <rps:ES2-EnableProfileRequest> element described in the following figure:

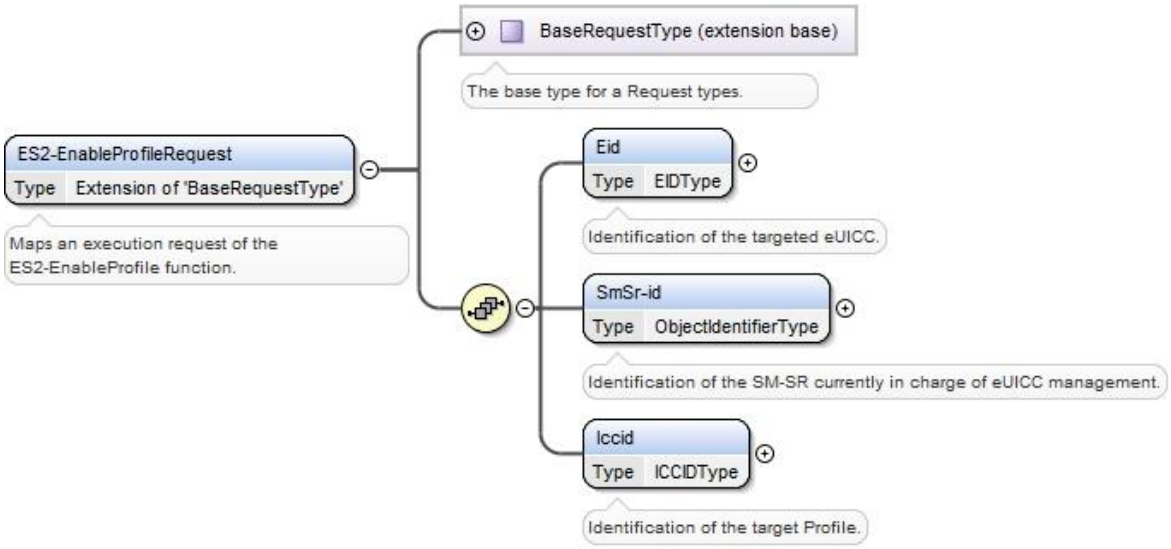


Figure 76: <rps:ES2-EnableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-EnableProfileRequest".

The output data of the "**ES2.EnableProfile**" function defined in section 5.3.5 shall be mapped to the <rps:EnableProfileResponse> element described in the following figure:

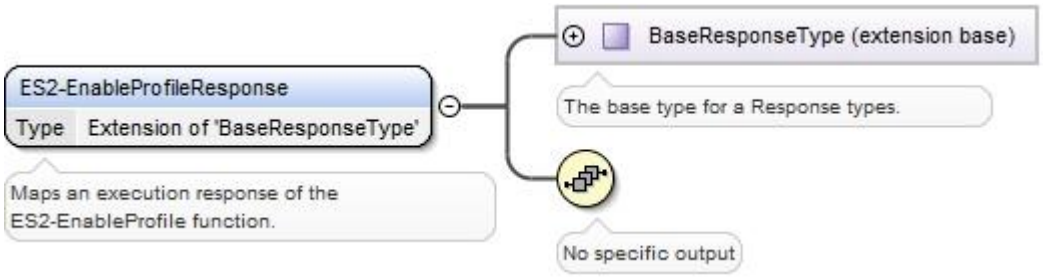


Figure 77: <rps:ES2-EnableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-EnableProfileResponse".

A.5.6 The “ES2.DisableProfile” Function

The input data of the “**ES2.DisableProfile**” function defined in section 5.3.6 shall be mapped to the <rps:ES2-DisableProfileRequest> element described in the following figure:

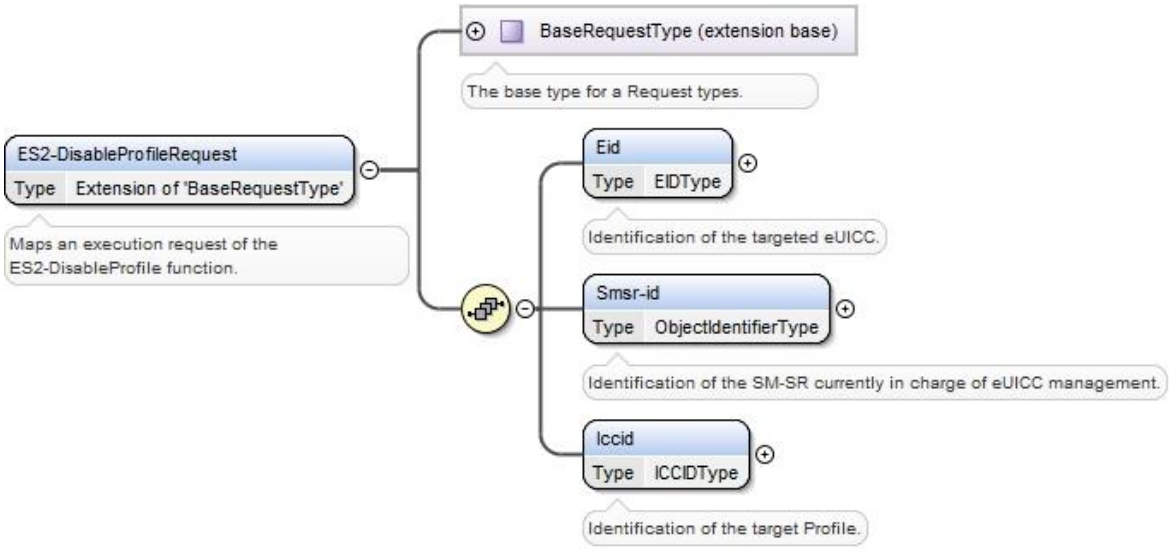


Figure 78: <rps:ES2-DisableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-DisableProfileRequest".

The output data of the “**ES2.DisableProfile**” function defined in section 5.3.6 shall be mapped to the <rps:DisableProfileResponse> element described in the following figure:

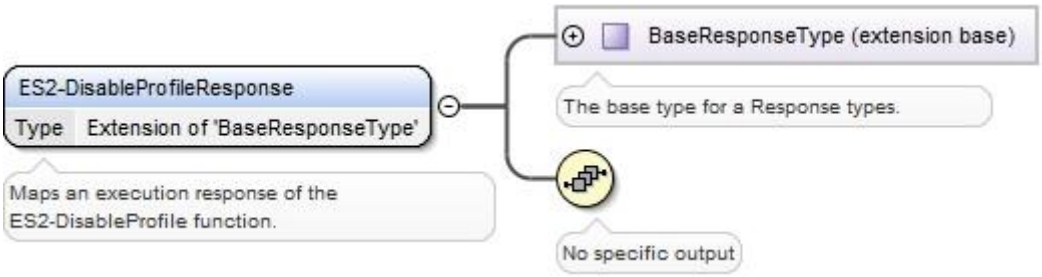


Figure 79: <rps:ES2-DisableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES2-DisableProfileResponse".

A.5.7 The “ES2.DeleteProfile” Function

The input data of the “**ES2.DeleteProfile**” function defined in section 5.3.7 shall be mapped to the <rps:ES2-DeleteProfileRequest> element described in the following figure:

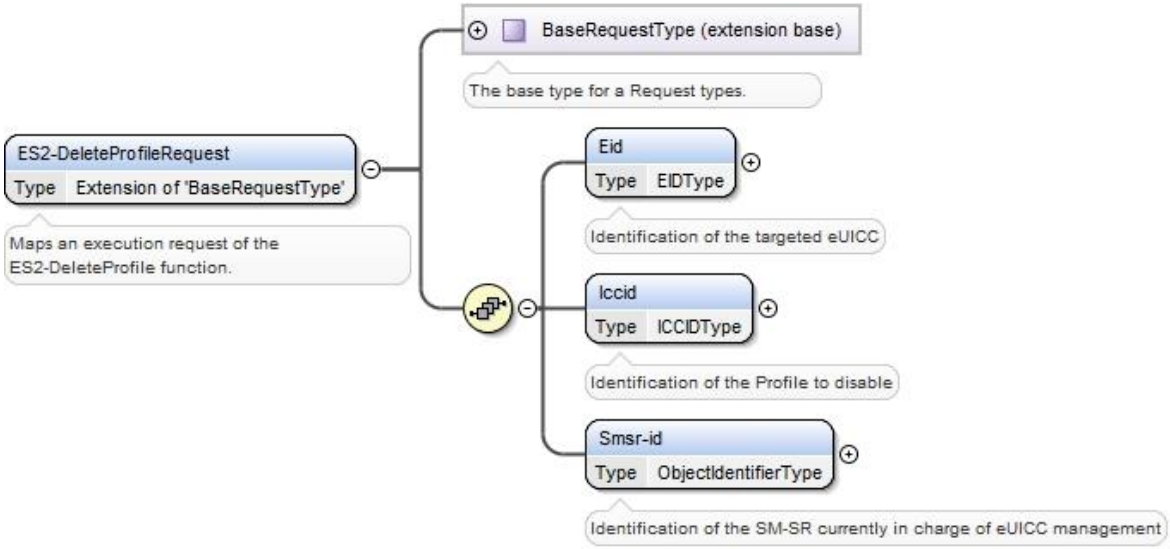


Figure 80: <rps:ES2-DeleteProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-DeleteProfileRequest".

The output data of the "**ES2.DeleteProfile**" function defined in section 5.3.7 shall be mapped to the <rps:DeleteProfileResponse> element described in the following figure:

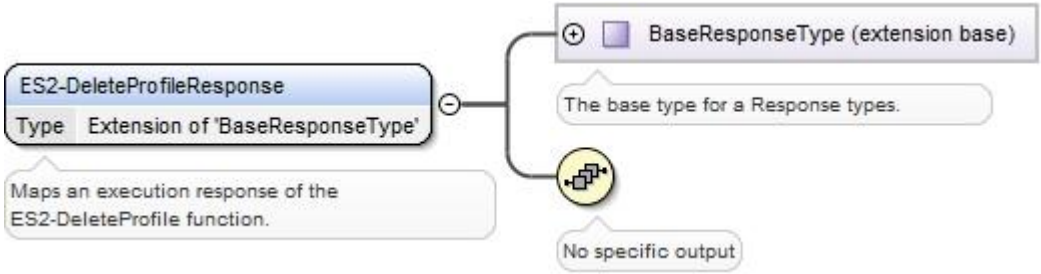


Figure 81: <rps:ES2-DeleteProfile Response>

This response doesn't carry any additional output data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-DeleteProfileResponse".

A.5.8 The "ES2.HandleProfileDisabledNotification" Function

The input data of the "**ES2.HandleProfileDisabledNotification**" function defined in section 5.3.8 shall be mapped to the <rps:ES2-HandleProfileDisabledNotification> element described in the following figure:

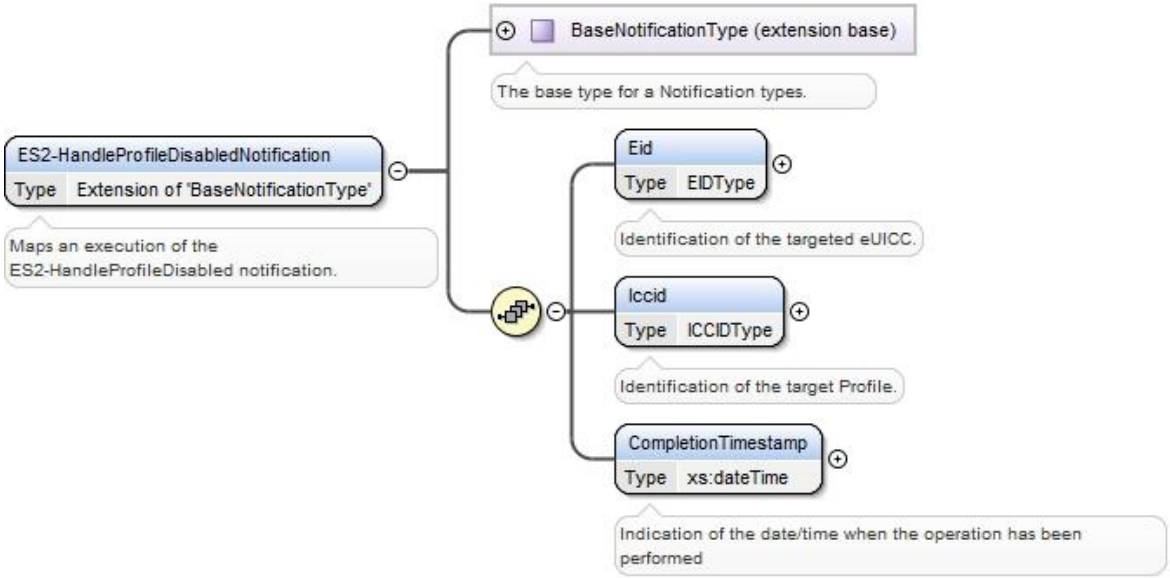


Figure 82: <rps:ES2- HandleProfileDisabledNotification >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES2-HandleProfileDisabledNotification".

A.5.9 The “ES2.HandleProfileEnabledNotification” Function

The input data of the “**ES2.HandleProfileEnabledNotification**” function defined in section 5.3.9 shall be mapped to the <rps:ES2-HandleProfileEnabledNotification> element described in the following figure:

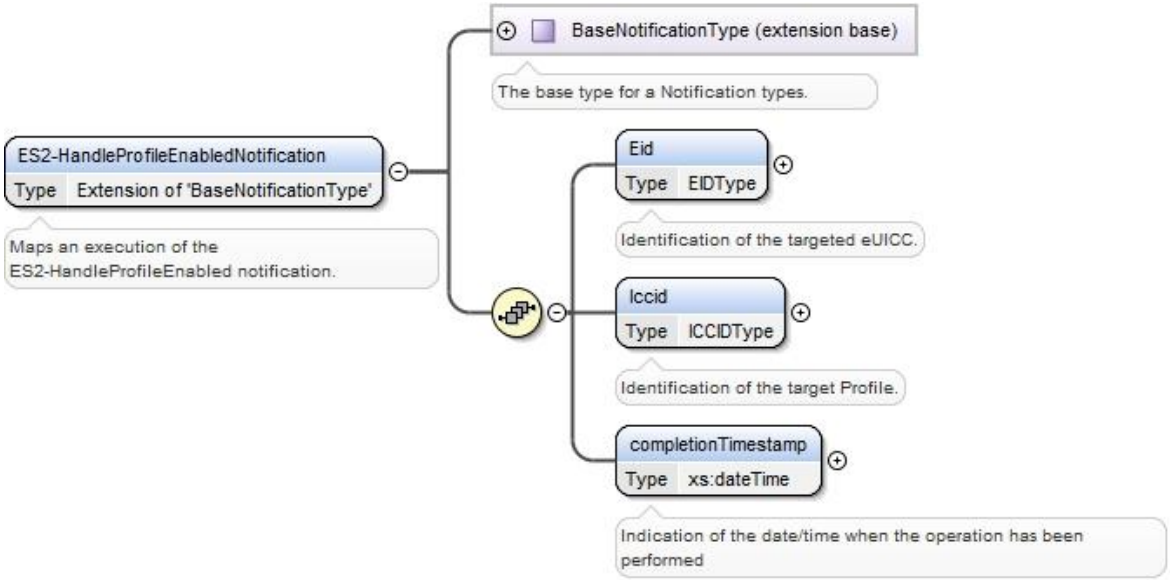


Figure 83: <rps:ES2-HandleProfileEnabledNotification >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES2-HandleProfileEnabledNotification".

A.5.10 The "ES2.HandleSMSRChangedNotification" Function

The input data of the "**ES2.HandleSMSRChangeNotification**" function defined in section 5.3.10 shall be mapped to the <rps:ES2-HandleSMSRChangeNotificationRequest> element described in the following figure:

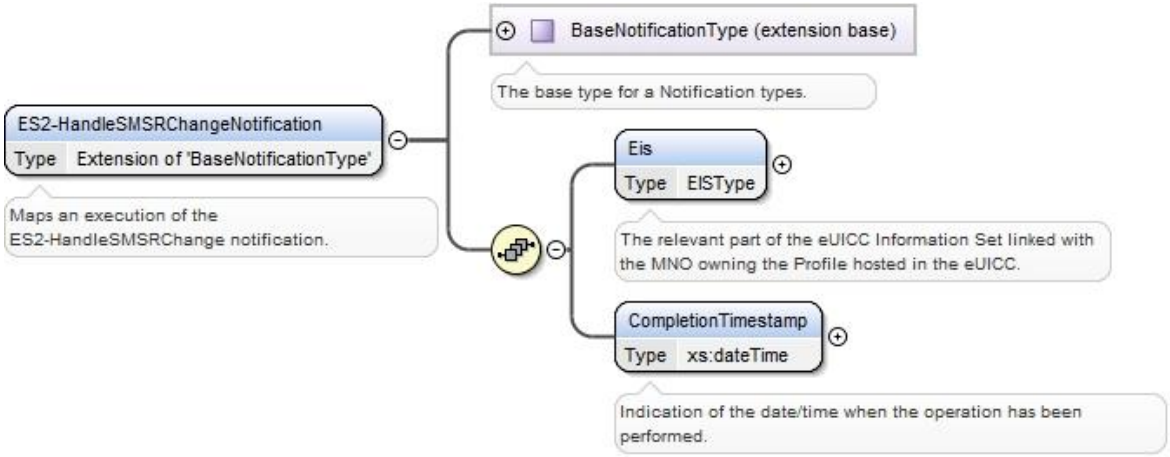


Figure 84: <rps:ES2-HandleSMSRChangeNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-HandleSMSRChangeNotification".

A.5.11 The "ES2.HandleProfileDeletedNotification" Function

The input data of the "**ES2.HandleProfileDeletedNotification**" function defined in section 5.3.11 shall be mapped to the <rps:ES2-HandleProfileDeletedNotification> element described in the following figure:

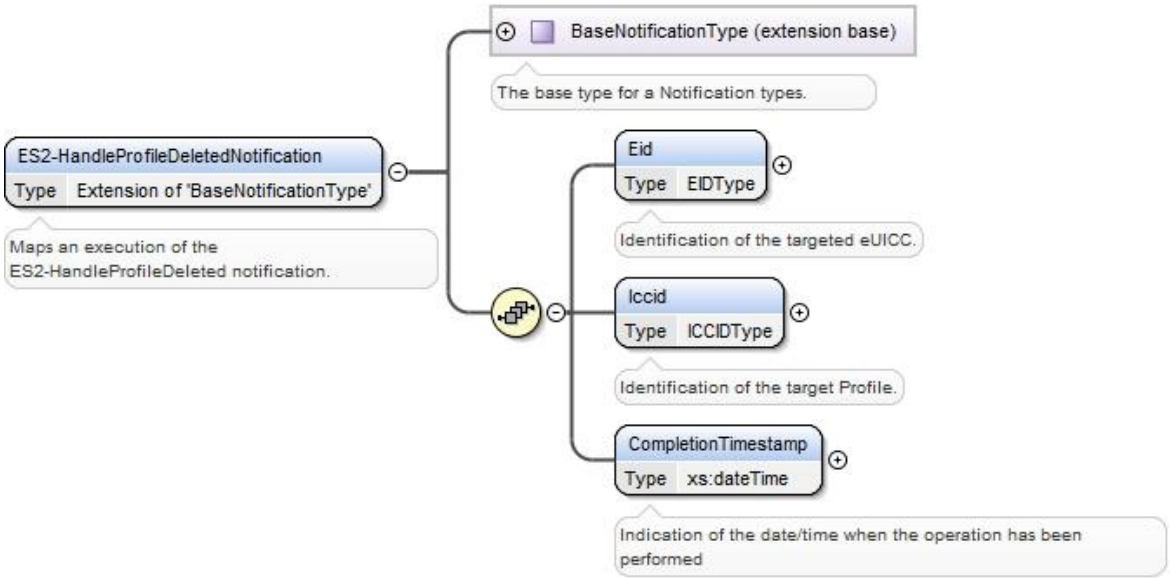


Figure 85: <rps:ES2-HandleProfileDeletedNotification >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES2-HandleProfileDeletedNotification".

A.6 The ES3 Interface Functions

A.6.1 The "ES3.GetEIS" Function

The input data of the "**ES3.GetEIS**" function defined in section 5.4.1 shall be mapped to the <rps:ES3-GetEISRequest> element described in the following figure:

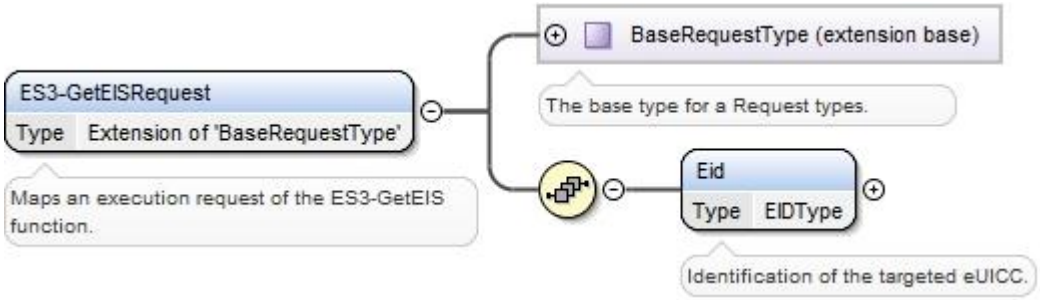


Figure 86: <rps:GetEISRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-GetEISRequest".

The output data of the "**ES3.GetEIS**" function defined in section 5.4.1 shall be mapped to the <rps:ES3-GetEISResponse> element described in the following figure:

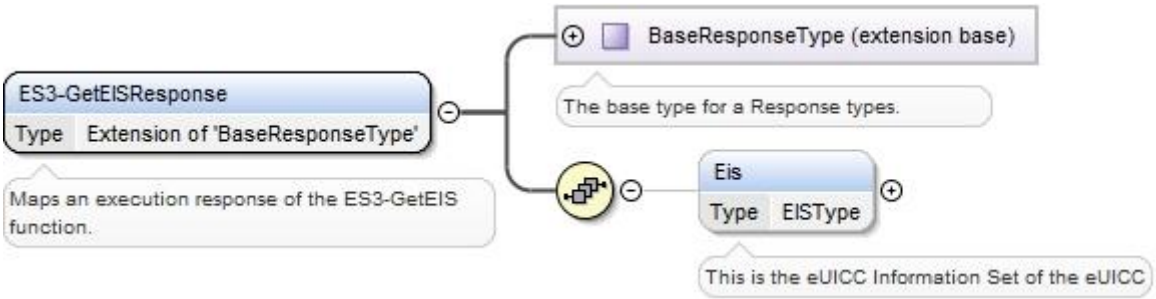


Figure 87: <rps:GetEISResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-GetEISResponse".

In case of function execution success or success with warning, the returned <rps:Eis> shall be filled with only:

- <rps:EumSignedInfo> (full content, including the information related to ECASD)
- <rps:EumSignature>
- <rps: RemainingMemory>
- <rps:AvailableMemoryForProfiles>
- <rps:LastAuditDate>; This element can be missing if no audit has been performed
- <rps:SmSr-id>, filled with the current SM-SR identification value

(No <rps:Profile> element, no <rps:AuditTrail> element).

In case of function execution failure orexpiration, no EIS shall be returned.

A.6.2 The “ES3.AuditEIS” Function

The input data of the “**ES3.AuditEIS**” function defined in section 5.4.2 shall be mapped to the <rps:ES3-AuditEISRequest> element described in the following figure:

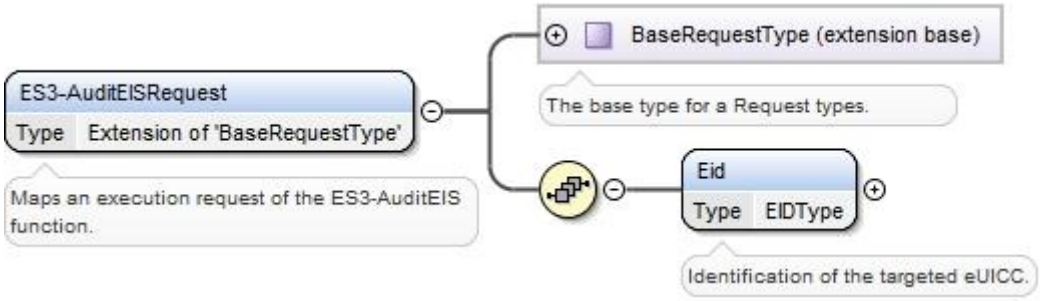


Figure 88: <rps:ES3-AuditEIS Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-AuditEISRequest".

The output data of the “**ES3.AuditEIS**” function defined in section 5.4.2 shall be mapped to the <rps:AuditEISResponse> element described in the following figure:

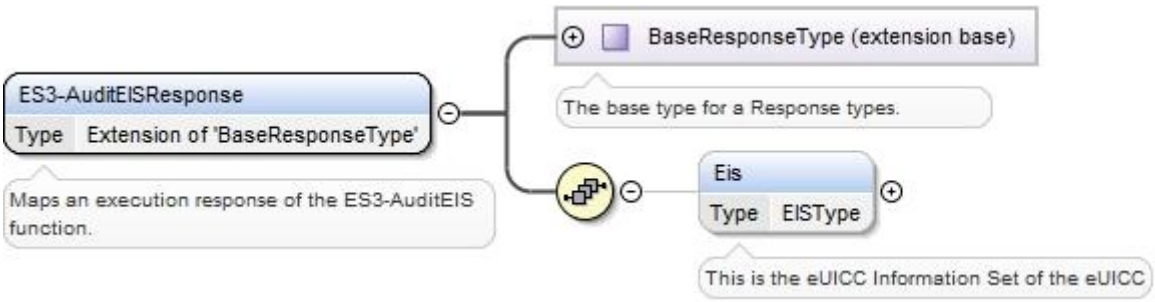


Figure 89: <rps:AuditEISResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3-AuditEISResponse".

In case of function execution success or success with warning, the returned <rps:Eis> shall be filled accordingly to what is described in section A.3.5.1 of this Annex.

In case of function execution failure or expiration, no EIS shall be returned.

A.6.3 The “ES3.CreateISDP” Function

The input data of the “**ES3.CreateISDP**” function defined in section 5.4.3 shall be mapped to the <rps: ES3-CreateISDPRequest> described in the following figure:

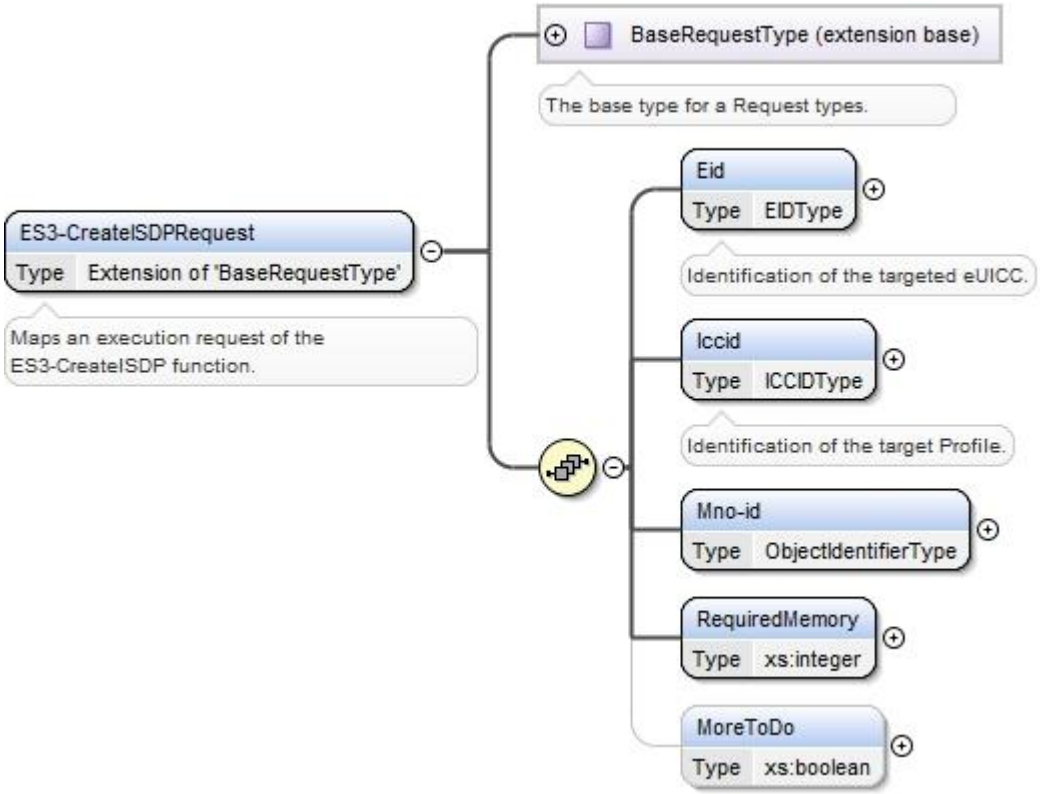


Figure 90: <rps:ES3-CreateISDPRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-CreateISDPRequest".

The output data of the "**ES3.CreateISDP**" function defined in section 5.4.3 shall be mapped to the <rps:ES3-CreateISDPResponse> element described in the following figure:

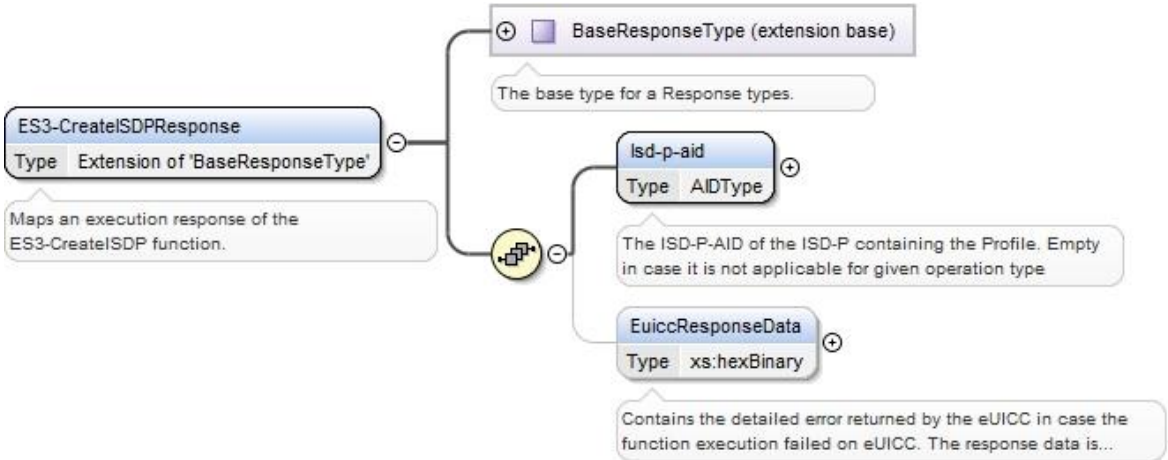


Figure 91: <rps:ES3-CreateISDPResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-CreateISDPResponse".

In case of function execution success or success with warning, the returned function shall return the ISD-P-AID value in the <rps:Isd-p-aid> element and the eUICC response in the <rps:EuiccResponseData> element.

In case of function execution failure of expiration, no ISP-P-AID shall be returned. The eUICC response may be provided.

A.6.4 The “ES3.SendData” Function

The input data of the “**ES3.SendData**” function defined in section 5.4.4 shall be mapped to the <rps:ES3-SendDataRequest> described in the following figure:

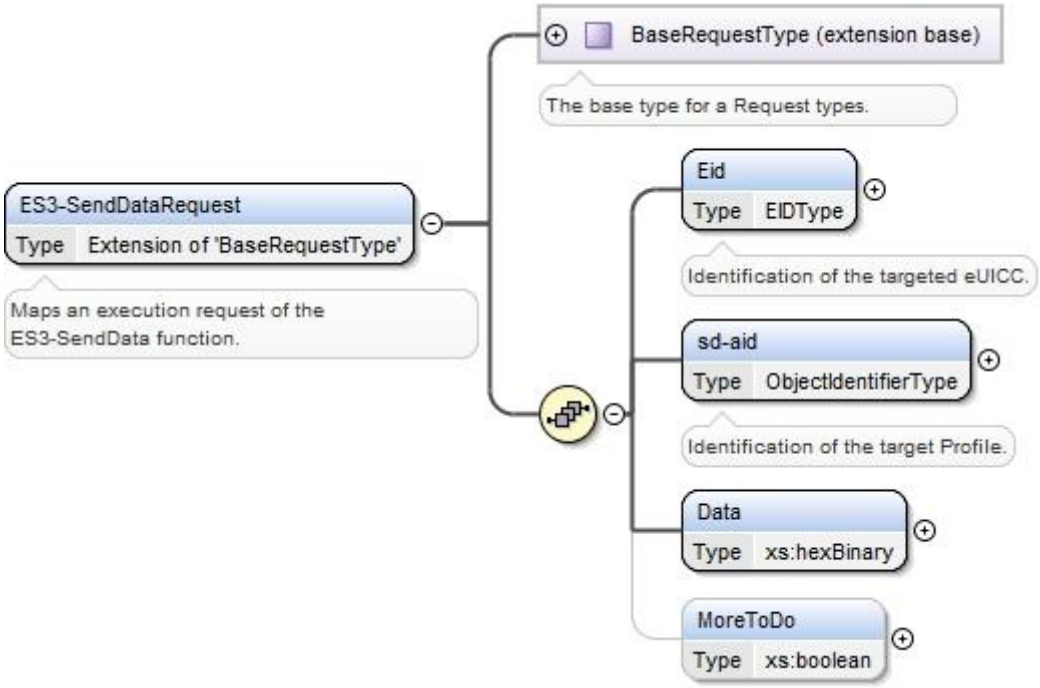


Figure 92: <rps:ES3-SendDataRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-SendDataRequest".

The output data of the “**ES3.SendData**” function defined in section 5.4.4 shall be mapped to the <rps:ES3-SendDataResponse> element described in the following figure:

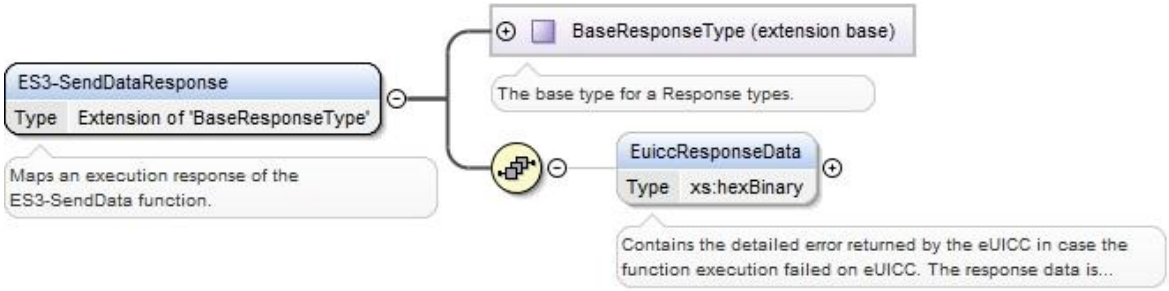


Figure 93: <rps:SendDataResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "SendDataResponse".

A.6.5 The "ES3.ProfileDownloadCompleted" Function

The input data of the "**ES3.ProfileDownloadCompleted**" function defined in section 5.4.5 shall be mapped to the <rps:ES3-ProfileDownloadCompletedRequest > element described in the following figure:

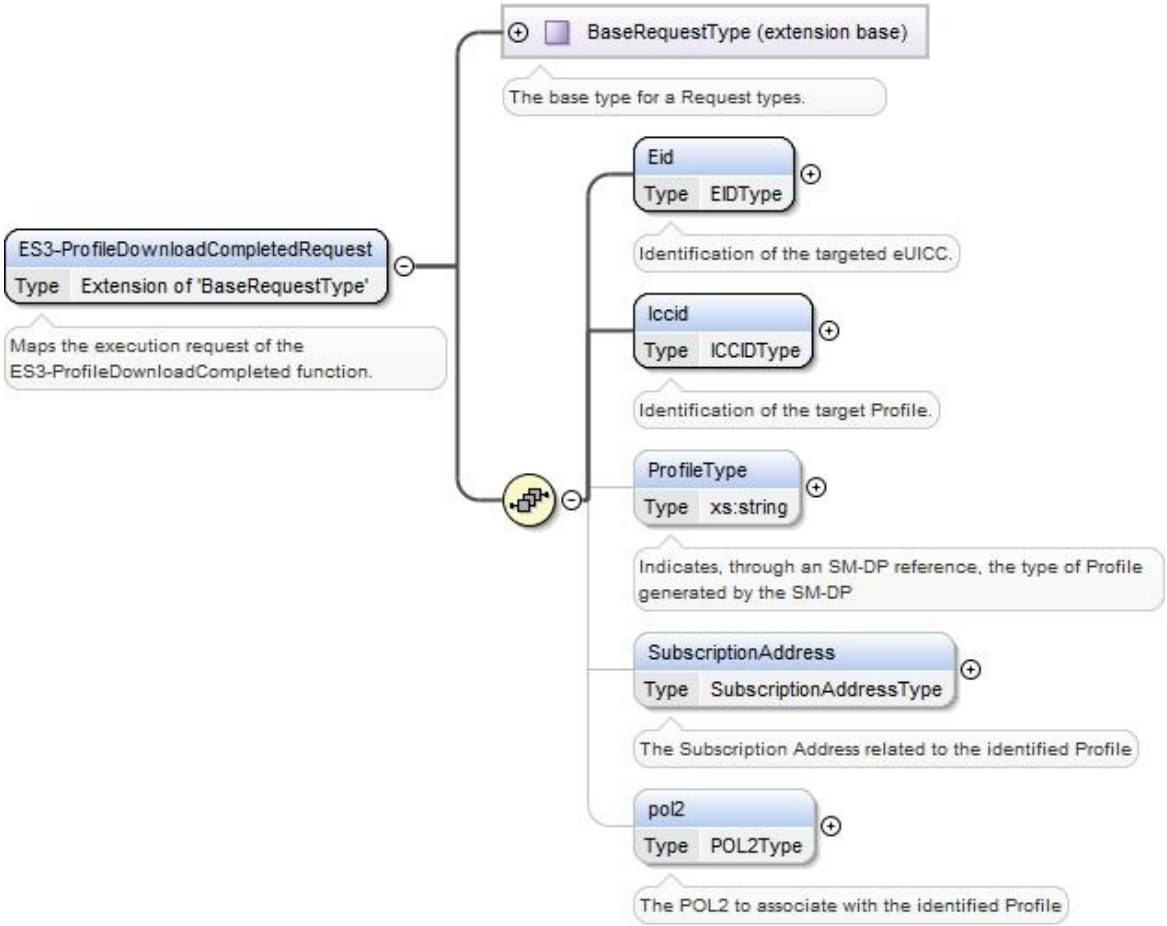


Figure 94: <rps:ES3-ProfileDownloadCompleted Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-ProfileDownloadCompletedRequest".

The output data of the "**ES3.ProfileDownloadCompleted**" function defined in section 5.4.5 shall be mapped to the <rps:ProfileDownloadCompletedResponse> element described in the following figure:

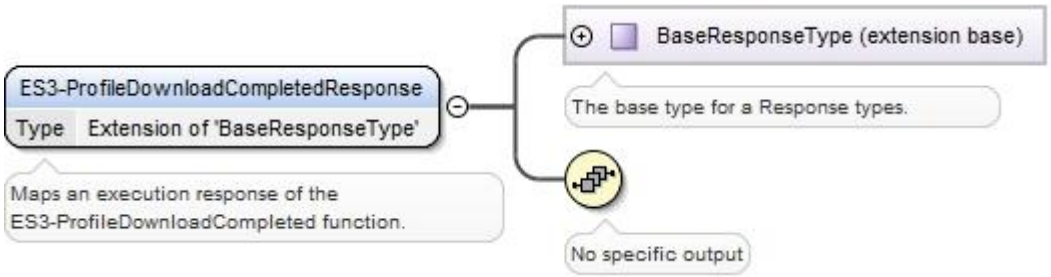


Figure 95: <rps:ES3-ProfileDownloadCompleted Response>

This response doesn't carry any additional output data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3-ProfileDownloadCompletedResponse".

A.6.6 The “ES3.UpdatePolicyRules” Function

The input data of the “**ES3.UpdatePolicyRules**” function defined in section 5.4.6 shall be mapped to the <rps:ES3-UpdatePolicyRulesRequest> element described in the following figure:

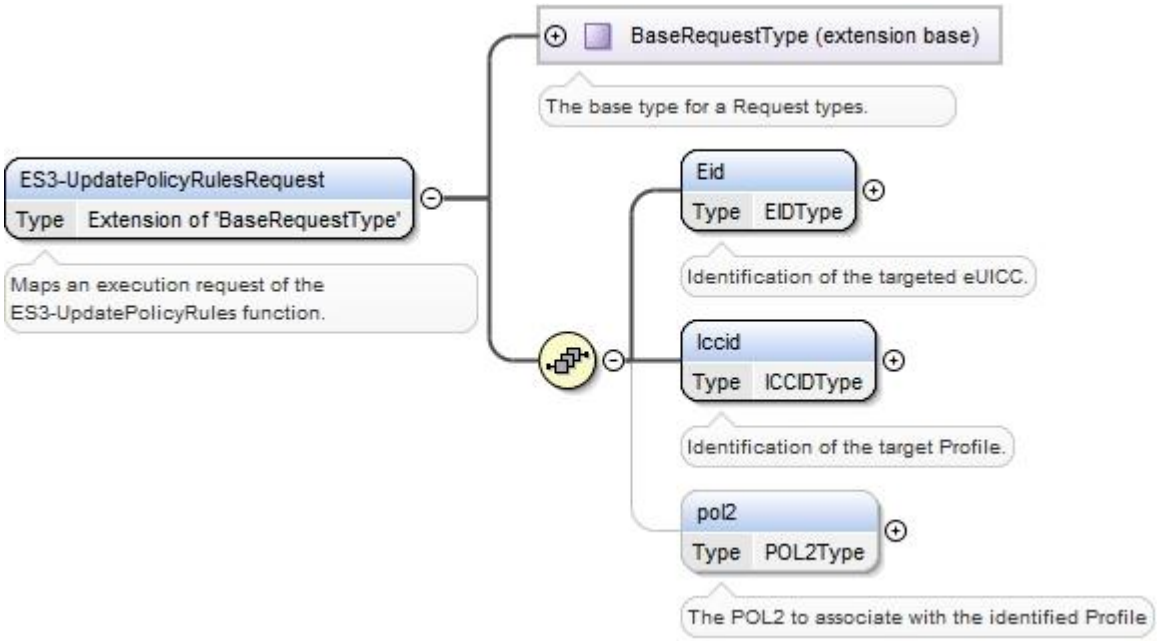


Figure 96: <rps:ES3-UpdatePolicyRulesRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-UpdatePolicyRules".

The output data of the “**ES3.UpdatePolicyRules**” function defined in section 5.4.6 shall be mapped to the <rps:ES3.UpdatePolicyRulesResponse> element described in the following figure:

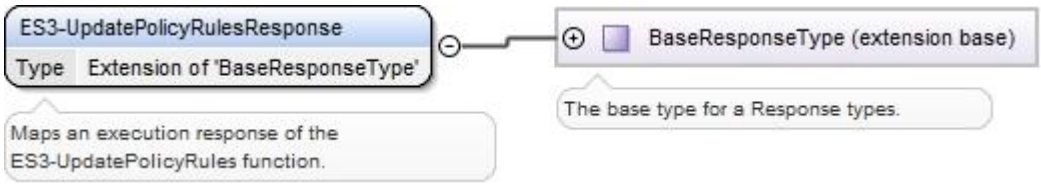


Figure 97: <rps:ES3-UpdatePolicyRulesResponse >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3-UpdatePolicyRulesResponse".

A.6.7 The “ES3.UpdateSubscriptionAddress” Function

The input data of the “**ES3.UpdateSubscriptionAddress**” function defined in section 5.4.7 shall be mapped to the <rps:ES3-UpdateSubscriptionAddressRequest> element described in the following figure:

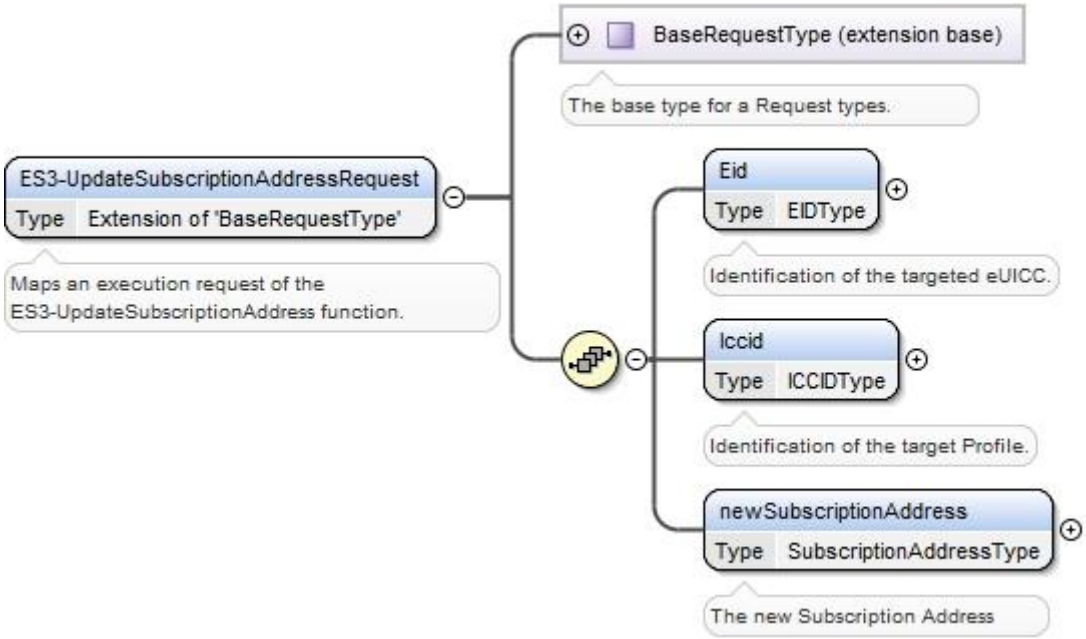


Figure 98: <rps:ES3-UpdateSubscriptionAddressRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-UpdateSubscriptionAddressRequest".

The output data of the "**ES3.UpdateSubscriptionAddress**" function defined in section 5.4.7 shall be mapped to the <rps:ES3-UpdateSubscriptionAddressResponse> element described in the following figure:

Figure 99: Void

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-UpdateSubscriptionAddressResponse".

A.6.8 The "ES3.EnableProfile" Function

The input data of the "**ES3.EnableProfile**" function defined in section 5.4.8 shall be mapped to the <rps:ES3-EnableProfileRequest> element described in the following figure:

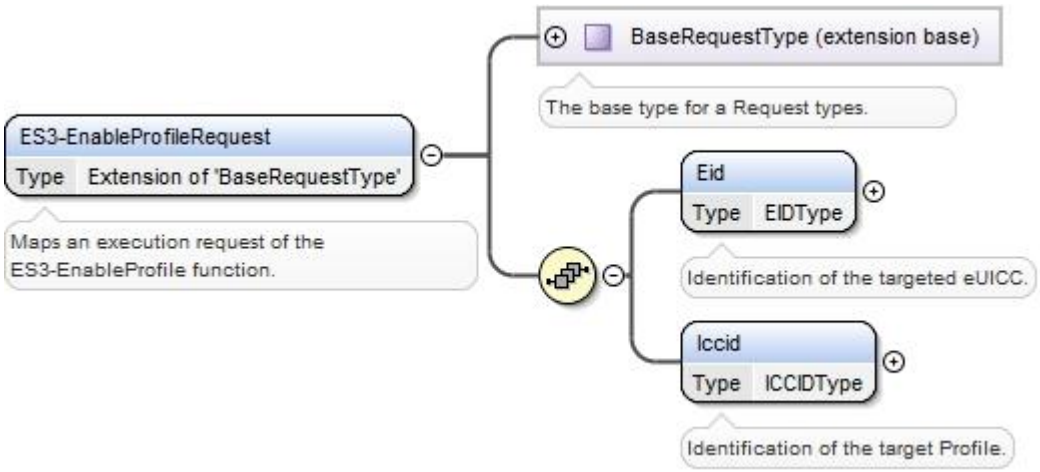


Figure 100: <rps:ES3-EnableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-EnableProfileRequest".

The output data of the "**ES3.EnableProfile**" function defined in section 5.4.8 shall be mapped to the <rps:EnableProfileResponse> element described in the following figure:

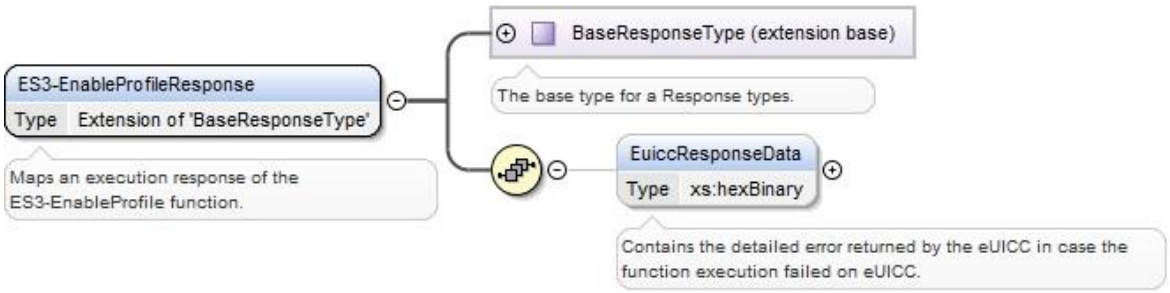


Figure 101: <rps:ES3-EnableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-EnableProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.6.9 The "ES3.DisableProfile" Function

The input data of the "**ES3.DisableProfile**" function defined in section 5.4.9 shall be mapped to the <rps:ES3-DisableProfileRequest> element described in the following figure:

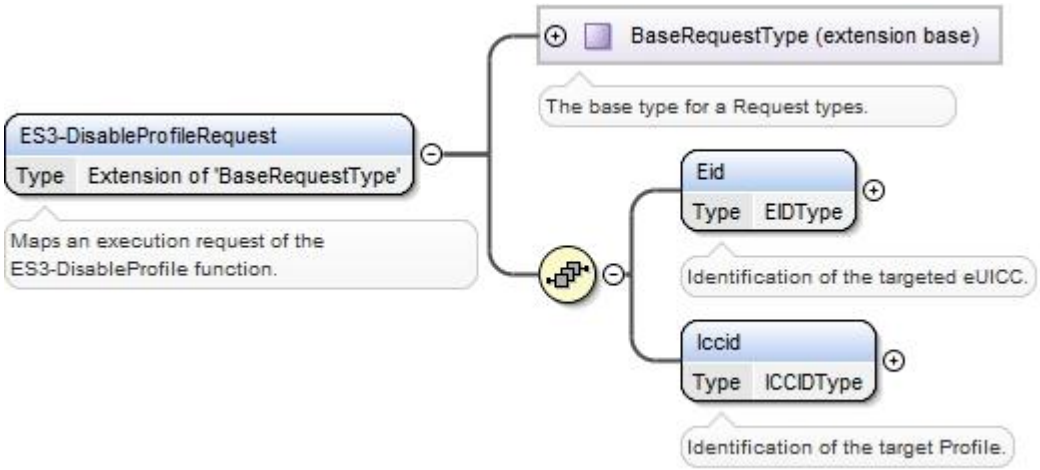


Figure 102: <rps:ES3-DisableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-DisableProfileRequest".

The output data of the "**ES3.DisableProfile**" function defined in section 5.4.9 shall be mapped to the <rps:DisableProfileResponse> element described in the following figure:

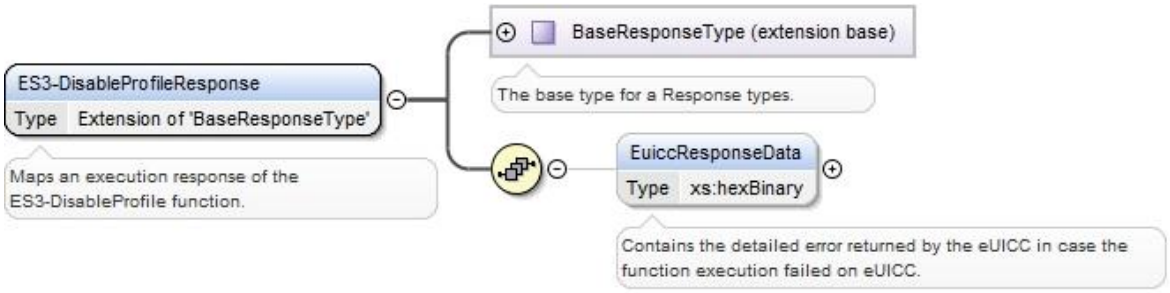


Figure 103: <rps:ES3-DisableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-DisableProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.6.10 The "ES3.DeleteISDP" Function

The input data of the "**ES3.DeleteISDP**" function defined in section 5.4.10 shall be mapped to the <rps:ES3-DeleteISDPRequest> element described in the following figure:

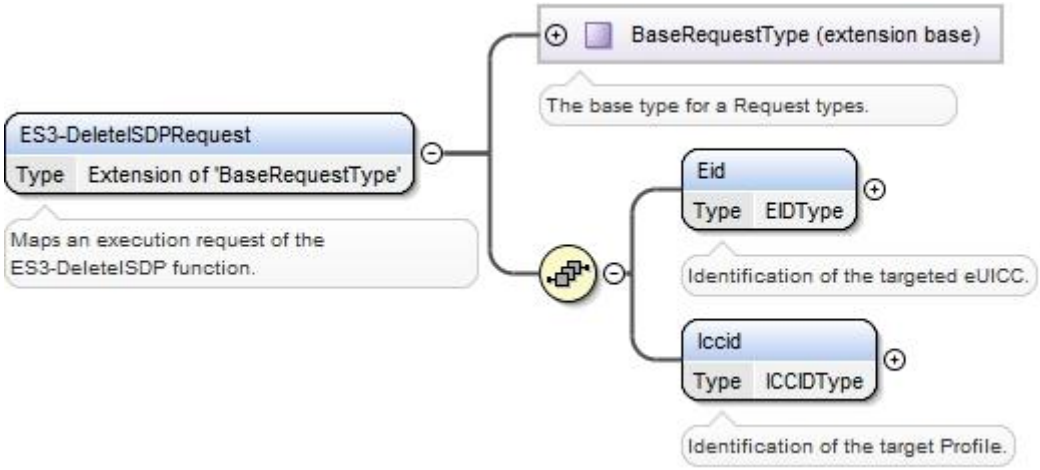


Figure 104: <rps:ES3-DeleteISDP Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-DeleteISDPRequest".

The output data of the "**ES3.DeleteISDP**" function defined in section 5.4.10 shall be mapped to the <rps:DeleteISDPResponse> element described in the following figure:

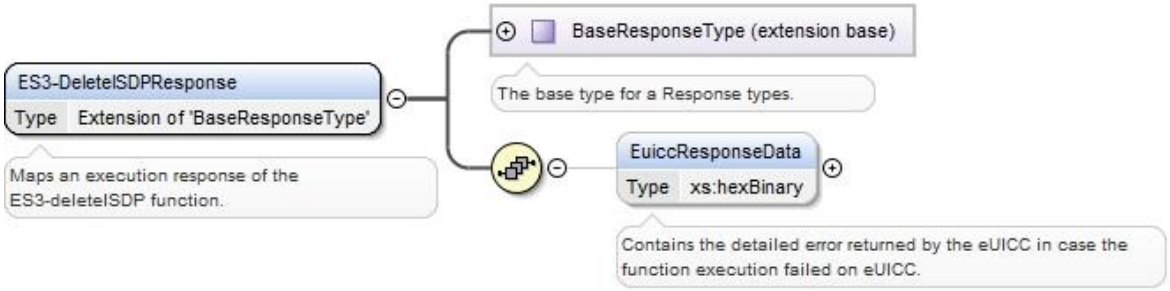


Figure 105: <rps:ES3-DeleteISDP Response>

This response doesn't carry any additional output data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-DeleteISDPResponse".

A.6.11 The "ES3. UpdateConnectivityParameters" Function

The input data of the "**ES3.UpdateConnectivityParameters**" function defined in section 5.4.11 shall be mapped to the <rps:ES3- UpdateConnectivityParametersRequest> element described in the following figure:

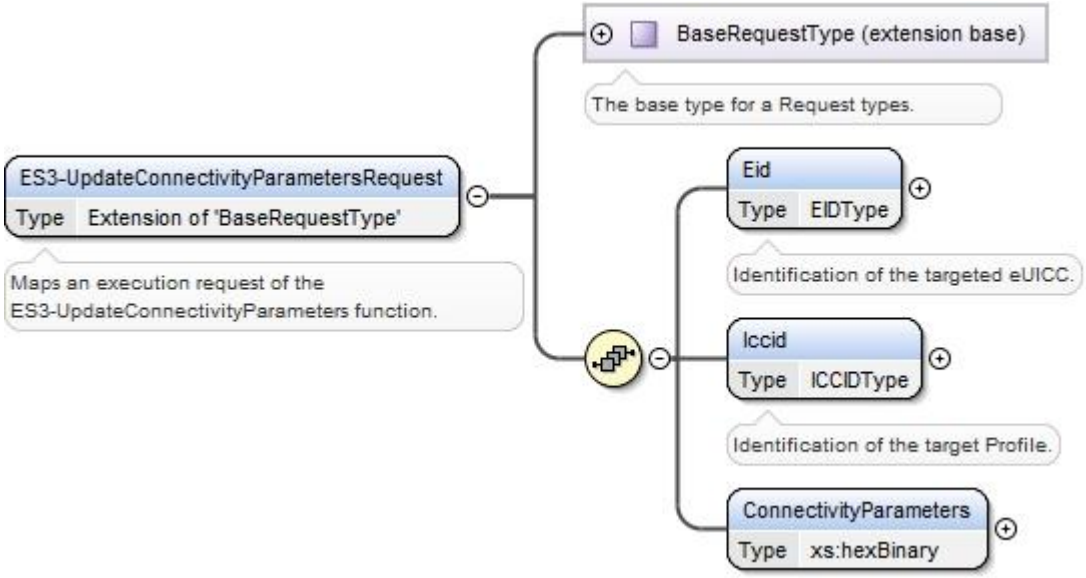


Figure 106: <rps:ES3- UpdateConnectivityParametersRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3- UpdateConnectivityParameters".

The output data of the "**ES3.UpdateConnectivityParameters**" function defined in section 5.4.11 shall be mapped to the <rps:ES3.UpdateConnectivityParametersResponse> element described in the following figure:

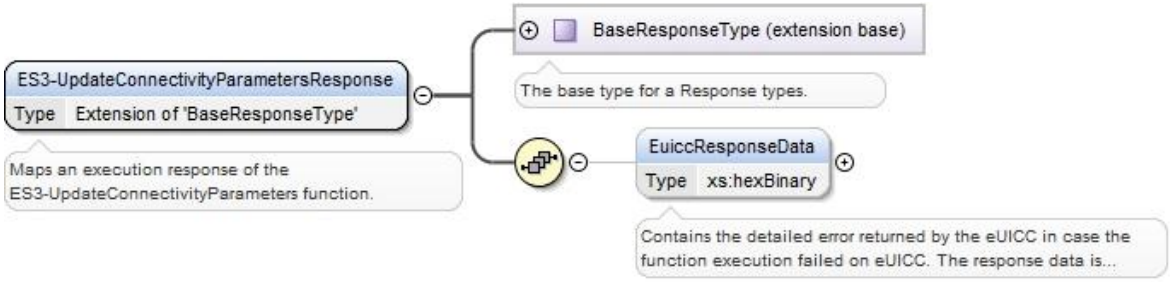


Figure 107: <rps:ES3- UpdateConnectivityParametersResponse >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3- UpdateConnectivityParameter Response".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.6.12 The “ES3.HandleProfileDisabledNotification” Function

The input data of the “**ES3.HandleProfileDisabledNotification**” function defined in section 5.4.12 shall be mapped to the <rps:ES3-HandleProfileDisabledNotification> element described in the following figure:

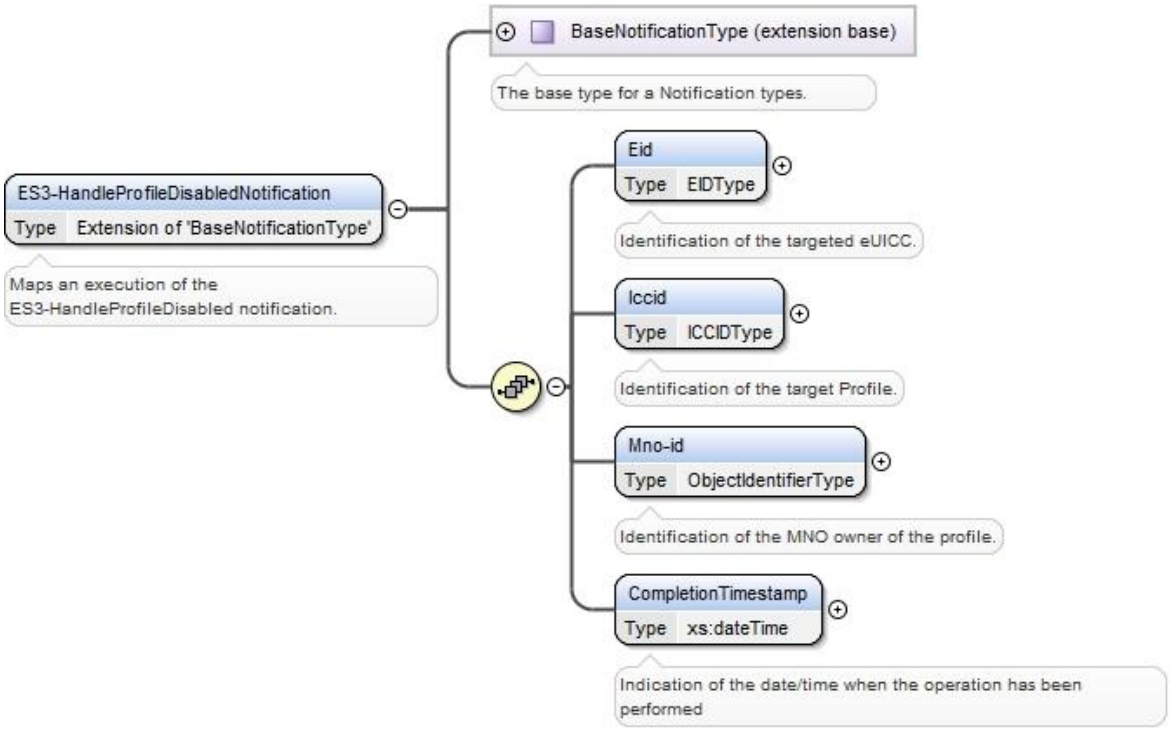


Figure 108: <rps:ES3-HandleProfileDisabledNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3-HandleProfileDisabledNotification".

A.6.13 The "ES3.HandleProfileEnabledNotification" Function

The input data of the "**ES3.HandleProfileEnabledNotification**" function defined in section 5.4.13 shall be mapped to the <rps:ES3-HandleProfileEnabledNotification> element described in the following figure:

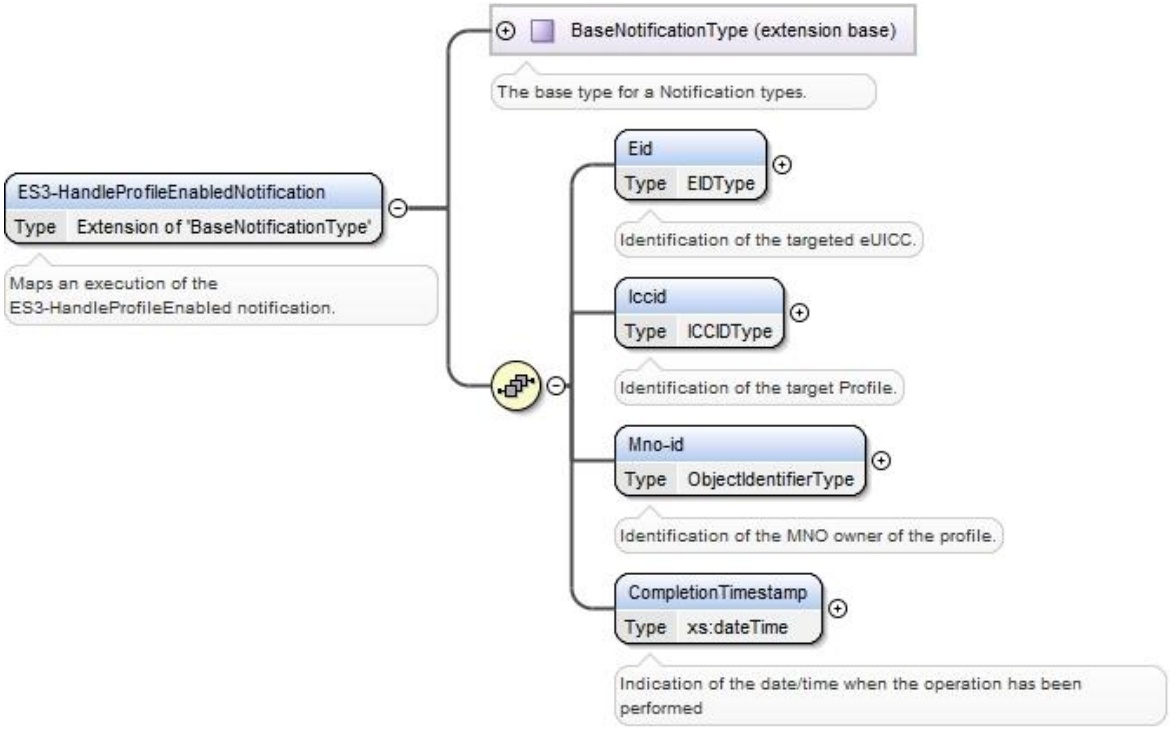


Figure 109: <rps:ES3-HandleProfileEnabledNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-HandleProfileEnabledNotification".

A.6.14 The "ES3.HandleSMSRChangeNotification " Function

The input data of the "**ES3.HandleSMSRChangeNotification**" function defined in section 5.4.14 shall be mapped to the <rps:ES3-HandleSMSRChangeNotificationRequest> element described in the following figure:

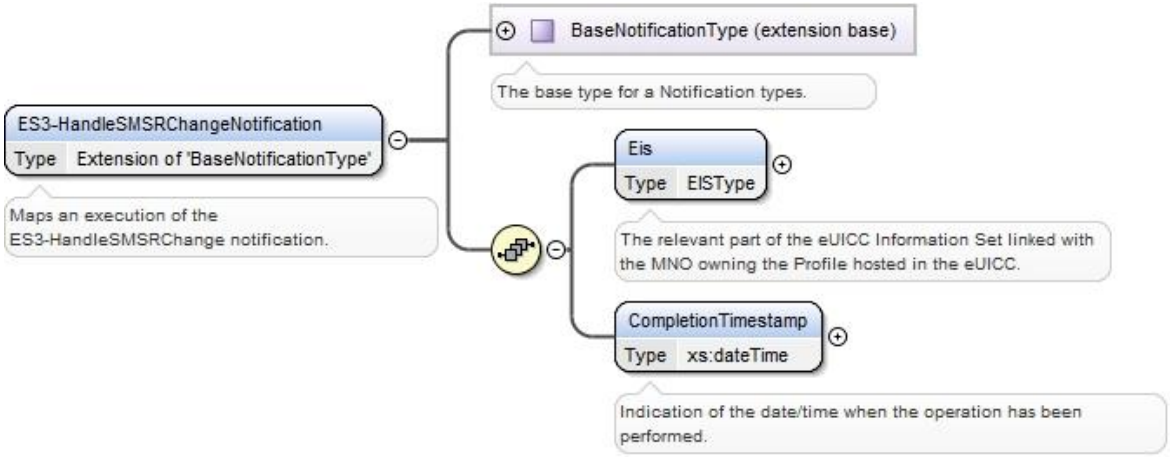


Figure 110: <rps:ES3-HandleSMSRChangeNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES3-HandleSMSRChangeNotification".

A.6.15 The “ES3. HandleProfileDeletedNotification” Function

The input data of the “**ES3.HandleProfileDeletedNotification**” function defined in section 5.4.15 shall be mapped to the <rps:ES3-HandleProfileDeletedNotification> element described in the following figure:

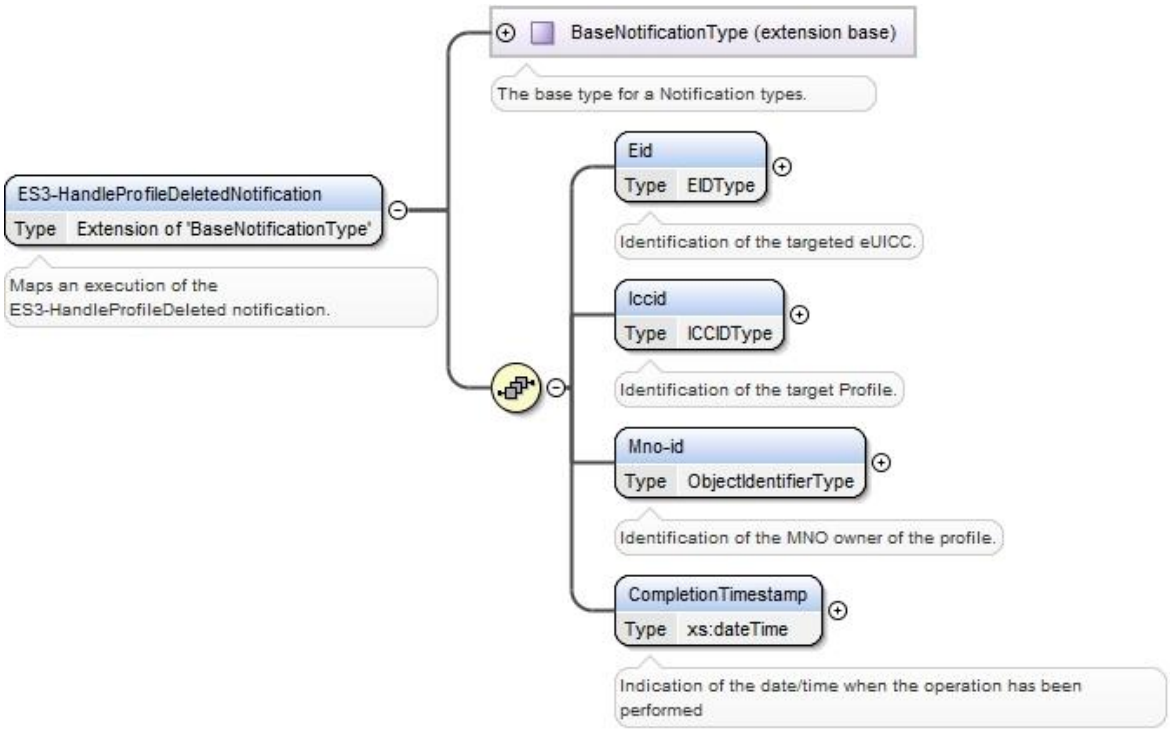


Figure 111: <rps:ES3-HandleProfileDeletedNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES3-HandleProfileDeletedNotification".

A.7 The ES4 Interface Functions

A.7.1 The “ES4.GetEIS” Function

The input data of the “**ES4.GetEIS**” function defined in section 5.5.1 shall be mapped to the <rps:ES4-GetEISRequest> element described in the following figure:

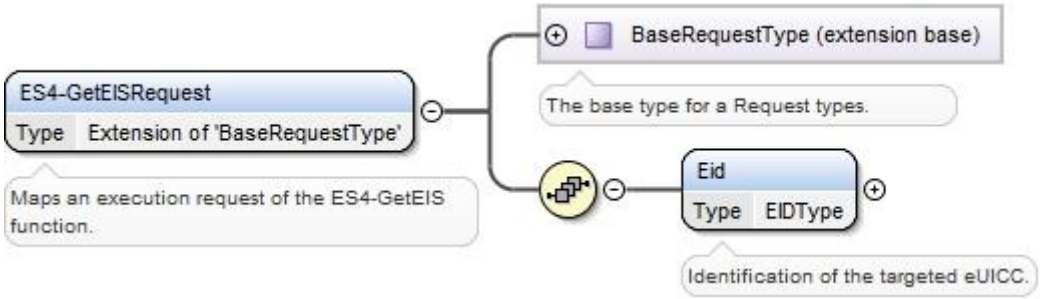


Figure 112: <rps:ES4-GetEISRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-GetEISRequest".

The output data of the "**ES4.GetEIS**" function defined in section 5.5.1 shall be mapped to the <rps:ES4-GetEISResponse> element described in the following figure:

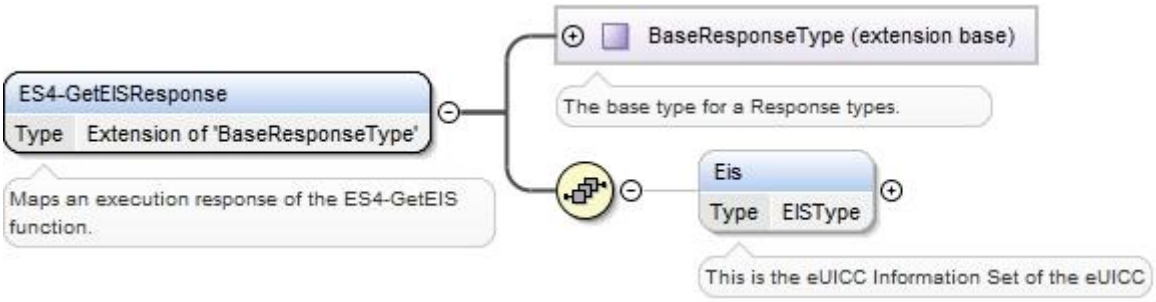


Figure 113: <rps:ES4-GetEISResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-GetEISResponse".

In case of function execution success or success with warning, the returned <rps:Eis> shall be filled with EIS as described in Annex E.

For <rps:Profile> element, only Profiles relevant to requesting MNO should be listed.

In case of function execution failure, no EIS shall be returned.

A.7.2 The "ES4.UpdatePolicyRules" Function

The input data of the "**ES4.UpdatePolicyRules**" function defined in section 5.5.2 shall be mapped to the <rps:ES4-UpdatePolicyRulesRequest> element described in the following figure:

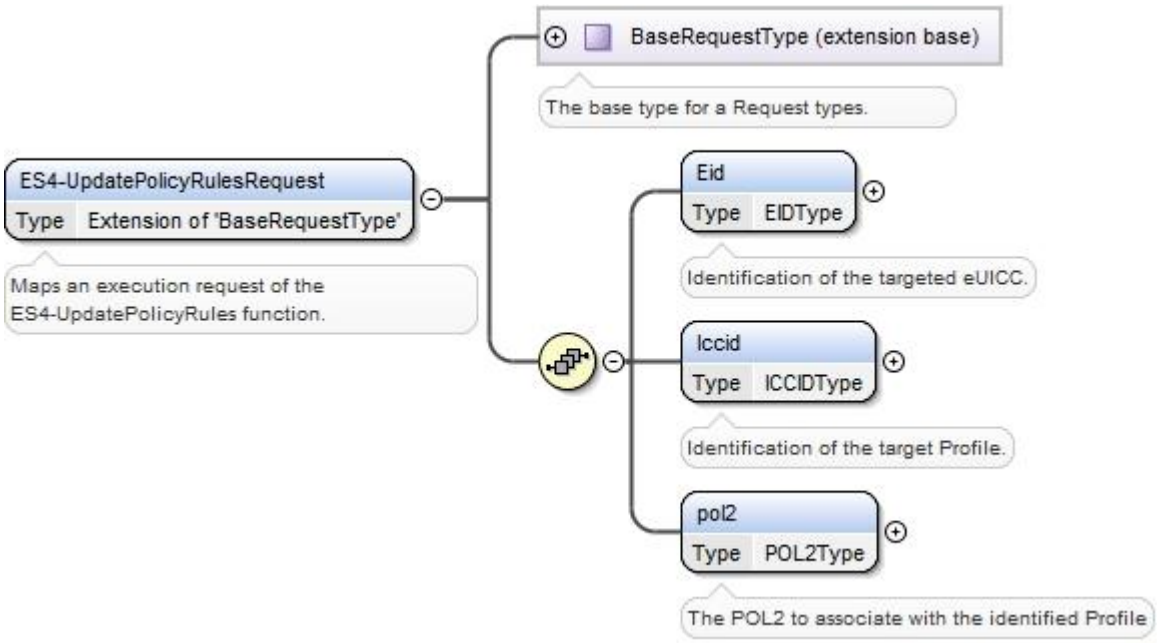


Figure 114: <rps:ES4-UpdatePolicyRulesRequest >

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-UpdatePolicyRules".

The output data of the "**ES4.UpdatePolicyRules**" function defined in section 5.5.2 shall be mapped to the <rps:ES4.UpdatePolicyRulesResponse> element described in the following figure:

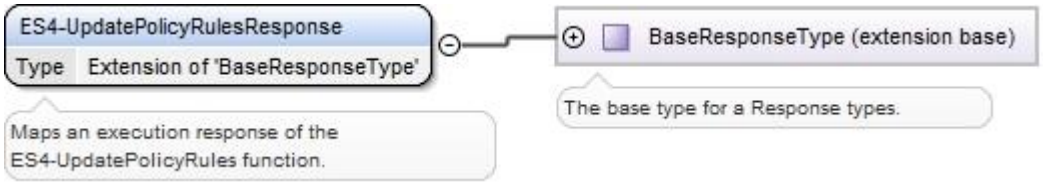


Figure 115: <rps:ES4-UpdatePolicyRulesResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-UpdatePolicyRulesResponse".

A.7.3 The "ES4. UpdateSubscriptionAddress" Function

The input data of the "**ES4.UpdateSubscriptionAddress**" function defined in section 5.5.3 shall be mapped to the <rps:ES4-UpdateSubscriptionAddressRequest> element described in the following figure:

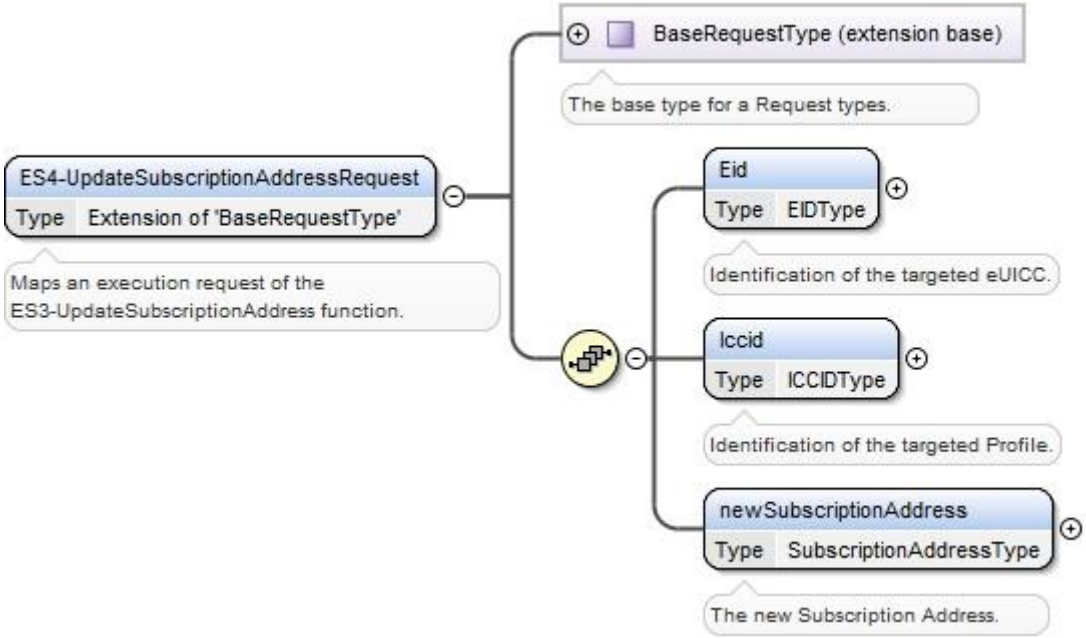


Figure 116: <rps:ES4-UpdateSubscriptionAddressRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-UpdateSubscriptionAddressRequest".

The output data of the "**ES4.UpdateSubscriptionAddress**" function defined in section 5.5.3 shall be mapped to the <rps:ES4-UpdateSubscriptionAddressResponse> element described in the following figure:

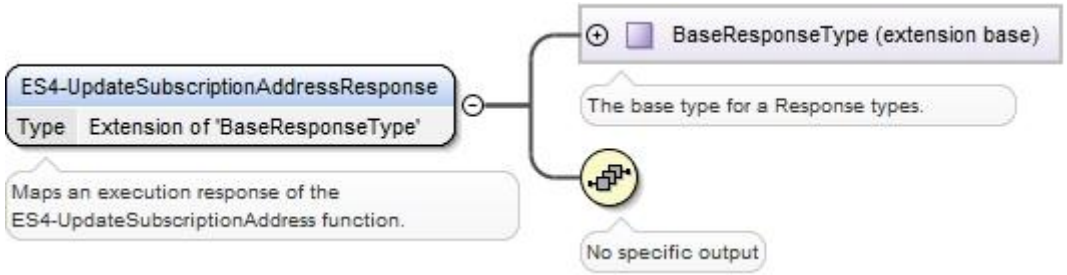


Figure 117: <rps:ES4-UpdateSubscriptionAddressResponse>

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-UpdateSubscriptionAddressResponse".

A.7.4 The "ES4.AuditEIS" Function

The input data of the "**ES4.AuditEIS**" function defined in section 5.5.4 shall be mapped to the <rps:ES4-AuditEISRequest> element described in the following figure:

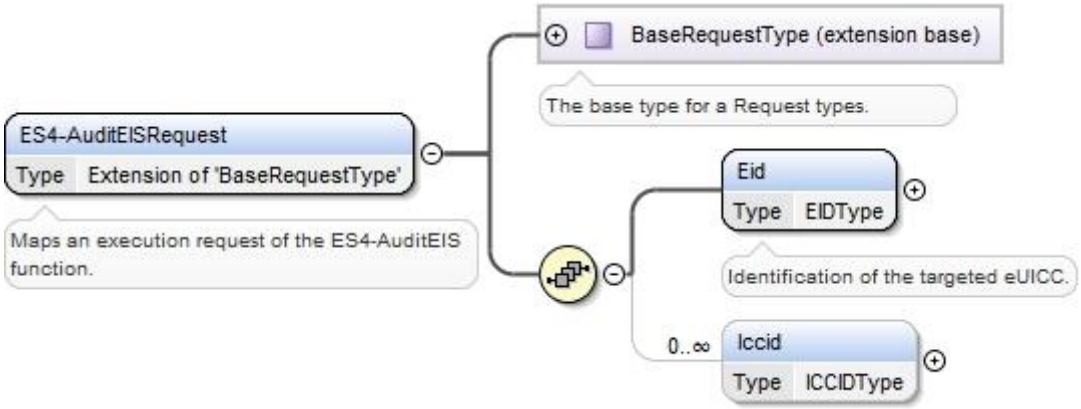


Figure 118: <rps:AuditEISRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-AuditEISRequest".

The output data of the "**ES4.AuditEIS**" function defined in section 5.5.4 shall be mapped to the <rps:ES4-AuditEISResponse> element described in the following figure:

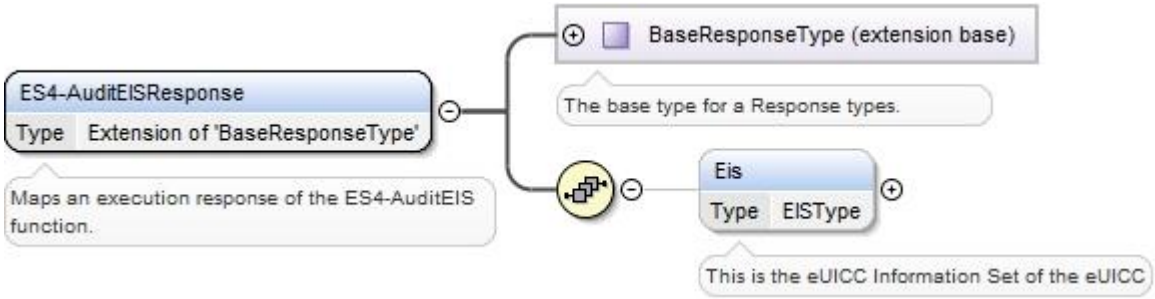


Figure 119: <rps:AuditEISResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES4-AuditEISResponse".

In case of function execution success or success with warning, the returned <rps:Eis> shall be filled accordingly to what is described in Annex E.

In case of function execution failure or expiration, no EIS shall be returned.

A.7.5 The “ES4.EnableProfile” Function

The input data of the “**ES4.EnableProfile**” function defined in section 5.5.5 shall be mapped to the <rps:ES4-EnableProfileRequest> element described in the following figure:

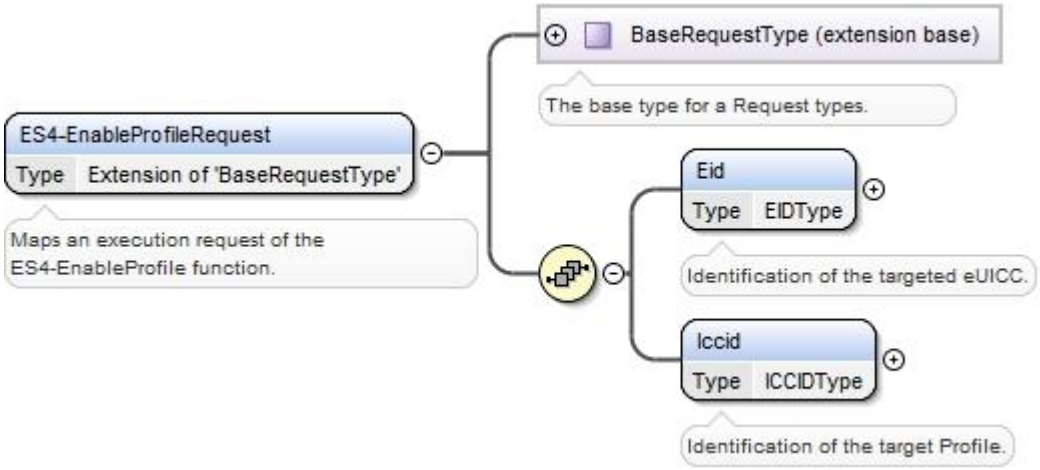


Figure 120: <rps:ES4-EnableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-EnableProfileRequest".

The output data of the “**ES4.EnableProfile**” function defined in section 5.5.5 shall be mapped to the <rps:ES4.EnableProfileResponse> element described in the following figure:

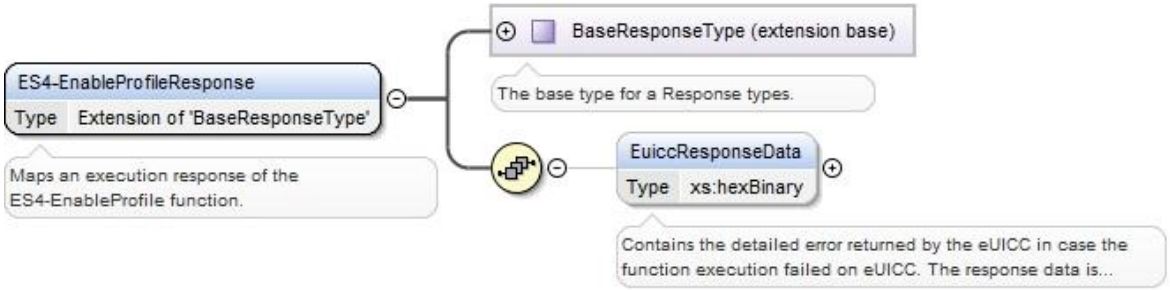


Figure 121: <rps:ES4-EnableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES4-EnableProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.7.6 The “ES4.DisableProfile” Function

The input data of the “**ES4.DisableProfile**” function defined in section 5.5.6 shall be mapped to the <rps:ES4-DisableProfileRequest> element described in the following figure:

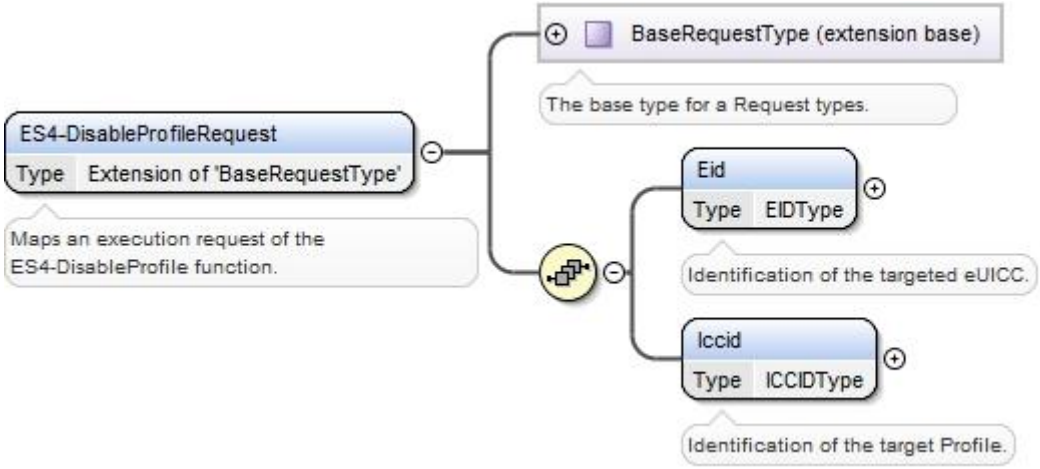


Figure 122: <rps:ES4-DisableProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-DisableProfileRequest".

The output data of the “**ES4.DisableProfile**” function defined in section 5.5.6 shall be mapped to the <rps:DisableProfileResponse> element described in the following figure:

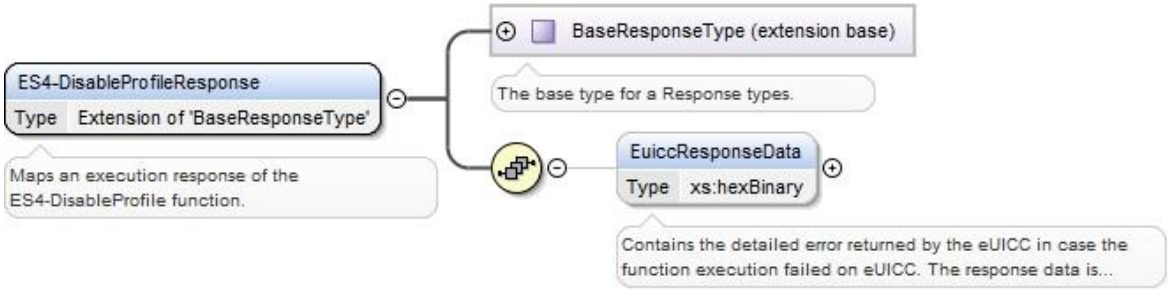


Figure 123: <rps:ES4-DisableProfileResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES4-DisableProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.7.7 The “ES4.DeleteProfile” Function

The input data of the “**ES4.DeleteProfile**” function defined in section 5.5.7 shall be mapped to the <rps:ES4-DeleteProfileRequest> element described in the following figure:

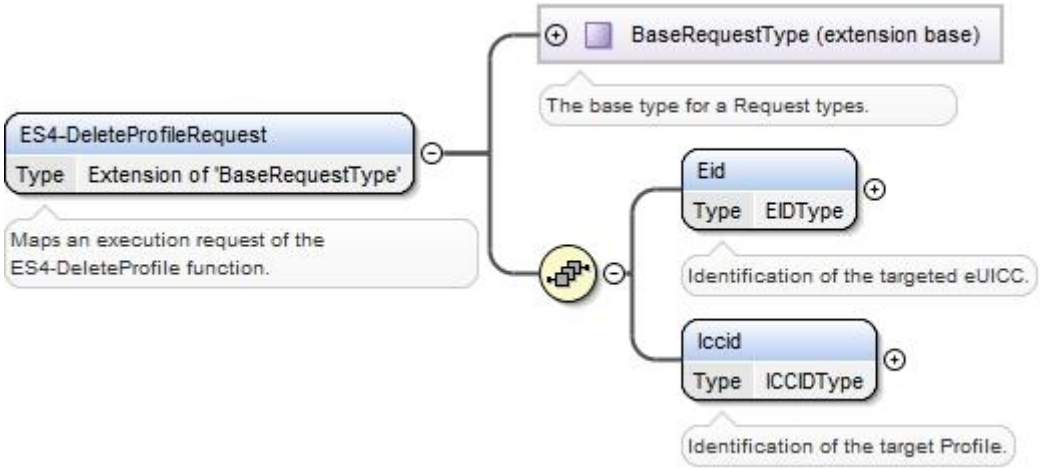


Figure 124: <rps:ES4-DeleteProfile Request>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-DeleteProfileRequest".

The output data of the "**ES4.DeleteProfile**" function defined in section 5.5.7 shall be mapped to the <rps:DeleteProfileResponse> element described in the following figure:

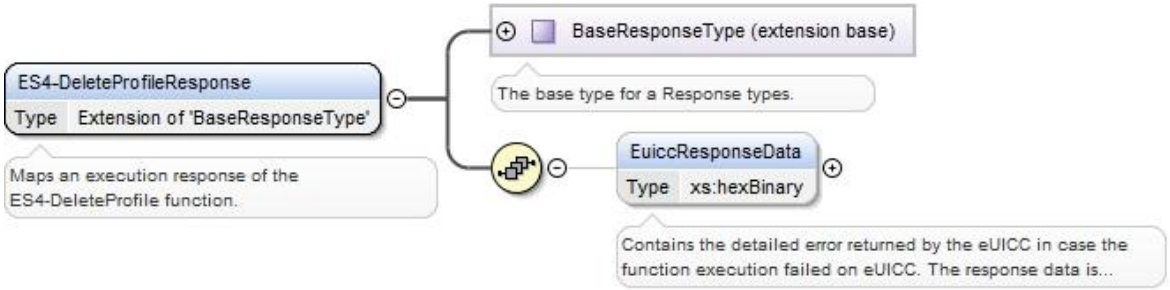


Figure 125: <rps:ES4-DeleteProfile Response>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-DeleteProfileResponse".

The response data may not be guaranteed to be provided, irrespective of the result of the function execution. If provided, the response data is in the <rps:euiccResponseData> element.

A.7.8 The "ES4.PrepareSMSRChange " Function

The input data of the "**ES4.PrepareSMSRChange**" function defined in section 5.5.8 shall be mapped to the <rps:ES4-PrepareSMSRChangeRequest> element described in the following figure:

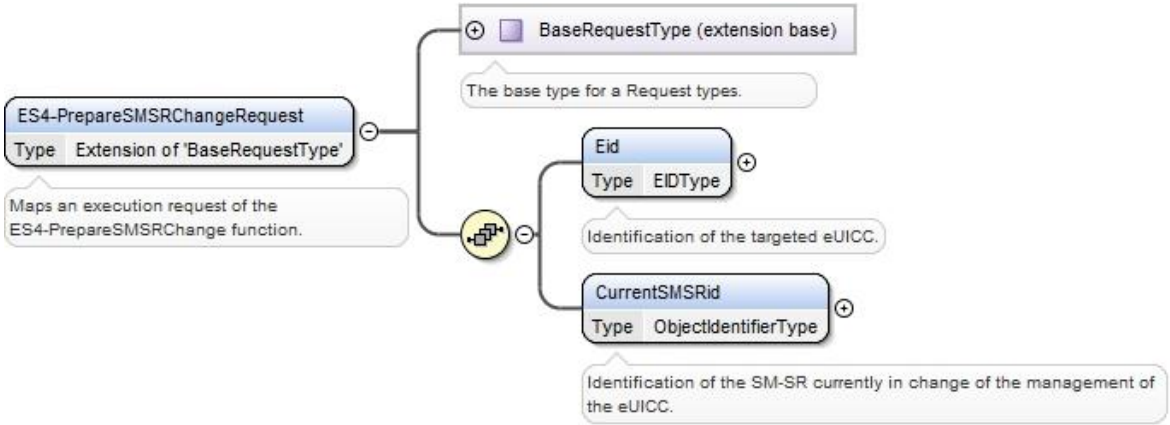


Figure 126: <rps:ES4-PrepareSMSRChangeRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-PrepareSMSRChangeRequest".

The output data of the "**ES4.PrepareSMSRChange**" function defined in section 5.5.8 shall be mapped to the <rps:ES4-PrepareSMSRResponse> element described in the following figure:

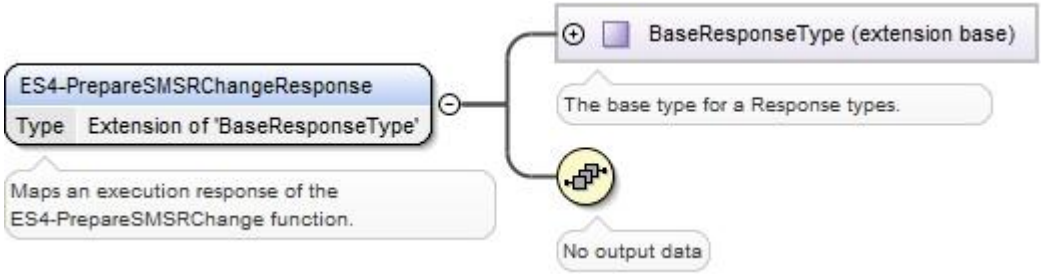


Figure 127: <rps:ES4-PrepareSMSRChangeResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-PrepareSMSRChangeResponse".

A.7.9 The "ES4.SMSRChange" Function

The input data of the "**ES4.SMSRChange**" function defined in section 5.5.9 shall be mapped to the <rps:ES4-SMSRChangeRequest> element described in the following figure:

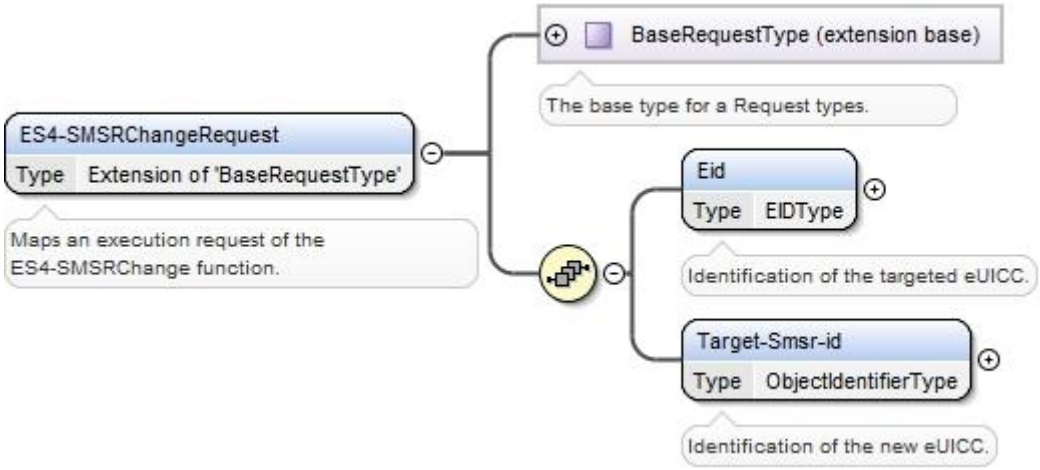


Figure 128: <rps:ES4-SMSRChangeRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-SMSRChangeRequest".

The output data of the "**ES4.SMSRChange**" function defined in section 5.5.9 shall be mapped to the <rps:ES4-SMSRResponse> element described in the following figure:

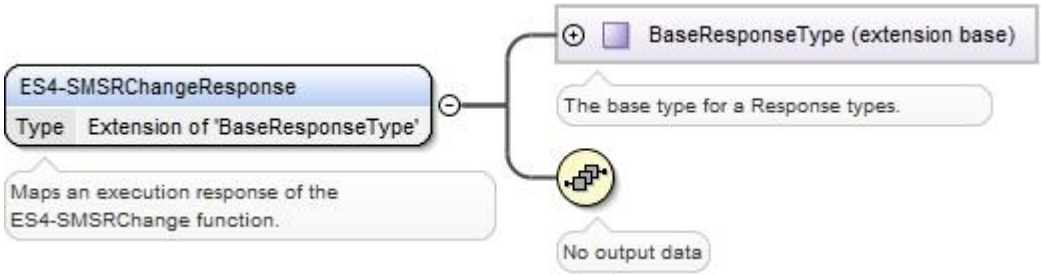


Figure 129: <rps:ES4-SMSRChangeResponse>

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-SMSRChangeResponse".

A.7.10 The "ES4.HandleProfileDisabledNotification" Function

The input data of the "**ES4.HandleProfileDisabledNotification**" function defined in section 5.5.10 shall be mapped to the <rps:ES4-HandleProfileDisabledNotification> element described in the following figure:

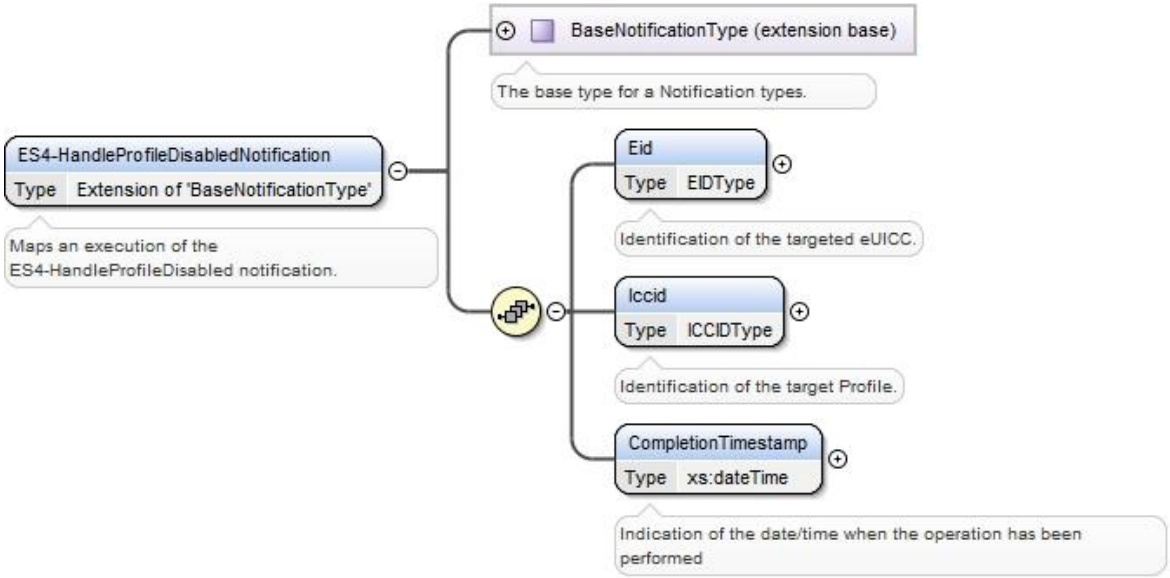


Figure 130: <rps:ES4-HandleProfileDisabledNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES4-HandleProfileDisabledNotification".

A.7.11 The “ES4.HandleProfileEnabledNotification” Function

The input data of the “**ES4.HandleProfileEnabledNotification**” function defined in section 5.5.11 shall be mapped to the <rps:ES4-HandleProfileEnabledNotification> element described in the following figure:

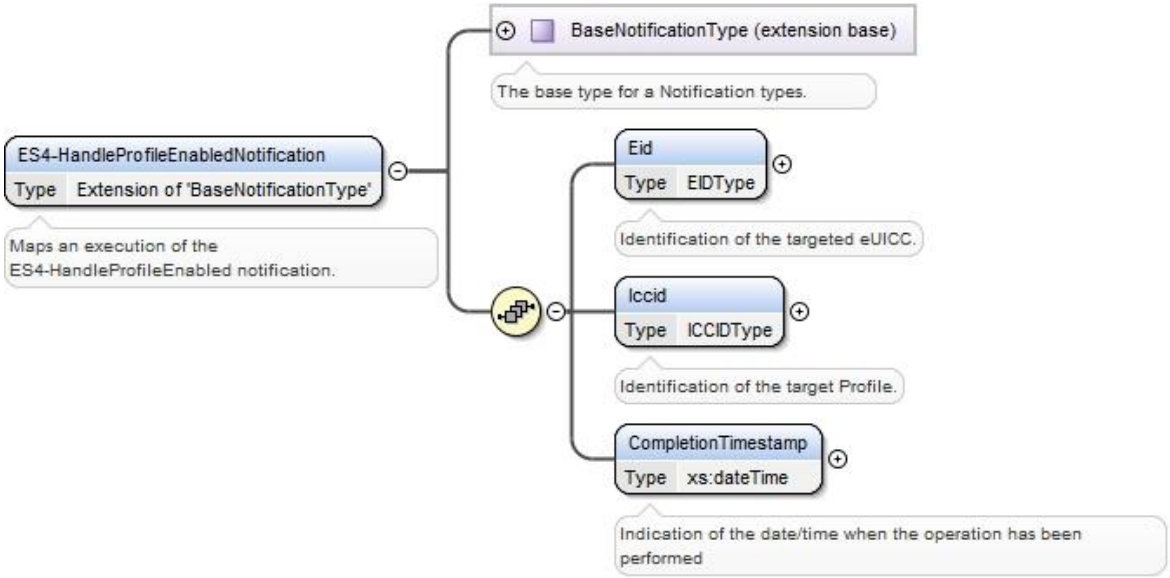


Figure 131: <rps:ES4-HandleProfileEnabledNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to “ES4-HandleProfileEnabledNotification".

A.7.12 The "ES4.HandleSMSRChangedNotification" Function

The input data of the "**ES4.HandleSMSRChangeNotification**" function defined in section 5.5.12 shall be mapped to the <rps:ES4-HandleSMSRChangeNotificationRequest> element described in the following figure:

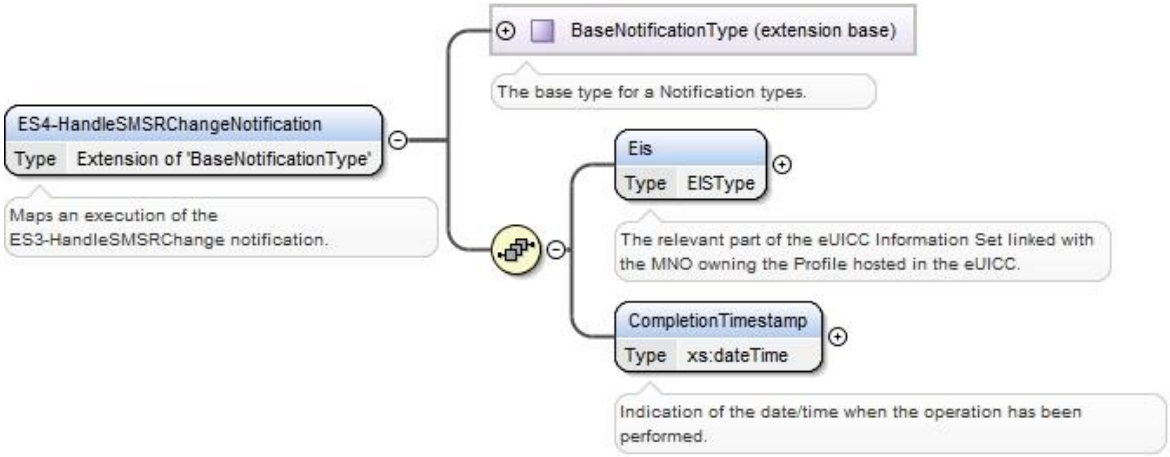


Figure 132: <rps:ES4-HandleSMSRChangeNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-HandleSMSRChangeNotification".

A.7.13 The "ES4. HandleProfileDeletedNotification" Function

The input data of the "**ES4.HandleProfileDeletedNotification**" function defined in section 5.5.13 shall be mapped to the <rps:ES4-HandleProfileDeletedNotification> element described in the following figure:

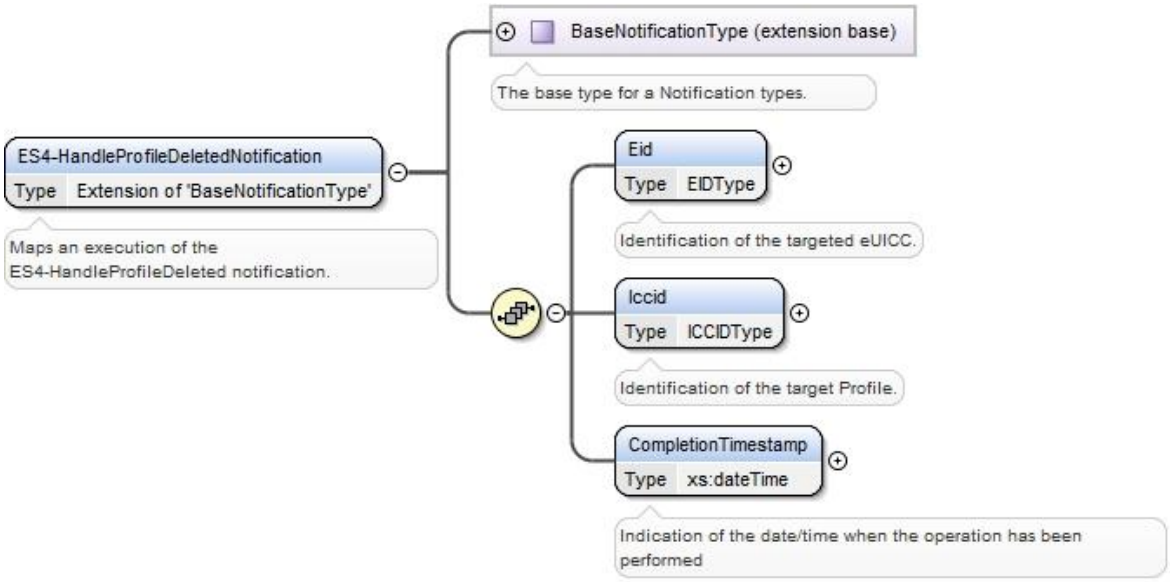


Figure 133: <rps:ES4-HandleProfileDeletedNotification>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES4-HandleProfileDeletedNotification".

A.8 The ES7 Interface Functions

A.8.1 The "ES7.CreateAdditionalKeySet" Function

The input data of the "**ES7.CreateAdditionalKeySet**" function defined in section 5.6.1 shall be mapped to the <rps:ES7-CreateAdditionalKeySet> element described in the following figure:

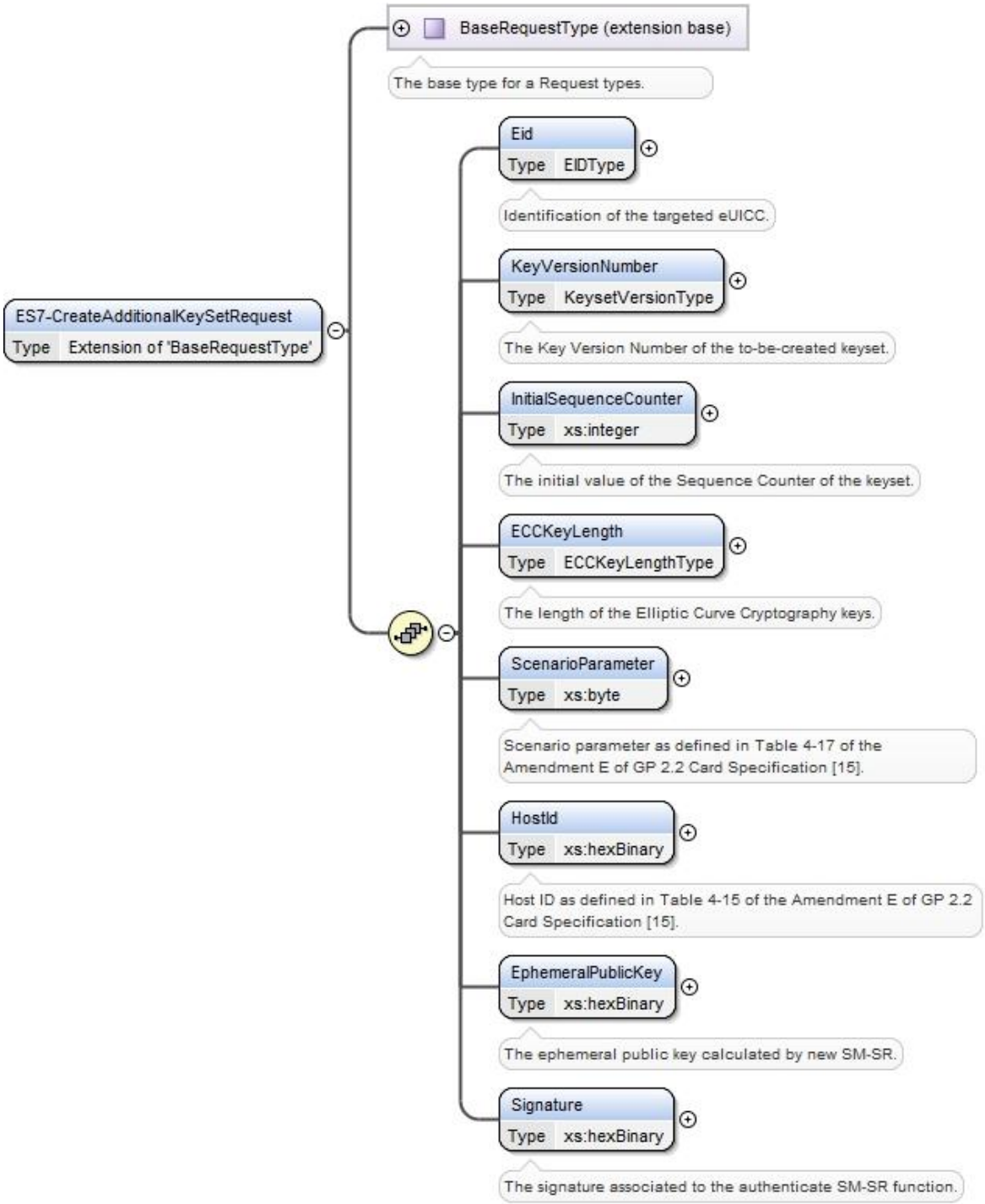


Figure 134: <rps:ES7-CreateAdditionalKeySetRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES7-CreateAdditionalKeySetRequest".

The output data of the "**ES7.CreateAdditionalKeySet**" function defined in section 5.6.1 shall be mapped to the <rps:ES7-CreateAdditionalKeySetEUICCResponse> element described in the following figure:

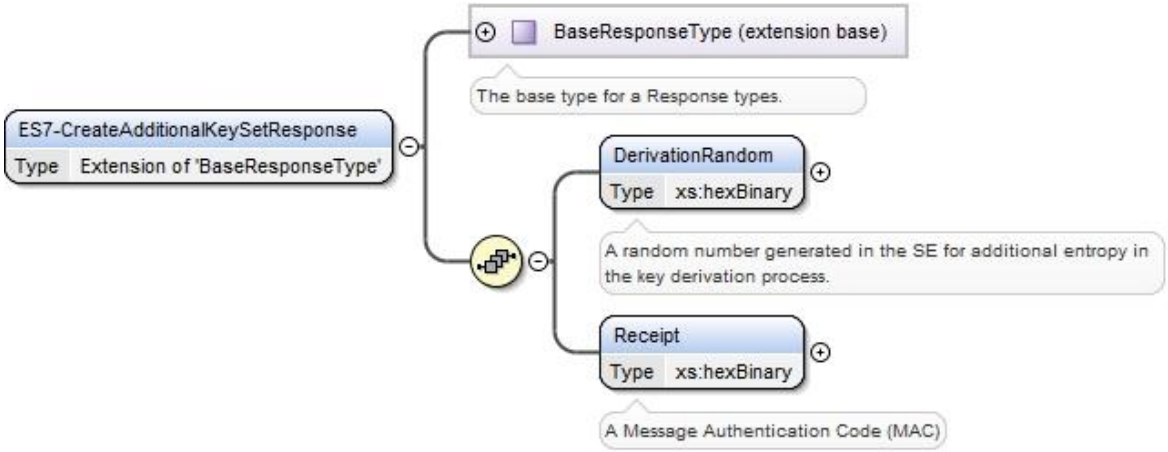


Figure 135: <rps:ES7-CreateAdditionalKeySetResponse>

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to"ES7-CreateAdditionalKeySetResponse"

A.8.2 The "ES7.HandoverEUICC" Function

The input data of the "**ES7.HandoverEUICC**" function defined in section 5.6.2 shall be mapped to the <rps:ES7-HandoverEUICCRequest> element described in the following figure:

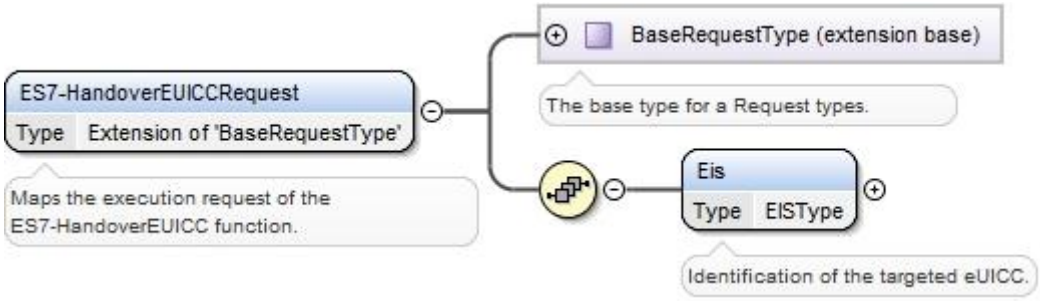


Figure 136: <rps:ES7-HandoverEUICCRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES7-HandoverEUICCRequest".

The output data of the "**ES7.HandoverEUICC**" function defined in section 5.6.2 shall be mapped to the <rps:ES7-HandoverEUICCResponse> element described in the following figure:

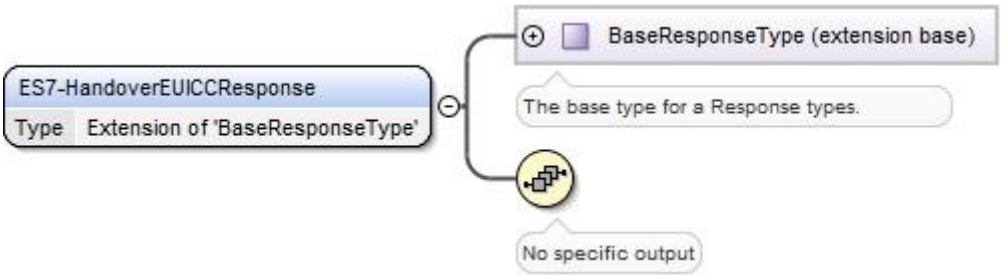


Figure 137: <rps:ES7-HandoverEUICCResponse>

This response doesn't carry any additional data

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES7-HandoverEUICCResponse".

A.8.3 The "ES7.AuthenticateSMSR" Function

The input data of the "**ES7.AuthenticateSMSR**" function defined in section 5.6.3 shall be mapped to the <rps:ES7-AuthenticateSMSRRequest> element described in the following figure:

Figure 138: <rps:ES7-AuthenticateSMSRRequest>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES7-AuthenticateSMSRRequest".

The output data of the "**ES7.AuthenticateSMSR**" function defined in section 5.6.3 shall be mapped to the <rps:ES7-AuthenticateSMSRResponse> element described in the following figure:

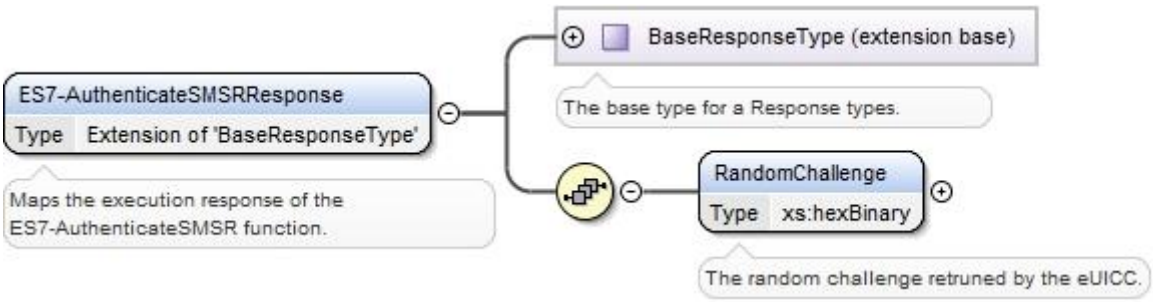


Figure 139: <rps:ES7-AuthenticateSMSRResponse>

The value of the <rps:RPSHeader>.<rps:MessageType> associated to this element shall be set to "ES7-AuthenticateSMSRResponse".

Annex B Binding to SOA Environment (Normative)

This section provides the binding of the messages defined in Annex A into a SOA infrastructure.

Web Services technology, following the OASIS and W3C WS-* standard, is the SOA environment recommended for the deployment of the off-card entities interfaces specified in this document. This technology provides interoperability and loose coupling between the interface provider and the interface consumer, also named respectively as "message receiver" and "message sender", “or “function provider” and “function requester”.

However this specification does not prevent from using another type of technology if it is suitable for a specific deployment. For sure, it implies that both message sender and message receiver uses the same technology and security around matches the level of expectation expressed in this document.

Nevertheless, in case Web Services is used, this section is normative and implementation shall comply with the requirements provided in this section.

B.1 General Recommendations

Systems are now highly multi-threaded. It is consequently possible for a function caller to perform massive parallel processing, and thus to call several Web Services in parallel. However, to avoid implementation and integration issues, this specification mandates that Function requester shall not perform parallel Web Services calls when they are targeting the same eUICC.

Web Services related to the same eUICC shall be serialised by the Function requester. For example to avoid key establishment to happen before ISD-P is created. Procedures described in section 3 shall be strictly followed regarding the sequence call.

If several Web Service calls are received by the Function provider for the same eUICC, then the Function provider could either:

- Return the following exception: 'Function for the same eUICC is already in process'.
- Or accept the new function execution request, and queue it to be executed after the already accepted function execution requests for this eUICC. This can only be applicable to asynchronous request (see B.2.3.3).

B.2 SOAP Binding

This section provides normative rules defining how to map the GSMA Embedded UICC Remote Provisioning messages (called RPS messages in the rest of section) defined in Annex A to a Web Services implementation, the rules being conditioned by Message Exchange Patterns (MEP), see B.2.3).

This specification mandates usage of SOAP v1.2 as the minimal version and specified in [40].

This section makes use of the following namespaces:

- wsa: the namespace for WS-Addressing message elements as defined in [41]
- wsmc: the namespace for WS-MakeConnection elements as defined in [43]

B.2.1 Message Binding

A RPS message consists of a body and a header (see A.2). This concept maps very well to the concept of SOAP messages that also contains a header and a body.

The binding of the messages defined in Annex A to SOAP shall follow the rules defined in this section.

- SOAP Header
  - The information contained in the RPSHeader of the message shall be transferred into the SOAP header. See also B.2.2.1
- SOAP Body
  - Only the element contained in the RPSBody structure shall be sent into the SOAP Body. It means that:
    The RPSMessage envelope shall not be sent.
    The full RPSHeader structure shall not be sent.
    The RPSBody envelope shall not be sent
  - The SOAP body shall contain the rps:MessageVersion attribute filled with the value of the <rps:RPSMessage>/<rps:MessageVersion> attribute.

As a consequence one RPS message corresponds to one SOAP message, and it is impossible to send several RPS messages in a single SOAP message.

Note that all information of the RPS message is bind to the SOAP message, so no information is lost during the binding.

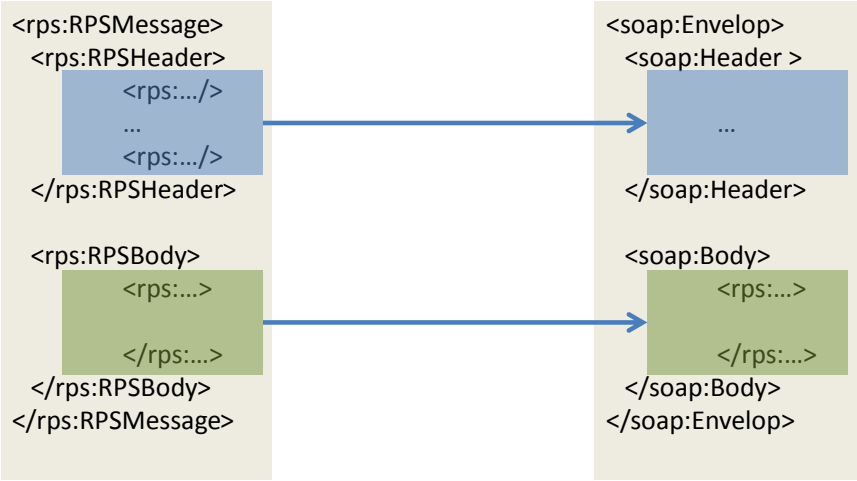


Figure 140: RPS Message Binding

B.2.1.1 RPS Header Binding to WS-Addressing Elements

This section describes the binding of RPS header into WS-Addressing properties. WS-Addressing properties are described in further detail in [41] and [42]. Only the elements that are used throughout this section are detailed here.

- `/wsa:From`

This OPTIONAL element (of type `wsa:EndpointReferenceType`) provides the value for the [source endpoint] property.

In the context of this specification this element is MANDATORY except in the synchronous response and defines the function requester. It shall be filled with:

- The sender URI. This value is not mapped from any value of the RPS Header, but it should be representative of the sender entity.
- A mandatory query parameter “EntityId” containing the `<rps.SenderEntity>/<rps:EntityId>` value
- An optional query parameter “EntityName” containing the `<rps.SenderEntity>/<rps:EntityName>` value
- An optional query parameter “UserName” containing the `<rps.SenderName>`

Example:

The following content:

```
<rsp:SenderEntity>
      <      rsp:EntityId>1.3.6.1.4.1.11111</rsp:EntityId><!--example value -->
      <rsp:EntityName>ACompany</rsp:EntityName>
</rsp:SenderEntity>
<rsp:SenderName>aSenderAccountId</rsp:SenderName>
```

Would be mapped into:

```
<wsa:From">

<wsa:Address>http://ACompany.com/RPS?EntityId=1.3.6.1.4.1.11111?EntityName = ACompany? UserName= aSenderAccountId</wsa:Address>
      </wsa:From>
```

- `/wsa:To`

This REQUIRED element (of type `xs:anyURI`) provides the value for the [destination] property.

In the context of this specification this element is MANDATORY and defines the function provider. It shall be filled with:

- The URL of the web service endpoint to which the message is sent. This value is not mapped from any value of the RPS Header, but it should be representative of the receiving entity.

- An optional query parameter “EntityId” containing the `<rps.ReceiverEntity>/<rps:EntityId>` value

Example:

The following content:

```
<rps:ReceiverEntity>
      <rps:EntityId>1.3.6.1.4.1.22222</rps:EntityId><!--      example value -->
</rps:ReceiverEntity>
```

Would be mapped into:

```
<wsa:To>http://ACompany.com/SMDP/ES2Services?EntityId=1.3.6.1.4.1.22222</wsa:To>
```

- `/wsa:ReplyTo`

This OPTIONAL element (of type `wsa:EndpointReferenceType`) provides the value for the [reply endpoint] property. If this element is NOT present, then the value of the [address] property of the [reply endpoint] EPR is "http://www.w3.org/2005/08/addressing/anonymous".

In the context of this specification this element is OPTIONAL. This element shall be present only when:

- MEP follows Asynchronous Request-Response with callback and
- When Message sender wants the response to be sent to a specific endpoint
- If missing, the response shall be sent to (in the preferred order):
- a well-known endpoint mutually agreed between message sender and message receiver
- or to the message originating endpoint.

If present, the /wsa:ReplyTo shall be filled with:

- The value set in `<rps:ResponseEndpoint>`
- An optional query parameter “EntityId” containing the `<rps.ReceiverEntity>/<rps:EntityId>` value

Example:

The following content:

```
<rps:ResponseEndpoint>http://ACompagny.com/SMDP/ES3Services</rps:ResponseEndpoint>

<rps:ReceiverEntity>
      <rps:EntityId>1.3.6.1.4.1.33333</rps:EntityId><!--      example value -->
</rps:ReceiverEntity>
```

Would be mapped into:

```
<wsa:ReplyTo>
```

```
<wsa:Address>http://ACompany.com/SMDP/ES3Services?EntityId=1.3.6.1.4.1.3333
3</wsa:Address> </wsa:ReplyTo>
```

- `/wsa:MessageID`

This OPTIONAL element (whose content is of type `xs:anyURI`) conveys the [message id] property.

In the context of this specification this element is MANDATORY whatever the MEP. This element shall be filled with:

- The value set in `<rps:MessageId>`.
- An optional query parameter "TransactionID" containing the `<rps:TransactionId>` value. This query parameter shall be present only if `<rps:TransactionId>` is present.
- An optional query parameter "ContextID" containing the `<rps:ContextId>` value. If this optional query parameter is present, it shall be included in any new request generated by the function provider entity for another functional provider entity. This identifier may be used to provide end-to-end logging management between the different web services.
- A mandatory query parameter "MessageDate" containing the `<rps:MessageDate>` value

Example:

The following content:

```
<rps:MessageId>//MySenderDomain/123</rps:MessageId><rps:TransactionId
>MyTansactionID1</rps:TransactionId>
<rps:ContextId>MyContextID1</rps:ContextId><rps:MessageDate>2013-04-
18T09:45:00Z</rps:MessageDate>
```

Would be mapped into:

```
<wsa:MessageID>//MySenderDomain/123?TransactionId=MyTansactionID1?Con
textId=MyContextID1?MessageDate=2013-04-18T09:45:00Z</wsa:MessageID>
```

- `/wsa:Action`

This REQUIRED element (whose content is of type `xs:anyURI`) conveys the value of the [action] property.

In the context of this specification this element is MANDATORY, and the format of this element shall be:

[target namespace] [delimiter][interface name] [delimiter][function group][delimiter][operation name][direction token]

Where:

- [target namespace]: 'http://gsma.com'
- [interface name]: One of the following label 'ES1', 'ES2' 'ES3', 'ES4' 'ES7',
- [function group]:

    - For Synchronous Request-Response MEP, for Notification MEP, and for Asynchronous with Polling MEP, the [function group] value shall be filled with the name of the functions group (see Table 95 and Table 96). Possible values are:
      eUICCManagement
      ProfileManagement
      PlatformManagement
    - For Asynchronous with callback MEP, the [function group] value SHALL be filled with the name of the functions group appended with the "CallBack" string. Possible values are:
      ProfileManagementCallBack
      PlatformManagementCallBack
      eUICCManagementCallBack
- [Operation name]: the name of the function as contained in the `/rps:RPSHeader/rps:MessageType` element
- [direction token] = Follows OASIS WS-* specifications, i.e.:
    - For Synchronous Request-Response MEP: the [direction token] is already part of the [Operation Name] as the "Request" string for the request, and as the "Response" string for the response. So no additional qualifier shall be added.
    - For Notification (One-Way MEP): no direction Token (empty string) needs to be added after the [Operation name]
    - For Asynchronous with callback MEP or Asynchronous with Polling: as these MEP are indeed mapped to two one-way service calls, then there is no need to have a direction token, neither for the request, nor for the asynchronous response (empty strings). The ‘Resquest’ and ‘Response’ qualifier shall be removed from the [Operation name].
- [delimiter]: “/”

Examples:

- For the ES2 ‘GetEIS’ part of the ‘Profile Management’ function group, the relevant /wsa:Action shall be (assumed to be called as a Synchronous Request-Response MEP):
    - For the request:

```
<wsa:Action>http://gsma.com/ES2/ProfileManagement/ES2-GetEISRequest</wsa:Action>
```

    - For the response:

```
<wsa:Action>http://gsma.com/ES2/ProfileManagement/ES2-GetEISResponse</wsa:Action>
```

- For the ES3 ‘HandleProfileDisabledNotification’ part of the ‘Platform Management’ function group, the relevant /wsa:Action shall be for the request (no response expected):

```
<wsa:Action>http://gsma.com/ES3/PlatformManagement/ES3-HandleProfileDisabledNotification</wsa:Action>
```

- For the ES3 ‘EnableProfile’ part of the ‘Platform Management’ function group, the relevant /wsa:Action shall be (assumed to be called as a Asynchronous Request-Response with callback MEP):

- For the request:

```
<wsa:Action>http://gsma.com/ES3/PlatformManagement/ES3-EnableProfile</wsa:Action>
```

- For the response:

```
<wsa:Action>http://gsma.com/ES3/PlatformManagementCallBack/ES3-EnableProfile</wsa:Action>
```

- For the ES3 'EnableProfile' part of the 'Platform Management' function group, the relevant /wsa:Action shall be (assumed to be called as a Asynchronous with Polling MEP):
  - For the request:

```
<wsa:Action>http://gsma.com/ES3/PlatformManagement/ES3-EnableProfile</wsa:Action>
```

  - For the response:

```
<wsa:Action>http://gsma.com/ES3/PlatformManagement/ES3-EnableProfile</wsa:Action>
```

- `/wsa:FaultTo`

This OPTIONAL element (of type wsa:EndpointReferenceType) provides the value for the [fault endpoint] property.

In the context of this specification this element SHALL NOT be used. Any fault shall be sent to (in the preferred order):

- The endpoint specified in the /wsa:ReplyTo, if present,
- Else, to a well know endpoint mutually agreed between message sender and message receiver
- Or to the message originating endpoint.

- /wsa:RelatesTo

This OPTIONAL (repeating) element information item contributes one abstract [relationship] property value, in the form of an (IRI, IRI) pair. The content of this element (of type `xs:anyURI`) conveys the [message id] of the related message.

In the context of this specification this element is MANDATORY if the message is a response. This element shall be filled with the value of the `<rps:RelatesTo>`

Example:

The following content:

```
<rps:RelatesTo>//MySenderDomain/123</rps:Relates>
```

Would be mapped into:

```
<wsa:RelatesTo>//MySenderDomain/123</wsa:RelatesTo>
```

B.2.1.2 Use of WS-MakeConnection

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). In this scenario, the sending party shall poll for a processed request using WS-MakeConnection [43]. This scenario is described in the Message Exchange Pattern: Asynchronous with Polling (Annex B-Section 2.3.3).

All the following elements are described in further detail in WS-MakeConnection [43], only the elements that are used throughout this document are detailed here.

To indicate to the Function provider that the Function requester is not addressable and will use Asynchronous with polling MEP (see B.2.3.3), the /wsa:ReplyTo header element shall indicate one of the two anonymous URL:

- The WS-Addressing anonymous URL 'http://www.w3.org/2005/08/addressing/anonymous'. This shall allow the function requester to poll for the first response message available for the function requester
- The WS-MakeConnection anonymous URL 'http://docs.oasis-open.org/ws-rx/wsmc/200702/anonymous?id=<value of <wsa:MessageId>'. This shall allow the Function requester to poll for the response for this specific message.

By using one of the two above anonymous /wsa:ReplyTo URL constructs, the Function provider knows that 'Asynchronous with Polling' mode is requested and shall answer with HTTP 202 (ACCEPT), see B.2.3.3.

To get a Function execution response, The Function provider shall send a new SOAP message with the /wsmc:MakeConnection element in the body; this new message establishes a contextualised back-channel for the transmission of the message response according to matching criteria (defined below).

In the context of this specification, the SOAP message allowing getting a function execution response message shall contain:

In the Header:

- `/wsa:Action` element with the specific value 'http://docs.oasis-open.org/ws-rx/wsmc/200702/MakeConnection'

In the body:

- `/wmc:MakeConnection` element with a sub element /`wsmc:Address` containing one of the anonymous URI defined here above and identifying the initiating endpoint contained in the /wsa:ReplyTo element of the original function execution request. Function provider shall not return message response in the HTTP response unless they have been addressed to this URI.
  - If the Function provider has not any response ready for the Function requester it shall answer with an empty response and HTTP 202 (ACCEPT)
  - If the Function provider has a response ready it shall return the response and use HTTP response code 200 (OK)

B.2.2 Security

To secure the messages being sent between Function requester and Function provider, one of the two following mechanisms shall be used:

11. 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.

B.2.2.1 Secure Channel Set-Up

The process of setting up secure channel is out of scope of this document. This process includes the exchange of the following information:

- Function requester and Function provider OIDs shall be registered and respective values have been communicated to each party
- Function requester and Function provider URL shall have been communicated to each party
- Function requester and Function provider shall agree on the MEP for response handling of asynchronous function: Asynchronous Request-Response with callback or Asynchronous with polling.
- Function requester and Function provider shall agree on the type of security mechanism used and respective credential:
  - WS-Security
    If UsernameToken Profile is used, the Username and Password shall be setup at receiving entities.
    If X509 Certificate Token Profile is used, the receiving entity shall trust the sending entity issued certificate.
  - Transport Level Security
    Function requester and Function provider party trust must have been established on a X509 certificate chain basis.

NOTE: Receiving entity and sending entity could either be the Function requester of the Function provider.

B.2.2.2 Identification/Authentication/Authorisation

Authentication of the sending party of a SOAP message shall rely on either the Transport layer security (using TLS certificate of the sending party) or the WS-Security [44]. In this case the SOAP message shall include specific WS-Security elements containing a security token, UserNameToken or X509Token as agreed during secure channel set-up (see 2.3.1).

Message receiver shall be able to process Web Service Security tokens as specified in the OASIS specification [44], specifically:

- UsernameToken Profile 1.1. as defined in [45]:

```
<S11:Envelope xmlns:S11="..." xmlns:wsse="...">
      <S11:Header>
          …
            <wsse:Security>
                  <wsse:UsernameToken>
                        <wsse:Username>ACompany</wsse:Username>
                        <wsse:Password>MyPassword</wsse:Password>
                  </wsse:UsernameToken>
            </wsse:Security>
    ….
```

- X509 Certificate Token Profile 1.1. as defined in [46], with '#X509v3' token type. The X509 certificate of the sender shall be included as a BinarySecurityToken.

```
<S11:Envelope xmlns:S11="...">
      <S11:Header>
            …
            <wsse:Security xmlns:wsse="..."  >
                  <wsse:BinarySecurityToken ValueType="…#X509v3"
                        EncodingType="…#Base64Binary">
                        MIIEZzCCA9CgAwIBAgIQEmtJZc0…
                  </wsse:BinarySecurityToken>
                  …
            </wsse:Security>
            …
```

B.2.2.3 Integrity

The integrity of the message shall exclusively rely on the transport level security (TLS).

B.2.2.4 Confidentiality

The confidentiality of the message shall exclusively rely on the transport level security (TLS).

B.2.3 Message Exchange Pattern (MEPs) – HTTPS Binding

B.2.3.1 MEP: Synchronous Request-Response

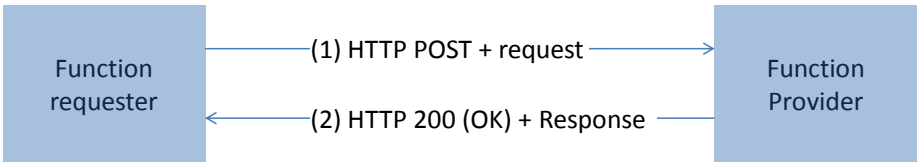


Figure 141: MEP: Synchronous Request-Response

(1) The SOAP header of the message sent in a HTTP POST from Function requester to Function provider shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

(2) The response to the message is on the HTTP(s) return channel with code 200 (OK) and the SOAP header shall contain:

`/wsa:From` (OPTIONAL)

`/wsa:To` (REQUIRED)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

`/wsa:RelatesTo` (Value of MessageId:value of the original message to which this is the response) (REQUIRED)

B.2.3.2 MEP: Asynchronous Request-Response With Callback

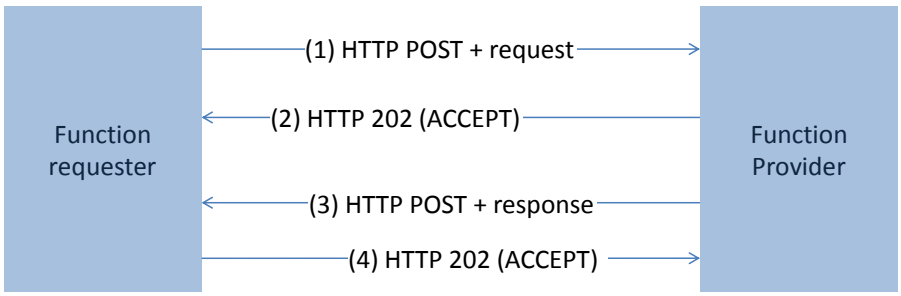


Figure 142: MEP: Asynchronous Request-Response With Callback

(1) The SOAP header of the message sent in a HTTP POST from Function requester to Function provider shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:ReplyTo` (OPTIONAL)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

(2) The Function requester shall be able to handle 202 (ACCEPT) HTTP response codes.

NOTE: In case the response is 200 (OK) steps (3) and (4) will be skipped if it is not a new session.

(3) The response to the message is sent in a HTTP POST from Function provider to Function requester, and the SOAP header shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

`/wsa:RelatesTo` (value of MessageId:value of the original message to which this is the response) (REQUIRED)

(4) Function requester shall reply with a HTTP 202 (ACCEPT).

B.2.3.3 MEP: Asynchronous With Polling

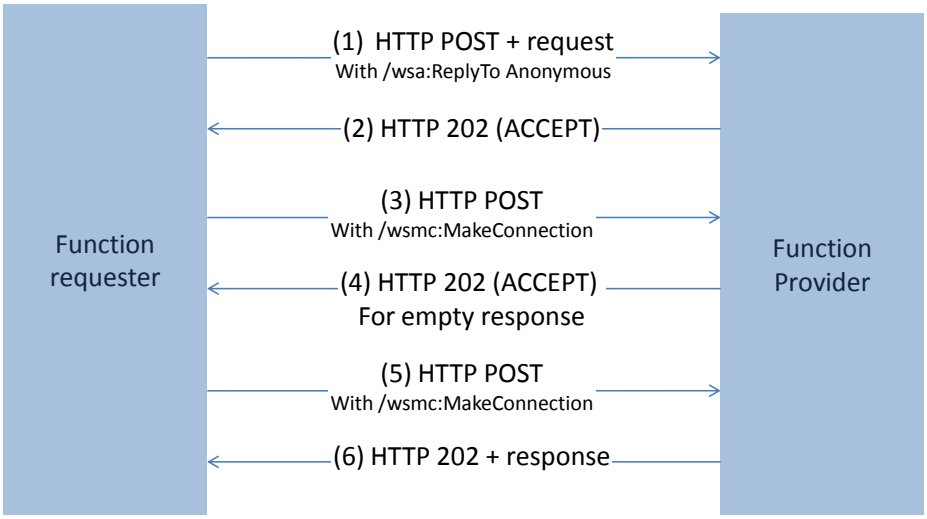


Figure 143: MEP: Asynchronous With Polling

(1) The SOAP header of the message sent in a HTTP POST from Function requester to Function provider shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:ReplyTo` (REQUIRED) containing one of the two possible anonymous URL (see Annex B-Section 2.1.2)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

(2) Function provider shall reply with a HTTP 202 (ACCEPT). (3 or 5) Function provider makes a WS-MakeConnection call as defined in Annex B-Section 2.1.2 with a header containing:

```
<wsa:Action>http://docs.oasis-open.org/ws-rx/wsmc/200702/MakeConnection<wsa:Action>
```

And a body containing:

```
<wsmc:MakeConnection ...>
    <wsmc:Address>AnonymousURL (same value as /wsa:ReplyTo above)</wsmc:Address>
</wsmc:MakeConnection>
```

(4 or 6) The response to the message is sent in a HTTP response from Function provider to Function requester, and the SOAP header shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

`/wsa:RelatesTo` (Value of MessageId:value of the original message to which this is the response) (REQUIRED)

B.2.3.4 MEP: Notification (One-Way)

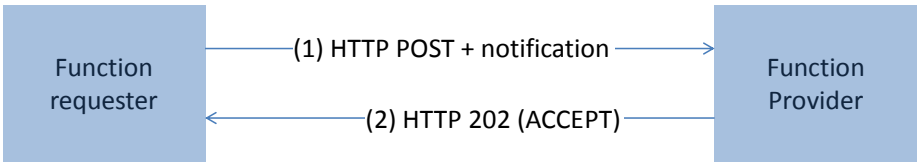


Figure 144: MEP: Synchronous Request-Response

(1) The SOAP header of the message sent in a HTTP POST from Function requester to Function provider shall contain:

`/wsa:From` (REQUIRED)

`/wsa:To` (REQUIRED)

`/wsa:MessageID` (REQUIRED)

`/wsa:Action` (REQUIRED)

(2) The response to the message is on the HTTP return channel with code 202 (ACCEPT) and with an empty body.

B.2.4 Binding Examples

B.2.4.1 Binding of a Message for ES4.EnableProfile Function Request

The xml hereunder illustrates an RPS message for requesting the execution of the **ES4.EnableProfile** function:

```
<?xml version="1.0" encoding="UTF-8"?>
<RPSMessage xmlns="http://namespaces.gsma.org/esim-messaging/1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
    MessageVersion="1.0.0">
    <RPSHeader>
        <SenderEntity>
            <EntityId>1.3.6.1.4.1.111111</EntityId><!-- Sample OID  -->
            <EntityName>ACompany</EntityName>
        </SenderEntity>
        <SenderName>aSenderAccountId</SenderName>
        <ReceiverEntity>
            <EntityId>1.3.6.1.4.1.222222</EntityId><!-- Sample OID -->
        </ReceiverEntity>

<ResponseEndpoint>http://ACompany.com/RPS/MyEndPoint</ResponseEndpoint>
        <TransactionId>MyTransID1</TransactionId>
        <MessageId>//MySenderDomain/123</MessageId>
        <MessageType>ES4-EnableProfileRequest</MessageType>
        <MessageDate>2013-04-18T09:30:47Z</MessageDate>
    </RPSHeader>
    <RPSBody>
        <ES4-EnableProfileRequest>
            <FunctionCallIdentifier>callId:1</FunctionCallIdentifier>
            <ValidityPeriod>3600</ValidityPeriod>
            <Eid>89001012012341234012345678901224</Eid>
            <ICCID>8933010000000000001</ICCID>
        </ES4-EnableProfileRequest>
    </RPSBody>
</RPSMessage>
```

In the case where:

- security is set with TLS, with mutual authentication based on certificate
- the MEP is : Asynchronous Request-Response with callback

This function execution request is bound to the following SOAP message:

```
<?xml version="1.0" encoding="UTF-8"?>
<s:Envelope               xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:rps="http://namespaces.gsma.org/esim-messaging/1">
    <s:Header>
        <wsa:From>

<wsa:Address>http://ACompany.com/RPS?EntityId=1.3.6.1.4.1.111111?EntityName
=ACompany?UserName=aSenderAccountID</wsa:Address>
        </wsa:From>

<wsa:To>http://AnotherCompany.com?EntityId=1.3.6.1.4.1.222222</wsa:To>

<wsa:MessageID>//MySenderDomain/123?TransactionId=MyTransID1?MessageDate=20
13-04-18T09:30:47Z</wsa:MessageID>
        <wsa:Action>http://gsma.com/ES4/ProfileManagement/ES4-
EnableProfile</wsa:Action>
        <wsa:ReplyTo>
            <wsa:Address>http://ACompany.com/RPS/MyEndPoint</wsa:Address>
        </wsa:ReplyTo>
    </s:Header>
    <s:Body rps:MessageVersion="1.0.0">
        <rps:ES4-EnableProfileRequest>

<rps:FunctionCallIdentifier>callID:1</rps:FunctionCallIdentifier>
            <rps:ValidityPeriod>3600</rps:ValidityPeriod>
            <rps:Eid>89001012012341234012345678901224</rps:Eid>
            <rps:ICCID>8933010000000000001</rps:ICCID>
        </rps:ES4-EnableProfileRequest>
    </s:Body>
</s:Envelope>
```

B.2.4.2 Binding of a Message for ES4.EnableProfile Function Response

The xml hereunder illustrates a possible message response for the **ES4.EnableProfile** function execution request illustrated in the example of the previous section 2.2.1:

```
<?xml version="1.0" encoding="UTF-8"?>
<RPSMessage xmlns="http://namespaces.gsma.org/esim-messaging/1"
    xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
```

```
    MessageVersion="1.0.0">


    <RPSHeader>
        <SenderEntity>
            <EntityId>1.3.6.1.4.1.222222</EntityId><!-- Sample OID  -->
        </SenderEntity>
        <SenderName>AnotherSenderAccountId</SenderName>
        <ReceiverEntity>
            <EntityId>1.3.6.1.4.1.111111</EntityId><!-- Sample OID -->
        </ReceiverEntity>
        <TransactionId>MyTransID1</TransactionId>
        <MessageId>//MyProviderDomain/99</MessageId>
        <MessageType>ES4-EnableProfileResponse</MessageType>
        <RelatesTo>//MySenderDomain/123</RelatesTo>
        <MessageDate>2013-04-18T09:45:00Z</MessageDate>
    </RPSHeader>
    <RPSBody>
        <ES4-EnableProfileResponse>
            <FunctionExecutionStatus>
                <Status>EXECUTED_SUCCESS</Status>
            </FunctionExecutionStatus>
        </ES4-EnableProfileResponse>
    </RPSBody>
</RPSMessage>
```

In the context described in the example of the previous section 2.2.1, the function execution response is bound to the following SOAP message:

```
<?xml                    version="1.0"                    encoding="UTF-8"?>
<s:Envelope               xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:wsa="http://schemas.xmlsoap.org/ws/2004/08/addressing"
    xmlns:rps="http://namespaces.gsma.org/esim-messaging/1">
    <s:Header>                                                  <wsa:From>

<wsa:Address>http://AnotherCompany.com/RPS?EntityId=1.3.6.1.4.1.222222?User
Name=AnotherSenderAccountId</wsa:Address>
        </wsa:From>

<wsa:To>http://AnotherCompany.com?EntityId=1.3.6.1.4.1.111111</wsa:To>

<wsa:MessageID>//MyProviderDomain/99?TransactionId=MyTransID1?MessageDate=2
013-04-18T09:45:00Z</wsa:MessageID>
        <wsa:Action>http://gsma.com/ES4/PlatformManagement/ES4-
EnableProfile</wsa:Action>
        <wsa:RelatesTo>//MySenderDomain/123</wsa:RelatesTo>
    </s:Header>
    <s:Body                                       rps:MessageVersion="1.0.0">
```

```
        <rps:ES4-EnableProfileResponse>
            <rps:FunctionExecutionStatus>
                <rps:Status>EXECUTED_SUCCESS</rps:Status>
            </rps:FunctionExecutionStatus>
        </rps:ES4-EnableProfileResponse>
    </s:Body>
</s:Envelope>
```

B.3 Function Binding

NOTE: In the tables below the Asynchronous Request-Response with Callback MEP can be replaced by an Asynchronous Request-Response with Polling MEP. In this case the /wsa:Action value has to be updated accordingly.

B.3.1 ES1

Function name	Binding Information	
RegisterEIS	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES1/eUICCManagement/ES1-RegisterEISRequest
	/wsa:Action Response	http://gsma.com/ES1/eUICCManagement/ES1-RegisterEISResponse

Table 225: ES1 Function Binding

B.3.2 ES2

Function name	Binding Information	
GetEIS	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES2/DataPreparation/ES2-GetEISRequest
	/wsa:Action Response	http://gsma.com/ES2/DataPreparation/ES2-GetEISResponse
DownloadProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES2/ProfileManagement/ES2-DownloadProfile
	/wsa:Action Response	http://gsma.com/ES2/ProfileManagementCallback/ES2-DownloadProfile
UpdatePolicyRules	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES2/ProfileManagement/ES2-UpdatePolicyRules
	/wsa:Action Response	http://gsma.com/ES2/ProfileManagement/ES2-UpdatePolicyRules
UpdateSubscriptionAddress	**MEP**	Synchronous Request-Response

	/wsa:Action Request	http://gsma.com/ES2/ProfileManagement/ES2-UpdateSubscriptionAddressRequest
	/wsa:Action Response	http://gsma.com/ES2/ProfileManagement/ES2-UpdateSubscriptionAddressResponse
EnableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-EnableProfile
	/wsa:Action Response	http://gsma.com/ES2/PlatformManagementCallback/ES2-EnableProfile
DisableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-DisableProfile
	/wsa:Action Response	http://gsma.com/ES2/PlatformManagementCallback/ES2-DisableProfile
DeleteProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-DeleteProfile
	/wsa:Action Response	http://gsma.com/ES2/PlatformManagementCallback/ES2-DeleteProfile
HandleProfileDisabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-HandleProfileDisabledNotification
	/wsa:Action Response	(none)
HandleProfileEnabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-HandleProfileEnabledNotification
	/wsa:Action Response	(none)
HandleSMSRChangeNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-HandleSMSRChangeNotification
	/wsa:Action Response	(none)
HandleProfileDeletedNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES2/PlatformManagement/ES2-HandleProfileDeletedNotification
	/wsa:Action Response	(none)

Table 226: ES2 Function Binding

B.3.3 ES3

Function name	Binding Information	
GetEIS	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-GetEISRequest
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagement/ES3-GetEISResponse
AuditEIS	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-AuditEIS
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagementCallBack/ES3-AuditEIS
CreateISDP	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-CreateISDP
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagemenCallBack/ES3-CreateISDP
SendData	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-SendData
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagementCallBack/ES3-SendData
ProfileDownloadCompleted	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-ProfileDownloadedCompletedRequest
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagement/ES3-ProfileDownloadedCompletedResponse
UpdatePolicyRules	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-UpdatePolicyRules
	/wsa:Action Response	http://gsma.com/ES3/ProfileManagement/ES3-UpdatePolicyRules
UpdateSubscriptionAddress	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES3/ProfileManagement/ES3-UpdateSubscriptionAddressRequest

	/wsa:Action Response	http://gsma.com/ES3/ProfileManagement/ES3-UpdateSubscriptionAddressResponse
EnableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-EnableProfile
	/wsa:Action Response	http://gsma.com/ES3/PlatformManagementCallBack/ES3-EnableProfile
DisableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-DisableProfile
	/wsa:Action Response	http://gsma.com/ES3/PlatformManagementCallBack/ES3-DisableProfile
DeleteISDP	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-DeleteISDP
	/wsa:Action Response	http://gsma.com/ES3/PlatformManagementCallBack/ES3-DeleteISDP
UpdateConnictivitParameters	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-UpdateConnectivityParameters
	/wsa:Action Response	http://gsma.com/ES3/PlatformManagementCallBack/ES3-UpdateConnectivityParameters
HandleProfileDisabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-HandleProfileDisabledNotification
	/wsa:Action Response	(none)
HandleProfileEnabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-HandleProfileEnabledNotification
	/wsa:Action Response	(none)
HandleSMSRChangeNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-HandleSMSRChangeNotification
	/wsa:Action Response	(none)
HandleProfileDeletedNotification	**MEP**	Notification (One-Way)

	/wsa:Action Request	http://gsma.com/ES3/PlatformManagement/ES3-HandleProfileDeletedNotification
	/wsa:Action Response	(none)

Table 227: ES3 Function Binding

B.3.4 ES4

Function name	Binding Information	
GetEIS	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES4/ProfileManagement/ES4-GetEISRequest
	/wsa:Action Response	http://gsma.com/ES4/ProfileManagement/ES4-GetEISResponse
UpdatePolicyRules	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES4/ProfileManagement/ES4-UpdatePolicyRules
	/wsa:Action Response	http://gsma.com/ES4/ProfileManagement/ES4-UpdatePolicyRules
UpdateSubscriptionAddress	**MEP**	Synchronous Request-Response
	/wsa:Action Request	http://gsma.com/ES4/ProfileManagement/ES4-UpdateSubscriptionAddressRequest
	/wsa:Action Response	http://gsma.com/ES4/ProfileManagement/ES4-UpdateSubscriptionAddressResponse
AuditEIS	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/ProfileManagement/ES4-AuditEIS
	/wsa:Action Response	http://gsma.com/ES4/ProfileManagementCallBack/ES4-AuditEIS
EnableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-EnableProfile
	/wsa:Action Response	http://gsma.com/ES4/PlatformManagementCallBack/ES4-EnableProfile
DisableProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-DisableProfile
	/wsa:Action	http://gsma.com/ES4/PlatformManagementCallBack/ES4-

	Response	DisableProfile
DeleteProfile	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-DeleteProfile
	/wsa:Action Response	http://gsma.com/ES4/PlatformManagementCallBack/ES4-DeleteProfile
PrepareSMSRChange	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/eUICCManagement/ES4-PrepareSMSRChange
	/wsa:Action Response	http://gsma.com/ES4/eUICCManagementCallBack/ES4-PrepareSMSRChange
SMSRChange	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES4/eUICCManagement/ES4-SMSRChange
	/wsa:Action Response	http://gsma.com/ES4/eUICCManagementCallBack/ES4-SMSRChange
HandleProfileDisabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-HandleProfileDisabledNotification
	/wsa:Action Response	(none)
HandleProfileEnabledNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-HandleProfileEnabledNotification
	/wsa:Action Response	(none)
HandleSMSRChangeNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES4/eUICCManagement/ES4-HandleSMSRChangeNotification
	/wsa:Action Response	(none)
HandleProfileDeletedNotification	**MEP**	Notification (One-Way)
	/wsa:Action Request	http://gsma.com/ES4/PlatformManagement/ES4-HandleProfileDeletedNotification
	/wsa:Action Response	(none)

Table 228: ES4 Functions Binding

B.3.5 ES7

Function name	Binding Information	
CreateAdditionalKeySet	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES7/eUICCManagement/ES7-CreateAdditionalKeySet
	/wsa:Action Response	http://gsma.com/ES7/eUICCManagementCallBack/ES7-CreateAdditionalKeySet
HandoverEUICC	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES7/eUICCManagement/ES7-HandoverEUICC
	/wsa:Action Response	http://gsma.com/ES7/eUICCManagementCallBack/ES7-HandoverEUICC
AuthenticateSMSR	**MEP**	Asynchronous Request-Response with CallBack
	/wsa:Action Request	http://gsma.com/ES7/eUICCManagement/ES7-AuthenticateSMSR
	/wsa:Action Response	http://gsma.com/ES7/eUICCManagementCallBack/ES7-AuthenticateSMSR

Table 229: ES7 Function Binding

B.4 Web Service Definition Language (WSDL)

The **Web Services Description Language** (**WSDL**) is an XML-based interface definition language that is used for describing the functionality offered by a web service. It provides a machine-readable description of how the service can be called, what parameters it expects, and what data structures it returns.

WSDL files are provided within the SGP.02 Remote Provisioning Architecture for Embedded UICC Technical Specification v3.1 WSDL ZIP package.

This package is composed of the following WSDL files:

- 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
- These WDSL files reference XML schemafiles (.xsd), which are also provided within the SGP.02 Remote Provisioning Architecture for Embedded UICC Technical Specification v3.1 WSDL ZIP package

Annex C Use of GlobalPlatform Privileges

GlobalPlatform defines the following privileges:

Privilege Number	Privilege	Description
0	Security Domain	Application is a Security Domain.
1	DAP Verification	Application is capable of verifying a DAP; Security Domain privilege shall also be set.
2	Delegated Management	Application is capable of Delegated Card Content Management: Security Domain privilege shall also be set.
3	Card Lock	Application has the privilege to lock the card.
4	Card Terminate	Application has the privilege to terminate the card.
5	Card Reset	Application has the privilege to modify historical bytes on one or more card interfaces. This privilege was previously labelled "Default Selected".
6	CVM Management	Application has the privilege to manage a shared CVM of a CVM Application.
7	Mandated DAP Verification	Application is capable of and requires the verification of a DAP for all load operations: Security Domain privilege and DAP Verification privilege shall also be set.
8	Trusted Path	Application is a Trusted Path for inter-application communication.
9	Authorized Management	Application is capable of Card Content Management; Security Domain privilege shall also be set.
10	Token Verification	Application is capable of verifying a token for Delegated Card Content Management.
11	Global Delete	Application may delete any Card Content.
12	Global Lock	Application may lock or unlock any Application.
13	Global Registry	Application may access any entry in the GlobalPlatform Registry.
14	Final Application	The only Application accessible in card Life Cycle State CARD_LOCKED and TERMINATED.
15	Global Service	Application provides services to other Applications on the card.
16	Receipt Generation	Application is capable of generating a receipt for Delegated Card Content Management.
17	Ciphered Load File	The Security Domain requires that the Load File being associated to it is to be loaded

	Data Block	ciphered.
18	Contactless Activation	Application is capable of activating and deactivating other Applications on the contactless interface.
19	Contactless Self-Activation	Application is capable of activating itself on the contactless interface without a prior request to the Application with the Contactless Activation privilege.

Table 230: GlobalPlatform Privileges

Privileges description in an eUICC:

The following rules apply for an eUICC with at least one Profile installed.

Security Domain Privilege:

GlobalPlatform Card Specification [6] states: “This privilege distinguishes a Security Domain from a 'normal' Application.”

DAP Verification Privilege:

GlobalPlatform Card Specification [6] states: “An application provider may require that their Application code to be loaded on the card shall be checked for integrity and authenticity. The DAP Verification privilege provides this service on behalf of an Application provider.”

Delegated Management:

GlobalPlatform Card Specification [6] states: “The privilege allows an Application Provider to manage Card Content with authorisation.” A “Security Domain having the Token Verification privilege controls such authorisation.”

Card Lock:

GlobalPlatform Card Specification [6] states: “This privilege allows an Application to set the card life cycle state to CARD_LOCKED.”

On the eUICC, the Card Lock privilege is not applicable and shall not be assigned to any security domain/Application. The equivalent mechanism of disabling a Profile shall be used.

Card Terminate:

GlobalPlatform Card Specification [6] states: “This privilege allows an Application to set the card life cycle state to TERMINATED.”

On the eUICC, the Card Terminate privilege is not applicable and shall not be assigned to any security domain/Application. The equivalent mechanism of deleting a Profile shall be used.

Card Reset:

GlobalPlatform Card Specification [6] states: “An Application installed or made selectable with the Card Reset privilege and no Implicit Selection parameter is registered in the GlobalPlatform Registry as the implicitly selectable Application on the Basic Logical Channel for all card I/O interfaces supported by the card if no other Application (other than the Issuer Security Domain) is already registered as implicitly selectable on the Basic Logical Channel of any card I/O interface”.

This privilege is relevant only when the Profile is enabled. Therefore, several Applications may have this privilege on the eUICC, but this privilege shall be unique within a Profile.

If the Application inside a Profile with the Card Reset privilege is deleted the privilege is reassigned to the corresponding MNO-SD.

CVM Management:

GlobalPlatform Card Specification [6] states: “The CVM Application, if present on a card, provides a mechanism for a Cardholder Verification Method (CVM), including velocity checking, that may be used by all Applications on the card”.

If an Application in a Profile has this privilege, it shall be relevant only when the Profile is enabled. In that case, several Applications may have this privilege on the card, but this privilege shall be unique within a Profile.

Mandated DAP Verification:

GlobalPlatform Card Specification [6] states: “A Controlling Authority may require that all Application code to be loaded onto the card shall be checked for integrity and authenticity. The Mandated DAP Verification privilege of the Controlling Authority's Security Domain detailed in this Specification provides this service on behalf of the Controlling Authority”.

If an Application in a Profile has this privilege, it shall be relevant only when the Profile is enabled. In that case, several Applications may have this privilege on the card, but this privilege shall be unique within a Profile.

The DAP verification is mandated only when loading an Application inside the Profile.

Trusted Path:

GlobalPlatform Card Specification [6] states: "The 'Trusted Path' privilege qualifies an Application as a Receiving Entity. Each Application present on the card playing the Role of a Receiving Entity shall: Enforce the Issuer's security rules for inter-application communication; Ensure that incoming messages are properly provided unaltered to the Trusted Framework; Ensure that any response messages are properly returned unaltered to the off-card entity”.

Authorised Management:

GlobalPlatform Card Specification [6] states: “Having a Security Domain with this privilege allows a Security Domain provider to perform Card Content management without authorisation (i.e. a token) in the case where the off-card entity is authenticated as the owner (Security Domain Provider) of the Security Domain”.

Token Verification:

GlobalPlatform Card Specification [6] states: “This privilege allows a Security Domain Provider, to authorize any Card Content management operation”.

Global Delete:

GlobalPlatform Card Specification [6] states: “This privilege provides the capability to remove any Executable Load File or Application from the card even if the Executable Load File or Application does not belong to this Security Domain”.

For MNO-SD and Applications inside a Profile, this privilege shall only allow deletion of Applications in the corresponding Profile.

Global Lock:

GlobalPlatform Card Specification [6] states: “This privilege provides the right to initiate the locking and unlocking of any Application on the card, independent of its Security Domain Association and hierarchy. It also provides the capability to restrict the Card Content Management functionality of OPEN”.

For MNO-SD and Applications inside a Profile, this privilege shall only allow locking of Applications in the corresponding Profile.

Global Registry:

GlobalPlatform Card Specification [6] states: “The search is limited to the Executable Load Files, Applications and Security Domains that are directly or indirectly associated with the eUICC entity receiving the command. When the eUICC entity receiving the command has the Global Registry privilege, the search applies to all Executable Load Files, Applications and Security Domains registered in the GlobalPlatform Registry”.

For ISD-P and Applications inside a Profile, this privilege shall only allow looking for Applications in the corresponding Profile.

Final Application:

GlobalPlatform Card Specification [6] states: “If a Security Domain has the Final Application privilege only the GET DATA command shall be processed, all other commands defined in this specification shall be disabled and shall return an error”.

On the eUICC, the Final Application privilege is not applicable and shall not be assigned to any security domain/Application.

Global Service:

GlobalPlatform Card Specification [6] states: “One or more Global Services Applications may be present on the card to provide services to other Applications on the card.

The MNO-SD or Applications inside a Profile with the Global Service privilege shall offer service only when the Profile is enabled. Therefore, it is possible to have several Applications registered on the same service in the same eUICC.

Receipt Generation:

GlobalPlatform Card Specification [6] states: “This privilege allows a Security Domain Provider, typically the Card Issuer, to provide a confirmation for the performed card content management. A Security Domain with Receipt Generation privilege requires the knowledge of keys and algorithms used for Receipts generation”.

Ciphered Load File Data Block:

GlobalPlatform Card Specification [6] states: “This privilege allows a Security Domain Provider to require that the Load File Data Block being associated to it shall be ciphered”.

Contactless Activation:

GlobalPlatform Card Specification [6] states: “The Contactless Activation privilege identifies the CRS Application. This Privilege allows:

- The Activation/Deactivation of Applications on the Contactless Interface
- The update of the Selection Priority
  - Manage the Volatile Priority
  - Reorder the GlobalPlatform Registry
- Notification by the OPEN when:
  - An Application is INSTALLED, LOCKED, unlocked or deleted
  - The availability state of an Application is changed between NON_ACTIVATABLE, ACTIVATED, or DEACTIVATED.
  - One of the Application’s contactless registry parameters is updated”.

If an Application in a Profile has this privilege, it shall be relevant only when the Profile is enabled. In that case, several Applications may have this privilege on the card, but this privilege shall be unique within a Profile.

Contactless Self-Activation:

GlobalPlatform Card Specification [6] states: “The Contactless Self-Activation Privilege allows an Application to activate itself without a prior request to the CRS Application”.

If an Application in a Profile has this privilege, it shall be relevant only when the Profile is enabled.

Privilege Number	Privilege	ISD-R	ISD-P	MNO-SD	Applications inside a Profile	ECASD
0	Security Domain	√	√	√		√
1	DAP Verification					
2	Delegated Management					
3	Card Lock					
4	Card Terminate					
5	Card Reset					
6	CVM Management			√ **		
7	Mandated DAP Verification					
8	Trusted Path	√	√	√		
9	Authorized Management		√ *	√		
10	Token Verification			√ **		
11	Global Delete			√ **		
12	Global Lock			√ **		
13	Global Registry			√ **		
14	Final Application					
15	Global Service					√
16	Receipt Generation			√ **		
17	Ciphered Load File Data Block					
18	Contactless Activation					
19	Contactless Self-Activation					

Table 231: GlobalPlatform Application Privileges

A tick (√) denotes the presence of the indicated privilege and its assignment to the Security Domain or Application.

A blank cell denotes that the assignment of the privilege is managed by the owner of the Application (according to GlobalPlatform Card Specification [6]) of the Security Domain.

A black cell denotes that the privilege cannot be assigned.

* Authorized Management privilege is only set when ISD-P is in CREATED state to allow Profile Download and Installation.

** These privileges are mandatory for cards compliant to GlobalPlatform Card Specification UICC Configuration [7].

Annex D Data Definitions

- Coding of the IMEI

The value of IMEI shall be directly copied from Terminal Response of the Provide Local Information command (see ETSI TS 102 223 [3] and ETSI TS 124 008[20]).

Annex E EIS Usage in Functions

This table gives additional information on the EIS usage depending on the function:

- Column 'EUM Signed': indicates if the data is part of the signature computed by the EUM at the initial registration time.
- **ES1.RegisterEIS**:
  - A 'X' indicates that the data shall to be provided
  - An empty cell indicates that the data shall not be provided
- **ES3.GetEIS**, **ES3.AuditEIS**, **ES4.GetEIS**, **ES4.AuditEIS**, **ES2. HandleSMSRChangeNotification**, **ES3. HandleSMSRChangeNotification**, **ES4. HandleSMSRChangeNotification** , ,**ES7.HandoverEUICC**:
  - A 'X' indicates that the data may be provided
  - An empty cell indicates that the data shall not be provided

Data name	EUM Signed	ES1.RegisterEIS	ES2.GetEIS	ES3.GetEIS ES3.AuditEIS	ES4.GetEIS ES4.AuditEIS	ES2.HandleSMSRChangeNotification ES3.HandleSMSRChangeNotification ES4.HandleSMSRChangeNotification	ES7.HandoverEUICC
eid	X	X	X	X	X	X	X
eum-id	X	X	X	X	X		X
productionDate	X	X	X	X	X		X
platformType	X	X	X	X	X		X
platformVersion	X	X	X	X	X		X
remainingMemory		X	X	X	X		X

Availablememoryforprofiles		X	X	X	X		X
lastAuditDate				X	X		X
smsr-id		X	X	X	X		X
isd-p-loadfile-aid	X	X		X	X		X
isd-p-module-aid	X	X		X	X		X
Profiles		X(1)			X(3)	X	X
iccid		X			X	X	X
isd-p-aid		X			X	X	X
mno-id		X			X	X	X
fallbackAttribute		X			X		X
subscriptionAddress		X			X		X
msisdn		X			X		X
imsi		X			X		X
state		X			X		X
smdp-id					X		X
ProfileType		X			X	X	X
allocatedMemory		X			X		X
freeMemory		X			X		X
pol2		X			X		X
rules		X			X		X
subject		X			X		X
action		X			X		X
qualification		X			X		X
ISD-R		X(2)		X(4)			X(4)
ECASD	X	X(2)	X(2)	X			X(4)

eUICC-Capabilities	X	X	X	X	X		X
CAT-TP-Support	X	X	X	X	X		X
CAT-TP-Version	X	X	X	X	X		X
HTTP-Support	X	X	X	X	X		X
HTTP-Version	X	X	X	X	X		X
secure-packet-version	X	X	X	X	X		X
Remote-provisioning-version	X	X	X	X	X		X
audit trail							X
eumCertificateId	X	X	X	X	X		X
signatureAlgorithm	X	X	X	X	X		X
signature		X(5)	X(5)	X(5)	X(5)		X(5)

Table 232: EIS Usage

NOTE 1: The initial EIS comes with the information of the Profile(s) loaded and installed by the EUM during the manufacturing.

NOTE 2: The initial EIS comes with the definition of the two Security Domains ISD-R and ECASD.

NOTE 3: The EIS shall only contain the Profile owned by the requesting MNO

NOTE 4: The EIS shall contain all Security Domains definition except the current Key set on ISD-R used by the current SM-SR.

NOTE 5: The EIS is signed using the private key of the EUM (see Figure 8).

Annex F Key Check Values

All key check values that have to be computed in the context of this specification shall follow the recommendation of GlobalPlatform Card Specification [6] section B5 and GlobalPlatform Card Specification Amendment B [8] section 3.8. Extract:

“For a DES key, the key check value is computed by encrypting 8 bytes, each with value '00', with the key to be checked and retaining the 3 highest-order bytes of the encrypted result.”

"For a AES key, the key check value is computed by encrypting 16 bytes, each with value '01', with the key to be checked and retaining the 3 highest-order bytes of the encrypted result."

"A key check value shall be computed as the three most significant bytes of the SHA-1 digest of the PSK TLS Key".

Annex G Device Requirements

Functional Device Requirements No.	Requirement

DEV1	For connectivity the Device shall support:
	• At least one of the network access technologies defined by 3GPP in the non-exhaustive following list:
	  ○ GERAN,
	  ○ UTRAN
	  ○ E-UTRAN.
	• UDP over IP [32] (subject to the right support of access network technology)
	• TCP over IP [33] (subject to the right support of access network technology)

DEV2	For Network connection control the Device shall support: • RPLMN details (LAC/TAC, NMR). • QoS (failures, duration, power, location). • SMS management. • New network selection after SIM/USIM update.
DEV3	For reporting to a server the Device shall support: • SMS-PP MO as defined in [3] and SMS-PP MO as defined [33] or [29] BIP as defined in DEV4 The Device should support: • USSD

DEV4	For Profile and Platform Management the Device shall support:
	• SMS-PP MT as defined in [3], and SMS-PP MT as defined [33] or [29]
	• BIP (subject to the support of the right network access technology) as defined in [3] including support of commands:
	  ○ OPEN CHANNEL (UDP and TCP over IP)
	  ○ CLOSE CHANNEL
	  ○ RECEIVE DATA
	  ○ SEND DATA
	  ○ GET CHANNEL STATUS

	○ ENVELOPE (EVENT DOWNLOAD - Data available) ○ ENVELOPE (EVENT DOWNLOAD – Channel status)
DEV5	The Device shall contain a unique IMEI (International Mobile Equipment Identity) value compliant with the format defined in ETSI TS 123 003 [31]. The value of IMEI shall be directly copied from TERMINAL RESPONSE of the Provide Local Information command (see ETSI TS 102 223 [3] and ETSI TS 124 008[20]).
DEV6	• The Device shall support, as a minimum, the following set of commands (in addition to BIP commands) as defined in ETSI TS 102 223 [3] and 3GPP TS 31.111 [27]. Basic SAT commands (TERMINAL PROFILE, FETCH, TERMINAL RESPONSE) • PROVIDE LOCAL INFORMATION (location information, IMEI, NMR, date and time, access technology, at least) • SEND SHORT MESSAGE • POLL INTERVAL, POLLING OFF, TIMER MANAGEMENT [at least one timer], ENVELOPE (TIMER EXPIRATION) • SET UP EVENT LIST and ENVELOPE (EVENT DOWNLOAD – location status, call connected, call disconnected, Access Technology Changed, Network Rejection) • ENVELOPE (SMS-PP DOWNLOAD) • REFRESH Command (At least mode 4 - “UICC reset”)
DEV7	The Device shall comply with the GSMA-EICTA document “Security Principles Related to Handset Theft” [30]
DEV8	The Device may retrieve the EID defined in section 2.2.2 of this specification from the eUICC and shall support the following commands as described in [35]: • AT+CCHO (Open Logical Channel) • AT+CCHC (Close Logical Channel) • AT+CGLA (Generic UICC Logical Channel Access)
DEV9	The Device shall support from the [35] the following commands for all generic purposes: • AT+CRSM (Restricted SIM access)

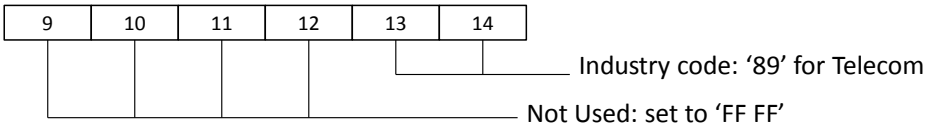
Annex H Coding of the PIX for ‘Embedded UICC Remote Provisioning and Management’ (Normative)

The following coding of the PIX, following ETSI TS 101 220 [2], applies for ISD-R, ISD-P and ECASD:

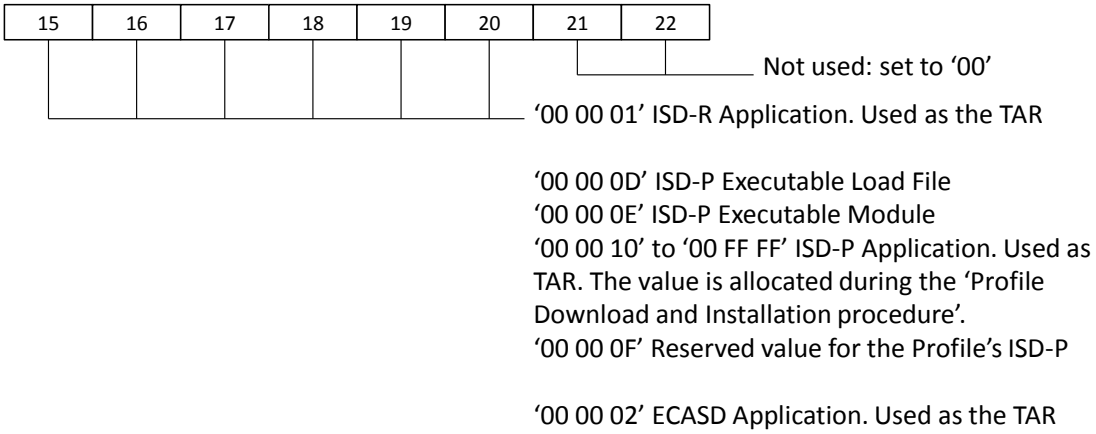
- **Digits 1 to 4** - Application code for ‘Embedded UICC Remote Provisioning and Subscription Management’
  - Coding: Fixed value '10 10'

- **Digits 5 to 8** - Not used
  - Coding: Fixed value 'FF FF'

- **Digits 9 to 14** - Application provider code



- **Digits 15 to 22** - Application Provider field 8 hexadecimal digits



Annex I List of Identifiers (Normative)

OIDs

The following identifiers for remote provisioning are created under a dedicated OID tree under ISO branch:

- ASN.1 notation: {iso(1) identified-organization(3) dod(6) internet(1) private(4) enterprise(1)}
- dot notation: 1.3.6.1.4.1
- IOD-IRI notation: /ISO/Identified-Organization/6/1/4/1

The private enterprise numbers may be found under the Internet Assigned Numbers Authority: http://www.iana.org/assignments/enterprise-numbers/enterprise-numbers

EUM Identifier

Identifier	Uniqueness	Registration Entity
EUM OID	within the ecosystem	ISO 1.3.6.1.4.1
SIN	within the ecosystem	ISO 7812

eUICC Identifier

Identifier	Uniqueness	Registration Entity
EID	within the ecosystem	GSMA ESIM Technical Specification
ECASD AID	within the eUICC	GSMA ESIM Technical Specification
ISD-R AID	within the eUICC	GSMA ESIM Technical Specification
ISD-P AID	within the eUICC	SM-SR within a range Defined in GSMA ESIM Technical Specification
ICCID	Global	ITU
ISD-R TAR	within the eUICC	GSMA ESIM Technical Specification
MNO-SD AID	Within the Profile	ETSI TS 101 220 (ISD AID)
MNO-SD TAR	Within the Profile	ETSI TS 101 220 (ISD TAR)

SM-SR Identifier

Identifier	Uniqueness	Registration Entity
SMSR OID (called SRID in Stage 2)	within the ecosystem	ISO 1.3.6.1.4.1

SM-DP Identifier

Identifier	Uniqueness	Registration Entity
SMDP OID (called DPID in Stage 2)	within the ecosystem	ISO 1.3.6.1.4.1

MNO Identifier

Identifier	Uniqueness	Registration Entity
MNO OID	within the ecosystem	ISO 1.3.6.1.4.1
MCC+MNC (IMSI)	Global	ITU-T for MCC and National Regulators for MNC

Annex J Verification of EID

Verification of an EID is performed as follows:

- Using the 32 digits as a decimal integer, compute the remainder of that number on division by 97.
- If the remainder of the division is 1, the verification is successful; else the EID is invalid.

NOTE: Examples of valid EIDs are:

- 8900 1012 0123 4123 4012 3456 7890 1224
- 8900 1567 01020304 0506 0708 0910 1152
- 8904 4011 1122 3344 1122 3344 1122 3321

Annex K Document Management

K.1 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
V1.0	19/12/2013	1st Release of Document, submitted to PSMC#119 for approval	GSMA Embedded SIM Leadership Team and PSMC	Ian Smith, GSMA
V2.0	16/09/2014	Revision of the document by SIMSG ESIM WG4, including the following CRs: 14ESIMWI401_03r1, 14ESIMWI402_06, 14ESIMWI402_08, 14ESIMWI402_09, 14ESIMWI402_10, 14ESIMWI402_12r1, 14ESIMWI403_06r1, 14ESIMWI403_07r1, 14ESIMWI403_08 r1, 14ESIMWI403_09, 14ESIMWI403_10, 14ESIMWI403_11 r1, 14ESIMWI403_13 r1, 14ESIMWI403_14 r1, 14ESIMWI403_18 r1, 14ESIMWI403_19 r1, 14ESIMWI403_20 r1, 14ESIMWI403_21 r1, 14ESIMWI405_5, 14ESIMWI405_14r1, 14ESIMWI405_15r2, 14ESIMWI405_20r1, 14ESIMWI405_21r1, 14ESIMWI405_22r1, 14ESIMWI405_23r1, 14ESIMWI405_24r1, 14ESIMWI405_25r1, 14ESIMWI405_26r1.	GSMA Embedded SIM Leadership Team and PSMC	Alexis Michel, Oberthur Technologies
V3.0	29/05/2015	14ESIMWI406_02, 14ESIMWI406_05r1,	GSMA Embedded SIM Leadership Team and PSMC	Alexis Michel, Oberthur Technologies

		14ESIMWI406_08r1, 14ESIMWI406_09r3, 14ESIMWI406_10r1, 14ESIMWI406_11r1, 14ESIMWI406_12r1, 14ESIMWI406_13r1, 14ESIMWI406_16r1, 14ESIMWI406_21r1, 14ESIMWI406_22r1, 14ESIMWI406_23r1, 14ESIMWI406_24r1, 14ESIMWI406_25r1, 14ESIMWI406_27r1, 14ESIMWI406_28r1, 14ESIMWI409_02, 14ESIMWI409_03, 14ESIMWI409_04r1, 14ESIMWI409_06, 14ESIMWI409_07r1, 14ESIMWI409_09, 14ESIMWI409_10, 14ESIMWI409_11r1, 14ESIMWI409_12, 14ESIMWI409_14r1, 14ESIMWI409_16r1, 14ESIMWI409_17r1, 14ESIMWI409_18, 14ESIMWI409_19r1, 14ESIMWI409_21r1, 14ESIMWI409_25r1, 14ESIMWI409_26r1, 14ESIMWI409_27, 14ESIMWI409_28, 14ESIMWI409_29, 14ESIMWI409_32, 14ESIMWI409_33r1, 14ESIMWI409_38r1, 14ESIMWI409_39, 14ESIMWI409_40, 14ESIMWI409_44r1, 14ESIMWI409_46, 14ESIMWI410_04r1, 14ESIMWI411_02r1, 14ESIMWI411_03r1, 14ESIMWI411_04r2,		

		14ESIMWI411_05r2, 14ESIMWI411_11r2, 14ESIMWI411_13r6, 14ESIMWI411_14r1, 14ESIMWI411_15, 14ESIMWI411_16, 14ESIMWI411_25, 14ESIMWI411_26r4, 14ESIMWI411_27, 14ESIMWI411_30r1, 14ESIMWI411_31, 14ESIMWI411_32, 14ESIMWI411_34, 14ESIMWI411_37r1.		
V3.1	12/05/16	15ESIMWI412_03r1, 15ESIMWI413_01, 15ESIMWI413_02, 15ESIMWI413_03r1, 15ESIMWI413_04, 15ESIMWI413_05r1, 15ESIMWI413_06, 15ESIMWI413_07, 15ESIMWI414_02r3, 15ESIMWI414_03r2, 15ESIMWI414_04r1, 15ESIMWI414_05r3, 15ESIMWI414_06r1, 15ESIMWI414_07r2, 15ESIMWI414_14r1, 15ESIMWI418_001r1, 16ESIMWI415_02r1, 16ESIMWI415_07r1, 16ESIMWI417_01r1, 16ESIMWI417_03r2, 15ESIMWI419_001.	GSMA Embedded SIM Leadership Team and PSMC	Alexis Michel, Oberthur Technologies

K.2 Other Information

Type	Description
Document Owner	Embedded SIM
Editor / Company	Alexis Michel, Oberthur Technologies

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

Your comments or suggestions & questions are always welcome.