

NFC Handset Testbook
Version 4.0
10 October 2014

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 © 2014 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 and Test Book structure 7
1.2.1 Test Book scope 8
1.3 Definition of Terms 9
1.3.1 Power mode definition 11
1.4 Document Cross-References 11
1.5 Conventions 13
2 Test environment 13
2.1 Applicability 13
2.1.1 Format of the table of optional features 13
2.1.2 Format of the applicability table 13
2.1.3 Status and Notations 14
2.1.4 Table of optional features 14
2.1.5 Applicability Table 15
2.2 General consideration 27
2.2.1 Test specifications 27
2.2.2 SIMalliance Open Mobile API 28
2.2.3 Pass criterion 28
2.2.4 Future study 28
2.3 Tests with measurement and physical settings 28
2.4 Reference Transaction 28
2.5 Test Equipment 29
2.5.1 UICC 29
2.5.2 Requirements for UMTS Network Simulator 31
2.5.3 Common applications 31
2.5.4 Tag Testing 34
2.5.5 Access Control Test bench 38
2.5.6 Reader equipment 39
2.5.7 NFC Controller and UI application triggering 39
2.5.8 Test Set-Up for OTA communication 39
3 NFC Features 41
3.1 General overview 41
3.2 Conformance requirements 41
3.3 Reader/Writer mode 43
3.3.1 General overview 43
3.3.2 Conformance requirements 43
3.3.3 Test Cases 43
3.4 Card emulation mode 76
3.4.1 General overview 76
3.4.2 Conformance requirements 76

3.4.3 Test Cases 76
3.5 Core and Common features 92
3.5.1 General overview 92
3.5.2 Conformance requirements 93
3.5.3 Test Cases 93
4 VOID 97
5 Secure Element Access Control 98
5.1 General overview 98
5.2 Conformance requirements 98
5.3 Test Cases 98
5.3.1 GP SE Access Control 98
5.3.2 GP SE Access Control - Refresh tag 110
5.3.3 GP SE Access Control – ADF_PKCS#15 and DF PKCS#15 112
5.3.4 GP SE Access Control – PKCS#15 selection via EF_DIR 114
5.3.5 GP SE Access Control – Configuration limits 115
5.3.6 GP SE Access Control – No access 118
5.4 GP SE Access Control – GP Test Plan 123
6 Secure Element Access API 124
6.1 General overview 124
6.2 Conformance requirements 124
6.3 Test Cases 124
6.3.1 SIMalliance APIs 124
6.3.2 Prevent access to basic channel. 124
6.3.3 VOID 125
6.3.4 VOID 125
6.3.5 VOID 125
7 Multiple Card Emulation Environment 126
7.1 General overview 126
7.2 Conformance requirements 126
7.3 Test Cases 126
7.3.1 SE Availabilities from mobile applications 126
7.3.2 SE selection 126
7.3.3 UICC as default SE 127
7.3.4 SE Selection API 128
7.3.5 SE Toggle switching API 128
7.3.6 Switch for Card Emulation Mode 128
7.3.7 Multiple CE Environments 129
7.3.8 Active Card Emulation in Multiple CE Environments / Card Emulation 134
8 UI Application triggering 137
8.1 General overview 137
8.2 Conformance requirements 137
8.3 Test Cases 137
8.3.1 EVT_TRANSACTION 137

8.3.2 Permissions 137
8.3.3 Intent management 138
8.3.4 Application's triggering order 140
8.3.5 Triggering on HCI event EVT_CARD_DEACTIVATED 143
8.3.6 Triggering on HCI event EVT_FIELD_OFF 144
9 VOID 145
10 VOID 146
11 Mobile Device APN management 147
11.1 General overview 147
11.2 Conformance requirements 147
11.3 Test Cases 147
11.3.1 OPEN CHANNEL 147
11.3.2 CLOSE CHANNEL 155
11.3.3 RECEIVE DATA 159
11.3.4 SEND DATA 173
12 Remote Management of NFC Services 185
12.1 General overview 185
12.2 Conformance requirements 185
12.3 Basic Remote Management 185
12.3.1 General overview 185
12.3.2 Conformance requirements 185
12.3.3 Test Cases 186
12.4 Remote Management use cases 233
12.4.1 General overview 233
12.4.2 Conformance requirements 233
12.4.3 Test Cases 233
13 General Device Support 278
13.1 General Overview 278
13.2 Conformance requirements 278
13.3 Test Cases 278
13.3.1 Secure Element Access API and Access control in Radio OFF State 278
13.3.2 Enabled / Disabled states 285
14 VOID 286
15 Android specific test cases 287
15.1 General overview 287
15.2 Conformance requirements 287
15.3 NFC Features 288
15.3.1 General overview 288
15.3.2 Conformance requirements 288
15.3.3 Test Cases 289
15.4 VOID 295
15.5 Secure Element Access Control 296
15.5.1 General overview 296

15.5.2 Conformance requirements 296
15.5.3 Test Cases 296
15.6 VOID 297
15.7 Multiple Card Emulation Environment 298
15.7.1 General overview 298
15.7.2 Conformance requirements 298
15.7.3 Test Cases 298
16 BlackBerry specific test cases 302
17 Windows Phone specific test cases 302
18 Document History 303
Annex A Reference Application 305
A.1 Description 305
A.2 AID 305
A.3 Structure FileGil 305
A.4 Commands Permitted 305
A.4.1 SELECT 305
A.4.2 READ BINARY 306
A.4.3 UPDATE BINARY 306
A.4.4 EXTERNAL AUTHENTICATE 306
A.5 Source Code (Java) 308
Annex B Reference to other test plan 313
B.1 SIMalliance 313
B.2 EMVCo 314
B.3 ISO 10373-6 314
B.4 ETSI TS 102 613 SWP 314
B.5 ETSI TS 102 622 HCI 317
B.6 ETSI TS 102.384, 3GPP 31.124 320
B.7 Void 326
B.8 GP Secure Element Access Control 326
Annex C Reference Tags - Real NFC Tags 342
Annex D Tables from the Test Book 343
D.1 List of test cases 343
D.2 Table of optional features 353
D.3 Applicability table 354
Annex E Reference PKCS#15 structures 365
Logically 365
E.1 PKCS#15 application for test case 5.3.1.1 365
E.2 PKCS#15 application for test case 5.3.1.2 367
E.3 PKCS#15 application for test case 5.3.1.3 368
E.4 PKCS#15 application for test case 5.3.1.4 369
E.5 PKCS#15 application for test case 5.3.1.5 370
E.6 PKCS#15 application for test case 5.3.1.6 371
E.7 PKCS#15 application for test case 5.3.1.7 372

E.8 PKCS#15 application for test case 5.3.1.8 373
E.9 PKCS#15 application for test case 5.3.1.9 374
E.10 PKCS#15 application for test case 5.3.2.1 375
E.11 PKCS#15 application for test case 5.3.2.1 Step5 376
E.12 PKCS#15 application for test case 5.3.2.2 377
E.13 PKCS#15 application for test case 5.3.3.1 378
E.14 PKCS#15 file system for test case 5.3.3.1 379
E.15 PKCS#15 file system for test case 5.3.4.1 380
E.16 PKCS#15 application for test case 5.3.5.1 381
E.17 PKCS#15 application for test case 5.3.5.2 383
E.18 PKCS#15 application for test case 5.3.6.2 385
E.19 PKCS#15 application for test case 5.3.6.3 386
E.20 PKCS#15 application for test case 5.3.6.4 387
E.21 PKCS#15 application for test case 5.3.6.5 388
E.22 PKCS#15 application for test case 8.3.4.1 389
E.23 PKCS#15 application for test case 8.3.4.2 390
E.24 PKCS#15 application for test case 8.3.4.3 391
E.25 PKCS#15 application for test case 13.3.1.1 392
E.26 PKCS#15 application for test case 13.3.1.2 393
E.27 PKCS#15 application for test case 13.3.1.3 394

1 Introduction

1.1 Overview

The main aim of the GSMA NFC activities is to accelerate the commercial launch of UICC – based NFC services in a number of markets by ensuring interoperability of services.

The NFC Test Book stream is part of GSMA NFC activities. The participating GSMA TSG members have developed a set of test cases to be used for testing the UICC based NFC functionality within a Mobile Device. These tests have been collated in this "Test Book" and provide test case descriptions against the requirements listed in the GSMA TS.26 NFC Handset Requirements document [1].

This document includes an applicability table providing an indication whether test cases are relevant for a specific device operating system.

The Test Book is developed in such a way that the test case descriptions are generic, but provide repeatable instructions so that any accredited Test Lab can implement these test cases without further clarification.

The Test Lab will be responsible for running the test cases (which are tool specific) as set out in the Test Book.

More information about the GSMA NFC activities can be found in [2].

1.2 Scope and Test Book structure

This document is intended for:

- Parties which develop test tools and platforms
- Test Labs / Test Houses which execute the testing
- Vendors, Device & chipset Manufacturers.
- Operators.

The Test Book consists of a set of test cases relevant for testing a device which is implementing UICC based NFC services. The testing scope is related to selected parts of the NFC enabled device and is further detailed below.

The test cases specified within the Test Book are either specified fully, step by step or referred to existing publicly available test standards. For the test cases from other organizations, a unique reference to the specification and test case is provided.

For each test case specified or referred to within this Test Book, there is a reference to one or more requirements from the TS.26 GSMA NFC Handset Requirements document.[1]

1.2.1 Test Book scope

The scope of testing is identified below with the reference architecture for a NFC enabled device with UICC NFC services.

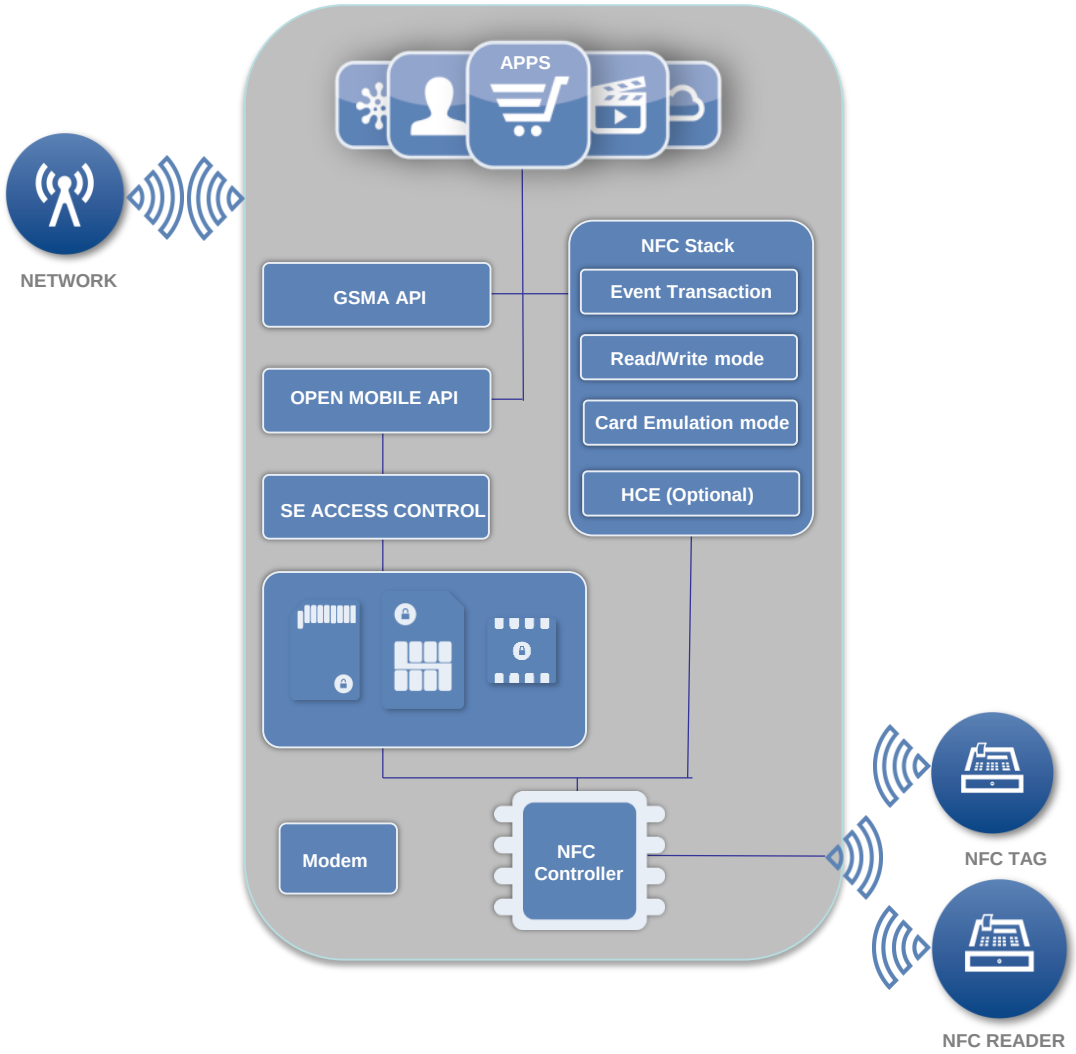

Figure 1: Reference architecture for a NFC enabled device with UICC NFC services

The overall structure of the Test Book is based on the interfaces as identified in the architecture showing relevant NFC related components. The first section starts with the Tag and Card reader interface, stepping through the different device components and ending at the Mobile network related features. This gives the following structure:

1. Introduction
2. Test Environment
3. NFC Features
   a) Reader / Writer mode
   b) Card emulation mode
   c) Core and common features
4. VOID (reserved for future test cases)
5. Secure Element Access Control
6. Secure Element Access API

7. Multiple Card Emulation Environment
8. UI Application Triggering
9. VOID (reserved for future test cases)
10. VOID (reserved for future test cases)
11. Mobile Device APN Management
12. Remote Management of NFC Services
    a) Basic Remote Management
    b) Remote Management use cases
13. General Device Support
14. VOID (reserved for future test cases)
15. Android Specific Test Cases
16. Blackberry Specific Test cases
17. Microsoft Windows Specific Test cases

Annexes

1.3 Definition of Terms

Term	Description
Device	In the context of this specification, the term Device is used to represent any electronic equipment supporting NFC functionality into which a UICC-based NFC Secure Element can be inserted, and that provides a capability for a server to reach the UICC through an Over The Air (OTA) channel.
Test Lab	This refers to a test lab which will run the test cases according to the Test Book for testing NFC Devices.
Operator	Refers to a Mobile Network Operator who provides the technical capability to access the mobile environment using an Over The Air (OTA) communication channel. The OPERATOR is also the UICC Issuer. An OPERATOR provides a UICC OTA Management System, which is also called the OTA Platform.
Vendor	Device manufacturer
Test Book	Document describing the test cases that allow testing the requirements listed in the GSMA TS.26 NFC Handset Requirements [1]
User	Describes any logical or physical entity which controls the device under test or the test equipment in a way that it is able to trigger activities of the DUT
Distance	This refers to the distance from the back of the device to Point of Sale NFC antenna or to Tag surface.

Acronyms	Description
AC	Access Control
ACCF	Access Control Conditions File
ACMF	Access Control Main File
ACRF	Access Control Rules File
ADF	Application Dedicated File
AID	Application Identifier

Acronyms	Description
API	Application Programming Interface
APDU	Application Protocol Data Unit
APN	Access Point Network
BIP	Bearer Independent Protocol
C-APDU	Command APDU
CE	Card Emulation
CEE	Card Emulation Environment
CLF	Contactless Frontend
DODF	Data Object Directory File
DUT	Device Under Test
EVT	Event
FFS	For Future Study
HCI	Host Controller Interface
JCP	Java Community Process
JVM	Java Virtual Machine
JSR	Java Specification Request
ME	Mobile Equipment
MIDP	Mobile Information Device Profile
MNO	Mobile Network Operator
NFC	Near Field Communication
ODM	Original Device Manufacturer
OEM	Original Equipment Manufacturer
OS	Operating System
PC/SC	PC SmartCard reader
PCD	Proximity Coupling Device
PKCS	Public Key Cryptographic Standard
PoS	Point of sale
R-APDU	Response APDU
RIL	Radio Interface Layer
RTD	Record Type Definition
SCWS	Smart Card Web Server
SE	Secure Element
SIM	Subscriber Identity Module
SP	Service Provider
SW	Status Word
SWP	Single Wire Protocol
UI	User Interface
UICC	Universal Integrated Circuit Card (USIM)

Acronyms	Description
USS	UMTS System Simulator

1.3.1 Power mode definition

This section gives the definition for different battery modes for the support NFC services as shown in Figure 2.

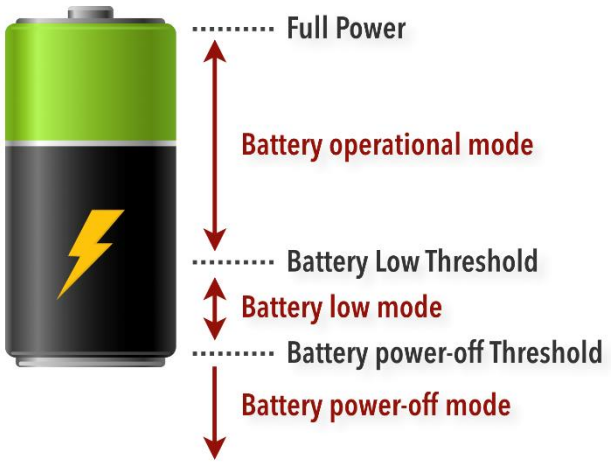

Figure 2: Battery power levels within the NFC mobile devices

Term	Description
Battery Operational Mode	The battery of the DUT has sufficient power to support all functions in the mobile devices.
Battery Low Mode	The battery of DUT has reached “Battery Low Threshold” at which the display and most functionalities of the DUT are automatically switched off, except the clock and a few remaining functions. The battery of the DUT only has sufficient power to support NFC controller to function.
Battery Power-off Mode	The battery of the DUT has reached “Battery Power-off threshold” at which there is no residual power to support NFC controller to function. No functions are available in the DUT. The NFC controller can function if power is provided via the contactless interface (i.e. power by the field).

1.4 Document Cross-References

Ref	Title
[1]	GSMA TS.26 v6.0 NFC Handset Requirements
[2]	GSMA Digital Commerce Programme : http://www.gsma.com/digitalcommerce/

Ref	Title
[3]	VOID
[4]	Public transport-Communication between contactless readers and fare media, AFIMB Implementation requirements for ISO/IEC 14443-v1.0,
[5]	SIMalliance OMAPI Transport API Test Specification V1.1
[6]	SIMalliance – Open Mobile API specification V2.05 or later (backwards compatible)
[7]	Global Platform – Secure Element Access Control V1.0
[8]	ETSI TS 102 221 (V9.2.0 or later) – UICC-Terminal interface – Physical and logical characteristics
[9]	ETSI TS 102 613 (V9.3.0 or later) – UICC – Contactless Front-end (CLF) Interface – Part 1: Physical and data link layer characteristics
[10]	ETSI TS 102 622 (V9.4.0 or later) – UICC – Contactless Front-end (CLF) Interface – Host Controller Interface (HCI)
[11]	ETSI TS 102 694-1 – Test specification for the Single Wire Protocol (SWP) interface; Part 1: Terminal features
[12]	ETSI TS 102 695-1 (V9.3.0 or later) - Test specification for the Host Controller Interface (HCI); Part 1: Terminal features
[13]	ETSI TS 102 384 – Card Application Toolkit (CAT) conformance specification
[14]	OMA – OMA-TS-Smartcard_Web_Server
[15]	GCF WI – 35 – USAT Testing
[16]	GCF WI – 133 – SWP/HCI
[17]	GCF WI – 126 – BIP TCP Server Mode
[18]	GCF WI – 116 Smart Card Web Server
[19]	NFCForum-TS-Type-1-Tag_1.0 (or later) NFCForum-TS-Type-2-Tag_1.0 (or later) NFCForum-TS-Type-3-Tag_1.0 (or later) NFCForum-TS-Type-4-Tag_1.0 (or later)
[20]	3GPP TS 31.121 – UICC-terminal interface; Universal Subscriber Identity Module (USIM) application test specification
[21]	3GPP TS 31.124 – Mobile Equipment (ME) conformance test specification; Universal Subscriber Identity Module Application Toolkit (USAT) conformance test specification
[22]	ETSI TS 102 223 (V9.4.0 or later) – Smart Cards;Card Application Toolkit (CAT)
[23]	ETSI TS 102 226 (V9.6.0 or later) – Smart Cards ;Remote APDU structure for UICC based applications
[24]	ETSI TS 102 127 – Smart Cards;Transport protocol for CAT applications;Stage 2
[25]	3GPP TS 34.108 – Common test environments for User Equipment (UE); Conformance testing
[26]	GCF WI – 190 – SWP/HCI Enhancements for UICC Based NFC Services
[27]	GlobalPlatform – SEAC DeviceSide Test Plan v1.0.6
[28]	ISO/IEC 18092:2013 Information technology -- Telecommunications and information exchange between systems -- Near Field Communication -- Interface and Protocol (NFCIP-1)

1.5 Conventions

As per IETF Requirements terminology, reference RFC 2119, the following terms have the following meaning.

Term	Description
SHALL	Denotes a mandatory requirement
SHOULD	Denotes a recommendation
MAY	Denotes Optional

2 Test environment

2.1 Applicability

2.1.1 Format of the table of optional features

The columns in Table 4 have the following meaning:

Column	Meaning
Option :	The optional feature supported or not by the implementation.
Status :	See clause 2.1.3 ‘Status and Notations’
Support :	The support columns are to be filled in by the supplier of the implementation. The following common notations are used for the support column in table 2.1. Y supported by the implementation. N not supported by the implementation. N/A no answer required (allowed only if the status is N/A, directly or after evaluation of a conditional status).
Mnemonic :	The mnemonic column contains mnemonic identifiers for each item.

Table 1: Format of the table of optional features

2.1.2 Format of the applicability table

The applicability of every test in Table 5 is expressed by the use of Boolean expression defined in the following clause.

The columns in Table 5 have the following meaning:

Column	Meaning
Test case :	The “Test case” column gives a reference to the test case number(s) detailed in the present document and required to validate the implementation of the corresponding item in the “Description” column
Description :	In the “Description” column a short non-exhaustive description of the requirement is found.
Test case applicability	The “Test case applicability” column indicates which test cases are applicable per given Device Operating System, for the item in the “Description” column

Device Operating System :	The corresponding “Device Operating System” column lists the tests required for a Device to be declared compliant to this Test case.
Support :	The “Support” column is blank in the pro-forma, and is to be completed by the manufacturer in respect of each particular requirement to indicate the choices, which have been made in the implementation. If an optional requirement is supported then it must pass the corresponding tests.

Table 2: Format of the applicability table

2.1.3 Status and Notations

The “Device Operating System” columns show the status of the entries as follows:

The following notations are used for the status column:

Status	Description
M	Mandatory – the capability is required to be supported.
O	Optional – the capability may be supported or not.
N/A	Not Applicable – in the given context, it is impossible to use the capability.
TNR	Test Not Ready – Test case is not available for the referenced OS in this version of the Test Book
Ci	conditional – the requirement on the capability (“M”, “O” or “N/A”) depends on the support of other optional or conditional items. “”" is an integer identifying an unique conditional status expression which is defined immediately following the table. For nested conditional expressions, the syntax“"IF ... THEN (IF ... THEN ... ELSE...) ELSE .. »" is to be used to avoid ambiguities.

Table 3: Status and Notations

For each possible item answer (answer in the support column) there exists a unique reference, used, for example, in the conditional expressions. It is defined as the table identifier, followed by a solidus character“"/"", followed by the item number in the table. If there is more than one support column in a table, the columns are to be discriminated by letters (a, b, etc.), respectively.

EXAMPLE: A.1/4 is the reference to the answer of item 4 in table A.1.

2.1.4 Table of optional features

The supplier of the implementation shall state the support of possible options in Table 4 . See clause 2.1.1 for the format of Table 4. Items indicated as O_XYZ (for example, O_SCWS) refer to features supported by the device.

Item	Option	Status	Support	Mnemonic
1	void			
2	Support of LTE/IMS	O		O_LTE/IMS

Item	Option	Status	Support	Mnemonic
3	Support of LTE with fallback to 2G/3G	O		O_LTE/2G-3G
4	Support of read/write NFC Tag at distance > 1,0cm and ≤ 2,0cm	O		O_TAG_DISTANCE 2CM
5	Support of read/write NFC Tag at distance > 2,0cm and ≤ 3,0cm, see note 5	O		O_TAG_DISTANCE_3CM
6	Support of read/write NFC Tag at distance > 3,0cm and ≤ 4,0cm, see note 4	O		O_TAG_DISTANCE_4CM
7	Support of battery low mode, see note 2	O		O_BAT_LOW
8	Support of battery off mode, see notes 2 and 3	O		O_BAT_OFF
9	Support of multiple SE	O		O_MUL_SE
10	Support of Multiple APN	O		O_MULTI_APN
11	Terminal executes User confirmation phase before sending PDP context activation request	O		O_User_Confirm_Before_PDP_Context_Request
12	Support of Multiple Active CEEs model in Battery Operational Mode	O		O_MULTI_CEE_ON
13	Single Active Card Emulation Environment is supported	O		O_Single_Active_CEE
14	ISO_DEP route selection mechanism is provided by the DUT	O		O_ISO_DEP_ROUTE_SELECTION _
15	The NFC status is persistent across DUT power off and power on	O		O_NFC _PERSISTENCE

Note 1: In order to reflect current industry implementation, test cases with read/write distance > 1cm are optional for this version

Note 2: For options O_BAT_LOW and O_BAT_OFF the DUT shall support at least one of these options or both.

Note 3: If the device supports O_BAT_OFF, the device manufacturer must supply details of how to enter this state

Note 4: If option O_TAG_DISTANCE_4CM is supported, then O_TAG_DISTANCE_2CM and O_TAG_DISTANCE_3CM must be supported.

Note 5: If option O_TAG_DISTANCE_3CM is supported, then O_TAG_DISTANCE_2CM must be supported

Table 4: Options

2.1.5 Applicability Table

Table 5 specifies the applicability of each test case to the device under test. See clause 2.1.2 for the format of Table.

Test	Description	Test case applicability	Support

case		Android	Windows	BlackBerry	
3.3.3.1	NFC Forum Tag Type 1 – Read NFC Tag	M	M	M	
3.3.3.2	NFC Forum Tag Type 2 – Read NFC Tag	M	M	M	
3.3.3.3	NFC Forum Tag Type 3 – Read NFC Tag	M	M	M	
3.3.3.4	NFC Forum Tag Type 4 – Read NFC Tag	M	M	M	
3.3.3.5	NFC Forum Tag Type 1 – Write NFC Tag	M	M	M	
3.3.3.6	NFC Forum Tag Type 2 – Write NFC Tag	M	M	M	
3.3.3.7	NFC Forum Tag Type 3 – Write NFC Tag	M	M	M	
3.3.3.8	NFC Forum Tag Type 4 – Write NFC Tag	M	M	M	
3.3.3.9.1	Distance for NFC Tag Type 1 reading Test Sequence No 1: Distance for NFC Tag Type 1 Reading – 0,0cm	M	M	M	
3.3.3.9.2	Distance for NFC Tag Type 1 reading Test Sequence No 2: Distance for NFC Tag Type 1 Reading – 0,5cm	M	M	M	
3.3.3.9.3	Distance for NFC Tag Type 1 reading Test Sequence No 3: Distance for NFC Tag Type 1 reading – 1,0cm	M	M	M	
3.3.3.9.4	Distance for NFC Tag Type 1 reading Test Sequence No 4: Distance for NFC Tag Type 1 Reading – 2,0cm	C015	C015	C015	
3.3.3.9.5	Distance for NFC Tag Type 1 reading Test Sequence No 5: Distance for NFC Tag Type 1 Reading – 3,0cm	C016	C016	C016	
3.3.3.9.6	Distance for NFC Tag Type 1 reading Test Sequence No 6: Distance for NFC Tag Type 1 Reading – 4,0cm	C017	C017	C017	

Test case	Description	Test case applicability			Support
3.3.3.10.1	Distance for NFC Tag Type 2 reading Test Sequence No 1: Distance for NFC Tag Type 2 Reading – 0,0cm	M	M	M	
3.3.3.10.2	Distance for NFC Tag Type 2 reading Test Sequence No 2: Distance for NFC Tag Type 2 Reading – 0,5cm	M	M	M	
3.3.3.10.3	Distance for NFC Tag Type 2 reading Test Sequence No 3: Distance for NFC Tag Type 2 reading – 1,0cm	M	M	M	
3.3.3.10.4	Distance for NFC Tag Type 2 reading Test Sequence No 4: Distance for NFC Tag Type 2 Reading – 2,0cm	C015	C015	C015	
3.3.3.10.5	Distance for NFC Tag Type 2 reading Test Sequence No 5: Distance for NFC Tag Type 2 Reading – 3,0cm	C016	C016	C016	
3.3.3.10.6	Distance for NFC Tag Type 2 reading Test Sequence No 6: Distance for NFC Tag Type 2 Reading – 4,0cm	C017	C017	C017	
3.3.3.11.1	Distance for NFC Tag Type 3 reading Test Sequence No 1: Distance for NFC Tag Type 3 Reading – 0,0cm	M	M	M	
3.3.3.11.2	Distance for NFC Tag Type 3 reading Test Sequence No 2: Distance for NFC Tag Type 3 Reading – 0,5cm	M	M	M	
3.3.3.11.3	Distance for NFC Tag Type 3 reading Test Sequence No 3: Distance for NFC Tag Type 3 reading – 1,0cm	M	M	M	
3.3.3.11.4	Distance for NFC Tag Type 3 reading Test Sequence No 4: Distance for NFC Tag Type 3 Reading – 2,0cm	C015	C015	C015	
3.3.3.11.5	Distance for NFC Tag Type 3 reading Test Sequence No 5: Distance for NFC Tag Type 3 Reading – 3,0cm	C016	C016	C016	
3.3.3.11.6	Distance for NFC Tag Type 3 reading Test Sequence No 6: Distance for NFC Tag Type 3 Reading – 4,0cm	C017	C017	C017	
3.3.3.12.1	Distance for NFC Tag Type 4A reading Test Sequence No 1: Distance for NFC Tag Type 4A Reading – 0,0cm	M	M	M	

Test case	Description	Test case applicability			Support
3.3.3.12.2	Distance for NFC Tag Type 4A reading Test Sequence No 2: Distance for NFC Tag Type 4A Reading – 0,5cm	M	M	M	
3.3.3.12.3	Distance for NFC Tag Type 4A reading Test Sequence No 3: Distance for NFC Tag Type 4A reading – 1,0cm	M	M	M	
3.3.3.12.4	Distance for NFC Tag Type 4A reading Test Sequence No 4: Distance for NFC Tag Type 4A Reading – 2,0cm	C015	C015	C015	
3.3.3.12.5	Distance for NFC Tag Type 4A reading Test Sequence No 5: Distance for NFC Tag Type 4A Reading – 3,0cm	C016	C016	C016	
3.3.3.12.6	Distance for NFC Tag Type 4A reading Test Sequence No 6: Distance for NFC Tag Type 4A Reading – 4,0cm	C017	C017	C017	
3.3.3.13.1	Distance for NFC Tag Type 4B reading Test Sequence No 1: Distance for NFC Tag Type 4B Reading – 0,0cm	M	M	M	
3.3.3.13.2	Distance for NFC Tag Type 4B reading Test Sequence No 2: Distance for NFC Tag Type 4B Reading – 0,5cm	M	M	M	
3.3.3.13.3	Distance for NFC Tag Type 4B reading Test Sequence No 3: Distance for NFC Tag Type 4B reading – 1,0cm	M	M	M	
3.3.3.13.4	Distance for NFC Tag Type 4B reading Test Sequence No 4: Distance for NFC Tag Type 4B Reading – 2,0cm	C015	C015	C015	
3.3.3.13.5	Distance for NFC Tag Type 4B reading Test Sequence No 5: Distance for NFC Tag Type 4B Reading – 3,0cm	C016	C016	C016	

Test case	Description	Test case applicability			Support
3.3.3.13.6	Distance for NFC Tag Type 4B reading Test Sequence No 5: Distance for NFC Tag Type 4B Reading – 4,0cm	C017	C017	C017	
3.3.3.14	NFC Tag type 1 reading performance	M	M	M	
3.3.3.15	NFC Tag type 2 reading performance	M	M	M	
3.3.3.16	NFC Tag type 3 reading performance	M	M	M	
3.3.3.17	NFC Tag type 4A reading performance	M	M	M	
3.3.3.18	NFC Tag type 4B reading performance	M	M	M	
3.3.3.19	NFC Tag handling during an active data transfer.	M	M	M	
3.4.3.1.1	Card Emulation enabled as soon as NFC hardware is on Test Sequence No.1	M	M	M	
3.4.3.1.2	Card Emulation enabled as soon as NFC hardware is on Test sequence No 2	C014	C014	C014	
3.4.3.2	NFC during Standby time	C005	C005	C005	
3.4.3.3.1	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 1: Card Emulation Mode Type A in Battery Power Off mode	C006	C006	C006	
3.4.3.3.2	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 2: Card Emulation Mode Type B in Battery Power Off mode	C006	C006	C006	

Test case	Description	Test case applicability			Support
3.4.3.3.4	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 4: Card Emulation Mode Type A in Battery Low Mode	C005	C005	C005	
3.4.3.3.5	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 5: Card Emulation Mode Type B in Battery Low Mode	C005	C005	C005	
3.4.3.4	Distance for card emulation	M	M	M	
3.4.3.5	Distance for card emulation in Battery Power-off Mode(0cm)	C006	C006	C006	
3.4.3.6	Distance for card emulation in Battery Power-off Mode (0.5cm)	C006	C006	C006	
3.4.3.7	Distance for card emulation in Battery Power-off Mode (1cm)	C006	C006	C006	
3.4.3.8	Distance for card emulation in Battery Power-off Mode (1.5cm)	C006	C006	C006	
3.4.3.9	Distance for card emulation in Battery Power-off Mode (2cm)	C006	C006	C006	
3.4.3.10	Distance for card emulation in Battery Power-low Mode(0cm)	C005	C005	C005	
3.4.3.11	Distance for card emulation in Battery Power-low Mode (0.5cm)	C005	C005	C005	
3.4.3.12	Distance for card emulation in Battery Power-low Mode (1cm)	C005	C005	C005	
3.4.3.13	Distance for card emulation in Battery Power-low Mode (1.5cm)	C005	C005	C005	
3.4.3.14	Distance for card emulation in Battery Power-low Mode (2cm)	C005	C005	C005	
3.4.3.15	Distance for card emulation in Battery Power- operational Mode(0cm)	M	M	M	

Test case	Description	Test case applicability			Support
3.4.3.16	Distance for card emulation in Battery Power- operational Mode (0.5cm)	M	M	M	
3.4.3.17	Distance for card emulation in Battery Power- operational Mode (1cm)	M	M	M	
3.4.3.18	Distance for card emulation in Battery Power- operational Mode (1.5cm)	M	M	M	
3.4.3.19	Distance for card emulation in Battery Power- operational Mode (2cm)	M	M	M	
3.5.3.1	SWP Compliance testing	M	M	M	
3.5.3.2	HCI Compliance testing	M	M	M	
3.5.3.3	SWP Stress test	M	M	M	
3.5.3.4	Switch mode	M	M	M	
3.5.3.5	RF Protocol compliance	M	M	M	
5.3.1	GP SE Access Control	M	TNR	TNR	
5.3.2	GP SE Access Control–- Refresh tag	M	TNR	TNR	
5.3.3	GP SE Access Control – ADF_PKCS#15 and DF PKCS#15	M	TNR	TNR	
5.3.4	GP SE Access Control – PKCS#15 selection via EF_DIR	M	TNR	TNR	
5.3.5	GP SE Access Control – Configuration limits	M	TNR	TNR	
5.3.6	GP SE Access Control – No access	M	TNR	TNR	
5.4	GP SE Access Control – GP Test Plan	M	M	M	
6.3.1	SIMalliance APIs	M	N/A	TNR	
7.3.2	SE selection	C007	C007	C007	
7.3.6	Switch for Card Emulation Mode	C012	TNR	TNR	
7.3.7	Multiple CE Environments	C019	N/A	N/A	
7.3.8.1	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 1	C018	C018	C018	
7.3.8.2	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 2	C018	C018	C018	
7.3.8.3	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 3	C019	C019	C019	

Test case	Description	Test case applicability			Support
7.3.8.4	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 4	C019	C019	C019	
7.3.8.5	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 5	C019	C019	C019	
7.3.8.6	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 6	C019	C019	C019	
8.3.1	EVT_TRANSACTION	M	M	M	
8.3.2	Permissions	M	N/A	N/A	
8.3.3	Intent management	M	N/A	N/A	
8.3.4	Application's triggering order	TNR	TNR	TNR	
8.3.5	Triggering on HCI event EVT_CARD_DEACTIVATED	M	TNR	TNR	
8.3.6	Triggering on HCI event EVT_FIELD_OFF	M	TNR	TNR	
11.3.1.1	OPEN CHANNEL Test Sequence 1: (OPEN CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN)	C008	C008	C008	
11.3.1.2	OPEN CHANNEL 11.3.1.2 Test Sequence 2: (OPEN CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN)	C009	C009	C009	
11.3.1.3	OPEN CHANNEL Test Sequence 3: (OPEN CHANNEL – Default APN Always-ON – APN empty)	M	M	M	
11.3.1.4	OPEN CHANNEL Test Sequence No 4: (OPEN CHANNEL – Default APN Always-ON – APN empty- Default Bearer Type used)	M	M	M	
11.3.2.1	CLOSE CHANNEL Test Sequence 1: (CLOSE CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN- SUCCESSFUL)	C008	C008	C008	

Test case	Description	Test case applicability			Support
11.3.2.2	CLOSE CHANNEL Test Sequence 2: (CLOSE CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN- SUCCESSFUL)	C009	C009	C009	
11.3.2.3	CLOSE CHANNEL Test Sequence 3: (CLOSE CHANNEL – Default APN Always-ON – APN empty- SUCCESSFUL)	M	M	M	
11.3.2.4	CLOSE CHANNEL Test Sequence No 4: (CLOSE CHANNEL – Default APN Always-ON – APN empty- SUCCESSFUL- Default Bearer Type Used)	M	M	M	
11.3.3.1	RECEIVE DATA Test Sequence 1: (RECEIVE DATA – Default APN Always-ON – Multiple APN supported – with different APN)	C008	C008	C008	
11.3.3.2	RECEIVE DATA Test Sequence 2: (RECEIVE DATA – Default APN Always-ON – Only Single APN supported – with different APN)	C009	C009	C009	
11.3.3.3	RECEIVE DATA Test Sequence 3: (RECEIVE DATA – Default APN Always-ON – APN empty)	M	M	M	
11.3.3.4	RECEIVE DATA Test Sequence No 4: (RECEIVE DATA –Default APN Always-ON – APN empty-Default Bearer Type used)	M	M	M	
11.3.4.1	SEND DATA Test Sequence 1: (SEND DATA – Default APN Always-ON – Multiple APN supported –with different APN- BUFFER FULLY USED)	C008	C008	C008	
11.3.4.2	SEND DATA Test Sequence 2: (SEND DATA – Default APN Always-ON – Only Single APN supported – with different APN- BUFFER FULLY USED)	C009	C009	C009	
11.3.4.3	SEND DATA Test Sequence 3: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED)	M	M	M	

Test case	Description	Test case applicability			Support
11.3.4.4	SEND DATA Test Sequence No 4: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED- Default Bearer Type used)	M	M	M	
12.3.3.1	Remote management in BIP	M	M	M	
12.3.3.2.1	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, No APN, immediate link establishment, Default Bearer for requested transport layer, No local address, no alpha identifier)	M	M	M	
12.3.3.2.2	OPEN CHANNEL Test sequence No : (OPEN CHANNEL, with APN, immediate link establishment, Default Bearer for requested transport layer, no alpha identifier)	M	M	M	
12.3.3.2.3	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, with alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.4	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, with null alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.5	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, command performed with modifications (buffer size), immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.6	OPEN CHANNEL Test Sequence No 6A : (OPEN CHANNEL, user rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier,)	C010	C010	C010	
12.3.3.2.7	OPEN CHANNEL Test Sequence No 6B : (OPEN CHANNEL, User rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier)	C011	C011	C011	
12.3.3.3	CLOSE CHANNEL	M	M	M	

Test case	Description	Test case applicability			Support
12.3.3.4	RECEIVE DATA	M	M	M	
12.3.3.5	SEND DATA	M	M	M	
12.3.3.6	GET CHANNEL STATUS	M	M	M	
12.3.3.7	Data available event	M	M	M	
12.3.3.8	Channel Status event	M	M	M	
12.3.3.10	Concurrent BIP channels	M	M	M	
12.3.3.11	Contents of the TERMINAL PROFILE	M	M	M	
12.4.3.1	Contactless transaction during BIP session	M	M	M	
12.4.3.2.1	Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.2	Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network only	C002	C002	C002	
12.4.3.2.3	Voice Call made from the device during BIP CAT-TP session on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.4	Voice Call made from the device during BIP CAT-TP session on LTE network only	C002	C002	C002	
12.4.3.2.5	BIP CAT-TP data transfer during a Voice Call is established on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.6	BIP CAT-TP data transfer during a Voice Call is established on LTE network only	C002	C002	C002	
12.4.3.3.1	Test Sequence No 1 : OTA data Loading without PoR requested by OTA server	M	M	M	
12.4.3.3.2	Test Sequence No 2 : OTA data Loading with PoR requested by OTA server	M	M	M	
12.4.3.3.3	Test Sequence No 3 : OTA data Loading with PoR requested by OTA server when the device is booting	M	M	M	

Test case	Description	Test case applicability			Support
12.4.3.3.4	Test Sequence No 4 : OTA data Loading with PoR requested by OTA server after the device comes back from airplane mode.	M	M	M	
12.4.3.3.5	Test Sequence No 5 : OTA data Loading with PoR requested by OTA server only on error	M	M	M	
12.4.3.4	Secure Element Access during BIP session	M	M	M	
12.4.3.5.	SMS and Internet Connection during OTA data Loading	M	M	M	
13.3.1	Secure Element Access API & Access control in Radio Off State	M	TNR	TNR	
13.3.2	Enabled / Disabled states	M	M	M	
15.3.3.1	Shared library	M	N/A	N/A	
15.3.3.2	Getting an instance of the “NFC controller” class	M	N/A	N/A	
15.3.3.3	Getting the status of the NFC Controller	M	N/A	N/A	
15.3.3.4	Enabling the NFC Controller	M	N/A	N/A	
15.5.3.1	Switching to “Multicast” Mode	M	N/A	N/A	
15.7.3.1	Dynamic AIDs Registration APIs when Multiple Active CEEs model is supported	C018	N/A	N/A	
15.7.3.2	APIs for handling Dynamic AID Registration	M	N/A	N/A	

Table 5: Applicability of tests

Conditional item	Condition
C001	void
C002	IF (O_LTE/IMS) THEN M ELSE N/A
C003	IF (O_LTE/2G-3G) THEN M ELSE N/A
C004	IF (O_EXT_RW_DISTANCE) THEN M ELSE N/A
C005	IF (O_BAT_LOW) THEN M ELSE N/A

Conditional item	Condition
C006	IF (O_BAT_OFF) THEN M ELSE N/A
C007	IF (O_MUL_SE) THEN M ELSE N/A
C008	IF (O_MULTI_APN) THEN M ELSE N/A
C009	IF (NOT_O_MULTI_APN) THEN M ELSE N/A
C010	IF (O_User_Confirm_Before_PDP_Context_Request) THEN M ELSE N/A
C011	IF NOT (O_User_Confirm_Before_PDP_Context_Request) THEN M ELSE N/A
C012	IF (O_ Single_Active_CEE) THEN M ELSE N/A
C013	void
C014	IF (O_NFC _PERSISTENCE) THEN M ELSE N/A
C015	IF (O_TAG_DISTANCE_2CM) THEN M ELSE N/A
C016	IF (O_TAG_DISTANCE_3CM) THEN M ELSE N/A
C017	IF (O_TAG_DISTANCE_4CM) THEN M ELSE N/A
C018	IF (O_MULTI_CEE_ON) THEN M ELSE N/A
C019	IF (O_MULTI_CEE_ON AND O_ISO_DEP_ROUTE_SELECTION) THEN M ELSE N/A

Table 6: Conditional items referenced by Table 5

2.2 General consideration

For the purpose of the test execution and unless specified, the UICC is the active Secure Element by default and the Access Control configuration provides full access to any AIDs from any mobile applications.

Test descriptions are independent.

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

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

2.2.1 Test specifications

The GSMA NFC Handset Test Book refers to test specifications developed by other organisations (EMVCo, ISO, ETSI, 3GPP, OMA). These organisations defined their own requirements for test benches, test applicability and pass criteria's.

The GSMA fully rely on these test specification for the purpose of the GSMA NFC Handset Test Book and requires these test to be performed. In the scope of the GSMA evaluation a list of tests will have to be conducted and are listed in Annex D.

2.2.2 SIMalliance Open Mobile API

The SIMalliance Open Mobile API specification is defined in an object oriented language-manner and may not be applicable for some OS platforms. Therefore, this Test Book is based on the SIMalliance specification for test steps description and pass criteria.

The mapping from Open Mobile API errors to Java based exceptions shall be as follows:

SIMalliance error	Java based exception
IOError	java.io.IOException
SecurityError	java.lang.SecurityException
NoSuchElementError	java.util.NoSuchElementException
IllegalStateError	java.lang.IllegalStateException
IllegalParameterError	java.lang.IllegalArgumentException

Table 7: Mapping from Open Mobile API errors to Java based exceptions

2.2.3 Pass criterion

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

A test execution is considered as failed if the tested feature provides an unexpected behaviour.

A test execution is considered as non-conclusive when the pass criteria cannot be evaluated due to issues during the setup of the initial conditions.

2.2.4 Future study

Some of the test cases described in this Test Book are FFS (For Future Study). This means that some clarifications are expected at the requirement level to conclude on a test method.

2.3 Tests with measurement and physical settings

Part of this testing refers to measurement or physical positions:

- Transaction duration measurement
- Power consumption measurement
- Distance between the DUT and a NFC tag or a contactless reader (reader and target are centred each other).

For test cases relative to these characteristics, all relevant information to allow identifying the severity of detected issues must be added in the test report.

2.4 Reference Transaction

To ascertain correct implementation by the DUT of the card emulation mode as described [1], a reference transaction will be used.

The **reference transaction** is executed using a contactless reader as follow:

The transaction always starts with putting DUT into reader RF field. Then the reader establishes the ISO14443 connection with the DUT. Afterwards following APDUs will be exchanged.

- Select by AID A0000000FF50000000000000052414441
- Select by File ID (1F00)
- Read Binary
- Update Binary (with 128 bytes with value 0xFF)
- Read Binary (again)
- Update Binary (with 128 bytes with value 0x00)
- External Authenticate

The transaction always ends with a DESELECT as per ISO14443 specification and finally the removal of DUT from reader RF field.

For this purpose, a UICC application will be used as a part of the test equipment.

Annex A of this document proposes a description of the application and its corresponding source code. In case of the simulated UICC the complete behaviour of this referenced application shall be simulated. The parts related to each single test shall be simulated according to the description given in the specific test case.

2.5 Test Equipment

This chapter aims at describing different test tools for evaluation of the subsequent test packages. Names assigned to these applications are also used in the test cases descriptions.

Implementation of these applications remains the responsibility of the provider. Nevertheless, a description of the test equipment used for testing (brand name, model name and version) will be provided as a part of the test report.

The .cap files mentioned within this document provides description of the UICC behaviour which can be either simulated or real UICC. The simulation of the behaviour remains language-independent. The test equipment/case manufacturer could use other means to gain the same behaviour as specified in the Java .cap files.

2.5.1 UICC

For all the tests described in this GSMA NFC Handset Test Book, a UICC must be used. For most of the test sequences described in this document the UICC have in important role in the test bench and should be managed by Test Labs as test tool.

The test environment can be implemented via use of real UICCs or via simulated environment for UICCs.

The following terms for test environment are used:

Real UICC: A real UICC is used during testing. Typically this is physically available UICCs provided by UICC manufacturers.

Simulated UICC: The UICC is emulated with a simulator which provides corresponding functionalities as a valid UICC.

In order to ensure best possible traceability and reproducibility of test results, the following sections define requirements for the different test environments.

2.5.1.1 Requirements for UICC environment

If the test cases in this NFC Handset Test Book is implemented using UICCs, the requirements for test environment described in this section shall be fulfilled.

The UICC (simulated or real) shall act as a valid UICC according to the following specifications:

- [8]: ETSI TS 102 221:"Smart Cards; UICC-Terminal interface; Physical and logical characteristic".
- [9]: ETSI TS 102 613:"Smart Cards; UICC–- Contactless Front-end (CLF) Interface; Part 1: Physical and data link layer characteristic".
- [10]: ETSI TS 102 622:"Smart Cards; UICC–- Contactless Front-end (CLF) Interface; Host Controller Interface (HCI".

In particular, during test procedure execution, the UICC shall respect the electrical and signalling conditions for all UICC contacts within the limits given by TS 102 613 [9], TS 102 221 [8] and TS 102 622 [10]). The accuracy of the UICC simulator's settings shall be taken into account when ensuring this.

The UICC shall be connected to the device under test (DUT) and shall provide functionalities specified below:

- Shall support card emulation, reader and connectivity gates as specified in [10].
- Shall support card emulation in both full power mode and low power mode.
- Shall support CLT mode in full power mode and in low power mode, as specified in ETSI TS 102 613[9] and ETSI TS 102 622[10].
- Shall support GlobalPlatform Secure Element Access Control both for ARA and ARF mechanism
- Shall support BIP and APN as specified in [21]
- Shall provide all necessary information (Specification, ADM codes) to manage the card content and the file system

The UICC simulator shall implement the following functionalities when required in the test cases in which the UICC simulator is used:

- Shall fulfil the requirements for SWP/HCI as specified in ETSI TS 102 694-1clause 4.4, and ETSI TS 102 695-1 clause 4.4
- Shall fulfil the requirements for Remote Management of NFC Services and for Mobile Device APN as specified in TS 31.121 [20] clause 4.1 and in TS 31.124 [21] in 27.22.2A, 27.22.2B and 27.22.2C.

2.5.1.2 UICC Form Factor

All UICC form factors, as specified in ETSI TS 102 221 [8] chapter 4.0; shall be provided by the simulated and real UICC environment.

2.5.2 Requirements for UMTS Network Simulator

For Remote Management of NFC Services and Mobile Device APN Management test execution, the test equipment shall fulfil the requirements specified in 3GPP TS 34.108 [25] clause 4.

2.5.3 Common applications

The following applications are common to different test packages.

APDU application: A software application running on a PC connected to a contactless reader. This application will be used to send C-APDU to the DUT and get the corresponding R-APDU.

ReferenceApplication.cap: A UICC application according to Annex A of this document. This application will be used to run the **reference transaction**.

MobileApplication: A mobile application allowing the following access to the UICC:

- Open Basic Channel
- Open Logical Channel via Select AID
  - SELECT_BY_DF_name on AID1
- Open Logical Channel via Manage Channel
  - Manage_Channel_Open to open another channel than channel 1
- Send APDU Case 1 => 0x0001[P1]00
  - Nominal expected response is SW1-SW2
- Send APDU Case 2 => 0x0002[P1]0000
  - Nominal expected response is [Data field of 0xFF bytes long] only if SW1 = 0x62 or 0x63 or 0x90 + SW1-SW2
- Send APDU Case 3 => 0x0003[P1]00FF [Data field of 0xFF bytes long]
  - Nominal expected response is SW1-SW2
- Send APDU Case 4 => 0x0004[P1]00FF [Data field of 0xFF bytes long] FF
  - Nominal expected response is [Data field of 0xFF bytes long] only if SW1 = 0x62 or 0x63 or 0x90 + SW1-SW2

Additionally the application will allow sending APDUs with all the other Class Instruction pairs [CLAINS] from 0x0000 to 0xFEFF excluding

- INS = 0x70, 0x6x, 0x9x for all CLA

- Send all CLA/INS pairs => 0x[CLAINS]000010 [Data field of 0x10 bytes long]
- Nominal expected response is [Data field of 0x10 bytes long] + SW1-SW2
- [P1] identifies the sub case.
- When not specified in the test case, [P1] equals 0x00 meaning default SW1-SW2 is 90 00.

For testing purpose, 2 or 3 occurrences of the application will be created

- **GSMA_Mobile_App_SP1_signed** signed with a private key corresponding to test certificate #1
- **GSMA_Mobile_App_SP2_signed** signed with a private key corresponding to test certificate #2
- **GSMA_Mobile_App_Unsigned** unsigned if managed by the DUT OS
- **MobileApplication** is considered as **launched** if it is selected and started by the User.
- **APDU_TestApplication.cap** providing the following features:
- Based on **ReferenceApplication.cap,** this application allows managing different APDU answers. The application sends EVT_TRANSACTION on the following events
- EVT_FIELD_OFF
- A modified version of the **APDU_TestApplication.cap is the APDU_TestApplication_card_deactivated.cap**.

The application sends EVT_TRANSACTION only on the following events

- EVT_CARD_DEACTIVATED
- Additionally, **APDU_TestApplication.cap** implements the sequence used by the **MobileApplication**
- On APDU Case 1 => 0x0001[P1]00
    - returns SW1-SW2
- On APDU Case 2 => 0x0002[P1]00FF
    - returns [Data field of 0xFF bytes long] only if SW1 = 0x62 or 0x63 or 0x90 + SW1-SW2
- On APDU Case 3 => 0x0003[P1]00FF [Data field of 0xFF bytes long]
    - returns SW1-SW2
- On APDU Case 4 => 0x0004[P1] 00FF [Data field of 0xFF bytes long and expected data length is 0xFF]
    - returns [Data field of 0xFF bytes long] only if SW1 = 0x62 or 0x63 or 0x90 + SW1-SW2

Depending of [P1] in the APDU command; the application will return the corresponding SW1-SW2.

[P1]	SW1-SW2
0x01	0x6200
0x02	0x6202
0x03	0x6280
0x04	0x6281
0x05	0x6282
0x06	0x6283
0x07	0x6284
0x08	0x6285
0x09	0x6286
0x0A	0x62F1
0x0B	0x62F2
0x0C	0x6300
0x0D	0x6381
0x0E	0x63C2
0x0F	0x6310
0x10	0x63F1
0x11	0x63F2
0x12	0x6400
0x13	0x6401
0x14	0x6402
0x15	0x6480
0x16	0x6500
0x17	0x6581
0x18	0x6800
0x19	0x6881
0x1A	0x6882
0x1B	0x6883
0x1C	0x6884
0x1D	0x6900
0x1E	0x6900
0x1F	0x6981
0x20	0x6982
0x21	0x6983
0x22	0x6984
0x23	0x6985
0x24	0x6986
0x25	0x6987
0x26	0x6988

[P1]	SW1-SW2
0x27	0x6A00
0x28	0x6A80
0x29	0x6A81
0x2A	0x6A82
0x2B	0x6A83
0x2C	0x6A84
0x2D	0x6A85
0x2E	0x6A86
0x2F	0x6A87
0x30	0x6A88
0x31	0x6A89
0x32	0x6A8A

Table 8: Status Word

Based on **APDU_TestApplication.cap, APDU_TestApplication_SW6283.cap** with AID1 answering to the SELECT_BY_DF_name command with a non-empty application FCI and Status Word 6283.

Based on **APDU_TestApplication.cap, APDU_TestApplication_SW6999.cap** with AID1 answering to the SELECT_BY_DF_name command with Status Word 6999.

Based on **APDU_TestApplication.cap, APDU_TestApplication_CLAINS.cap** with AID1 handling all **CLAINS** from 0x0000 to 0xFEFF excluding INS = 0x70, 0x6x, 0x9x for all CLA

The APDU is as follow

- On [CLAINS]000010 [Data field of 0x10 bytes long] with the following answer [Data field of 0x10 bytes long] + 90 00

Based on **APDU_TestApplication.cap, APDU_TestApplication_SW61XX.cap** with AID1 answering to the Case 2 and Case 4 commands with Status Word 61XX.

2.5.3.1 Logically

The reference PKCS#15 structures are using the following AID-s:

AID1 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41’

AID2 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42’

AID3 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 43’

2.5.4 Tag Testing

The test environment described in this GSMA NFC Handset Test Book can be implemented to use real Tags or simulated Tags.

The following terms for test environment are used:

Real Tags: A real Tag is used during testing. Typically this is physically available Tag provided by Tags manufacturers. A list of reference Real Tags are defined in Annex C.

Simulated Tags: The Tag is emulated with a simulator which provides corresponding functionalities as the specified by NFC Forum. It is provided by test tool manufacturers.

2.5.4.1 Common positioning of Device and Tag

A number of the test cases require the use of a Tag which shall be positioned relative to the DUT. Contactless communication between the device and the Tag is part of the verdict evaluation of the test cases. Therefore it is essential that a minimum set of positions are defined in order to ensure the test cases are executed in a reproducible way.

The following are definitions for DUT and Tag:

- DUT antenna reference point:
  - This is the position on the DUT which will provide the optimal performance of the NFC antenna. This point is provided by the device manufacturer. The reference point shall be marked on the outside cover of the device.
- Tag antenna reference point:
  - This is the position at the Tag where the antenna performance is optimal. For a real Tag this point is provided by the Tag vendor or measured by the test laboratory. For a reader/listener antenna, the point is provided by the vendor of the antenna.

Positioning of DUT and Tag for test cases where there is no requirement to the distance between DUT and Tag, the DUT and Tag are positioned as follows:

- The DUT and Tag are placed with their antenna reference points located as close as possible to each other as to the form factor of DUT will allow.
- The DUT and Tag are positioned both in a vertical position as default position. I.e. with a traditional DUT form factor and a Tag with ID1 form factor, the positioning will be as below:

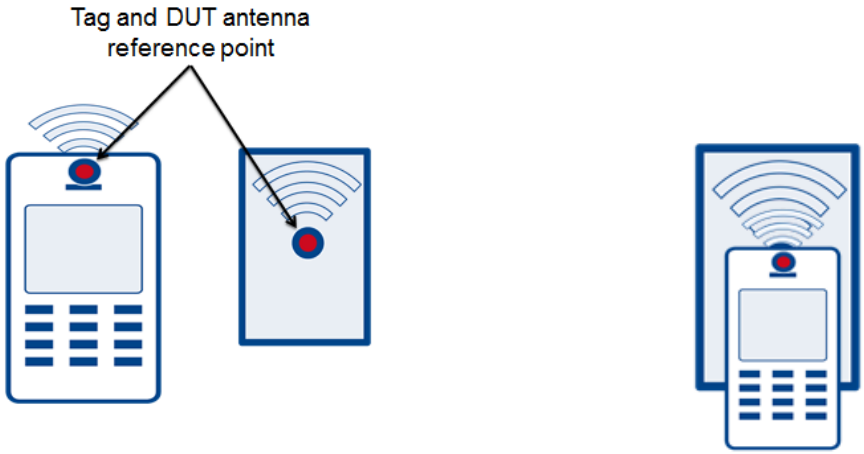

Figure 3: Tag and DUT antenna reference point

- The DUT and Tag is positioned in parallel plans as possible due to form factor of the DUT. Ideally the position will looks like:

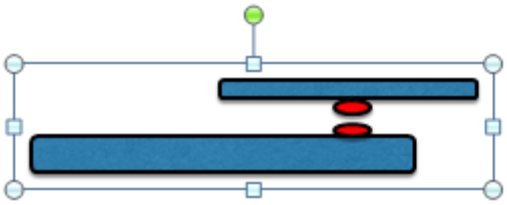

Figure 4: Antenna positioning

- The positioning shall provide optimal antenna coupling between DUT and Tag.

The following conditions shall be fulfilled to limit the impact of external noise by executing all contactless tests in the present test specification:

The external interferences sources:

Metal objects or other any other interference elements shall be kept at least 15cm from the Test System.

Any magnetic field shall not be present in a volume of 1 meter around the Test System; e.g. no other antennas, contactless terminals, cell phones, etc.

The DUT and the Tag must be placed so that the radio communication can correctly take place.

2.5.4.2 Distance specific positioning

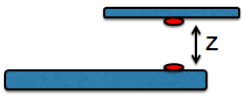

Figure 5: “z” distance

For the test cases specifying exact distance between DUT and Tag, the distance is the vertical distance between DUT and Tag antenna reference points. The following distances are used during distance testing:

- z = 0,0cm
- z = 0,5cm
- z = 1,0cm
- z = 2,0cm
- z = 3,0cm
- z = 4,0cm

The distance setting accuracy: +/- 0,05cm

The distance z is measured from the device outside cover to the Tag independent if the antenna is located inside the DUT.

For test cases not specifying a distance between DUT and Tag, the default distance z = 0,0cm between DUT and Tag antenna reference point.

2.5.4.3 Tag requirements

NFC Forum Tag Type 1:

- Provide the functionality specified in NFCForum TS Type 1 Tag [19]

NFC Forum Tag Type 2:

- Provide the functionality specified in NFCForum TS Type 2 Tag [19]

NFC Forum Tag Type 3:

Provide the functionality specified in NFCForum TS Type 3 Tag [19]

NFC Forum Tag Type 4A:

- Provide the functionality specified in NFCForum TS Type 4 Tag [19]

NFC Forum Tag Type 4B:

Provide the functionality specified in NFCForum TS Type 4 Tag [19]

2.5.4.4 Applications

The following applications are dedicated to NFC tag relative test cases.

NFC Tag application: An external tag reader and writer application for tag content verification purpose.

NFC Tag mobile application: A mobile application based on the operating system standardized APIs for tag reading and writing.

Reference NFC Tags: A set of reference NFC tags (Type 1, 2, 3 and 4)

2.5.4.5 Reference NFC tag content

The following NFC Tag content will be used when not otherwise specified

Reference	NFC Tag Content
“vCard” see note 2)	Type: “text/x-vCard” BEGIN: VCARD VERSION: 2.1 N: ;John Smith;;; FN: John Smith TEL;CELL; 332312345678 END: VCARD
“URI” see note 1)	Type: “U” file://test
“Text” see note 1)	Type: “T”

Reference	NFC Tag Content
	"Hello, world!"
"SmartPoster" (launch browser) see note 2)	Type: "Sp" Text Type: "T" Encoding: UTF-8 Lang: "" Test: "GSMA Website" URI Type: "U" http://www.gsma.com
"SmartPoster" (SMS Sending) see note 1)	Type: "Sp" URI Type: "U" sms:332312345678?body=Hello, world!
"SmartPoster" (phone call) see note 2)	Type: "Sp" Text Type: "T" Encoding: UTF-8 Lang: "" Test: "John Smith" URI Type: "U" Tel: 442312345678
"SmartPoster" (email) see Note see note 2)	Type: "Sp" URI Type: "U" mailto:john.smith@gsma.com?subject=email subject&body=email content Text Type: "T" Encoding: UTF-8 Lang: "en" Test: "email title"

Table 9: NFC Tags content

Note 1: This Tag shall represent static memory layout for NFC Tag type 1 and type 2 only.

Note 2: This Tag shall represent dynamic memory layout for NFC Tag type 1 and type 2 only.

2.5.5 Access Control Test bench

The following test description applies to test packages evaluating the Access Control mechanism and application management.

The test bench consists in a single mobile application provided with different certificates to ensure the DUT manages signatures by different service providers.

Test package will consist in managing the PKCS#15 structure inside the UICC to ensure the access control rights are granted or not.

Two instances of the UICC application **APDU_TestApplication.cap** with AID1 and AID2 are available for Access Control testing.

For that purpose, Test Labs will use the **MobileApplication** registered for EVT_TRANSACTION handling from AID1 and AID2 and implementing the following functions:

- “Select AID1” function: sends SELECT command with AID1 to the UICC
- “Select AID2” function: sends SELECT command with AID2 to the UICC

The application is duplicated with different signature configurations as it is specified above in sec. 2.5.2 “Common Applications”and respectively named

- GSMA_AC_Mobile_App_SP1_signed
- GSMA_AC_Mobile_App_SP2_signed

2.5.6 Reader equipment

The contactless reader shall support the ISO/IEC 14443 series for type A and B functionality

2.5.7 NFC Controller and UI application triggering

For NFC Controller and UI application triggering, specific test applications will be defined in the initial conditions of the tests.

Unless otherwise specified, when EVT_TRANSACTION is used for triggering a certain application, the event shall be received by the application within 30 seconds from the point that this event has been sent by the UICC. In the case where no application is expected to receive the event, the test tool shall wait for 60 seconds.

(Note – These times are specified for this version of the test book for test implementation purposes. Normative times are expected to be defined by the TSG NFC Handset Requirement Group in TS.26 version 7.0)

2.5.8 Test Set-Up for OTA communication

A real OTA Platform connected to the network’s backend communicates through the Radio Access Network and the Device with the UICC.

To allow for testing in a lab environment, some of the real world components may be replaced by simulations:

- OTA Server may be replaced by a software simulation.
- Radio Access Network may be replaced by a system simulator.
- UICC may be replaced by a simulated UICC.

Such a setup does not require any Internet or Intranet connection. It allows for deep diagnosis insights into all involved components. It also enables manipulation of any of the components, e.g. for failure simulation.

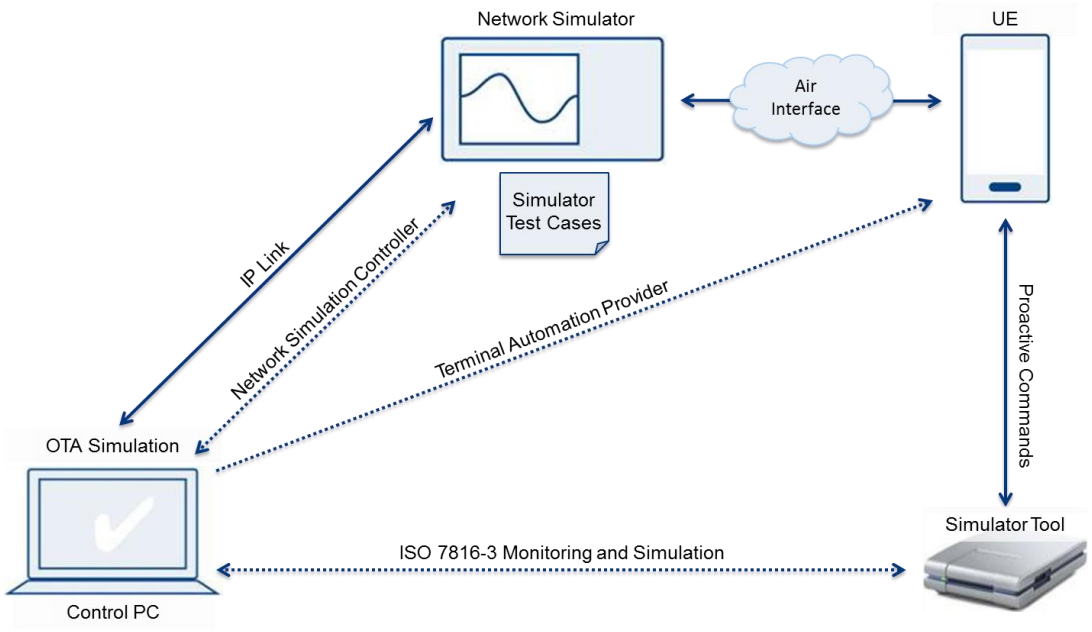

Figure 6: Test Environment

For delivering the SMS push to the UICC, the real world OTA platform will use an SMPP gateway. For ease of testing the real world OTA platform can be replaced by a simulated environment, this should also be simulated by the control PC.

There might be high volume data transmissions through a data channel between the UICC and the OTA Platform, e.g. when deploying an applet of ~100k from the OTA platform to the UICC.

3 NFC Features

3.1 General overview

This chapter addresses the NFC features covering the contactless interfaces between the device and NFC Tag and Reader respectively as well as the interface between NFC controller and UICC (SWP/HCI).

The test cases are grouped in three sub sections covering respectively NFC Tag testing, NFC Reader and core NFC functions including the SWP/HCI testing.

The list of conformance requirements tested within this section is listed in the table in section 3.2.

3.2 Conformance requirements

TS26_NFC_REQ_006	The NFC controller SHALL support SWP (Single Wire Protocol) interface with the UICC as per ETSI TS 102.613.
TS26_NFC_REQ_007	The NFC controller SHALL support HCI with the UICC as per ETSI TS 102.622.
TS26_NFC_REQ_008	Contactless tunnelling (CLT=A) mode SHALL be supported for SWP (per ETSI TS 102.613).
TS26_NFC_REQ_009	Contactless tunnelling (CLT=F) mode SHOULD be supported for SWP (per ETSI TS 102.613). Note This requirement is effective until 31st July 2015
TS26_NFC_REQ_009.1	Contactless tunnelling (CLT=F) mode SHALL be supported for SWP (per ETSI TS 102 613). Note This mandatory requirement will become effective from 1st August 2015.
TS26_NFC_REQ_010	The device/ NFC controller interface with UICC SHOULD support Class B
TS26_NFC_REQ_011	The device/ NFC controller interface with UICC SHALL support Class C full power mode
TS26_NFC_REQ_012	The device/ NFC controller interface with UICC SHALL support Class C low power mode if the device supports battery power off mode
TS26_NFC_REQ_013	The device/ NFC controller interface with UICC SHOULD support Class C low power mode if the device supports battery power low mode
TS26_NFC_REQ_014	The device/NFC Controller interface with UICC SHALL support DEACTIVATED followed by subsequent SWP interface activation in full power mode.
TS26_NFC_REQ_015	The NFC controller SHOULD support both windows size set to 3 and set to 4.
TS26_NFC_REQ_020	When the mobile device is automatically switched off, and enters battery low mode, the mobile device SHALL be able to perform 15 transactions in card emulation within the following 24 hours
TS26_NFC_REQ_021	NFC transactions SHALL be possible either in battery power off or battery low mode.
TS26_NFC_REQ_025	The mobile device SHALL support Card-emulation as per ISO/IEC 14443 Type A and Type B PICC and SHOULD support Card Emulation as per ISO/IEC 18092 Type F (Note This is effective until 31st July 2015)
TS26_NFC_REQ_025.1	The mobile device SHALL support Card-emulation as per ISO/IEC 14443 Type A, Type B PICC and ISO/IEC 18092 Type F

	(Effective as of 1st August 2015)
TS26_NFC_REQ_026	Card Emulation mode SHALL be enabled when the NFC is turned on.
TS26_NFC_REQ_027	For Card emulation mode the read distance SHALL be in the 0cm – 2cms range for battery operational mode, battery low or power off mode.
#TS26_NFC_REQ_033	The mobile device SHALL support Reader/Writer Mode as per (ISO/IEC 14443 Type A and Type B PCD and ISO/IEC 18092 Type F NOTE 1: For Type F implementation see JIS X 6319-4 which provides more details
TS26_NFC_REQ_034	The mobile device SHALL support NFC Forum Tag Type 1 as specified in NFC Forum Type 1 Tag Operation Specification.
TS26_NFC_REQ_035	The mobile device SHALL support NFC Forum Tag Type 2 as specified in NFC Forum Type 2 Tag Operation Specification.
TS26_NFC_REQ_036	The mobile device SHALL support NFC Forum Tag Type 3 as specified in NFC Forum Type 3 Tag Operation Specification.
TS26_NFC_REQ_037	The mobile device SHALL support NFC Forum Tag Type 4 as specified in NFC Forum Type 4 Tag Operation Specification.
TS26_NFC_REQ_038	Reader mode events SHALL be routed exclusively to the UICC or the Application processor.
TS26_NFC_REQ_039	The default routing for the reader mode events SHALL be via the Application processor.
TS26_NFC_REQ_040	The NFC Controller SHOULD support Reader Mode as per ETSI TS 102 622
TS26_NFC_REQ_041	The device SHALL support automatic and continuous switching between card emulation and reader mode. Note The impact on the standby time should be as minimal as possible. Further work will be progressed in TSG BLM to define power consumption test cases
TS26_NFC_REQ_042	A transaction time SHALL take 500ms or less for TAG message length not exceeding 100 bytes. The transaction time is defined from the start of the frame of the first RF command receiving an answer, to the end of the frame of the response to the last received RF command by a device, where the RF command is used to read the content in a tag
TS26_NFC_REQ_043	The mobile device SHALL be able to read/write the NFC Forum Smart Poster RTD.
TS26_NFC_REQ_044	The TAG SHALL be read at a distance of 1 cm and at distances between 0 to 1 cm. Note This requirement will be tested with a TAG Test Reference system agreed in the Test Book group.
TS26_NFC_REQ_110	The Tag SHOULD be read at a distance from 1 cm to 4 cms. Note: This requirement will be tested with a Tag Test Reference system agreed in the Test Book group.

Note: # - This indicates the requirement is still work in progress according to TS.26 NFC Handset Requirements

3.3 Reader/Writer mode

3.3.1 General overview

This chapter addresses the functions of the device for NFC Tag reading and writing according to the NFC Forum specification and under a limited set of distances between device and NFC Tag. General reader mode requirements are also covered in this section.

The list of conformance requirements tested within this section is listed in the table in section 3.3.2.

3.3.2 Conformance requirements

TS26_NFC_REQ_034	The mobile device SHALL support NFC Forum Tag Type 1, as specified in NFC Forum Type 1 Tag Operation Specification.
TS26_NFC_REQ_035	The mobile device SHALL support NFC Forum Tag Type 2, as specified in NFC Forum Type 2 Tag Operation Specification.
TS26_NFC_REQ_036	The mobile device SHALL support NFC Forum Tag Type 3, as specified in NFC Forum Type 3 Tag Operation Specification.
TS26_NFC_REQ_037	The mobile device SHALL support NFC Forum Tag Type 4, as specified in NFC Forum Type 4 Tag Operation Specification.
TS26_NFC_REQ_038	A reader mode events SHALL be routed exclusively to the UICC or the Application processor.
TS26_NFC_REQ_039	The default routing for the reader mode events SHALL be via the Application processor.
TS26_NFC_REQ_040	The NFC Controller SHOULD support Reader Mode as per ETSI TS 102 622
TS26_NFC_REQ_042	A transaction time SHALL take 500ms or less for TAG message length not exceeding 100 bytes. The transaction time is defined from the start of the frame of the first RF command receiving an answer, to the end of the frame of the response to the last received RF command by a device, where the RF command is used to read the content in a tag
TS26_NFC_REQ_043	The mobile device SHALL be able to read/write the NFC Forum Smart Poster RTD.
TS26_NFC_REQ_044	The Tag SHALL be read at a distance of 1 cm and at distances between 0 to 1 cm. Note: This requirement will be tested with a Tag Test Reference system
TS26_NFC_REQ_110	The Tag SHOULD be read at a distance from 1 cm to 4 cms. Note: This requirement will be tested with a Tag Test Reference system agreed in the Test Book group.

3.3.3 Test Cases

3.3.3.1 NFC Forum Tag Type 1 – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Tag Type 1 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 1 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_034
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to read the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag contents should be configured and used in the following order to perform the test:
    - NFC Tag #1–- NFC Tag is personalized with a “vCard”
    - NFC Tag #2–- NFC Tag is personalized with a “URI”
    - NFC Tag #3–- NFC Tag is personalized with a “Text”
        - In case of using reference tag: configuration and personalization of tags shall be performed independently of the DUT.
        - The DUT is not placed in the Read Range (more than 50cm from the Tag).

Procedure

3.3.3.1.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in NFC read range	If the DUT requests Tag contents to be accepted, the user shall accept this request.
2	DUT → Application	The DUT read the tag content	The tag content is correctly received by the application.
3	User → DUT	Remove the DUT from the read range	None
4	User → DUT	Repeat the above sequence to read the next tag	None

3.3.3.2 NFC Forum Tag Type 2 – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Tag Type 2 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 2 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_035
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to read the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag content should be configured and used in the following order to perform the test:
    - NFC T–g #1 - NFC Tag is personalized with a “SmartPoster” (launch browser)
    - In case of using reference tag: configuration and personalization of tags shall be performed independently of the DUT.
- The DUT is not placed in the Read Range (more than 50cm from the Tag).

Procedure

3.3.3.2.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in NFC read range	If the DUT requests Tag contents to be accepted, the user shall accept this request.
2	DUT → Application	The DUT read the tag content	The tag content is correctly received by the application.
3	User → DUT	Remove the DUT from the read range	None

3.3.3.3 NFC Forum Tag Type 3 – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Tag Type 3 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 3 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_036
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to read the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag content should be configured and used in the following order to perform the test:
- NFC T–g #1 - NFC Tag is personalized with a “SmartPoster” (SMS Sending)
    - In case of using reference tag: configuration and personalization of tags shall be performed independently of the DUT.
- The DUT is not placed in the Read Range (more than 50cm from the Tag).

Procedure

3.3.3.3.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in NFC read range	If the DUT requests Tag contents to be accepted, the user shall accept this request.
2	DUT → Application	The DUT to read the tag content	The tag content is correctly received by the application
3	User → DUT	Remove the DUT from the read range	None

3.3.3.4 NFC Forum Tag Type 4 – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Tag Type 4A and Type 4B platforms with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 4A & 4B Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_037
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to read the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag content should be configured and used in the following order to perform the test:
  - Initial conditions for test sequence #1 NFC Tag Type 4A
    - NFC Tag #1A - NFC Tag is personalized with a “SmartPoster” (phone call)
  - Initial conditions for test sequence #2 NFC Tag Type 4B
    - NFC Tag #1B - NFC Tag is personalized with a “SmartPoster” (email)
  - In case of using reference tag: configuration and personalization of tags shall be performed independently of the DUT.

The DUT is not placed in the Read Range (more than 50cm from the Tag).

Procedure

3.3.3.4.1 Test sequence No 1

Initial Conditions

Read the following tag content:

NFC Tag #1A - NFC Type A Tag is personalized with a “SmartPoster” (phone call)

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in NFC read range	If the DUT requests Tag contents to be accepted, the user shall accept this request.
2	DUT → Application	The DUT read the tag content	The tag content is correctly received by the application.
3	User → DUT	Remove the DUT from the read range	None

3.3.3.4.2 Test sequence No 2

Initial Conditions

Read the following tag content:

3.3.3.5 NFC Tag #1B - NFC Type B Tag is personalized with a “SmartPoster” (email)NFC Forum Tag Type 1 – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Tag Type 1 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 1 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_034
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to write the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag contents shall be configured to perform the test as following:
    - Initial conditions for test sequence #1: Tag empty initialized with Dynamic memory layout
    - Initial conditions for test sequence #2: Tag empty initialized with static memory layout

- The DUT is not placed in the Write Range (more than 50cm from the Tag).

Procedure

3.3.3.5.1 Test sequence No 1

Initial Conditions

Write the following tag content:

NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (SMS sending)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it.	The tag content is correctly written by the DUT. The Tag content is verified independently of the DUT

3.3.3.5.2 NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (SMS sending)Test sequence No 2

Initial Conditions

Write the following tag content:

NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (launch Browser)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None

Step	Direction	Sequence	Expected Result
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.6 NFC Forum Tag Type 2 – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Tag Type 2 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 2 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_035
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to write the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The tag contents shall be configured to perform the test as following:
  - Initial conditions for test sequence #1: Tag empty initialized with Dynamic memory layout
  - Initial conditions for test sequence #2: Tag empty initialized with static memory layout
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

Procedure

3.3.3.6.1 Test sequence No 1

Initial Conditions

Write the following tag content:

NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (phone call)

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.6.2 Test sequence No 2

Initial Conditions

Write the following tag content:

NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (email)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.7 NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (email)NFC Forum Tag Type 3 – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Tag Type 3 with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 3 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_036
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to write the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Method of Test

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- Tag empty initialized
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

Procedure

3.3.3.7.1 Test sequence No 1

Initial Conditions

Write the following tag content:

NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (Vcard)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.8 NFC Tag #1 - NFC Tag will be personalized with a “SmartPoster” (Vcard)NFC Forum Tag Type 4 – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Tag Type 4A and Type 4B with SmartPoster RTD (Record Type Definition) as specified in NFC Forum Type 4 Tag Operation Specification.

Referenced requirement

- TS26_NFC_REQ_037
- TS26_NFC_REQ_043

Test execution:

- This test case should be executed using reference NFC tag or simulated NFC tag.
- An application is installed on the DUT able to write the specified Tag format. This application is provided with the default DUT software or a reference application is installed

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag contents shall be configured to perform the test as following :
  - Initial conditions for test sequence #1: Tag empty initialized for NFC Tag Type 4A
  - Initial conditions for test sequence #2: Tag empty initialized for NFC Tag Type 4B
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

Procedure

3.3.3.8.1 Test sequence No 1

Initial Conditions

Write the following tag content:

	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.

	Direction	Sequence	Expected Result
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.8.2 Test sequence No 2

Initial Conditions

Write the following tag content:

NFC–Type 4B Tag #1 - NFC Tag is blank and will be personalized with a “SmartPoster” (Text)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	Application → DUT DUT → Tag	Use the application to write NFC Tag#1	The DUT write NFC Tag#1.
3	User → DUT	Remove the DUT from the write range	None

Step	Direction	Sequence	Expected Result
4	User	Verify that tag content was written correctly by reading it	The tag content is correctly written by the DUT. The Tag content shall be verified independently of the DUT

3.3.3.9 Distance for NFC Tag Type 1 reading

Test Purpose

This test case verifies the correct interpretation of NFC Tag Type 1 with RTD (Record Type Definition) by the DUT from 0 to 1cm optional 2 to 4cm

Referenced requirement

- TS26_NFC_REQ_044
- TS26_NFC_REQ_110

Method of Test

Initial Conditions

- Related Core Specifications / Reference
- Documents ISO/IEC 14443 ISO/IEC 18092.
- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 1 with RTD “SmartPoster” (launch browser) is available

Procedure

3.3.3.9.1 Test sequence No 1: Distance for NFC Tag–Type 1 Reading - 0,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None

Step	Direction	Sequence	Expected Result
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.9.2 Test sequence No 2: Distance for NFC Tag–Type 1 Reading - 0,5cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0,5 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.9.3 Test sequence No 3: Distance for NFC Tag–Type 1 Reading - 1,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 1 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None

Step	Direction	Sequence	Expected Result
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.9.4 Test sequence No 4: Distance for NFC Tag–Type 1 Reading - 2cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 2 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.9.5 Test sequence No 5: Distance for NFC Tag–Type 1 Reading - 3,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 3 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.9.6 Test sequence No 6: Distance for NFC Tag Type 1 Reading - 4,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 4 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10 Distance for NFC Tag Type 2 reading

Test Purpose

This test case verifies the correct interpretation of NFC Tag Type 2 with RTD (Record Type Definition) by the DUT from 0 to 1cm optional 2 to 4cm

Referenced requirement

- TS26_NFC_REQ_044
- TS26_NFC_REQ_110

Method of Test

Initial Conditions

- Related Core Specifications / Reference
- Documents ISO/IEC 14443 ISO/IEC 18092.
- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 2 with RTD “SmartPoster” (launch browser) is available

Procedure

3.3.3.10.1 Test sequence No 1: Distance for NFC Tag–Type 2 Reading - 0,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None

Step	Direction	Sequence	Expected Result
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10.2 Test sequence No 2: Distance for NFC Tag–Type 2 Reading - 0,5cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0,5 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10.3 Test sequence No 3: Distance for NFC Tag–Type 2 Reading - 1,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out

Step	Direction	Sequence	Expected Result
3		Place the DUT with the best coupling point at 1 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10.4 Test sequence No 4: Distance for NFC Tag–Type 2 Reading - 2,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 2 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10.5 Test sequence No 5: Distance for NFC Tag–Type 2 Reading - 3,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 3 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.10.6 Test sequence No 6: Distance for NFC Tag Type 2 Reading - 4,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 4 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11 Distance for NFC Tag Type 3 reading

Test Purpose

This test case verifies the correct interpretation of NFC Tag Type 3 with RTD (Record Type Definition) by the DUT from 0 to 1cm optional 2 to 4cm

Referenced requirement

- TS26_NFC_REQ_044
- TS26_NFC_REQ_110

Method of Test

Initial Conditions

- Related Core Specifications / Reference
- Documents ISO/IEC 14443 ISO/IEC 18092.
- Antenna reference point may be marked on the outside of the DUT
- **NFC Tags Type 3 with RTD “SmartPoster” (launch browser) is available**

Procedure

3.3.3.11.1 Test sequence No 1: Distance for NFC Tag–Type 3 Reading - 0,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11.2 Test sequence No 2: Distance for NFC Tag–Type 3 Reading - 0,5cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0,5 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11.3 Test sequence No 3: Distance for NFC Tag–Type 3 Reading - 1,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out

Step	Direction	Sequence	Expected Result
3		Place the DUT with the best coupling point at 1 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11.4 Test sequence No 4: Distance for NFC Tag–Type 3 Reading - 2cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 2 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11.5 Test sequence No 5: Distance for NFC Tag–Type 3 Reading - 3,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 3 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.11.6 Test sequence No 6: Distance for NFC Tag Type 3 Reading – 4,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 4 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12 Distance for NFC Tag Type 4A reading

Test Purpose

This test case verifies the correct interpretation of NFC Tag Type 4A with RTD (Record Type **Definition) by the DUT from 0 to 1cm optional 2 to 4cm**

Referenced requirement

- TS26_NFC_REQ_044
- TS26_NFC_REQ_110

Method of Test

Initial Conditions

- Related Core Specifications / Reference
- Documents ISO/IEC 14443 ISO/IEC 18092.
- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 4A with RTD “SmartPoster” (launch browser) is available

Procedure

3.3.3.12.1 Test sequence No 1: Distance for NFC Tag Type 4A Reading - 0,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12.2 Test sequence No 2: Distance for NFC Tag Type 4A Reading - 0,5cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0,5 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12.3 Test sequence No 3: Distance for NFC Tag Type 4A Reading - 1,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out

Step	Direction	Sequence	Expected Result
3		Place the DUT with the best coupling point at 1 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12.4 Test sequence No 4: Distance for NFC Tag Type 4A Reading - 2cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 2 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12.5 Test sequence No 5: Distance for NFC Tag Type 4A Reading - 3,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 3 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.12.6 Test sequence No 6: Distance for NFC Tag Type 4a Reading – 4,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 4 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13 Distance for NFC Tag Type 4B reading

Test Purpose

This test case verifies the correct interpretation of NFC Tag Type 4B with RTD (Record Type Definition) by the DUT from 0 to 1cm optional 2 to 4cm

Referenced requirement

- TS26_NFC_REQ_044
- TS26_NFC_REQ_110

Method of Test

Initial Conditions

- Related Core Specifications / Reference
- Documents ISO/IEC 14443 ISO/IEC 18092.
- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 4B with RTD “SmartPoster” (launch browser) is available

Procedure

3.3.3.13.1 Test sequence No 1: Distance for NFC Tag –ype 4B Reading - 0,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13.2 Test sequence No 2: Distance for NFC Tag –ype 4B Reading - 0,5cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 0,5 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13.3 Test sequence No 3: Distance for NFC Tag –ype 4B Reading - 1,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out

Step	Direction	Sequence	Expected Result
3		Place the DUT with the best coupling point at 1 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13.4 Test sequence No 4: Distance for NFC Tag –ype 4B Reading - 2cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 2 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13.5 Test sequence No 5: Distance for NFC Tag –ype 4B Reading - 3,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 3 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.13.6 Test sequence No 6: Distance for NFC Tag Type 4b Reading – 4,0cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using **NFC Tag application**, read the tag content	None
2		Remove the NFC Tag from the DUT and close the application **NFC Tag application**	The NFC Tag is read out
3		Place the DUT with the best coupling point at 4 cm from the NFC Tag with RTD “SmartPoster” (launch browser)	None
4		Confirm tag reading	The tag is automatically read and the information retrieved is identical to step 2

3.3.3.14 NFC Tag type 1 reading performance

Test Purpose

To ensure a tag reading takes 500ms or less on a NFC Tag type 1 for a Tag message length not exceeding 100 bytes

Method of Test

Referenced requirement

- TS26_NFC_REQ_042

Initial Conditions

RF spy tool able to measure the transaction time.

Time for transaction is measured between:

- The SoF of the first RF command receiving an answer (for ex: Wake Up)
- The EoF of the answer of the last RF command used to read the content.

The way to present the DUT in front of the tag is done in such a way that the number of communication issues is minimized. For the purpose of this testing, tag content exchanged will have a length of 100 bytes.

Procedure

3.3.3.14.1 Test sequence No 1

Initial Conditions

NFC Tag type 1 is personalized with RTD “SmartPoster” (launch browser)**Step**	**Direction**	**Sequence**	**Expected Result**
1		Start the RF spy	None
2		Read a NFC tag Type 1	NFC Tag content is read
3		As soon as the DUT prompts the end user, stops the RF spy	Time for transaction is less than 500ms

3.3.3.15 NFC Tag type 2 reading performance

Test Purpose

To ensure a tag reading takes 500ms or less on a NFC Tag type 2 for a Tag message length not exceeding 100 bytes

Method of Test

Referenced requirement

- TS26_NFC_REQ_042

Initial Conditions

RF spy tool able to measure the transaction time.

Time for transaction is measured between:

- The SoF of the first RF command receiving an answer (for ex: Wake Up)
- The EoF of the answer of the last RF command used to read the content.

The way to present the DUT in front of the tag is done in such a way that the number of communication issues is minimized.

For the purpose of this testing, tag content exchanged will have a length of 100 bytes.

Procedure

3.3.3.15.1 Test sequence No 1

Initial Conditions

NFC Tag type 2 is personalized with RTD “SmartPoster” (launch browser)

Step	**Direction**	**Sequence**	**Expected Result**
1		Start the RF spy	None
2		Read a NFC tag Type 2	NFC Tag content is read

Step	Direction	Sequence	Expected Result
3		As soon as the DUT prompts the end user, stops the RF spy	Time for transaction is less than 500ms

3.3.3.16 NFC Tag type 3 reading performance

Test Purpose

To ensure a tag reading takes 500ms or less on a NFC Tag type 3 for a Tag message length not exceeding 100 bytes

Method of Test

Referenced requirement

- TS26_NFC_REQ_042

Initial Conditions

RF spy tool able to measure the transaction time.

Time for transaction is measured between:

- The SoF of the first RF command receiving an answer (for ex: Wake Up)
- The EoF of the answer of the last RF command used to read the content.

The way to present the DUT in front of the tag is done in such a way that the number of communication issues is minimized.

For the purpose of this testing, tag content exchanged will have a length of 100 bytes.

Procedure

3.3.3.16.1 Test sequence No 1

Initial Conditions

NFC Tag type 3 is personalized with RTD “SmartPoster” (launch browser)

Step	Direction	Sequence	Expected Result
1		Start the RF spy	None
2		Read a NFC tag Type 3	NFC Tag content is read
3		As soon as the DUT prompts the end user, stops the RF spy	Time for transaction is less than 500ms

3.3.3.17 NFC Tag type 4A reading performance

Test Purpose

To ensure a tag reading takes 500ms or less on a NFC Tag type 4A for a Tag message length not exceeding 100 bytes

Method of Test

Referenced requirement

- TS26_NFC_REQ_042

Initial Conditions

RF spy tool able to measure the transaction time.

Time for transaction is measured between:

- The SoF of the first RF command receiving an answer (for ex: Wake Up)
- The EoF of the answer of the last RF command used to read the content.

The way to present the DUT in front of the tag is done in such a way that the number of communication issues is minimized.

For the purpose of this testing, tag content exchanged will have a length of 100 bytes.

Procedure

3.3.3.17.1 Test sequence No 1

Initial Conditions

NFC Tag type 4A is personalized with RTD “SmartPoster” (launch browser)

Step	Direction	Sequence	Expected Result
1		Start the RF spy	None
2		Read a NFC tag Type 4A	NFC Tag content is read
3		As soon as the DUT prompts the end user, stops the RF spy	Time for transaction is less than 500ms

3.3.3.18 NFC Tag type 4B reading performance

Test Purpose

To ensure a tag reading takes 500ms or less on a NFC Tag type 4B for a Tag message length not exceeding 100 bytes

Method of Test

Referenced requirement

- TS26_NFC_REQ_042

Initial Conditions

RF spy tool able to measure the transaction time.

Time for transaction is measured between:

- The SoF of the first RF command receiving an answer (for ex: Wake Up)
- The EoF of the answer of the last RF command used to read the content.

The way to present the DUT in front of the tag is done in such a way that the number of communication issues is minimized.

For the purpose of this testing, tag content exchanged will have a length of 100 bytes.

Procedure

3.3.3.18.1 Test sequence No 1

Initial Conditions

NFC Tag type 4B is personalized with RTD “SmartPoster” (launch browser)

Step	Direction	Sequence	Expected Result
1		Start the RF spy	None
2		Read a NFC tag Type 4B	NFC Tag content is read
3		As soon as the DUT prompts the end user, stops the RF spy	Time for transaction is less than 500ms

3.3.3.19 NFC Tag handling during an active data transfer.

Test Purpose

To ensure that during an active data transfer (data exchanged over the mobile network) the DUT SHOULD still be able to handle NFC tags accordingly and inform the user of read tags.

Referenced requirement

- TS26_NFC_REQ_043

Method of Test

Initial Conditions

- NFC Forum Tag Type 1 with content as described in Section 2.5.4.5 is available for testing.
- Initial setup is performed.
- The DUT is prepared for data transmission over the mobile network.

Procedure

3.3.3.19.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Initiate data transmission over the mobile network	None
2		Read a NFC tag	NFC Tag is read
3		At the end of the tag reading ensure the data transmission is going on	Data transmission ends successfully

3.3.3.20 Reader mode via UICC

Test Purpose

To ensure the reader mode events are correctly routed exclusively to the UICC as soon as a card application registers on tag reading event

Referenced requirement

- TS26_NFC_REQ_038
- TS26_NFC_REQ_040

Method of Test

As some clarification are expected on the reader mode via UICC, this test is FFS

3.3.3.21 Reader mode via Application Processor

Test Purpose

To ensure the reader mode events are correctly routed by default to the application processor

Referenced requirement

- TS26_NFC_REQ_039

Method of Test

As some clarification are expected on the reader mode via UICC, this test is FFS

3.3.3.22 Reader mode events routing

Test Purpose

Ensure the reader mode events are routed to the application processor until no UICC application activate the reader mode

Referenced requirement

- TS26_NFC_REQ_040

Method of Test

As some clarification is expected on the reader mode via UICC, this test is FFS

3.3.3.23 NFC Forum RTD signature

Test Purpose

Check correct management of the NFC Forum RTD signature.

Referenced requirement

Method of Test

As some clarification is expected on the NFC Forum RTD signature management by the DUT, this test is FFS.

3.4 Card emulation mode

3.4.1 General overview

This section addresses the requirements for card emulation mode. This includes basic test cases for card emulation in normal mode as well under different battery modes and distances.

The list of conformance requirements tested within this section is listed in the table in section 3.4.2.

3.4.2 Conformance requirements

TS26_NFC_REQ_020	When the mobile device is automatically switched off, and enters battery low mode, the mobile device SHALL be able to perform 15 transactions in card emulation within the following 24 hours
TS26_NFC_REQ_021	NFC transactions SHALL be possible either in battery power off or battery low mode.
TS26_NFC_REQ_026	Card Emulation mode SHALL be enabled as soon as the NFC is turned on.
TS26_NFC_REQ_027	For Card emulation mode the read distance SHALL be in the 0cm – 2cms range for battery operational mode, battery low or power off mode.

3.4.3 Test Cases

3.4.3.1 Card Emulation enabled as soon as NFC hardware is on

Test Purpose

To verify if card emulation mode works on the device as soon as the device is on.

Referenced requirement

- TS26_NFC_REQ_026

Method of Test

Initial Conditions

- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.

- APDU Application to send APDUs according to the reference transaction.

Procedure

3.4.3.1.1 Test sequence No 1

Initial Conditions

None.

Step	Direction	Sequence	Expected Result
1	User → DUT DUT → UICC	Power On the DUT and wait until the UICC has completed HCI session initialization	The HCI initialization is performed correctly
2	User → DUT	Enable NFC in the DUT	DUT indicates NFC is on
3	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	None
4	User → PCD	Power on the field	None
5	PCD → DUT DUT → UICC	Perform the reference transaction using a contactless reader	The reference transaction is performed successfully

3.4.3.1.2 Test sequence No 2

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT DUT → UICC	Power On the DUT and wait until the UICC has completed HCI session initialization	The HCI initialization is performed correctly
2	User → DUT	Enable the NFC on the DUT	None
3	User → DUT	Power off the DUT	None
4	User → DUT	Power on the DUT	None
5	User → DUT	Check that NFC is on	DUT indicates NFC is on

Step	Direction	Sequence	Expected Result
6	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	None
7	User → PCD	Power on the field	None
8	PCD → DUT DUT → UICC	Perform the reference transaction using a contactless reader	The reference transaction is performed successfully

3.4.3.2 NFC during Standby time

Test Purpose

To ensure the NFC transaction in card emulation mode is possible during 24 hours after the DUT automatically switched off due to a low battery level.

DUT SHALL accept 15 correct reference transactions

Referenced requirement

- TS26_NFC_REQ_020

Method of Test

Initial Conditions

- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.

Procedure

3.4.3.2.1 Test sequence No 1

Initial Conditions

The DUT enters Battery Low Mode

Step	Direction	Sequence	Expected Result
1		Execute the reference transaction in loop mode	The DUT must manage the reference transaction 15 times Note: The 15th transaction shall be performed within the last 5 minutes before the expiry of the 24 hours.

3.4.3.3 Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes

Test Purpose

To ensure the NFC transaction in card emulation mode is possible in Battery Power off Mode or in Battery Low Mode.

Referenced requirement

- TS26_NFC_REQ_021

Method of Test

Initial Conditions

- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.

Procedure

3.4.3.3.1 Test sequence No 1: Card Emulation Mode Type A in Battery Power Off mode

Initial Conditions

The device is in Battery Power Off mode.

Step	Direction	Sequence	Expected Result
1		Perform the reference transaction type A using a contactless reader	Verify that the reference transaction is successfully performed

3.4.3.3.2 Test sequence No 2: Card Emulation Mode Type B in Battery Power Off mode

Initial Conditions

The device is in Battery Power Off mode.

Step	Direction	Sequence	Expected Result
1		Perform the reference transaction type B using a contactless reader	Verify that the reference transaction is successfully performed

3.4.3.3.3 Test sequence No 3:CLT (A) in Battery Power Off mode –

FFS

3.4.3.3.4 Test sequence No 4: Card Emulation Mode Type A in Battery Low Mode

Initial Conditions

- . The device is in Battery Low mode.

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Perform the reference transaction type A using a contactless reader	Verify that the reference transaction is successfully performed

3.4.3.3.5 Test sequence No 5: Card Emulation Mode Type B in Battery Low Mode

Initial Conditions

- . The device is in Battery Low mode

Step	Direction	Sequence	Expected Result
1		Perform the reference transaction type B using a contactless reader	Verify that the reference transaction is successfully performed

3.4.3.3.6 Test sequence No 6: CLT (A) in Battery Low Mode

FFS

3.4.3.4 Distance for card emulation

Test Purpose

- To ensure that in card emulation mode, the communication is ok in a range from 0cm to 2cm (antenna side) in Battery Operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- Related Specs/Docs: ISO/IEC 14443

Procedure

- Distance for card emulation is tested as part of the test cases referenced in Annex B.2 and tested in 3.5.3.5

3.4.3.5 Distance for card emulation in Battery Power-off Mode (0cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0cm (antenna side) with Battery Power-off Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI Initialization was correctly performed in previous operating session
- NFC is enabled in DUT
- Card emulation is enabled in the DUT
- DUT is in Battery Power-off Mode
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 0cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.5.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.6 Distance for card emulation in Battery Power-off Mode (0.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0.5cm (antenna side) with Battery Power-off Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- DUT is in Battery Power-off Mode.
- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.

- While the field is off, the DUT set at 0.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.6.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.7 Distance for card emulation in Battery Power-off Mode (1cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1cm (antenna side) with Battery Power-off Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT
- DUT is in Battery Power-off Mode
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.7.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.8 Distance for card emulation in Battery Power-off Mode (1.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1.5cm (antenna side) with Battery Power-off Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT
- DUT is in Battery Power-off Mode
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.8.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.9 Distance for card emulation in Battery Power-off Mode (2cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 2cm (antenna side) with Battery Power-off Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT
- DUT is in Battery Power-off Mode
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 2cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.9.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed fully

3.4.3.10 Distance for card emulation in Battery Power-low Mode (0cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0cm (antenna side) with Battery Power-low Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- DUT is in Battery Power-low Mode.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 0cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.10.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed fully

3.4.3.11 Distance for card emulation in Battery Power-low Mode (0.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0.5cm (antenna side) with Battery Power-low Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- DUT is in Battery Power-low Mode
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.

- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 0.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.11.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.12 Distance for card emulation in Battery Power-low Mode (1cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1cm (antenna side) with Battery Power-low Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation for is enabled in the DUT.
- DUT is in Battery Power-low Mode.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.12.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.13 Distance for card emulation in Battery Power-low Mode (1.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1.5cm (antenna side) with Battery Power-low Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- DUT is in Battery Power-low Mode.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.13.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.14 Distance for card emulation in Battery Power-low Mode (2cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 2cm (antenna side) with Battery Power-low Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- HCI initialization was correctly performed in previous operating session
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- DUT is in Battery Power-low Mode.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 2cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.14.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.15 Distance for card emulation in Battery Power-operational Mode (0cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0cm (antenna side) with Battery Power- operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- DUT is powered on and the DUT is in Battery Power- operational Mode
- HCI initialization is correctly performed.
- NFC is enabled in the DUT.
- Card emulation is enabled in the DUT.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 0cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.15.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.16 Distance for card emulation in Battery Power-operational Mode (0.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 0.5cm (antenna side) with Battery Power- operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- DUT is powered on and the DUT is in Battery Power- operational Mode
- HCI initialization is correctly performed.
- NFC is enabled in the DUT.
- Card emulation is enabled in the DUT.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.

- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 0.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.16.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.17 Distance for card emulation in Battery Power-operational Mode (1cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1cm (antenna side) with Battery Power- operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- DUT is powered on and the DUT is in Battery Power- operational Mode
- HCI initialization is correctly performed.
- NFC is enabled in the DUT.
- Card emulation is enabled in the DUT.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.17.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.18 Distance for card emulation in Battery Power-operational Mode (1.5cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 1.5cm (antenna side) with Battery Power- operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- DUT is powered on and the DUT is in Battery Power- operational Mode.
- HCI initialization is correctly performed.
- NFC is enabled in the DUT.
- Card emulation is enabled in the DUT.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 1.5cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.18.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.4.3.19 Distance for card emulation in Battery Power- operational Mode (2cm)

Test Purpose

To ensure that in card emulation mode, the communication is ok at 2cm (antenna side) with Battery Power- operational Mode

Referenced requirement

- TS26_NFC_REQ_027

Method of Test

Initial Conditions

- DUT is powered on and the DUT is in Battery Power- operational Mode
- HCI initialization is correctly performed
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- ReferenceApplication.cap managing the reference transaction with AID1 selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT set at 2cm of the reference contactless reader at the best coupling point between DUT and contactless reader. In order to support testing - the antenna reference point may be marked on the DUT.

Procedure

3.4.3.19.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power on the field	None
2	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully

3.5 Core and Common features

3.5.1 General overview

This section addresses the requirements for the core NFC controller and for the common functions between Reader/Writer and Card emulation mode. This also includes the SWP/HCI and RF protocol compliance.

The list of conformance requirements tested within this section is listed in the table in section 3.5.2.

3.5.2 Conformance requirements

TS26_NFC_REQ_006	The NFC controller SHALL support SWP (Single Wire Protocol) interface with the UICC as per ETSI TS 102.613.
TS26_NFC_REQ_007	The NFC controller SHALL support HCI with the UICC as per ETSI TS 102.622.
TS26_NFC_REQ_008Contactless tunnelling (CLT=A) mode SHALL be supported for SWP (per ETSI TS 102.613).	
TS26_NFC_REQ_009Contactless tunnelling (CLT=F) mode SHOULD be supported for SWP (per ETSI TS 102.613). Note This requirement is effective until 31st July 2015	
TS26_NFC_REQ_009.1	Contactless tunnelling (CLT=F) mode SHALL be supported for SWP (per ETSI TS 102 613). Note This mandatory requirement will become effective from 1st August 2015.
TS26_NFC_REQ_010	The device/ NFC controller interface with UICC SHOULD support Class B
TS26_NFC_REQ_011	The device/ NFC controller interface with UICC SHALL support Class C full power mode
TS26_NFC_REQ_012	The device/ NFC controller interface with UICC SHALL support Class C low power mode if the device supports battery power off mode
TS26_NFC_REQ_013	The device/ NFC controller interface with UICC SHOULD support Class C low power mode if the device supports battery power low mode
TS26_NFC_REQ_014	The device/NFC Controller interface with UICC SHALL support DEACTIVATED followed by subsequent SWP interface activation in full power mode.
TS26_NFC_REQ_015	The NFC controller SHOULD support both windows size set to 3 and set to 4.
TS26_NFC_REQ_025	The mobile device SHALL support Card-emulation as per ISO/IEC 14443 Type A and Type B PICC and SHOULD support Card Emulation as per ISO/IEC 18092 Type F NOTE 1: For Type F implementation see JIS X 6319-4 which provides more details (This is effective until 31st July 2015)
TS26_NFC_REQ_025.1	The mobile device SHALL support Card-emulation as per ISO/IEC 14443 Type A, Type B PICC and ISO/IEC 18092 Type F (Effective as of the 1st August 2015)
#TS26_NFC_REQ_033	The mobile device SHALL support Reader/Writer Mode as per (ISO/IEC 14443 Type A and Type B PCD and ISO/IEC 18092 Type F NOTE 1: For Type F implementation see JIS X 6319-4 which provides more details
TS26_NFC_REQ_041	The device SHALL support automatic and continuous switching between card emulation and reader mode. Note The impact on the standby time should be as minimal as possible.

Note: # - This indicates the requirement is still work in progress according to TS.26 NFC Handset Requirements

3.5.3 Test Cases

3.5.3.1 SWP Compliance testing

Test Purpose

To ensure the device conforms to Single Wire Protocol specification

Referenced requirement

- TS26_NFC_REQ_006
- TS26_NFC_REQ_008
- TS26_NFC_REQ_009 - Valid to 31st July 2015
- TS26_NFC_REQ_009.1 - Valid from 1st August 2015
- TS26_NFC_REQ_010
- TS26_NFC_REQ_011
- TS26_NFC_REQ_012
- TS26_NFC_REQ_013
- TS26_NFC_REQ_014
- TS26_NFC_REQ_015

Method of Test

Related Specs/Docs: ETSI TS 102.613 |9]

Procedure

The DUT shall pass all applicable test cases referenced in Annex B.4 and in Table 15, Table 16.

3.5.3.2 HCI Compliance testing

Test Purpose

To ensure the device conforms to Host Controller Interface specification

Referenced requirement

- TS26_NFC_REQ_007

Method of Test

Related Specs/Docs: ETSI TS 102.622 [10]

Procedure

The DUT shall pass all applicable test cases referenced in B.5, Table 17 and Table 18.

3.5.3.3 SWP Stress test

Test Purpose

To ensure the DUT manages 100 transactions consecutively

Referenced requirement

- TS26_NFC_REQ_006

Method of Test

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully.
- NFC is enabled on the DUT
- Card Emulation is enabled in the DUT
- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.

Procedure

3.5.3.3.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	None
2	User → PCD	Power on the field	None
3	PCD → DUT DUT → UICC	Execute the **reference transaction** in loop mode (100 loops)	The reference transaction is performed correctly 100 times consecutively.

3.5.3.4 Switch mode

Test Purpose

To ensure the DUT is able to automatically and continuous switch between card emulation mode and reader emulation mode

Referenced requirement

- TS26_NFC_REQ_041

Method of Test

Initial Conditions

- The DUT is on
- HCI initialization has been correctly performed
- UICC application with AID1 selectable
- A Tag type 1 with the RTD “Text” content
- The Tag and the reader are separated by at least 50cm
- The NFC is enabled

Procedure

3.5.3.4.1 Test sequence No 1

Initial Conditions

Backlight is on. Device not locked.

Step	Direction	Sequence	Expected Result
1		Place the DUT in front of the Tag to read	Tag reading ok
2		Set the DUT in front of the contactless reader then send a SELECT_BY_DF_name AID1	**APDU application** receives Status word 90 00
3		Place the DUT in front of the Tag to read	Tag reading ok
4		Set the DUT in front of the contactless reader then send a SELECT_BY_DF_name AID1	**APDU application** receives Status word 90 00
5		Place the DUT in front of the Tag to read	Tag reading ok
6		Set the DUT in front of the contactless reader then send a SELECT_BY_DF_name AID1	**APDU application** receives Status word 90 00
7		Place the DUT in front of the Tag to read	Tag reading ok
8		Set the DUT in front of the contactless reader then send a SELECT_BY_DF_name AID1	**APDU application** receives Status word 90 00
9		Place the DUT in front of the Tag to read	Tag reading ok
10		Set the DUT in front of the contactless reader then send a SELECT_BY_DF_name AID1	**APDU application** receives Status word 90 00

3.5.3.5 RF Protocol compliance

Test Purpose

To ensure the mobile device is compliant with ISO/IEC 14443 A & B for card and reader emulation modes

To ensure the mobile device is compliant with ISO/IEC 18092 type F for card and reader emulation modes – (Required from 1st Jan 2015 onwards)

Referenced requirement

- TS26_NFC_REQ_025 – Valid to 31st July 2015

- TS26_NFC_REQ_025.1 – Valid from 1st August 2015
- TS26_NFC_REQ_033

Method of Test

Related Specs/Docs: ISO/IEC 14443, ISO/IEC 18092

Procedure

The DUT shall pass all the test cases referenced in annex B.2, and annex B.3.

3.5.3.6 VOID

4 VOID

5 Secure Element Access Control

5.1 General overview

This chapter addresses the implementation of the Secure Element Access Control mechanism according to the *GlobalPlatform Access Control* [7] standard. It will grant or refuse the communication to/from applets stored in the UICC SE

The list of conformance requirements tested within this section is listed in the table in section 5.2.

Note: The current version of this test book covers usage of Access Rule Files in some selected aspects.

5.2 Conformance requirements

TS26_NFC_REQ_082	Open OS devices SHALL provide access control as per Global Platform, Secure Element Access Control specification.
TS26_NFC_REQ_083	When no access control data (files or applets) is found on the UICC the API SHALL deny access to the UICC.

5.3 Test Cases

Following initial conditions are applicable for all SE Access Control tests in this section, unless it is otherwise specified for a particular test case.

General Initial Conditions

Two instances of the UICC application **APDU_TestApplication.cap** with AID1 and AID2 are selectable.

For that purpose, **MobileApplication** is registered for EVT_TRANSACTION handling from AID1 and AID2 and implements the functions “Select AID1” and “Select AID2” as it is specified in section 2.

The application is duplicated with different signature configurations as it is specified in section 2 and respectively named:

- GSMA_AC_Mobile_App_SP1_signed
- GSMA_AC_Mobile_App_SP2_signed

Note1: GSMA_AC_Mobile_App_SP1_signed is installed first.

Note2: Steps performed through the contactless interface (e.g. step 17 and 25 in Test Sequence 1) ensure for each test that the application on the mobile is correctly triggered by an NFC event.

Initial state: RF field is powered off and no applications should be started manually on the DUT. APDU_TestApplication.cap is not selected on UICC.

5.3.1 GP SE Access Control

Test Purpose

To ensure the Open OS device provide API for Access Control as per Global Platform Specification GPD_SE_Access_Control for:

- Secure Element Access API
- NFC Event

Referenced requirement

- TS26_NFC_REQ_082
- TS26_NFC_REQ_083

Method of Test

Procedure

5.3.1.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF (Application Dedicated File) with a DODF (Data Object Directory File) present and valid
- an ACMF (Access Control Main File) is present and valid
- an ACRF (Access Control Rules File) is present and valid and contains a rule for “all other AIDs” (a rule for all Secure Element applications that are not explicitly protected by a specific rule) and a path for one ACCF containing rules for AID1 and AID2 and an SP1 hash condition

  ⇨ SP1 has full access to all AID

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	SELECT command is successful and call to "Select AID1" function returns successfully
3		Call "Select AID2" function	SELECT command is successful and call to "Select AID1" function returns successfully
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	

Step	Direction	Sequence	Expected Result
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13a		RF field is powered off	
13b		The mobile sends EVT_FIELD_OFF	
14		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
15		Close GSMA_AC_Mobile_App_SP1_signed	
16		repeat steps 9 to 14 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.2 Test sequence No 2

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF. The ACCF is present and contains no hash condition (access allowed for mobile apps)

  ⇨ AID1 is always accessible, no access allowed for any other AID

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is successful
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.1.3 Test sequence No 3

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid

- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF. The ACCF is present and contains no hash condition (access allowed for mobile apps)

⇨ all applications have full access to all AID

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is successful
7		Call "Select AID2" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 16 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.4 Test sequence No 4

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing SP1 hash condition

⇨ only access to AID1 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	

Step	Direction	Sequence	Expected Result
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.1.5 Test sequence No 5

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains
  - one specific target rule for AID1 and a path for one ACCF containing SP1 hash condition
  - one specific target rule for AID1 and a path for one ACCF containing SP2 hash condition

⇨ only access to AID1 by SP1 and SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is successful
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted

Step	Direction	Sequence	Expected Result
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.1.6 Test sequence No 6

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
    - an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing SP1 hash condition and SP2 hash condition.

⇨ SP1 and SP2 can access AID1 only

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted

Step	Direction	Sequence	Expected Result
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is successful
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched (first installed application)
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.1.7 Test sequence No 7

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains
  - one rule “8200” and a path for one ACCF containing SP1 hash condition
  - one rule “8200” and a path for one ACCF containing SP2 hash condition

⇨ SP1 and SP2 has full access to all AID

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is successful
7		Call "Select AID2" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched (first installed application)
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 16 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.8 Test sequence No 8

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains

- one specific target rule for AID1 and a path for one ACCF containing SP1 hash condition
- one specific target rule for AID2 and a path for the same ACCF

⇨ SP1 have access to AID1 and AID2

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	

Step	Direction	Sequence	Expected Result
17		repeat steps 9 to 16 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.9 Test sequence No 9

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains
  - one specific target rule for AID1 and a path for one ACCF containing SP1 hash condition
  - one specific target rule for AID1 and a path for one ACCF. The ACCF contains no hash condition (access allowed for mobile apps)

⇨ only access to AID1 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	

Step	Direction	Sequence	Expected Result
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.2 GP SE Access Control - Refresh tag

Test Purpose

To ensure the DUT does not read all the Access Control rules when the refresh tag is not set.

Referenced requirement

- TS26_NFC_REQ_082
- TS26_NFC_REQ_083

Method of Test

Initial Conditions

- An instance of the UICC application **APDU_TestApplication.cap** with AID1 is selectable.
- **MobileApplication** is installed on the device and implements a function “Select AID1”.
- The application is signed with test certificate SP1 (**GSMA_Mobile_App_SP1_signed).**

Procedure

5.3.2.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid

- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing an empty hash condition

⇨ only access to AID1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Using the **MobileApplication**, select AID1	Call is successful
2		Start the ISO7816 spy	
3		Using the **MobileApplication**, select AID1	Call is successful
4		Stop the spy.	The log can be used to verify whether the DUT checks the "refresh tag". If after reading the PKCS#15 structure, a logical channel has been opened then check the DUT closes the logical channel at the end of the reading. The whole content of the PKCS#15 is not read.
5		Change the UICC configuration with the following: • PKCS#15 ADF with a DODF present and valid • an ACMF is present and valid • an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing an entry with a corrupted certificate (wrong length) The reference PKCS#15 structure is in Annex E.	
6		Start the ISO7816 spy	
7		Using the **MobileApplication**, select AID1	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Stop the spy.	

5.3.2.2 Test sequence No 2

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid

- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing an empty hash condition

⇨ only access to AID1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Using the **MobileApplication**, select AID1	Call is successful
2		Switch off the DUT	
3		Start the ISO7816 spy	
4		Switch on the DUT	
5		Using the **MobileApplication**, select AID1	Call is successful
6		Stop the spy.	The log can be used to verify whether the DUT read the whole content during the first access to the PKCS#15 content.

5.3.3 GP SE Access Control – ADF_PKCS#15 and DF PKCS#15

Test Purpose

To ensure the DUT correctly manages card configuration with a PKCS#15 ADF selectable and another DF PKCS#15 available in EF_DIR

Referenced requirement

- TS26_NFC_REQ_082

Method of Test

Initial Conditions

Only the following versions of the MobileApplication are used for these tests:

- **GSMA_AC_Mobile_App_SP1_signed**
- **GSMA_AC_Mobile_App_SP2_signed**

Procedure

5.3.3.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid

- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing a SP1 hash condition
    - EF_DIR contains a reference to PKCS#15 DF structure containing a specific target rule for AID2 and a path for one ACCF containing a SP2 hash condition

⇨ only access to AID1 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched

Step	Direction	Sequence	Expected Result
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.4 GP SE Access Control – PKCS#15 selection via EF_DIR

Test Purpose

To ensure the DUT correctly manages card configuration without PKCS#15 AID. According to GP specification, it the selection of the PKCS#15 AID fails, the DUT selects the EF_DIR to locate a PKCS#15 DF

Referenced requirement

- TS26_NFC_REQ_082

Method of Test

Initial Conditions

Only the following versions of the MobileApplication are used for these tests:

- **GSMA_AC_Mobile_App_SP1_signed**
- **GSMA_AC_Mobile_App_SP2_signed**

Procedure

5.3.4.1 Test sequence No 1

Initial Conditions for test# 1

The following configuration is loaded into the UICC:

- ADF PKCS#15 is absent
- EF_DIR contains a reference to PKCS#15 DF structure containing a specific target rule for AID1 and a path for one ACCF containing a SP1 hash condition

⇨ only access to AID1 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted

Step	Direction	Sequence	Expected Result
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 15 with AID2 instead of AID1	No application is triggered

5.3.5 GP SE Access Control – Configuration limits

Test Purpose

To ensure the DUT correctly manages card configuration with large contents

Referenced requirement

- TS26_NFC_REQ_082

Method of Test

Initial Conditions

Only the following versions of the MobileApplication are used for these tests:

- **GSMA_AC_Mobile_App_SP1_signed**
- **GSMA_AC_Mobile_App_SP2_signed**

Procedure

5.3.5.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains
  - one specific target rule for AID1 and a path for one ACCF containing 10 dummy hashes conditions and a SP1 hash condition
  - one specific target rule for AID2 and a path for one ACCF containing 10 dummy hashes conditions and a SP2 hash condition

⇨ access to AID1 by SP1 is allowed – access to AID2 by SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	

Step	Direction	Sequence	Expected Result
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 16 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP2_signed** is launched

5.3.5.2 Test sequence No 2

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains
  - one specific target rule for AID1 and a path for one ACCF containing 1 dummy hash condition and a SP1 hash condition
  - one specific target rule for AID2 and a path for one ACCF containing 1 dummy hash condition and a SP2 hash condition
  - 48 rules “A0XX04XX[dummy AIDs]” and a path for one ACCF containing 2 dummy hash conditions

⇨ access to AID1 by SP1 is allowed – access to AID2 by SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID1" function	Call is successful
3		Call "Select AID2" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted

Step	Direction	Sequence	Expected Result
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID2" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9		RF field is powered on	
10		Mobile starts card emulation session	
11		Start **APDU application** by sending a SELECT command with AID1	
12		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
13		RF field is powered off	
14		The mobile sends EVT_FIELD_OFF	
15		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**GSMA_AC_Mobile_App_SP1_signed** is launched
16		Close GSMA_AC_Mobile_App_SP1_signed	
17		repeat steps 9 to 16 with AID2 instead of AID1	**GSMA_AC_Mobile_App_SP2_signed** is launched

5.3.6 GP SE Access Control – No access

Test Purpose

To ensure the DUT deny the access to

- Secure Element Access API
- NFC Event when no PKCS#15 structure is available

Referenced requirement

- TS26_NFC_REQ_083

Method of Test

Initial Conditions

An instance of the UICC application **APDU_TestApplication.cap** with AID1 is selectable.

For that purpose, **MobileApplication** is registered for EVT_TRANSACTION handling from AID1 and implements a function “Select AID1”

The application is signed with test certificate SP1 (**GSMA_AC_Mobile_App_SP1_signed).**

Procedure

5.3.6.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- ADF PKCS#15 is absent
- EF_DIR does not contain references to PKCS15 structure

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
3		Close **GSMA_AC_Mobile_App_SP1_signed**	
4		RF field is powered on	
5		Mobile starts card emulation session	
6		Start **APDU application** by sending a SELECT command with AID1	
7		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
8		RF field is powered off	
9		The mobile sends EVT_FIELD_OFF	
10		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	No application is triggered

5.3.6.2 Test sequence No 2

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- ACRF is absent

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		**Launch GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
3		Close **GSMA_AC_Mobile_App_SP1_signed**	
4		RF field is powered on	
5		Mobile starts card emulation session	
6		Start **APDU application** by sending a SELECT command with AID1	
7		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
8		RF field is powered off	
9		The mobile sends EVT_FIELD_OFF	
10		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	No application is triggered

5.3.6.3 Test sequence No 3

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- ACRF is present but without any rule entry

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	

Step	Direction	Sequence	Expected Result
2		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
3		Close **GSMA_AC_Mobile_App_SP1_signed**	
4		RF field is powered on	
5		Mobile starts card emulation session	
6		Start **APDU application** by sending a SELECT command with AID1	
7		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
8		RF field is powered off	
9		The mobile sends EVT_FIELD_OFF	
10		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	No application is triggered

5.3.6.4 Test sequence No 4

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing an entry with a corrupted certificate (wrong length)

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
3		Close **GSMA_AC_Mobile_App_SP1_signed**	
4		RF field is powered on	

Step	Direction	Sequence	Expected Result
5		Mobile starts card emulation session	
6		Start **APDU application** by sending a SELECT command with AID1	
7		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
8		RF field is powered off	
9		The mobile sends EVT_FIELD_OFF	
10		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	No application is triggered

5.3.6.5 Test sequence No 5

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a specific target rule for AID1 and a path for one ACCF containing an entry with a corrupted certificate (original ACCF padded with two 0x00 bytes)

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID1" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
3		Close **GSMA_AC_Mobile_App_SP1_signed**	
4		RF field is powered on	
5		Mobile starts card emulation session	
6		Start **APDU application** by sending a SELECT command with AID1	
7		**APDU_TestApplication.cap** with AID1 sends response to SELECT	
8		RF field is powered off	

Step	Direction	Sequence	Expected Result
9		The mobile sends EVT_FIELD_OFF	
10		The **APDU_TestApplication.cap** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	No application is triggered

5.4 GP SE Access Control – GP Test Plan

Test Purpose

To ensure the device provide API for Access Control as per Global Platform Specification GPD_SE_Access_Control

Referenced requirement

- TS26_NFC_REQ_082
- TS26_NFC_REQ_083

Related Specs/Docs: –GlobalPlatform - SEAC DeviceSide Test Plan v1.0.6" [27]

The DUT shall pass the Test Cases with ID REQ from GlobalPlatform –GlobalPlatform - SEAC DeviceSide Test Plan v1.0.6" [27], the set of applicable test cases is referenced in Annex B.18.

6 Secure Element Access API

6.1 General overview

This chapter addresses the implementation of the Mobile Device APIs according to the SIMalliance *Open Mobile API* specification or equivalent. The objective is to verify mobile applications can access different Secure Elements in a mobile device such as SIMs.

The list of conformance requirements tested within this section is listed in the table in section 6.2.

6.2 Conformance requirements

TS26_NFC_REQ_047	OS implementations SHALL provide an API for communicating with the UICC
TS26_NFC_REQ_047.1	Communication with the UICC SHALL be done through the logical channels.
TS26_NFC_REQ_047.2	Communication with the UICC SHALL prevent access to basic channel (channel 0).
TS26_NFC_REQ_047.3	The API SHALL implement the SIMalliance Open Mobile API transport layer or provide an equivalent set of features.

6.3 Test Cases

6.3.1 SIMalliance APIs

Test Purpose

To ensure the DUT follows the SIMalliance specification for the Transport API part of the Open Mobile API.

Referenced requirement

- TS26_NFC_REQ_047
- TS26_NFC_REQ_047.1
- TS26_NFC_REQ_047.3

Related Specs/Do–s: SIMalliance - Open Mobile API specification [6]

The DUT shall pass the test cases referenced in annex B.1

6.3.2 Prevent access to basic channel.

Test Purpose

APDU APIs SHALL prevent access to basic channel (channel 0).

Referenced requirement

- TS26_NFC_REQ_047.2

Method of Test

For devices supporting the Open Mobile API, the DUT shall pass the Test Case ID7 in Clause 6.4.6 from Open Mobile API test specification, the full set of applicable test cases is referenced in Annex B.1.

6.3.3 VOID

6.3.4 VOID

6.3.5 VOID

7 Multiple Card Emulation Environment

7.1 General overview

This chapter addresses the requirements for Multiple Card Emulation Environment support when the device has the capacity to handle further Secure Elements to the UICC.

The list of conformance requirements tested within this section is listed in the table in section 7.2.

7.2 Conformance requirements

TS26_NFC_REQ_051	The mobile device software platform SHOULD expose an API to select and switch the Active CEE. A change in the Active CEE selection SHALL not imply a power cycle of the device, modem or the UICC.
TS26_NFC_REQ_053	Only one CEE SHALL be active at any one time.
TS26_NFC_REQ_054	If there is more than one CEE, the default Active CEE SHALL be the UICC.
TS26_NFC_REQ_116	The OS implementation SHALL provide an API that allows an application to enable & disable Card Emulation Mode
TS26_NFC_REQ_117	In Multiple Active CEE model, the UICC SHALL be one of the active CEEs.
TS26_NFC_REQ_118.1	At first power up or factory reset the device SHALL set the route for NFCA and NFCB technologies (as defined in NFC Forum specification) to the UICC.
TS26_NFC_REQ_118.2	At first power up or factory reset the device SHALL set the route for ISO_DEP protocol (as defined in NFC Forum specification) to the UICC.
TS26_NFC_REQ_118.3	At first power up or factory reset the device SHOULD set the Default AID route (AID routing mechanism is defined in NFC Forum specification) to the UICC. Note: The Default AID Route" is the route used by the NFC Controller when a NFC reader is selecting explicitly a NFC Service by its AID but no NFC application registered this AID.
TS26_NFC_REQ_119	The device MAY provide a mechanism for changing the route for technology and protocol. Note: No API details are provided at this stage. Further work will be progressed, to provide with the next release further details.

7.3 Test Cases

7.3.1 SE Availabilities from mobile applications

Test Purpose

Check that only one SE is available at a time from a mobile application

Referenced requirement

- TS26_NFC_REQ_053

Method of Test

Some precision are necessary for this requirement, test is FFS.

7.3.2 SE selection

Test Purpose

Check the DUT allows selecting the SE available over the contactless interface

Referenced requirement

- TS26_NFC_REQ_051

Method of Test

Initial Conditions

- **ReferenceApplication.cap** is installed with AID1 on the UICC
- **APDU Application** to send APDUs according to the reference transaction.

No application is installed on the other SE.

Procedure

7.3.2.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Using the **APDU application,** perform the reference transaction	Reference transaction is performed successfully
2		Select the other SE by any mean	The selection of another SE is possible
3		Using the **APDU application,** perform the reference transaction	Reference transaction fails (at the application selection).
4		Select the UICC by any mean	The selection of the UICC is possible. The list of all available SEs is displayed and the DUT doesn't power cycle the DUT, the modem or the UICC
5		Using the **APDU application,** perform the reference transaction	Reference transaction is performed successfully
Note: This test case considers only one additional SE. Test must be run for all other SE than the UICC if the DUT supports more than one additional SE.			

7.3.3 UICC as default SE

Test Purpose

Check that the UICC is defined as SE by default

Referenced requirement

- TS26_NFC_REQ_054

Method of Test

As some clarification is expected on this requirement, this test is FFS.

7.3.4 SE Selection API

Test Purpose

API to test not yet available

Referenced requirement

- TS26_NFC_REQ_051

As some clarification is expected on this requirement, this test is FFS.

7.3.5 SE Toggle switching API

Test Purpose

API to test not yet available

Referenced requirement

As some clarification is expected on this element, this test is FFS.

7.3.6 Switch for Card Emulation Mode

Test Purpose

Test that in Single Active Card Emulation Environment, there is an OS API that allows for enabling & disabling of Card Emulation Mode

Referenced requirement

- TS26_NFC_REQ_116

Method of Test

Initial Conditions

- ReferenceApplication.cap is installed
- APDU Application to send APDUs according to the reference transaction.

Procedure

7.3.6.1 Test sequence No 1

Initial Conditions

- NFC is enabled
- No AIDs are registered in the CLF routing table
- The active SE is the UICC
- The mobile application must be signed with one of the certificates stored in the PKCS#15 topology of the UICC

Step	Direction	Sequence	Expected Result
1		Call the API to enable Card Emulation Mode	
2		Using the **APDU application,** perform the reference transaction while the device and screen is on	Reference Transaction is passed.
3		Call the API to disable Card Emulation Mode	
4		Using the **APDU application,** perform the reference transaction while device and screen is on.	Reference transaction performed unsuccessfully

7.3.7 Multiple CE Environments

Test Purpose

Check the UICC is an active Card Emulation Environment in Multiple Card Emulation Environments models.

Referenced requirement

- TS26_NFC_REQ_117

Method of Test

Initial Conditions

- ReferenceApplication.cap is installed
- APDU Application to send APDUs according to the reference transaction.

Procedure

7.3.7.1 Test sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to UICC. Do not register any AIDs in the CLF routing table.	The default Card Emulation Environment is now UICC.

Step	Direction	Sequence	Expected Result
2		Using the **APDU application,** perform the reference transaction	Reference transaction passes as the CLF route table does not contain any explicit routes, the UICC applet selection by AID should be successful, due to the default/fallback (ISO_DEP) route in the CLF, which points to the UICC in this test

7.3.7.2 Test sequence No 2

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to HCE. Do not register any AIDs in the CLF routing table.	The default Card Emulation Environment is now HCE.
2		Using the **APDU application,** perform the reference transaction	Reference transaction fails as the CLF route table does not contain any explicit routes, the UICC applet selection by AID should fail, due to the default/fallback (ISO_DEP) route in the CLF, which points to the HCE in this test

7.3.7.3 Test sequence No 3

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to UICC. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now UICC.
2		Using the **APDU application,** perform the reference transaction while Device is on and Screen is on.	Reference transaction passes as the CLF route table does not contain any explicit routes, the UICC applet selection by AID should be successful, due to the default/fallback (ISO_DEP) route in the CLF, which points to the UICC in this test

7.3.7.4 Test sequence No 4

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to HCE. Register the AID of the UICC applet in the CLF routing table	The default Card Emulation Environment is now HCE.
2		Using the **APDU application,** perform the reference transaction while Device is on and Screen is on.	Reference transaction fails as the CLF route table does not contain any explicit routes, the UICC applet selection by AID should fail, due to the default/fallback (ISO_DEP) route in the CLF, which points to the HCE in this test

7.3.7.5 Test sequence No 5

Initial Conditions

- Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to UICC. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now UICC.
2		Also register the same AID to an OnHost (HCE) destination from reference APK #2	In cases of conflict, only when the device is on and the screen is available, a pop-up should appear.
3		Using the **APDU application,** perform the reference transaction and when the DUT prompts the user asking to select the desired application the tester selects the reference APK and performs the reference transaction again.	Reference transaction passes as there is a conflict of AIDs (the same AID is registered towards both UICC and HCE), the OS will prompt the user with a pop-up, asking to select the desired application. After the tester confirms the APDU application the selection shall go through to the UICC applet.

7.3.7.6 Test sequence No 6

Initial Conditions

- Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to HCE. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now HCE.
2		Also register the same AID to an On Host (HCE) destination from reference APK #2	In cases of conflict, only when the device is on and the screen is available, a pop-up should appear.
3		Using the **APDU application,** perform the reference transaction and when the DUT prompts the user asking to select the desired application the tester selects the reference APK and performs the reference transaction again.	Reference transaction passes as there is a conflict of AIDs (the same AID is registered towards both UICC and HCE), the OS will prompt the user with a pop-up, asking to select the desired application. After the tester confirms the APDU application the selection shall go through to the UICC applet.

7.3.7.7 Test sequence No 7

Initial Conditions

- Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to UICC. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now UICC.
2		Also register the same AID to an OnHost (HCE) destination from reference APK #2	“
3		Mark both applications as "payment" category and select the UICC APK in the Tap&Pay menu	UICC APK is selected as Tap&Pay Option.
4		Using the **APDU application,** perform the reference transaction	Reference transaction passes and the DUT will NOT prompt the user with a pop-up, since the UICC applet will answer to the AID Select.

7.3.7.8 Test sequence No 8

Initial Conditions

- -Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to HCE. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now HCE.
2		Also register the same AID to an OnHost (HCE) destination from reference APK #2	“
3		Mark both applications as "payment" category and select the UICC APK in the Tap&Pay menu	UICC APK is selected as Tap&Pay Option.
4		Using the **APDU application,** perform the reference transaction	Reference transaction passes and the DUT will NOT prompt the user with a pop-up, since the UICC applet will answer to the AID Select.

7.3.7.9 Test sequence No 9

Initial Conditions

- Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to UICC. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now UICC.
2		Also register the same AID to an OnHost (HCE) destination from reference APK #2	“
3		Mark both applications as "payment" category and select the HCE APK in the Tap&Pay menu	HCE APK is selected as Tap&Pay Option.
4		Using the **APDU application #2,** perform the reference transaction	Reference transaction passes and the DUT will NOT prompt the user with a pop-up, since the HCE applet will answer to the AID Select.

7.3.7.10 Test sequence No 10

Initial Conditions

- Reference APK#2 ready which is just a variant of the reference APK

Step	Direction	Sequence	Expected Result
1		Set the ISO_DEP route in the Android settings to HCE. Register the AID of the UICC applet in the CLF routing table.	The default Card Emulation Environment is now HCE.
2		Also register the same AID to an OnHost (HCE) destination from reference APK #2	“
3		Mark both applications as "payment" category and select the HCE APK in the Tap&Pay menu	HCE APK is selected as Tap&Pay Option.
4		Using the **APDU application #2,** perform the reference transaction	Reference transaction passes and the DUT will NOT prompt the user with a pop-up, since the HCE applet will answer to the AID Select.

7.3.8 Active Card Emulation in Multiple CE Environments / Card Emulation

Test Purpose

Test that in Multiple Active CEE, the UICC CEE is one of the active CEEs

Referenced requirement

- TS26_NFC_REQ_118.1
- TS26_NFC_REQ_118.2
- TS26_NFC_REQ_118.3
- TS26_NFC_REQ_119

Method of Test

Initial Conditions

- ReferenceApplication.cap is installed
- **APDU Application** to send APDUs according to the reference transaction.
- If the phone supports a mechanism to change the NFCA and/or NFCB route, then do a factory reset before this test.

Procedure

7.3.8.1 Test sequence No 1

Initial Conditions

If the device is not supporting multiple active CEE model the default CEE shall be the UICC

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCA	-
2		Use a contactless reader to access this applet	The access to the UICC applet is successful

7.3.8.2 Test sequence No 2

Initial Conditions

- The UICC contains a selectable applet

Note – Applets for NFCA & NFCB need to be defined in the test environment section.

- If the device is not supporting multiple active CEE model the default CEE shall be the UICC

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCB	-
2		Use a contactless reader to access this applet	The access to the UICC applet is successful

7.3.8.3 Test sequence No 3

Initial Conditions

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCA	-
2		Set the ISO_DEP route to the UICC using the mechanism offered by the device.	-
3		Use a contactless reader to access this applet	The access to the UICC applet is successful

7.3.8.4 Test sequence No 4

Initial Conditions

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCB	-
2		Set the ISO_DEP route to the UICC using the mechanism offered by the device	-
3		Use a contactless reader to access this applet	The access to the UICC applet is successful

7.3.8.5 Test sequence No 5

Initial Conditions

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCA	-
2		Set the ISO_DEP route to a CEE which is not the UICC using the mechanism offered by the device	-
3		Use a contactless reader to access this applet	The access to the UICC applet is successful

7.3.8.6 Test sequence No 6

Initial Conditions

Step	Direction	Sequence	Expected Result
1		Prepare an applet on the UICC, implicitly selectable via NFCB	-
2		Set the ISO_DEP route to a CEE which is not the UICC using the mechanism offered by the device	-
3		Use a contactless reader to access this applet	The access to the UICC applet is successful

8 UI Application triggering

8.1 General overview

This chapter addresses the UI application triggering. The aim is to ensure the NFC controller is able to trigger the appropriate UI application.

The list of conformance requirements tested within this section is listed in the table in section 8.2.

8.2 Conformance requirements

TS26_NFC_REQ_071	The device SHALL support HCI event EVT_TRANSACTION as per ETSI TS 102 622
TS26_NFC_REQ_073	The AID parameter SHALL be used during the process of triggering the UI application.

8.3 Test Cases

8.3.1 EVT_TRANSACTION

Test Purpose

To ensure the DUT correctly handles the EVT_TRANSACTION event as per the ETSI 102.622 specification

Referenced requirement

- TS26_NFC_REQ_071

Method of Test

Initial Conditions

Related Specs/Docs: ETSI TS 102 622 Rel 7. (latest)

Procedure

The DUT shall pass the Test Case 5.8.2.3.5.2 from ETSI TS 102 695-1, the full set of applicable test cases is referenced in Annex B4

8.3.2 Permissions

Test Purpose

To ensure the DUT correctly manages the Android mechanism of permissions relative to NFC

Referenced requirement

- TS26_NFC_REQ_071

Method of Test

Initial Conditions

Test applicable only for DUT with an Android OS

The application is registered for EVT_TRANSACTION handling from AID1 with *com.gsma.services.nfc.action.TRANSACTION_EVENT*

Procedure

8.3.2.1 Test sequence No 1

Initial Conditions
The following permission is declared in the Mobile Application manifest:
android.permission.NFC
com.gsma.services.nfc.permission.TRANSACTION_EVENT

Step	Direction	Sequence	Expected Result
1		Install the Android application	DUT shall prompt the user to authorize the application installation indicating the NFC permission

8.3.2.2 Test sequence No 2

Initial Conditions
The following permission is declared in the Mobile Application manifest:
com.gsma.services.nfc.permission.TRANSACTION_EVENT

The following permission is NOT declared in the Mobile Application manifest:
android.permission.NFC

Step	Direction	Sequence	Expected Result
1		Install the Android application	DUT does not require the user to authorize the NFC permission

8.3.2.3 Test sequence No 3

Initial Conditions
The following permission is declared in the Mobile Application manifest:
android.permission.NFC

The following permission is NOT declared in the Mobile Application manifest:
com.gsma.services.nfc.permission.TRANSACTION_EVENT

Step	Direction	Sequence	Expected Result
1		Install the Android application	DUT does not require the user to authorize the NFC permission

8.3.3 Intent management

Test Purpose
To ensure the DUT correctly manages the Android mechanism of intents

Referenced requirement
- TS26_NFC_REQ_071

- TS26_NFC_REQ_073

Method of Test

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully
- NFC is enabled in the DUT
- Two instances of the UICC application **APDU_TestApplication.cap** with AID1 and AID2 are selectable.

The application is registered for EVT_TRANSACTION handling from AID1 and AID2 with *com.gsma.services.nfc.action.TRANSACTION_EVENT*

- No applications should be started manually on the DUT

Procedure

8.3.3.1 Test sequence No 1

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	Start card emulation session	Card emulation session is started
4	User → PCD	Power off the field	
5	DUT → UICC	DUT sends EVT_FIELD_OFF	
6	UICC → DUT	UICC sends EVT_TRANSACTION with AID1	Application is triggered and receives the transaction event and URI format is the following: • nfc://secure:0/SE_Name/AID with • AID equals to AID1 • SE_Name according to SIMalliance open mobile API specification

8.3.3.2 Test sequence No 2

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	DUT starts card emulation session	Contactless Session is started
4	User → PCD	Power off the field	
5	DUT → UICC	DUT sends EVT_FIELD_OFF	
6	UICC → DUT	UICC sends EVT_TRANSACTION with AID2 with data (0x20 bytes long)	Application receives the transaction event with additional data (0x20 bytes long) retrieved from com.gsma.services.nfc.extra.DATA

8.3.4 Application's triggering order

Test Purpose

Check the launch, and order of launch of the applications; check that for the same HCI_Event, which calls 2 applications, only the first installed one answers, and that no application is launched when an event does not refer to any AID.

Referenced requirement

- TS26_NFC_REQ_073

Method of Test

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully
- NFC is enabled in the DUT

Install GSMA_Mobile_App_UIA #1 first

- **GSMA_Mobile_App_UIA #1** signed with a private key corresponding to test certificate #1 defined to be woken up on HCI Events from AID1 and AID2

Install **GSMA_Mobile_App_UIA #2**

- **GSMA_Mobile_App_UIA_#2** signed with a private key corresponding to test certificate #2 defined to be woken up on HCI Events from AID2

Three instances of the **APDU_TestApplication.cap** (respectively with AID1, AID2 and AID3).

- No applications should be started manually on the DUT

Procedure

8.3.4.1 Test sequence No 1

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing an empty hash condition

⇨ all applications have full access to all AID
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	DUT starts card emulation session	Contactless session is started
4	User → PCD	Power off the field	
5	DUT → UICC	DUT sends EVT_FIELD_OFF	
6	UICC → DUT	UICC sends EVT_TRANSACTION with AID1	**GSMA_Mobile_App_UIA_#1** is launched
7	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
8	User → DUT	Repeat steps 2 – 5	
9	UICC → DUT	UICC sends EVT_TRANSACTION with AID2	**GSMA_Mobile_App_UIA_#1** is launched
10	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
11	User → DUT	Repeat steps 2 – 5	
12	UICC → DUT	UICC sends EVT_TRANSACTION with AID3	No mobile application is launched

8.3.4.2 Test sequence No 2

Initial Conditions

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing an empty hash condition

⇨ all applications have full access to all AID

Install another **GSMA_Mobile_App_UIA_#3**
- **GSMA_Mobile_App_UIA_#3** signed with a private key corresponding to test certificate #1 defined to be woken up on HCI Events from all AIDs
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	DUT starts card emulation session	Contactless session is started
4	User → PCD	Power off the field	
5	DUT → UICC	DUT sends EVT_FIELD_OFF	
6		UICC sends EVT_TRANSACTION with AID1	**GSMA_Mobile_App_UIA_#1** is launched
7	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
8	User → DUT	Repeat steps 2 – 5	
9	UICC → DUT	UICC sends EVT_TRANSACTION with AID2	**GSMA_Mobile_App_UIA_#1**is launched
10	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
11	User → DUT	Repeat steps 2 – 5	
12	UICC → DUT	UICC sends EVT_TRANSACTION with AID3	**GSMA_Mobile_App_UIA_#3**is launched

8.3.4.3 Test sequence No 3

Initial Conditions
The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing SP1 hash condition
  AID2 is always accessible, no access allowed for any other AID
  The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	DUT starts card emulation session	Contactless session is started
4	User → PCD	Power off the field	
5	DUT → UICC	DUT sends EVT_FIELD_OFF	
6	UICC → DUT	UICC sends EVT_TRANSACTION with AID1	No mobile application is launched
7	User → DUT	Repeat steps 2 – 5	
8	UICC → DUT	UICC sends EVT TRANSACTION with AID2	**GSMA_Mobile_App_UIA_#2** is launched
9	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
10	User → DUT	Repeat steps 2 – 5	
11	UICC → DUT	UICC sends EVT TRANSACTION with AID3	No mobile application is launched

8.3.5 Triggering on HCI event EVT_CARD_DEACTIVATED

Test Purpose
To ensure the device is able to launch the mobile application on EVT_TRANSACTION when a HCI EVT_CARD_DEACTIVATED event is processed by the CLF.

Referenced requirement
- *TS26_NFC_REQ_071*

- *TS26_NFC_REQ_073*

Method of Test

Initial Conditions

- The DUT is powered on
- HCI initialisation has been performed successfully
- NFC is enabled in the DUT
- **APDU_TestApplication_card_deactivated** is installed on the UICC and is selectable with AID1
- **MobileApplication** is installed on the DUT and is launched on EVT_TRANSACTION from AID1
- No applications should be started manually on the DUT.

Procedure

8.3.5.1 Test sequence No 1

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on.	None
2	User → PCD	Power on the field	None
3	PCD → DUT DUT → UICC	Starts card emulation session	Card Emulation Session is started correctly
4	PCD → DUT DUT → UICC	Start the **APDU application**, send a SELECT_BY_DF_name command with AID1	Status word 90 00 is returned
5	PCD → DUT	The **APDU application** sends a contactless DESELECT	None
6	DUT → UICC	The DUT sends EVT_ CARD_DEACTIVED	None
7	UICC → DUT	UICC sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	MobileApplication is launched and the intent received indicates reception of EVT_TRANSACTION with AID1

8.3.6 Triggering on HCI event EVT_FIELD_OFF

Test Purpose
To ensure the device is able to launch the mobile application on EVT_TRANSACTION when a HCI EVT_FIELD_OFF event is processed by the CLF

Referenced requirement

- *TS26_NFC_REQ_071*

- TS26_NFC_REQ_073

Method of Test

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully
- NFC is enabled in the DUT
- **APDU_TestApplication** is installed on the UICC and is selectable with AID1
- **MobileApplication** is installed on the DUT and is launched on EVT_TRANSACTION from AID1
- No applications should be started manually on the DUT.
- APDU_TestApplication is not selected on UICC.

Procedure

8.3.6.1 Test sequence No 1

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, the DUT is placed in the area where the field will be powered on	None
2	User → PCD	Power on the field	None
3	PCD → DUT DUT → UICC	Starts card emulation session	Card Emulation Session is started
4	PCD → DUT DUT → UICC	Start the **APDU application**, send a SELECT_BY_DF_name command with AID1	Status word 90 00 is returned
5	User → DUT	Power off the RF field	None
6	DUT → UICC	DUT sends EVT_ FIELD_OFF	None
7	UICC → DUT	The **UICC** sends EVT_TRANSACTION to **GSMA_AC_Mobile_App_SP1_signed**	**MobileApplication** is launched and the intent received indicates the reception of EVT_TRANSACTION with AID1

9 VOID

10 VOID

11 Mobile Device APN management

11.1 General overview

This chapter addresses the APN management by the device according to ETSI specifications.

The list of conformance requirements tested within this section is listed in the table in section 11.2.

11.2 Conformance requirements

TS26_NFC_REQ_075	For mobile devices supporting multiple APNs, the device SHALL be able to set-up a SIM OTA channel using the APN information that is provided in the OPEN CHANNEL command.
TS26_NFC_REQ_076	For devices which are configured as "Always-ON" and only support a single APN, the APN information provided in the OPEN CHANNEL command SHALL be ignored and the device SHALL use the device default APN.
TS26_NFC_REQ_077	If the APN information provided by the network in the OPEN CHANNEL command is empty the device SHALL always use the device default APN

11.3 Test Cases

11.3.1 OPEN CHANNEL

Test Purpose
To verify OPEN CHANNEL related to Default APN Always

Referenced requirement

- TS26_NFC_REQ_075
- TS26_NFC_REQ_076
- TS26_NFC_REQ_077

Method of Test

Initial Conditions

One default APN is configured on the DUT and the related PDN connection to this APN has been already established.

Procedure

11.3.1.1 Test Sequence No 1: (OPEN CHANNEL - Default APN Always-ON - Multiple APN supported - with different APN)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.1.1	
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.1.1	
5	ME → user	The ME may display channel opening information	
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 11.1.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: ME

Bearer

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

Buffer size: 1024

Network access name (APN):web99.test-nfc1.com

UICC/ME interface transport level

Transport format: UDP

Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 11.1.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel status: Channel identifier 1 and link established or PDP context activated

Bearer Description:

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

11.3.1.2 Buffer size 1024Test Sequence No 2: (OPEN CHANNEL - Default APN Always–ON - Only Single APN supported - with different APN)

Initial Conditions

None

Step		Direction	Sequence	Expected Result
1		ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2		UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.2.1	
3		ME → UICC	FETCH	
4		UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.2.1	
5		ME → user	The ME may display channel opening information	
6		ME → USS	The terminal shall not send a PDP context activation request	
7		ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.2.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 11.2.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: ME

Bearer

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

Buffer size: 1024

Network access name (APN): web99.test-nfc1.com

UICC/ME interface transport level

Transport format: UDP

Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 11.2.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel status: Channel identifier 1 and link established or PDP context activated

Bearer Description:

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

Buffer size 1024

11.3.1.3 Test Sequence No 3: (OPEN CHANNEL - Default APN Always-ON - APN empty)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.3.1	
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.3.1	
5	ME → user	The ME may display channel opening information	
6	ME → USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.3.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 11.3.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: ME

Bearer

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

Buffer size: 1024

UICC/ME interface transport level

Transport format: UDP

Port number: 44444

Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 11.3.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel status Channel identifier 1 and link established or PDP context activated

Bearer description

Bearer type: GPRS/UTRAN packet service/ E-UTRAN

Bearer parameter:

Precedence Class: 02

Delay Class: 04

Reliability Class: 02

Peak throughput class: 05

Mean throughput class: 31

Packet data protocol: 02 (IP)

Buffer

Buffer size: 1024

11.3.1.4 Test Sequence No 4: (OPEN CHANNEL - Default APN Always-ON - APN empty- Default Bearer Type used)

Initial Conditions

None

Step	Direction	Message / Action	Comments
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.4.1	
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.4.1	
5	ME → user	The ME may display channel opening information	
6	ME →USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.4.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 11.4.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: ME

Bearer

Bearer type: Default Bearer Type

Buffer

Buffer size: 1024

UICC/ME interface transport level

Transport format: UDP

Port number: 44444

Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 11.4.1

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel status Channel identifier 1 and link established or PDP context activated

Bearer description

Bearer type: Default Bearer Type

Buffer

Buffer size: 1024

11.3.2 CLOSE CHANNEL

Test Purpose

To verify CLOSE CHANNEL related to Default APN Always-ON

Referenced requirement

- TS26_NFC_REQ_075
- TS26_NFC_REQ_076
- TS26_NFC_REQ_077

Method of Test

Initial Conditions

One default APN is configured on the DUT and the related PDN connection to this APN has been already established.

Procedure

11.3.2.1 Test Sequence No 1: (CLOSE CHANNEL - Default APN Always-ON - Multiple APN supported - with different APN- SUCCESSFUL)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.1.1	See OPEN CHANNEL SEQ 11.1.1
3	ME → UICC	FETCH	

Step	Direction	Sequence	Expected Result
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.1.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.1.1	[Command performed successfully]
9	UICC → ME	PROACTIVE COMMAND: PENDING CLOSE CHANNEL 11.1.1	
10	ME → UICC	FETCH	
11	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 11.1.1	
12	ME → USS	PDP context deactivation request	
13	USS → ME	PDP context deactivation accept	
14	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 11.1.1	[Command performed successfully]

PROACTIVE COMMAND: CLOSE CHANNEL 11.1.1

Logically:

Command details

Command number: 1

Command type: CLOSE CHANNEL

Command qualifier: RFU

Device identities

Source device: UICC

Destination device: Channel 1

TERMINAL RESPONSE: CLOSE CHANNEL 1.1

Logically:

Command details

Command number: 1

Command type: CLOSE CHANNEL

Command qualifier: RFU

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

11.3.2.2 Test Sequence No 2: (CLOSE CHANNEL - Default APN Always–ON - Only Single APN supported - with different APN- SUCCESSFUL)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.2.1	Please See OPEN CHANNEL SEQ 11.2.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.2.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.2.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 11.1.1	
11	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 11.1.1	[Command performed successfully]

11.3.2.3 Test Sequence No 3: (CLOSE CHANNEL - Default APN Always-ON - APN empty- SUCCESSFUL)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.3.1	See OPEN CHANNEL SEQ 11.3.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.3.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.3.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 11.1.1	
11	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 11.1.1	[Command performed successfully]

11.3.2.4 Test Sequence No 4: (CLOSE CHANNEL - Default APN Always-ON - APN empty- SUCCESSFUL- Default Bearer Type Used)

Initial Conditions

None

Step	Direction	MESSAGE / Action	Comments
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.4.1	See OPEN CHANNEL SEQ 11.4.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.4.1	
5	ME → USER	The ME may display channel opening information	
6	ME →USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.4.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 11.1.1	
11	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 11.1.1	[Command performed successfully]

11.3.3 RECEIVE DATA

Test Purpose

To verify RECEIVE DATA related to Default APN Always-ON

Referenced requirement

- TS26_NFC_REQ_075
- TS26_NFC_REQ_076
- TS26_NFC_REQ_077

Method of Test

Initial Conditions

One default APN is configured on the DUT and the related PDN connection to this APN has been already established.

Procedure

11.3.3.1 Test Sequence No 1: (RECEIVE DATA - Default APN Always-ON - Multiple APN supported - with different APN)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 PENDING	
3	ME→ UICC	FETCH	
4	UICC→ ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1	
5	ME→ UICC	TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1	
6	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.1.1	See OPEN CHANNEL SEQ 11.1.1
7	ME → UICC	FETCH	
8	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.1.1	
9	ME → USER	The ME may display channel opening information	
10	ME → USS	PDP context activation request	
11	USS → ME	PDP context activation accept	
12	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.1.1	[Command performed successfully]
13	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
14	ME → UICC	FETCH	
15	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.1	
16	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
17	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.1	[Command performed successfully]

Step	Direction	Sequence	Expected Result
18	USS → ME	Transfer of 1024 Bytes of data to the M' through channel 1 using the ME's port number, which was retrieved in step 16	
19	ME – UICC	ENVELOPE: EVENT DOWNLOAD - Data available 11.1.1	(1024 Bytes of data in the ME buffer)
20	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.1	
21	ME → UICC	FETCH	
22	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.1	205 Bytes
23	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.1	
24	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.2	
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.2	205 Bytes
27	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.2	
28	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.3	
29	ME → UICC	FETCH	
30	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.3	205 Bytes
31	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.3	
32	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.4	
33	ME → UICC	FETCH	
34	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.4	205 Bytes
35	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.4	
36	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.5	
37	ME → UICC	FETCH	
38	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.5	204 Bytes

Step	Direction	Sequence	Expected Result
39	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.5	

PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1

Logically:

Command details
- Command number: 1
- Command type: SET UP EVENT LIST
- Command qualifier: RFU

Device identities
- Source device: UICC
- Destination device: ME

Event list: Data available

TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1

Logically:

Command details
- Command number: 1
- Command type: SET UP EVENT LIST
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

PROACTIVE COMMAND: SEND DATA 11.1.1

Logically:

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Send Immediately

Device identities
- Source device: UICC
- Destination device: Channel 1

Channel Data
- Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 11.1.1

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Send Immediately

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: More than 255 bytes of space available in the Tx buff–r

ENVELOPE: EVENT DOWNLOAD - Data available 11.1.1

Logically:

Event list

Event: Data available

Device identities

Source device: ME

Destination device: UICC

Channel status

Channel status: Channel 1 open, link established

Channel Data Length

Channel data length: FF (more than 255 bytes are available)

PROACTIVE COMMAND: RECEIVE DATA 11.1.1

Logically:

Command details

Command number: 1

Command type: RECEIVE DATA

Command qualifier: RFU

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data Length

Channel Data Length: 205

PROACTIVE COMMAND: RECEIVE DATA 11.1.2

Logically:

Command details
Command number: 2
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 205

PROACTIVE COMMAND: RECEIVE DATA 11.1.3

Logically:

Command details
Command number: 3
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 205

PROACTIVE COMMAND: RECEIVE DATA 11.1.4

Logically:

Command details
Command number: 4
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 205

PROACTIVE COMMAND: RECEIVE DATA 11.1.5

Logically:

Command details
Command number: 5
Command type: RECEIVE DATA

Command qualifier: RFU

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data Length

Channel Data Length: 204

TERMINAL RESPONSE: RECEIVE DATA 11.1.1

Logically:

Command details

Command number: 1

Command type: RECEIVE DATA

Command qualifier: RFU

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel Data: 00 01 02 .. CC (205 Bytes of data)

Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 11.1.2

Logically:

Command details

Command number: 2

Command type: RECEIVE DATA

Command qualifier: RFU

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel Data: CD CE CF .. FF 00 01 .. 99(205 Bytes of data)

Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 11.1.3

Logically:

Command details

Command number: 3

Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data: 9A 9B .. FF 00 01 – 66 (205 Bytes of data)
Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 11.1.4

Logically:

Command details
Command number: 4
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data : 67 68 .. FF 00 01 .. 33 (205 Bytes of data)
Channel data length: CC

TERMINAL RESPONSE: RECEIVE DATA 11.1.5

Logically:

Command details
Command number: 5
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data: 34 35 .. FF (204 Bytes of data)
Channel data length: 00

11.3.3.2 Test Sequence No 2: (RECEIVE DATA - Default APN Always–ON - Only Single APN supported - with different APN)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 PENDING	
3	ME→ UICC	FETCH	
4	UICC→ ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1	
5	ME→ UICC	TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1	
6	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.2.1	See OPEN CHANNEL SEQ 11.2.1
7	ME → UICC	FETCH	
8	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.2.1	
9	ME → USER	The ME may display channel opening information	
10	ME → USS	The terminal shall not send a PDP context activation request	
11	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.2.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.1	

Step	Direction	Sequence	Expected Result
15	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
16	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.1	[Command performed successfully]
17	USS → ME	Transfer of 1024 Bytes of data to the M' through channel 1 using the ME's port number, which was retrieved in step 15	
18	ME – UICC	ENVELOPE: EVENT DOWNLOAD - Data available 11.1.1	(1024 Bytes of data in the ME buffer)
19	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.1	
20	ME → UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.1	205 Bytes
22	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.1	
23	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.2	
24	ME → UICC	FETCH	
25	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.2	205 Bytes
26	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.2	
27	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.3	
28	ME → UICC	FETCH	
29	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.3	205 Bytes
30	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.3	

Step	Direction	Sequence	Expected Result
31	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.4	
32	ME → UICC	FETCH	
33	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.4	205 Bytes
34	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.4	
35	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.5	
36	ME → UICC	FETCH	
37	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.5	204 Bytes
38	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.5	

11.3.3.3 Test Sequence No 3: (RECEIVE DATA - Default APN Always-ON - APN empty)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 PENDING	
3	ME→ UICC	FETCH	
4	UICC→ ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1	
5	ME→ UICC	TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1	

Step	Direction	Sequence	Expected Result
6	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.3.1	See OPEN CHAANEL SEQ 11.3.1
7	ME → UICC	FETCH	
8	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.3.1	
9	ME → USER	The ME may display channel opening information	
10	ME → USS	The terminal shall not send a PDP context activation request	
11	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.3.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.1	
15	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
16	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.1	[Command performed successfully]
17	USS → ME	Transfer of 1024 Bytes of data to the M' through channel 1 using the ME's port number, which was retrieved in step 15	
18	ME – UICC	ENVELOPE: EVENT DOWNLOAD - Data available 11.1.1	(1024 Bytes of data in the ME buffer)
19	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.1	
20	ME → UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.1	205 Bytes

Step	Direction	Sequence	Expected Result
22	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.1	
23	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.2	
24	ME → UICC	FETCH	
25	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.2	205 Bytes
26	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.2	
27	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.3	
28	ME → UICC	FETCH	
29	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.3	205 Bytes
30	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.3	
31	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.4	
32	ME → UICC	FETCH	
33	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.4	205 Bytes
34	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.4	
35	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.5	
36	ME → UICC	FETCH	
37	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.5	204 Bytes
38	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.5	

11.3.3.4 Test Sequence No 4: (RECEIVE DATA - Default APN Always-ON - APN empty-Default Bearer Type used)

Initial Conditions

None

Step	Direction	Message / Action	Comments
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 PENDING	
3	ME→ UICC	FETCH	
4	UICC→ ME	PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1	
5	ME→ UICC	TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1	
6	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.4.1	See OPEN CHAANEL SEQ 11.4.1
7	ME → UICC	FETCH	
8	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.4.1	
9	ME → USER	The ME may display channel opening information	
10	ME →USS	The terminal shall not send a PDP context activation request	
11	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.4.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.1	
15	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
16	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.1	[Command performed successfully]
17	USS → ME	Transfer of 1024 Bytes of data to the M’ through channel 1 using the ME's port number, which was retrieved in step 15	
18	ME – UICC	ENVELOPE: EVENT DOWNLOAD - Data available 11.1.1	(1024 Bytes of data in the ME buffer)
19	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.1	

20	ME → UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.1	205 Bytes
22	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.1	
23	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.2	
24	ME → UICC	FETCH	
25	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.2	205 Bytes
26	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.2	
27	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.3	
28	ME → UICC	FETCH	
29	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.3	205 Bytes
30	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.3	
31	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.4	
32	ME → UICC	FETCH	
33	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.4	205 Bytes
34	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.4	
35	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 11.1.5	
36	ME → UICC	FETCH	
37	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 11.1.5	204 Bytes
38	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 11.1.5	

11.3.4 SEND DATA

Test Purpose

To verify SEND DATA related to Default APN Always-ON

Referenced requirement

- TS26_NFC_REQ_075
- TS26_NFC_REQ_076
- TS26_NFC_REQ_077

Method of Test

Initial Conditions

One default APN is configured on the DUT and the related PDN connection to this APN has been already established.

11.3.4.1 Test Sequence No 1: (SEND DATA - Default APN Always-ON - Multiple APN supported -with different APN- BUFFER FULLY USED)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.1.1	See OPEN CHANNEL SEQ 11.1.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.1.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.1.1	[Command performed successfully]
9	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
10	ME → UICC	FETCH	
11	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.1	Send 1024 Bytes of data by packet of 200 Bytes
12	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.1	[Command performed successfully]
13	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.2	
14	ME → UICC	FETCH	
15	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.2	[205 Bytes]

Step	Direction	Sequence	Expected Result
16	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.2	[Command performed successfully]
17	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.3	
18	ME → UICC	FETCH	
19	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.3	[205 Bytes]
20	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.3	[Command performed successfully]
21	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.4	
22	ME → UICC	FETCH	
23	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.4	[205 Bytes]
24	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.4	[Command performed successfully]
25	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.5	
26	ME → UICC	FETCH	
27	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[204 Bytes]
28	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
29	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.5	[Command performed successfully]

PROACTIVE COMMAND: SEND DATA 11.1.1

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data

Channel Data: 00 01 02 .. CC (205 Bytes of data)

TERMINAL RESPONSE: SEND DATA 11.1.1

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 11.1.2

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data

Channel Data: CD CE CF .. FF 00 01 .. 99(205 Bytes of data

TERMINAL RESPONSE: SEND DATA 11.1.2

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 11.1.3

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data

Channel Data: 9A 9B .. FF 00 01 .. 66 (205 Bytes of data)

TERMINAL RESPONSE: SEND DATA 11.1.3

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 11.1.4

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data

Channel Data : 67 68 .. FF 00 01 .. 33 (205 Bytes of data)

TERMINAL RESPONSE: SEND DATA 11.1.4

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Store mode

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: 204 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 11.1.5

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Send Immediately

Device identities

Source device: UICC

Destination device: Channel 1

Channel Data

Channel Data: 34 35 .. FF (204 Bytes of data)

TERMINAL RESPONSE: SEND DATA 11.1.5

Logically:

Command details

Command number: 1

Command type: SEND DATA

Command qualifier: Send Immediately

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Channel data length: More than 255 bytes of space available in the Tx buffer

11.3.4.2 Test Sequence No 2: (SEND DATA - Default APN Always–ON - Only Single APN supported - with different APN- BUFFER FULLY USED)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.2.1	SEE OPEN CHANNEL SEQ 11.2.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.2.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.2.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.1	Send 1024 Bytes of data by packet of 205 Bytes
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.2	
13	ME → UICC	FETCH	

Step	Direction	Sequence	Expected Result
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.2	[200 Bytes]
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.3	[200 Bytes]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.3	[Command performed successfully]
20	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.4	
21	ME → UICC	FETCH	
22	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.4	[200 Bytes]
23	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.4	[Command performed successfully]
24	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.5	
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[200 Bytes]
27	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
28	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.5	[Command performed successfully]

11.3.4.3 Test Sequence No 3: (SEND DATA - Default APN Always-ON - APN empty- BUFFER FULLY USED)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.3.1	See OPEN CHANNEL SEQ 11.3.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.3.1	
5	ME → USER	The ME may display channel opening information	
6	ME → USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.3.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.1	Send 1024 Bytes of data by packet of 205 Bytes
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.2	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.2	[205 Bytes]
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.3	[205 Bytes]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.3	[Command performed successfully]
20	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.4	
21	ME → UICC	FETCH	

Step	Direction	Sequence	Expected Result
22	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.4	[205 Bytes]
23	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.4	[Command performed successfully]
24	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.5	
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[204 Bytes]
27	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
28	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.5	[Command performed successfully]

11.3.4.4 Test Sequence No 4: (SEND DATA - Default APN Always-ON - APN empty- BUFFER FULLY USED- Default Bearer Type used)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	ME is connected to the USS and the first PDN to the APN for “Always on connection” (web.network.com) has been already established.	Indication to the test operator required to configure the ME for the establishment of the first PDN connection to the desired APN after registration.
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 11.4.1	See OPEN CHANNEL SEQ 11.4.1
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 11.4.1	
5	ME → USER	The ME may display channel opening information	
6	ME →USS	The terminal shall not send a PDP context activation request	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 11.4.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.1	Send 1024 Bytes of data by packet of 205 Bytes
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.2	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.2	[205 Bytes]
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.3	[205 Bytes]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.3	[Command performed successfully]
20	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.4	

21	ME → UICC	FETCH	
22	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 11.1.4	[205 Bytes]
23	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 11.1.4	[Command performed successfully]
24	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 11.1.5	
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[204 Bytes]
27	ME → USS	Transfer of 1024 Bytes of data to the USS through channel 1	
28	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 11.1.5	[Command performed successfully]

12 Remote Management of NFC Services

12.1 General overview

This chapter addresses the remote management of NFC services. The objective is to ensure that the device allows remote application management according to GSM requirements.

The test cases are grouped in a sub section testing the basic remote management functions of the device and a sub section covering use cases with approach to handle end-2-end functionalities.

The list of conformance requirements tested within this section is listed in the table in section 12.2.

12.2 Conformance requirements

TS26_NFC_REQ_078	The mobile device SHALL support BIP in UICC client mode for UDP.
TS26_NFC_REQ_079	The mobile device SHALL support BIP in UICC client mode for TCP.
TS26_NFC_REQ_080	The mobile device SHALL support two concurrent channels, BIP in UICC client mode.
TS26_NFC_REQ_081	The mobile device SHALL support SMS push (per ETSI TS 102.226 and 3GPP TS 31.116) to establish an open BIP channel as per ETSI TS 102.223 Open Channel Command
TS26_NFC_REQ_088	The device SHALL support the letter class “e” with the following commands and events[4]: • Proactive command: OPEN CHANNEL ((UICC in client mode and with the support of UDP/TCP bearer)) • Proactive command: CLOSE CHANNEL • Proactive command: RECEIVE DATA • Proactive command: SEND DATA • Proactive command: G ET CHANNEL STATUS • Event download: Data available • Event download: Channel status
TS26_NFC_REQ_120	The device SHALL support the BIP session regardless of incoming or outgoing calls, incoming or outgoing MMS, SMS. Note: This is not applicable if the device is on a 2G network

12.3 Basic Remote Management

12.3.1 General overview

This section addresses the testing of the Bearer Independent protocol used in remote management of NFC services.

The list of conformance requirements tested within this section is listed in the table in section 12.3.2.

12.3.2 Conformance requirements

TS26_NFC_REQ_078	The mobile device SHALL support BIP in UICC client mode for UDP.

TS26_NFC_REQ_079	The mobile device SHALL support BIP in UICC client mode for TCP.
TS26_NFC_REQ_080	The mobile device SHALL support two concurrent channels, BIP in UICC client mode.
TS26_NFC_REQ_081	The mobile device SHALL support SMS push (per ETSI TS 102.226) to establish an open BIP channel as per ETSI TS 102.223 Open Channel Command
TS26_NFC_REQ_088	The device SHALL support the letter class "e" with the following commands and events[4]: • Proactive command: OPEN CHANNEL ((UICC in client mode and with the support of UDP/TCP bearer)) • Proactive command: CLOSE CHANNEL • Proactive command: RECEIVE DATA • Proactive command: SEND DATA • Proactive command: G ET CHANNEL STATUS • Event download: Data available • Event download: Channel status

12.3.3 Test Cases

12.3.3.1 Remote management in BIP

Test Purpose

To ensure the DUT allows remote management over the Bearer Independent Protocol

Referenced requirement

- TS26_NFC_REQ_078
- TS26_NFC_REQ_079
- TS26_NFC_REQ_080
- TS26_NFC_REQ_088

Method of Test

Related Specs/Docs: ETSI TS 102 223

Procedure

The DUT shall pass all test cases referenced in Annex B.6 and in Table 19, Table 20, Table 21, Table 22, and Table 23.

12.3.3.2 OPEN CHANNEL

Test Purpose

To verify OPEN CHANNEL related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.2.1 Test Sequence No 1: (OPEN CHANNEL, No APN, immediate link establishment, Default Bearer for requested transport layer, No local address, no alpha identifier)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	USER → ME	Set and activate APN "TestGp.rs" in the terminal configuration if required	[see initial conditions]
2	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	
3	ME → UICC	FETCH	
4	UICC → ME	PROACTIVE COMMAND : OPEN CHANNEL 1.1	
5	ME → user	The ME may display channel opening information	
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE : OPEN CHANNEL 1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: ME
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
Command number: 1

Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400

12.3.3.2.2 Test sequence No 2: (OPEN CHANNEL, with APN, immediate link establishment, Default Bearer for requested transport layer, no alpha identifier)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 2.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 2.1	
4	ME → user	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 2.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 2.1
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: ME
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400
Network access name (APN): TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 2.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel status: Channel identifier 1 and link established or PDP context activated

Bearer Description:
- Bearer Type: Default Bearer for requested transport layer

Buffer
- Buffer size 1400

12.3.3.2.3 Test Sequence No 3: (OPEN CHANNEL, with alpha identifier, immediate link establishment, Default Bearer for requested transport layer)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 3.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 3.1	
4	ME → user	Confirmation phase with alpha ID	“Open ID”
5	user → ME	The user confirms	
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 3.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Alpha Identifier Open ID

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1400

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

12.3.3.2.4 Test Sequence No 4: (OPEN CHANNEL, with null alpha identifier, immediate link establishment, Default Bearer for requested transport layer)

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 4.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 4.1	
4	ME → user	Confirmation phase	[The ME should not give any information]
5	user → ME	The user confirms	[Only if the ME asks for user confirmation]
6	ME → USS	PDP context activation request	
7	USS → ME	PDP context activation accept	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 4.1
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source de vice: UICC
Destination device: ME
Alpha Identifier Null
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400
Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

12.3.3.2.5 Test Sequence No 5: (OPEN CHANNEL, command performed with modifications (buffer size), immediate link establishment, Default Bearer for requested transport layer)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 5.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 5.1	
4	ME → user	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 5.1	[Command performed with modification]

PROACTIVE COMMAND: OPEN CHANNEL 5.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 65535

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 5.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed with modifications (07)

Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
- Bearer type: Default Bearer for requested transport layer

Buffer

Buffer size: The buffer size TLV shall be attached and contain the value stated in table A.2/29 "Preferred buffer size supported by the terminal for Open Channel command".

12.3.3.2.6 Test Sequence No 6A: (OPEN CHANNEL, user rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier,)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 6.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 6.1	
4	ME → user	Confirmation phase with alpha ID	[The ME shall display “Open ID”]
5	user → ME	The user rejects	
6	ME → USS	No PDP context activation request is sent to the USS	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 6.1	[User did not accept the proactive command]

PROACTIVE COMMAND: OPEN CHANNEL 6.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: ME
Alpha Identifier "Open ID"
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400
Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 6.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities

Source device: ME
Destination device: UICC

Result
General Result: User did not accept the proactive command

Channel status The presence and content of this TLV shall not be verified

Bearer description
Bearer type: Default Bearer for requested transport layer

Buffer
Buffer size: Because the value depends in this case on the terminal's implementation, it shall be ignored.

12.3.3.2.7 Test Sequence No 6B: (OPEN CHANNEL, User rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 6.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 6.1	
4	ME → USS	PDP context activation request	
5	USS → ME	PDP context activation accept	
6	ME → user	Confirmation phase with alpha ID	[The ME shall display “Open ID”]
7	user → ME	The user rejects	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 6.1	[User did not accept the proactive command]

PROACTIVE COMMAND: OPEN CHANNEL 6.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment

Device identities
Source device: UICC
Destination device: ME

Alpha Identifier "Open ID"

Bearer
Bearer type: Default Bearer for requested transport layer

Buffer
Buffer size: 1400

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level

Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 6.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: ME
Destination device: UICC
Result
General Result: User did not accept the proactive command
Channel status The presence and content of this TLV shall not be verified
Bearer description
Bearer type: Default Bearer for requested transport layer

Buffer
Buffer size: Because the value depends in this case on the terminal's implementation, it shall be ignored.

12.3.3.3 CLOSE CHANNEL

Test Purpose

To verify CLOSE CHANNEL related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.3.1 Test Sequence No 1: (CLOSE CHANNEL, successful)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	

Step	Direction	Sequence	Expected Result
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 1.1	
11	ME → USS	PDP context deactivation request	
12	USS → ME	PDP context deactivation accept	
13	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1000

PROACTIVE COMMAND: CLOSE CHANNEL 1.1
Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1

TERMINAL RESPONSE: CLOSE CHANNEL 1.1
Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully

12.3.3.3.2 Test Sequence No 2: (CLOSE CHANNEL, with an invalid channel identifier)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 2.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 2.1	

Step	Direction	Sequence	Expected Result
11	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 2.1	[Invalid channel number]

PROACTIVE COMMAND: CLOSE CHANNEL 2.1

Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 2

TERMINAL RESPONSE: CLOSE CHANNEL 2.1

Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Bearer Independent Protocol error
Additional Result: Channel identifier not valid

12.3.3.3.3 Test Sequence No 3: (CLOSE CHANNEL, on an already closed channel)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 1.1	

Step	Direction	Sequence	Expected Result
11	ME → USS	PDP context deactivation request	
12	USS → ME	PDP context deactivation accept	
13	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 1.1	[Command performed successfully]
14	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 1.1	
15	ME → UICC	FETCH	
16	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 1.1	
17	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 3.1A or TERMINAL RESPONSE CLOSE CHANNEL 3.1B	[Channel closed] [Channel identifier invalid]

TERMINAL RESPONSE: CLOSE CHANNEL 3.1A

Command details
- Command number: 1
- Command type: CLOSE CHANNEL
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Bearer Independent Protocol error
- Additional Result: Channel closed

TERMINAL RESPONSE: CLOSE CHANNEL 3.1B

Command details
- Command number: 1
- Command type: CLOSE CHANNEL
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Bearer Independent Protocol error
- Additional Result: Channel identifier invalid

12.3.3.4 RECEIVE DATA

Test Purpose

To verify RECEIVE DATA related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.4.1 Test Sequence No 1: (RECEIVE DATA)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 1.1 PENDING	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 1.1	
4	ME → UICC	TERMINAL RESPONSE: SET UP EVENT LIST 1.1	
5	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
6	ME → UICC	FETCH	
7	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
8	ME → USER	The ME may display channel opening information	
9	ME → USS	PDP context activation request	
10	USS → ME	PDP context activation accept	
11	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 1.1	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 1.1	
15	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
16	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.1	[Command performed successfully]

Step	Direction	Sequence	Expected Result
17	USS → ME	Transfer of 1000 Bytes of data to the M' through channel 1 using the ME's port number, which was retrieved in step 15	
18	ME – UICC	ENVELOPE: EVENT DOWNLOAD - Data available 1.1	(1000 Bytes of data in the ME buffer)
19	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1.1	
20	ME → UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1.1	200 Bytes
22	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1.1	
23	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1.2	
24	ME → UICC	FETCH	
25	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1.2	200 Bytes
26	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1.2	
27	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1.3	
28	ME → UICC	FETCH	
29	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1.3	200 Bytes
30	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1.3	
31	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1.4	
32	ME → UICC	FETCH	
33	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1.4	200 Bytes
34	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1.4	
35	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1.5	

Step	Direction	Sequence	Expected Result
36	ME → UICC	FETCH	
37	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1.5	200 Bytes
38	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1.5	

PROACTIVE COMMAND: SET UP EVENT LIST 1.1

Command details
- Command number: 1
- Command type: SET UP EVENT LIST
- Command qualifier: RFU

Device identities
- Source device: UICC
- Destination device: ME

Event list: Data available

TERMINAL RESPONSE: SET UP EVENT LIST 1.1

Command details
- Command number: 1
- Command type: SET UP EVENT LIST
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

Network access name: TestGp.rs

Text String: UserLog (User login)

Text String: UserPwd (User password)

UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result

General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer

Buffer
Buffer size: 1000

PROACTIVE COMMAND: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

ENVELOPE: EVENT DOWNLOAD - Data available 1.1
Event list
Event: Data available
Device identities
Source device: ME
Destination device: UICC
Channel status
Channel status: Channel 1 open, link established
Channel Data Length
Channel data length: FF (more than 255 bytes are available)

PROACTIVE COMMAND: RECEIVE DATA 1.1
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

PROACTIVE COMMAND: RECEIVE DATA 1.2
Command details
Command number: 2
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1

Channel Data Length
Channel Data Length: 200

PROACTIVE COMMAND: RECEIVE DATA 1.3
Command details
Command number: 3
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

PROACTIVE COMMAND: RECEIVE DATA 1.4
Command details
Command number: 4
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

PROACTIVE COMMAND: RECEIVE DATA 1.5
Command details
Command number: 5
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

TERMINAL RESPONSE: RECEIVE DATA 1.1
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data : 00 01 02 .. C7 (200 Bytes of data)
Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 1.2
Command details
Command number: 2
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data : C8 C9 CA .. FF 00 01 .. 8F (200 Bytes of data)
Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 1.3

Command details
- Command number: 3
- Command type: RECEIVE DATA
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel Data : 90 91 .. FF 00 01 – 57 (200 Bytes of data)
Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 1.4

Command details
- Command number: 4
- Command type: RECEIVE DATA
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel Data : 58 59 .. FF 00 01 .. 1F (200 Bytes of data)
Channel data length: C8

TERMINAL RESPONSE: RECEIVE DATA 1.5

Command details
- Command number: 5
- Command type: RECEIVE DATA
- Command qualifier: RFU Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel Data: 20 21 .. E7 (200 Bytes of data)
Channel data length: 00

12.3.3.5 SEND DATA

Test Purpose

To verify SEND DATA related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type ‘03’= default bearer for requested transport layer.

Procedure

12.3.3.5.1 Test sequence No 1 (SEND DATA, immediate mode)

Initial Conditions

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 1.1	
11	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	
12	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL

Command qualifier: immediate link establishment
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1000

PROACTIVE COMMAND: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

12.3.3.5.2 Test sequence No 2 (SEND DATA, Store mode)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]

Step	Direction	Sequence	Expected Result
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 2.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 2.1	Send 500 Bytes of data (200 + 200 + 100)
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 2.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 2.2	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 2.2	[200 Bytes]
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 2.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 2.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (Immediate mode) 2.3	[100 Bytes]
19	ME → USS	Transfer of 500 Bytes of data to the USS through channel 1	
20	ME → UICC	TERMINAL RESPONSE: SEND DATA (Immediate mode) 2.3	[Command performed successfully]

PROACTIVE COMMAND: SEND DATA 2.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: UICC
- Destination device: Channel 1

Channel Data
- Channel Data : 00 01 .. C7 (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 2.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: ME
- Destination device: UICC

Result

General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 2.2
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data : C8 C9 .. FF 00 01 .. 8F (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 2.2
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 2.3
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Immediate mode
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data : 90 91 .. F3 (100 Bytes of data)

TERMINAL RESPONSE: SEND DATA 2.3
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Immediate mode
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

12.3.3.5.3 Test Sequence No 3: (SEND DATA, Tx buffer fully used, Store mode)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions

Step	Direction	Sequence	Expected Result
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.1	Send 1000 Bytes of data by packet of 200 Bytes
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.2	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.2	[200 Bytes]
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.3	[200 Bytes]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.3	[Command performed successfully]
20	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.4	
21	ME → UICC	FETCH	
22	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.4	[200 Bytes]

Step	Direction	Sequence	Expected Result
23	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.4	[Command performed successfully]
24	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.5	
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 3.5	[200 Bytes]
27	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
28	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 3.5	[Command performed successfully]

PROACTIVE COMMAND: SEND DATA 3.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: UICC
- Destination device: Channel 1

Channel Data
- Channel Data : 00 01 02 .. C7 (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 3.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully
- Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 3.2

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: UICC
- Destination device: Channel 1

Channel Data
- Channel Data : C8 C9 CA .. FF 00 01 .. 8F (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 3.2

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Store mode

Device identities
- Source device: ME

Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 3.3
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data : 90 91 .. FF 00 01 .. 57 (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 3.3
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 3.4
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data : 58 59 .. FF 00 01 .. 1F (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 3.4
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Store mode
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully

Channel data length: 200 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 3.5
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data

Channel Data: 20 21 .. E7 (200 Bytes of data)

TERMINAL RESPONSE: SEND DATA 3.5

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Send Immediately

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully
- Channel data length: More than 255 bytes of space available in the Tx buffer

12.3.3.5.4 Test Sequence No 4: (SEND DATA, 2 consecutive SEND DATA Store mode)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1..1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.1	Send 1000 Bytes of data by packet of 200 Bytes
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.1	[Command performed successfully]
12	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.2	
13	ME → UICC	FETCH	
14	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.2	[200 Bytes]

Step	Direction	Sequence	Expected Result
15	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.2	[Command performed successfully]
16	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.3	
17	ME → UICC	FETCH	
18	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.3	[200 Bytes]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.3	[Command performed successfully]
20	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.4	
21	ME → UICC	FETCH	
22	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.4	[200 Bytes]
23	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.4	[Command performed successfully]
24	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.5	…
25	ME → UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 3.5	[200 Bytes]
27	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
28	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 3.5	[Command performed successfully]
29	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.1	
30	ME → UICC	FETCH	
31	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.1	Send 1000 Bytes of data by packet of 200 Bytes
32	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.1	[Command performed successfully]
33	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.2	
34	ME → UICC	FETCH	
35	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.2	[200 Bytes]

Step	Direction	Sequence	Expected Result
36	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.2	[Command performed successfully]
37	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.3	
38	ME → UICC	FETCH	
39	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.3	[200 Bytes]
40	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.3	[Command performed successfully]
41	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.4	
42	ME → UICC	FETCH	
43	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 3.4	[200 Bytes]
44	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.4	[Command performed successfully]
45	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 3.5	…
46	ME → UICC	FETCH	
47	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 3.5	[200 Bytes]
48	ME → USS	Transfer of 1000 Bytes of data to the USS through channel 1	
49	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 3.5	[Command performed successfully]

12.3.3.5.5 Test Sequence No 5: (SEND DATA, immediate mode with a bad channel identifier)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	

Step	Direction	Sequence	Expected Result
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 5.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 5.1	
11	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 5.1	[Invalid channel number]

PROACTIVE COMMAND: SEND DATA 5.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Send Immediately

Device identities
- Source device: UICC
- Destination device: Channel 2

Channel Data
- Channel Data : 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 5.1

Command details
- Command number: 1
- Command type: SEND DATA
- Command qualifier: Send Immediately

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Bearer Independent Protocol error (3A)
- Additional Result: Channel identifier not valid (03)

12.3.3.6 GET CHANNEL STATUS

Test Purpose

To verify GET CHANNEL STATUS related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.6.1 Test Sequence No 1: (GET STATUS, without any BIP channel opened)

Initial Conditions

No channel has been opened.

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: GET CHANNEL STATUS 1.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: GET STATUS 1.1	
4	ME → UICC	TERMINAL RESPONSE GET STATUS 1.1A Or TERMINAL RESPONSE: GET STATUS 1.1B Or TERMINAL RESPONSE: GET STATUS 1.1C	[Command performed successfully]

PROACTIVE COMMAND: GET STATUS 1.1

Command details
- Command number: 1
- Command type: GET STATUS
- Command qualifier: RFU

Device identities
- Source device: UICC
- Destination device: ME

TERMINAL RESPONSE: GET STATUS 1.1A

Command details
- Command number: 1
- Command type: GET STATUS
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

TERMINAL RESPONSE: GET STATUS 1.1B

Command details
- Command number: 1
- Command type: GET STATUS
- Command qualifier: RFU

Device identities
- Source device: ME

Destination device: UICC
Result
General Result: Command performed successfully
Channel status
Channel status: No Channel available, link not established or PDP context not activated

TERMINAL RESPONSE: GET STATUS 1.1C
Command details
Command number: 1
Command type: GET STATUS
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status
Channel 1 status: Channel identifier 1, Link not established or PDP context not activated
Channel 2 status: Channel identifier 2, Link not established or PDP context not activated
.

Channel n status: Channel identifier n, Link not established or PDP context not activated
The number of channel status data objects shall be same as the number of channels(n) supported by the ME

12.3.3.6.2 Test Sequence No 2: (GET STATUS, with a BIP channel currently opened)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
4	ME → USS	PDP context activation request	
5	USS → ME	PDP context activation accept	
6	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
7	UICC → ME	PROACTIVE COMMAND PENDING: GET CHANNEL STATUS 2.1	
8	ME → UICC	FETCH	
9	UICC → ME	PROACTIVE COMMAND: GET STATUS 2.1	

Step	Direction	Sequence	Expected Result
10	ME → UICC	TERMINAL RESPONSE GET STATUS 2.1A Or TERMINAL RESPONSE: GET STATUS 2.1B	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

PROACTIVE COMMAND: GET STATUS 2.1

Command details
- Command number: 1
- Command type: GET STATUS
- Command qualifier: RFU

Device identities
- Source device: UICC
- Destination device: ME

TERMINAL RESPONSE: GET STATUS 2.1A

Command details
- Command number: 1
- Command typ e: GET STATUS
- Command qualifier: RFU

Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status
Channel status: Channel 1 open, link established or PDP context activated

TERMINAL RESPONSE: GET STATUS 2.1B
Command details
Command number: 1
Command type: GET STATUS
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status
Channel 1 status: Channel identifier 1 open, Link established or PDP context activated
Channel 2 status: Channel identifier 2, Link not established or PDP context not activated

Channel n status: Channel identifier n, Link not established or PDP context not activated
The number of channel status data objects shall be same as the number of channels(n) supported by the ME

12.3.3.6.3 Test sequence No 3: (GET STATUS, after a link dropped)

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: SET UP EVENT LIST 1.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 1.1	
4	ME → UICC	TERMINAL RESPONSE: SET UP EVENT LIST 1.1	[Command performed successfully]
5	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
6	ME → UICC	FETCH	
7	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
8	ME → USS	PDP context activation request	
9	USS → ME	PDP context activation accept	
10	ME →	TERMINAL RESPONSE: OPEN	[Command performed successfully]

Step	Direction	Sequence	Expected Result
	UICC	CHANNEL 1.1	
11	USS → ME	DROP LINK	
12	ME → UICC	ENVELOPE EVENT DOWNLOAD: CHANNEL STATUS 1.1	[Link dropped]
13	UICC → ME	PROACTIVE COMMAND PENDING: GET STATUS 1.1	
14	ME → UICC	FETCH	
15	UICC → ME	PROACTIVE COMMAND: GET STATUS 1.1	
16	ME → UICC	TERMINAL RESPONSE: GET STATUS 3.1A Or TERMINAL RESPONSE: GET STATUS 3.1B Or TERMINAL RESPONSE: GET STATUS 3.1C Or TERMINAL RESPONSE: GET STATUS 3.1D Or TERMINAL RESPONSE: GET STATUS 3.1E	[Command performed successfully]

TERMINAL RESPONSE: GET STATUS 3.1A

Same as TERMINAL RESPONSE: GET STATUS 1.1A

TERMINAL RESPONSE: GET STATUS 3.1B

Same as TERMINAL RESPONSE: GET STATUS 1.1B

TERMINAL RESPONSE: GET STATUS 3.1C

Same as TERMINAL RESPONSE: GET STATUS 1.1C

TERMINAL RESPONSE: GET STATUS 3.1D

Command details
- Command number: 1
- Command type: GET STATUS
- Command qualifier: RFU

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel status
- Channel status: Channel 1, link dropped

TERMINAL RESPONSE: GET STATUS 3.1E

Command details
Command number: 1
Command type: GET STATUS
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status
Channel 1 status: Channel identifier 1, link dropped
Channel 2 status: Channel identifier 2, Link not established or PDP context not activated
.
Channel n status: Channel identifier n, Link not established or PDP context not activated
The number of channel status data objects shall be same as the number of channels(n) supported by the ME

PROACTIVE COMMAND: SET UP EVENT LIST 1.1
Command details
Command number: 1
Command type: SET UP‘EV’NT LIST
Command qualifier: '00'
Device identities
Source device: UICC
Destination device: ME
Event list
Event 1: Channel Status

TERMINAL RESPONSE: SET UP EVENT LIST 1.1
Command details
Command number: 1
Command type: SET UP‘EV’NT LIST
Command qualifier: '00'
Device identities
Source device: ME
Destination device: UICC
Result General Result: Command performed successfully

ENVELOPE EVENT DOWNLOAD: CHANNEL STATUS 1.1
Event list
Event list: Channel Status
Device identities
Source device: ME
Destination device: UICC
Channel status
Channel status: Channel 1, link dropped

12.3.3.7 Data available event

Test Purpose
To verify Data available event related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement
- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.7.1 Test–Sequence No 1: (EVENT DOWNLOAD - Data available)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	[Command performed successfully]
4	ME → USER	The ME may display channel opening information	
5	ME → USS	PDP context activation request	
6	USS → ME	PDP context activation accept	
7	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	
8	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 1.1	
9	ME → UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 1.1	
11	ME → USS	Transfer of 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
12	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.1	[Command performed successfully]
13	USS → ME	Data sent through the BIP channel using the ME's port number, which was retrieved in step 11	
14	ME → UICC	ENVELOPE 1.1 (Event-Data Available)	

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details

Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment

Device identities
Source device: UICC
Destination device: ME
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1000
Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1000

PROACTIVE COMMAND: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 1.1
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

ENVELOPE: EVENT DOWNLOAD - Data available 1.1
Event list
Event: Data available
Device identities
Source device: ME
Destination device: UICC
Channel status

Channel status: Channel 1 open, link established
Channel Data Length
Channel data length: 8 Bytes available in Rx buffer

12.3.3.8 Channel Status event

Test Purpose

To verify Channel Status event related to Default (network) Bearer, for UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '03'= default bearer for requested transport layer.

Procedure

12.3.3.8.1 Test–Sequence No 1: (EVENT DOWNLOAD - Channel Status on a link dropped)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: SET UP EVENT LIST 1.1	
2	ME → UICC	FETCH	
3	UICC → ME	PROACTIVE COMMA ND: SET UP EVENT LIST 1.1	[EVENT: channel status]
4	ME → UICC	TERMINAL RESPONSE: SET UP EVENT LIST 1.1	[command performed successfully]
5	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 1.1	See initial conditions
6	ME → UICC	FETCH	
7	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 1.1	
8	ME → USER	The ME may display channel opening information	
9	ME → USS	PDP context activation request	
10	USS → ME	PDP context activation accept	
11	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]

Step	Direction	Sequence	Expected Result
12	USS → ME	Link dropped	
13	ME → UICC	ENVELOPE 1.1 (Event-Channel Status)	

PROACTIVE COMMAND: SET UP EVENT LIST 1.1

Command details
- Command number: 1
- Command type: SET UP'EV'NT LIST
- Command qualifier: '00'

Device identities
- Source device: UICC
- Destination device: ME

Event list
- Event 1: Channel Status

TERMINAL RESPONSE: SET UP EVENT LIST 1.1

Command details
- Command number: 1
- Command type: SET UP'EV'NT LIST
- Command qualifier: '00'

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1000

Network access name: TestGp.rs
Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
- General Result: Command performed successfully

Channel status Channel identifier 1 and link established or PDP context activated

Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size– 1000

ENVELOPE: EVENT DOWNLOAD - Chan nel Status 1.1
Event list
Event: Channel Status
Device identities
Soure device: ME
Destination device: UICC
Channel status
Channel status: Channel 1, link dropped

12.3.3.9 SMS-PP Data Download

Test Purpose
To verify SMS-PP Data Download related to GPRS, for UICC in client mode for UDP

Referenced requirement
- TS26_NFC_REQ_078
- TS26_NFC_REQ_081

Method of Test

Initial Conditions

All TCs are define by making use of Bearer Type '02'= GPRS bearer for requested transport layer.

Procedure

12.3.3.9.1 Test Sequence No 1: (SMS-PP - followed by Open channel - Send/Receive data)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Test Procedure SMS-PP Data Download	as specified in 12.3.3.9.4.
2		Test Procedure Open Channel	as specified in 12.3.3.9.4.
3		Test Procedure Send Data	as specified in 12.3.3.9.4.
4		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.
5		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4.
6		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.
7		Repeat Steps 5 to 6	5 times
8		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4.
9		Test Procedure Receive Data 1	repeat twice, as specified in 12.3.3.9.4–

12.3.3.9.2 Test Sequence No 2: (SMS-PP- Send SM -followed by Open channel - Send/Receive data)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Test Procedure SMS-PP Data Download	as specified in 12.3.3.9.4
2		Test Procedure Send Short Message	as specified in 12.3.3.9.4
3		Test Procedure Open Channel	as specified in 12.3.3.9.4
4		Test Procedure Send Data	as specified in 12.3.3.9.4
5		Test Procedure Receive Data 1	as specified in 12.3.3.9.4
6		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4
7		Test Procedure Receive Data 2	as specified in 12.3.3.9.4
8		Repeat Steps 6 to 7	5 times
9		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4
10		Test Procedure Receive Data 1	repeat twice, as specified in 12.3.3.9.–

12.3.3.9.3 Test Sequence No 3: (SMS-PP - Send SM -followed by Open channel - Send/Receive data with timer management)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Test Procedure SMS-PP Data Download	as specified in 12.3.3.9.4.
2		Test Procedure Send Short Message	as specified in 12.3.3.9.4.
3		Test Procedure Open Channel	as specified in 12.3.3.9.4.
4		Test Procedure Send Data	as specified in 12.3.3.9.4.
5		Test Procedure Timer Management	as specified in 12.3.3.9.4.
6		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.
7		Test Procedure Timer Management	as specified in 12.3.3.9.4.
8		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4.
9		Test Procedure Timer Management	as specified in 12.3.3.9.4.

Step	Direction	Sequence	Expected Result
10		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.
11		Test Procedure Timer Management	as specified in 12.3.3.9.4.
12		Repeat Steps 8 to 11	5 times
13		Test Procedure Send Data	repeat twice, as specified in 12.3.3.9.4.
14		Test Procedure Timer Management	as specified below
15		Test Procedure Receive Data 1	repeat twice, as specified in 12.3.3.9.4.
16		Test Procedure Timer Management	as specified in 12.3.3.9.4.

12.3.3.9.4 Reference Test Procedures

Test Procedure Open Channel (OPEN CHANNEL, immediate link establishment, GPRS, no local address)

FFS

Test Procedure Send Data (SEND DATA, immediate mode)

FFS

Test Procedure Receive Data 1 (RECEIVE DATA)

FFS

Test Procedure Receive Data 2 (RECEIVE DATA)

FFS

Test Procedure Timer Management (Timer Management)

FFS

Test Procedure Send Short Message

FFS

Test Procedure SMS-PP Data Download

FFS

12.3.3.10 Concurrent BIP channels

Test Purpose
To verify that the DUT supports two concurrent channels, BIP in client mode.

Referenced requirement

- TS26_NFC_REQ_080

Method of Test

Initial Conditions
None

Procedure

12.3.3.10.1 Test sequence No 1

Initial Conditions
None

Step	Direction	Sequence	Expected Result
1		The 3GPP TS 31.124"27.22.2 Contents of the TERMINAL PROFILE command" test SHALL be performed in order to check that the DUT declare to support two concurrent channels, BIP in client mode.	
2		The 3GPP TS 31.124 "27.22.4.27." Open Channel (related to GPRS)" test SHALL be performed in order to open a first channel BIP in client mode.	The Channel is correctly opened
3		Before the first channel is closed, and in order to open a second channel the 3GPP TS 31.124 "27.22.4.27." Open Channel (related to GPRS)" test SHALL be performed again in order to open a second channel BIP in client mode.	T is correctly opened

12.3.3.11 Contents of the TERMINAL PROFILE

Direction: terminal to UICC.

Test Purpose

To verify the contents of TERMINAL PROFILE for BIP in UDP client mode.

Referenced requirement

- TS26_NFC_REQ_080

Initial Conditions

The ME is connected to the UICC Simulator. All elementary files are coded as the default UICC Application Toolkit personalization.

Procedure

12.3.3.11.1 Test Sequence No 1: (TERMINAL PROFILE – command for BIP in UDP, client mode)

Initial Conditions

None

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1	User → ME	Power on the ME.	
2	ME → UICC	sends the TERMINAL PROFILE command	After the ME sends the TERMINAL PROFILE command to the UICC Simulator, the UICC Simulator shall record the content of the TERMINAL PROFILE as mentioned below in Profile section
3	UICC → ME	UICC sends SW1 / SW2 of '90 00'.	The contents of the TERMINAL PROFILE is recorded and compared to the corresponding Byte 1, Byte 12, Byte 13 and Byte 17 as explained below.

The test is terminated upon the ME sending the TERMINAL PROFILE command to the UICC Simulator

Command parameters/data:

Description	Clause	M/O/C	Length
Profile	-	M	length

Profile:

- Contents:
  - The list of CAT facilities that are supported by the terminal.
- Coding:
  - 1 bit is used to code each facility:
    bit = 1: facility supported by terminal;
    bit = 0: facility not supported by terminal.
    (bit = x: not checked by the UICC Simulator)

The terminal shall indicate Profile download as SUPPORTED in the content of TERMINAL PROFILE at First byte to be used for BIP in UDP, client mode as shown below:

First byte (Download):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	-	-	1	Profile download
-	-	-	-	-	-	X	-	Reserved by 3GPP (SMS-PP data download)
-	-	-	-	-	X	-	-	Reserved by 3GPP (Cell Broadcast data download)
-	-	-	-	X	-	-	-	Menu selection
-	-	-	X	-	-	-	-	Reserved by 3GPP (SMS-PP data download)
-	-	X	-	-	-	-	-	Timer expiration
-	X	-	-	-	-	-	-	Reserved by 3GPP and 3GPP2 (USSD string data object support in Call Control by USIM)
X	-	-	-	-	-	-	-	Call Control by NAA

The terminal shall indicate OPEN CHANNEL, CLOSE CHANNEL, RECEIVE DATA and SEND DATA as SUPPORTED in the content of TERMINAL PROFILE at 12th byte to be used for BIP in UDP, client mode as shown below:

Twelfth byte (Bearer Independent protocol proactive commands, class "e"):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	-	-	1	Proactive UICC: OPEN CHANNEL
-	-	-	-	-	-	1	-	Proactive UICC: CLOSE CHANNEL
-	-	-	-	-	1	-	-	Proactive UICC: RECEIVE DATA
-	-	-	-	1	-	-	-	Proactive UICC: SEND DATA Proactive
-	-	-	X	-	-	-	-	UICC: GET CHANNEL STATUS Proactive
-	-	X	-	-	-	-	-	UICC: SERVICESEARCH
-	X	-	-	-	-	-	-	Proactive UICC: GET SERVICE INFORMATION
X	-	-	-	-	-	-	-	Proactive UICC: DECLARE Service

The terminal shall indicate GPRS as SUPPORTED and Number of channels supported by terminal (Mini= 1) in the content of TERMINAL PROFILE at 13th byte to be used for BIP in UDP, client mode as shown below:

Thirteenth byte (Bearer Independent protocol supported bearers, class "e"):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	-	-	X	CSD
-	-	-	-	-	-	1	-	GPRS
-	-	-	-	-	X	-	-	Bluetooth
-	-	-	-	X	-	-	-	IrDA
-	-	-	X	-	-	-	-	RS232
-	-	x1	-	-	-	-	-	Number of channels supported by terminal
-	x2	-	-	-	-	-	-	Number of channels supported by terminal
x3	-	-	-	-	-	-	-	Number of channels supported by terminal

Number of channels coded by x1, x2 and x3 must be >0

The terminal shall indicate UDP, UICC in client mode as SUPPORTED in the content of TERMINAL PROFILE at 17th byte to be used for BIP in UDP, client mode as shown below:

Seventeenth byte (Bearer independent protocol supported transport interface/bearers, class "e"):

b8	b7	b6	b5	b4	b3	b2	b1	Meaning
-	-	-	-	-	-	-	X	TCP, UICC in client mode, remote connection
-	-	-	-	-	-	1	-	UDP, UICC in client mode, remote connection
-	-	-	-	-	X	-	-	TCP, UICC in server mode
-	-	-	-	X	-	-	-	TCP, UICC in client mode, local connection (i.e. class "k" is supported)
-	-	-	X	-	-	-	-	UDP, UICC in client mode, local connection (i.e. class "k" is supported)
-	-	X	-	-	-	-	-	Direct communication channel (i.e. class "k" is supported)
-	X	-	-	-	-	-	-	Reserved by 3GPP (E-UTRAN)
X	-	-	-	-	-	-	-	Reserved by 3GPP (HSDPA)

12.4 Remote Management use cases

12.4.1 General overview

This section addresses testing of selected use cases for remote management of NFC services in environment with possible real data transfer in place.

The list of conformance requirements tested within this section is listed in the table in section 12.4.2.

12.4.2 Conformance requirements

TS26_NFC_REQ_078	The mobile device SHALL support BIP in UICC client mode for UDP.
TS26_NFC_REQ_079	The mobile device SHALL support BIP in UICC client mode for TCP.
TS26_NFC_REQ_081	The mobile device SHALL support the SMS push (per ETSI TS 102 226 and 3GPP TS 31.116) to establish an open BIP channel as per ETSI TS 102 223 Open Channel Command
TS26_NFC_REQ_120	The device SHALL support the BIP session regardless of incoming or outgoing calls, incoming or outgoing MMS, SMS. Note: This is not applicable if the device is on a 2G network

12.4.3 Test Cases

12.4.3.1 Contactless transaction during BIP session

Test Purpose

To ensure that the device is able to perform contactless transaction during a CAT-TP/BIP session

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

12.4.3.1.1 Test sequence No 1 – Receiving and send a SMS during BIP data transfer

Initial Conditions

- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.

- **APDU Application** to send APDUs according to the reference transaction.

Procedure

Step	Direction	Sequence	Expected Result
1	Device → UICC	Send Fetch OPEN CHANNEL command	
2	UICC → Device	OPEN CHANNEL 1.1	
3	Device →	TERMINAL RESPONSE: OPEN	TR Open Channel successful + SW

Step	Direction	Sequence	Expected Result
	UICC	CHANNEL	= 91xx
4		Fetch Send Data (CATTP SYN command for Link establishment)	TR Successful + 90 00
		Event Data Available is sent to the card (Reception of CATTP SYN-ACK)	91 XX
	Device → UICC	Fetch Receive Data	TR Successful + 91 XX
		Fetch Send Data (ACK-PDU)	Ask server for downloading data
5	Device → UICC	Event Data Available is sent to the card (Reception of data from the server)	91 FF
6	Device → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
7	Device → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
8		Execute the **reference transaction** in loop mode (5 loops)	The DUT must manage the reference transaction at least 5 transaction done consecutively without any loss.
9	Device → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
10		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
11		Fetch Send Data store data in Tx buffer (with 0x'zz' data)	TR Successful + 90 00
12		Event Data Available is sent to the card	
13		Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
14	Device → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
15		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
16		Fetch Send Data immediate	TR Successful + 90 00
17		Event Data Available is sent to the card	

Step	Direction	Sequence	Expected Result
18		Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
19		Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
20		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
21		Fetch Send Data immediate	TR Successful + 91 xx
22		Fetch Close Channel	TR Successful + 90 00

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: UICC
- Destination device: ME

Bearer
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1400

Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
- Transport format: UDP
- Port number: 44444

Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
- Command number: 1
- Command type: OPEN CHANNEL
- Command qualifier: immediate link establishment

Device identities
- Source device: ME
- Destination device: UICC

Result
General Result: Command performed successfully
- Channel status: Channel identifier 1 and link established or PDP context activated

Bearer description
- Bearer type: Default Bearer for requested transport layer

Buffer
- Buffer size: 1400

12.4.3.2 OTA Data Loading during a phone call

Test Purpose

Ensure that the Baseband can support the OTA data Loading during a phone call

Referenced requirement

- TS26_NFC_REQ_078
- TS26_NFC_REQ_079
- TS26_NFC_REQ_081
- TS26_NFC_REQ_120

Method of Test

Initial Conditions
A test data with a size of 60k Bytes to induce OTA Load duration in CAT-TP

Set up a network simulator for the appropriate radio access technology as defined in chapter 2.5.8.

Also, a test phone number which may be called and which permits to maintain the call during several minutes is necessary.

Simulated UICC is connected to the DUT

Prior to this test the device shall have been powered ON and ISO7816 initialization has been completed.

Test shall be made based on the capability of the DUT (Example: For LTE device, test shall use LTE; otherwise, use 3G).

12.4.3.2.1 Test Sequence No 1: Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network with 2G/3G fallback

Initial Conditions

Set up a network simulator for LTE/IMS.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
2.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	

Step	Direction	Sequence	Expected Result
2.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 12.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
18		During CATTP data transfer, Receive and accept an incoming voice call. Operate the Call for the whole test sequence.	Voice call established
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

PROACTIVE COMMAND: OPEN CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: DUT
Bearer
Bearer type: Default Bearer for requested transport layer

Buffer
Buffer size: 1400
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400

PROACTIVE COMMAND: SEND DATA 12.1

Logically:
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 12.1

Logically:
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx buffer

ENVELOPE: EVENT DOWNLOAD - Data available 12.1

Logically:

Event list
Event: Data available
Device identities
Source device: DUT
Destination device: UICC
Channel status
Channel status: Channel 1 open, link established
Channel Data Length
Channel data length: FF (more than 255 bytes are available)

PROACTIVE COMMAND: RECEIVE DATA 12.1

Logically:
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

TERMINAL RESPONSE: RECEIVE DATA 12.1

Logically:
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data : 00 01 02 .. C7 (Segmented Bytes of data)
Channel data length: FF (for the last TERMINAL RESPONSE: RECEIVE DATA thischannel data length should be 00)

PROACTIVE COMMAND: CLOSE CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel

TERMINAL RESPONSE: CLOSE CHANNEL 12.1

Logically:

Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully

12.4.3.2.2 Test Sequence No 2: Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network only

Initial Conditions

Set up a network simulator for LTE/IMS.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
2.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	
2.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	

Step	Direction	Sequence	Expected Result
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
18		During CATTP data transfer, Receive and accept an incoming voice call in 4G only. Operate the Call for the whole test sequence	Voice call established
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	

Step	Direction	Sequence	Expected Result
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as, ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

12.4.3.2.3 Test Sequence No 3: Voice Call made from the device during BIP CAT-TP session on LTE network with 2G/3G fallback

Initial Conditions

Set up a network simulator for LTE with 2G /3G fallback.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1.1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
2.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	
2.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	

Step	Direction	Sequence	Expected Result
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
18		During CATTP data transfer , Start a voice call to a test phone number and Receive the call . Operate the Call for the whole test sequence	Voice call established
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	

Step	Direction	Sequence	Expected Result
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

PROACTIVE COMMAND: OPEN CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: DUT
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400
UICC/ME interface transport level
Transport format: TCP
Port number: 44444
Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully

Channel status	Channel identifier 1 and link established or PDP context activated
Bearer description	
Bearer type:	Default Bearer for requested transport layer
Buffer	
Buffer size:	1400

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as, ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

12.4.3.2.4 Test Sequence No 4: Voice Call made from the device during BIP CAT-TP session on LTE network only

Initial Conditions

Set up a network simulator for LTE/IMS.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
2.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	

Step	Direction	Sequence	Expected Result
2.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
18		During CATTP data transfer , Start a voice call to a test phone number and Receive the call over 4G. Operate the Call for the whole test sequence.	Voice call established
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.3.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as, ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.
Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.
Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

12.4.3.2.5 Test Sequence No 5: BIP CAT-TP data transfer during a Voice Call is established on LTE network with 2G/3G fallback

Initial Conditions

Set up a network simulator for LTE with 2G /3G fallback.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1		Start the test by Receive and accept an incoming voice call over 2G/3G . Operate the Call for the whole test sequence.	Voice call established
2	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
3.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	
3.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
4	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	

Step	Direction	Sequence	Expected Result
5	DUT → UICC	FETCH	
6	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
7	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
8	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
9	DUT → UICC	FETCH	
10	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
11	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
12	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
13	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
14	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
15	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
16	DUT → UICC	FETCH	
17	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
18	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	

Step	Direction	Sequence	Expected Result
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.3.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as, ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

12.4.3.2.6 Test Sequence No 6: BIP CAT-TP data transfer during a Voice Call is established on LTE network only

Initial Conditions

Set up a network simulator for LTE/IMS.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1		Start the test by Receive and accept an incoming voice call over 4G . Operate the Call for the whole test sequence.	Voice call established
2	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
3.1	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	
3.2	DUT→ Server	[Command performed successfully]	SW1 SW2 90 00
4	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
5	DUT → UICC	FETCH	
6	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
7	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
8	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
9	DUT → UICC	FETCH	
10	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1	

Step	Direction	Sequence	Expected Result
		(For DUT port number)	
11	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
12	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
13	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
14	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
15	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
16	DUT → UICC	FETCH	
17	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
18	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	

Step	Direction	Sequence	Expected Result
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.3.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as, ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

12.4.3.3 OTA Data Loading with and without *proof of Receipt* (PoR)

Test Purpose

Ensure that the mobile device supports the OTA data Loading with and without proof of Receipt (PoR) request by the OTA server.

Referenced requirement

- TS26_NFC_REQ_078
- TS26_NFC_REQ_081

Method of Test

Initial Conditions

A test data with a size of 60k Bytes to induce OTA data transfer

Set up a network simulator for the appropriate radio access technology as defined in chapter 2.5.8.

Also, a test phone number which may be called and which permits to maintain the call during several minutes is necessary.

Simulated UICC is connected to the DUT

Prior to this test the device shall have been powered ON and ISO7816 initialization has been completed.

Test shall be made based on the capability of the DUT (Example: For LTE device, test shall use LTE; otherwise, use 3G).

12.4.3.3.1 Test Sequence No 1: OTA data loading without PoR requested by OTA server.

Initial Conditions

Set up a network simulator for LTE with 2G/3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test UICC should be configured to No PoR.

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223 The SPI parameter is set to "12 00" in the sms open channel push	

Step	Direction	Sequence	Expected Result
2	Server → DUT DUT→ Server	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	[Command performed successfully] DUT should send “SW1 SW2 90 00” to the server
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
18	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
19	DUT → UICC	FETCH	
20	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
21	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
22	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
23	DUT → UICC	FETCH	
24	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
25	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
26	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
27	DUT → UICC	FETCH	
28	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
29	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically:

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1

Same as ENVELOPE: EVENT DOWNLOAD – Data available 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

12.4.3.3.2 Test Sequence No 2 : OTA data loading with PoR requested by OTA server.

Initial Conditions

Set up a network simulator for LTE with 2G/3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223. The SPI parameter is set to "12 21" in the sms open channel push	
2	Server → DUT DUT→ Server	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	DUT should send “SW1 SW2 90 00” with Binary SMS (Ack PoR) to the server
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server →	Transfer of 60k Bytes of data to the	

Step	Direction	Sequence	Expected Result
	DUT	DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
18	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
19	DUT → UICC	FETCH	
20	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
21	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
22	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
23	DUT → UICC	FETCH	
24	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
25	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
26	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
27	DUT → UICC	FETCH	
28	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	

Step	Direction	Sequence	Expected Result
29	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically:

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1

Same as ENVELOPE: EVENT DOWNLOAD – Data available 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

12.4.3.3.3 Test Sequence No 3: OTA data loading with PoR requested by OTA server when the device is booting.

Initial Conditions

Set up a network simulator for LTE with 2G/3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1		Switch Off the device	
2	Server → DUT	Power on the device and during the boot sequence and Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223. The SPI parameter is set to "12 21" in the sms open channel push	(Check that once the device is turned on, the command is delivered properly)
3	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in	DUT should send “SW1 SW2 90

Step	Direction	Sequence	Expected Result
	DUT → Server	TS 102 226 and TS 102 127	00" with Binary SMS (Ack PoR) to the server
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
18	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
19	DUT → UICC	FETCH	
20	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
21	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
22	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
23	DUT → UICC	FETCH	
24	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
25	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
26	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
27	DUT → UICC	FETCH	
28	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
29	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1

Same as ENVELOPE: EVENT DOWNLOAD – Data available 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL REPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL REPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

12.4.3.3.4 Test Sequence No 4: OTA data loading with PoR requested by OTA server after the device comes back from airplane mode.

Initial Conditions

Set up a network simulator for LTE with 2G/3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result
1		Activate the airplane mode of the test device.	
2	Server → DUT	Deactivate the airplane mode and Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223. The SPI parameter is set to "12 21" in the sms open channel push	(Check that once the airplane mode is turned off (data is reactivated), the command is delivered)
3	Server → DUT DUT → Server	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	DUT should send “SW1 SW2 90 00” with Binary SMS (Ack PoR) to the server
4	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
5	DUT → UICC	FETCH	
6	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
7	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
8	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
9	DUT → UICC	FETCH	
10	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1	

Step	Direction	Sequence	Expected Result
		(For DUT port number)	
11	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
12	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
13	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
14	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
15	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
16	DUT → UICC	FETCH	
17	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
18	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	

Step	Direction	Sequence	Expected Result
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL REPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1

Same as ENVELOPE: EVENT DOWNLOAD – Data available 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

12.4.3.3.5 Test Sequence No 5: OTA data loading with PoR requested by OTA server only on error

Initial Conditions

Set up a network simulator for LTE with 2G/3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

(With APN, immediate link establishment, Bearer type 03 (Default Bearer for requested transport layer), no alpha identifier)

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223. The SPI parameter is set to "12 22" in the sms open channel push	
2	Server → DUT DUT → Server	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	DUT should send “SW1 SW2 90 00” to the server
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	

Step	Direction	Sequence	Expected Result
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
18	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
19	DUT → UICC	FETCH	
20	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
21	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
22	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
23	DUT → UICC	FETCH	
24	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
25	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
26	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
27	DUT → UICC	FETCH	
28	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
29	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: SEND DATA 12.1 in clause 12.4.3.2.1

Same as ENVELOPE: EVENT DOWNLOAD – Data available 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1

12.4.3.4 Secure Element Access during BIP session

Test Purpose

To ensure that the device is able to perform Secure Element Access *during a BIP session*

Referenced requirement

- TS26_NFC_REQ_078

Method of Test

12.4.3.4.1 Test sequence No 1

Initial Conditions

Set up a network simulator according to the capability of the DUT.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

- **APDU_TestApplication.cap** implements the sequence used by the **MobileApplication.**

- **MobileApplication** to call Secure Element Access APIs for open channel and Send APDU. This Application have full access to all AIDs.

The UICC simulator is connected to the DUT

The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing an empty hash condition.

Test Procedure

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	[Command performed successfully] SW1 SW2 90 00

Step	Direction	Sequence	Expected Result
2	Server → DUT	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	[Command performed successfully] SW1 SW2 90 00
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
18		Execute the **MobileApplication** in loop mode (20 loops) sending APDUs simultaneously ,APDU Case 1, Case 2, Case 3, Case 4 .	APDU Case 1 and 3: 90 00 APDU Case 2 and 4 :Data field of 0xFF bytes+ 9000
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

PROACTIVE COMMAND: OPEN CHANNEL 1.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: UICC
Destination device: DUT
Bearer
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400

Text String: UserLog (User login)
Text String: UserPwd (User password)
UICC/ME interface transport level
Transport format: UDP
Port number: 44444
Data destination address 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 1.1

Command details
Command number: 1
Command type: OPEN CHANNEL
Command qualifier: immediate link establishment
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully
Channel status Channel identifier 1 and link established or PDP context activated
Bearer description
Bearer type: Default Bearer for requested transport layer
Buffer
Buffer size: 1400

PROACTIVE COMMAND: SEND DATA 12.1

Logically:
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data
Channel Data: 00 01 .. 07 (8 Bytes of data)

TERMINAL RESPONSE: SEND DATA 12.1

Logically:
Command details
Command number: 1
Command type: SEND DATA
Command qualifier: Send Immediately
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully
Channel data length: More than 255 bytes of space available in the Tx b–ffer

ENVELOPE: EVENT DOWNLOAD - Data available 12.1

Logically:

Event list
Event: Data available
Device identities
Source device: DUT
Destination device: UICC
Channel status
Channel status: Channel 1 open, link established
Channel Data Length
Channel data length: FF (more than 255 bytes are available)

PROACTIVE COMMAND: RECEIVE DATA 12.1

Logically:
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: UICC
Destination device: Channel 1
Channel Data Length
Channel Data Length: 200

TERMINAL RESPONSE: RECEIVE DATA 12.1

Logically:
Command details
Command number: 1
Command type: RECEIVE DATA
Command qualifier: RFU
Device identities
Source device: ME
Destination device: UICC
Result
General Result: Command performed successfully
Channel Data: 00 01 02 .. C7 (Segmented Bytes of data)
Channel data length: FF (for the last TERMINAL RESPONSE: RECEIVE DATA this channel data length should be 00)

PROACTIVE COMMAND: CLOSE CHANNEL 12.1

Logically:
Command details
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: UICC
Destination de vice: Channel

TERMINAL RESPON SE: CLOSE CHANNEL 12.1

Logically:

Command detail s
Command number: 1
Command type: CLOSE CHANNEL
Command qualifier: RFU
Device identities
Source device: DUT
Destination device: UICC
Result
General Result: Command performed successfully

12.4.3.5 SMS and Internet Connection during OTA data loading.

Test Purpose
Ensure that the mobile device supports the OTA data Loading during receiving /sending SMS

Referenced requirement
- TS26_NFC_REQ_078
- TS26_NFC_REQ_081
- TS26_NFC_REQ_120

Method of Test
Initial Conditions
A test data with a size of 60k Bytes to induce OTA data transfer

Set up a network simulator for the appropriate radio access technology as defined in chapter 2.5.8.

Also, a test phone number which may be called and which permits to maintain the call during several minutes is necessary.

Simulated UICC is connected to the DUT

Prior to this test the device shall have been powered ON and ISO7816 initialization has been completed.

Test shall be made based on the capability of the DUT (Example: For LTE device, test shall use LTE; otherwise, use 3G).

12.4.3.5.1 Test Sequence No 1: Receiving and send a SMS during BIP data transfer

Initial Conditions

Set up a network simulator for LTE/IMS with 3G.

One default APN is configured on the DUT and the related PDN connection to this APN has been already established. Else Logically APN information should be set as

Network access name:	Gateway
User login:	UserLog
User password:	UserPwd

Test Procedure

Step	Direction	Sequence	Expected Result
1	Server → DUT	Send “PUSH” command with Parameter “Request for BIP channel opening” as defined in TS 102 226 and TS 102 223	
2	Server → DUT DUT → Server	Send “PUSH” command with Parameter “Request for CAT_TP link establishment” as defined in TS 102 226 and TS 102 127	[Command performed successfully] DUT should send “SW1 SW2 90 00” to the server
3	UICC → DUT	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.1	
4	DUT → UICC	FETCH	
5	UICC → DUT	Send PROACTIVE COMMAND: OPEN CHANNEL command	
6	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.1	TR Open Channel successful + SW = 91xx
7	UICC → DUT	PROACTIVE COMMAND PENDING: SEND DATA 12.1	
8	DUT → UICC	FETCH	
9	UICC → DUT	PROACTIVE COMMAND: SEND DATA (immediate) 12.1 (For DUT port number)	
10	DUT → Server	Transfer of 8 Bytes of data to the server through channel 1 to retrieve DUT's port number	
11	DUT → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 1.2.1	[Command performed successfully] SW1 SW2 90 00
12	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved in step 10	
13	DUT → UICC	ENVELOPE: EVENT DOWNLOAD –Data Available is sent to the card (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 FF
14	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
15	DUT → UICC	FETCH	

Step	Direction	Sequence	Expected Result
16	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data) 12.1	
17	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 FF
18		Send SMS from DUT to test phone number. Receive SMS from test phone number on DUT.	SMS Sent and Received
19	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0xFF data) 12.1	
20	DUT → UICC	FETCH	
21	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0xFF data)	
22	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful + 91 yy (last Bytes)
23	UICC → DUT	PROACTIVE COMMAND PENDING: Receive Data (with 0x'yy' data) 12.1	
24	DUT → UICC	FETCH	
25	UICC → DUT	PROACTIVE COMMAND: Receive Data (with 0x'yy' data)	
26	DUT→ UICC	TERMINAL RESPONSE: RECEIVE DATA 12.2	TR Successful + 91 zz
27	UICC → DUT	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.1	
28	DUT → UICC	FETCH	
29	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
30	DUT → UICC	TERMINAL RESPONSE : CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: SEND DATA 12.1 in clause 12.4.3.2.1.
Same as TERMINAL REPONSE: SEND DATA 12.1 in clause 12.4.3.2.1.

Same as ENVELOPE: EVENT DOWNLOAD - Data available 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.
Same as TERMINAL RESPONSE: RECEIVE CHANNEL 12.1 in clause 12.4.3.2.1.

Same as PROACTIVE COMMAND: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.
Same as TERMINAL RESPONSE: CLOSE CHANNEL 12.1 in clause 12.4.3.2.1.

13 General Device Support

13.1 General Overview

This chapter addresses requirements for general device features which cannot be grouped under previous specific section. This includes general UI requirements, modem requirements and general device related requirements.

The list of conformance requirements tested within this section is listed in the table in section 13.2.

13.2 Conformance requirements

TS26_NFC_REQ_022	There SHOULD be an API to ask the system to enable the NFC functionality. User input SHALL be required to enable NFC. This UI dialogue SHALL be generated by the OS and not by the calling application.
TS26_NFC_REQ_046	Access to the UICC (logical channel) SHALL be allowed even when the mobile device is in a Radio OFF state, i.e. flight mode, airplane mode etc.
TS26_NFC_REQ_108	The device SHALL automatically turn off NFC, when the user activates Radio Off Mode, in line with regulation.
TS26_NFC_REQ_109	The OS SHALL allow the user to turn NFC on and off when the device is in Radio Off Mode.

13.3 Test Cases

13.3.1 Secure Element Access API and Access control in Radio OFF State

Test Purpose
Access to the UICC (logical channel) SHALL be allowed even when the DUT device is in a Radio OFF state, i.e. flight mode, airplane mode etc.

Referenced requirement
- TS26_NFC_REQ_046
- TS26_NFC_REQ_108
- TS26_NFC_REQ_109

Method of Test

Initial Conditions

An instance of the UICC application **APDU_TestApplication.cap** with AID1 is selectable.

MobileApplication is installed on the device and implements a function “Select AID1”. The application is duplicated with different signature configurations and respectively named:
- **GSMA_Mobile_App_SP1_signed** signed with a test certificate #1
- **GSMA_Mobile_App_SP2_signed** signed with a test certificate #2

Note: When the DUT is set in Radio OFF State (e.g. Flight Mode, Airplane Mode etc.), if NFC is automatically deactivated, it is necessary to switch on the NFC again.

Procedure

13.3.1.1 Test sequence No 1

Initial Conditions
DUT is powered on in operational mode and the NFC is enabled.
The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF is present and contains an empty hash condition

⇨ all applications have full access to all AID
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	Put the DUT in Radio OFF state (e.g. Flight mode, Airplane Mode, etc.)	None
2	User → DUT	Check if the NFC is enabled in the DUT.	NFC is not enabled in the DUT
3	User → DUT	Enable NFC in the DUT	NFC is enabled in the DUT
4	User → DUT	Launch **GSMA_Mobile_App_SP1_signed**	None
5	DUT→ UICC	Call the “Select AID1” function	The expected ISO command(s) (C-APDU) sent by the DUT to the UICC: CMD 1: APDU_MANAGE_CHANNEL_OPEN ('0X 70 00 00 01') CMD 2: APDU_SELECT_BY_DF_NAME – the CLA contains the Channel Number returned by the UICC as a response to APDU_MANAGE_CHANNEL_OPEN; Data = ‘AID1’; Le=00, or empty
6	UICC → DUT	UICC successfully opens a new logical channel to AID1	Expected result of the API called: call to “Select AID1” function returns successfully
7	DUT → UICC	Send **APDU Case 1** fifty times in a loop from P1 = 0x01 to 0x32	The expected C-APDUs in the loop: CMD 1: C-APDU (´XX 01 01 00´) …. CMD 50: C-APDU (´XX 01 32 00´)

Step	Direction	Sequence	Expected Result
8	UICC → DUT	**APDU_TestApplication.cap** returns SW1-SW2 according to the P1 in the C-APDU. RESP 1: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: RESP 1: R-APDU - SW1, SW2 ... RESP 50: R-APDU - SW1, SW2
9	DUT → UICC	Send **APDU Case 2** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: From P1 = 0x01 to 0x11: CMD 1: C-APDU ('XX 02 01 00 FF') …. CMD 17: C-APDU ('XX 02 11 00 FF') From P1 = 0x12 to 0x32: CMD 18: C-APDU ('XX 02 12 00') …. CMD 50: C-APDU ('XX 02 32 00')
10	DUT → UICC	**APDU_TestApplication.cap** returns : From P1 = 0x01 to 0x11: RESP 1: R-APDU – <data field of 255 bytes> SW1-SW2 ... RESP 17: R-APDU – <data field of 255 bytes> SW1-SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: From P1 = 0x01 to 0x11: RESP 1: R-APDU - data field of 255 byte, SW1, SW2 ... RESP 17: R-APDU - data field of 255 byte, SW1, SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU - SW1, SW2 … RESP 50: R-APDU - SW1, SW2
11		Send **APDU Case 3** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: CMD 1: C-APDU ('XX 03 01 00 FF' <Data field of 255 bytes>) …. CMD 50: C-APDU ('XX 03 32 00 FF' <Data field of 255 bytes>)

Step	Direction	Sequence	Expected Result
12	UICC → DUT	**APDU_TestApplication.cap** returns SW1-SW2 according to the P1 in the C-APDU. RESP 1: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: RESP 1: R-APDU - SW1, SW2 ... RESP 50: R-APDU - SW1, SW2
13	DUT → UICC	Send **APDU Case 4** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: From P1 = 0x01 to 0x11: CMD 1: C-APDU ('XX 04 01 00 FF' <Data field of 255 bytes> FF) …. CMD 17: C-APDU ('XX 04 11 00 FF' <Data field of 255 bytes> FF) From P1 = 0x12 to 0x32: CMD 18: C-APDU ('XX 04 12 00 FF' <Data field of 255 bytes>)
14	UICC → DUT	**APDU_TestApplication.cap** returns : From P1 = 0x01 to 0x11: RESP 1: R-APDU – <data field of 255 bytes> SW1-SW2 ... RESP 17: R-APDU – <data field of 255 bytes> SW1-SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: From P1 = 0x01 to 0x11: RESP 1: R-APDU - data field of 255 bytes, SW1, SW2 ... RESP 17: R-APDU - data field of 255 bytes, SW1, SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU - SW1, SW2 … RESP 50: R-APDU - SW1, SW2

13.3.1.2 Test sequence No 2

Initial Conditions
DUT is powered on in operational mode and the NFC is enabled
The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing an SP1 hash condition

⇨ SP1 has full access to all AID
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	Put the DUT in Radio OFF state (e.g. Flight mode, Airplane Mode, etc.)	None
2	User → DUT	Enable NFC in the DUT	NFC is enabled in the DUT
3	User → DUT	Launch **GSMA_Mobile_App_SP1_signed**	None
4	DUT → UICC	Call "Select AID1" function	The expected ISO command(s) (C-APDU) sent by the DUT to the UICC as a result of the API call: CMD 1: APDU_MANAGE_CHANNEL_OPEN ('0X 70 00 00 01') CMD 2: APDU_SELECT_BY_DF_NAME – the CLA contains the Channel Number returned by the UICC as a response to APDU_MANAGE_CHANNEL_OPEN; Data = 'AID1'; Le=00, or empty
5	UICC → DUT	UICC successfully opens a new logical channel to AID1	Expected result of the API called: call to "Select AID1" function returns successfully
6	DUT → UICC	Send **APDU Case 1** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: CMD 1: C-APDU ('XX 01 01 00') … CMD 50: C-APDU ('XX 01 32 00')
7	UICC → DUT	**APDU_TestApplication.cap** returns SW1-SW2 according to the P1 in the C-APDU. RESP 1: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: RESP 1: R-APDU - SW1, SW2 ... RESP 50: R-APDU - SW1, SW2

Step	Direction	Sequence	Expected Result
8	DUT →UICC	Send **APDU Case 2** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: From P1 = 0x01 to 0x11: CMD 1: C-APDU ('XX 02 01 00 FF') …. CMD 17: C-APDU ('XX 02 11 00 FF') From P1 = 0x12 to 0x32: CMD 18: C-APDU ('XX 02 12 00') …. CMD 50: C-APDU ('XX 02 32 00')
9	UICC → DUT	**APDU_TestApplication.cap** returns : From P1 = 0x01 to 0x11: RESP 1: R-APDU – <data field of 255 bytes> SW1-SW2 ... RESP 17: R-APDU – <data field of 255 bytes> SW1-SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: From P1 = 0x01 to 0x11: RESP 1: R-APDU - data field of 255 byte, SW1, SW2 … RESP 17: R-APDU - data field of 255 byte, SW1, SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU - SW1, SW2 … RESP 50: R-APDU - SW1, SW2
10	DUT → UICC	Send **APDU Case 3** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: CMD 1: C-APDU ('XX 03 01 00 FF' <Data field of 255 bytes>) …. CMD 50: C-APDU ('XX 03 32 00 FF' <Data field of 255 bytes>)
11	UICC →DUT	**APDU_TestApplication.cap** returns SW1-SW2 according to the P1 in the C-APDU. RESP 1: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: RESP 1: R-APDU - SW1, SW2 ... RESP 50: R-APDU - SW1, SW2

Step	Direction	Sequence	Expected Result
12	DUT → UICC	Send **APDU Case 4** fifty times in a loop From P1 = 0x01 to 0x32	The expected C-APDUs in the loop: From P1 = 0x01 to 0x11: CMD 1: C-APDU ('XX 04 01 00 FF' <Data field of 255 bytes> FF) …. CMD 17: C-APDU ('XX 04 11 00 FF' <Data field of 255 bytes> FF) From P1 = 0x12 to 0x32: CMD 18: C-APDU ('XX 04 12 00 FF' <Data field of 255 bytes>) …. CMD 50: C-APDU ('XX 04 32 00 FF' <Data field of 255 bytes>)
13	UICC → DUT	**APDU_TestApplication.cap** returns : From P1 = 0x01 to 0x11: RESP 1: R-APDU – <data field of 255 bytes> SW1-SW2 ... RESP 17: R-APDU – <data field of 255 bytes> SW1-SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU – SW1-SW2 ... RESP 50: R-APDU – SW1-SW2	Expected result of the API called: From P1 = 0x01 to 0x11: RESP 1: R-APDU - data field of 255 byte, SW1, SW2 ... RESP 17: R-APDU - data field of 255 byte, SW1, SW2 From P1 = 0x12 to 0x32: RESP 18: R-APDU - SW1, SW2 … RESP 50: R-APDU - SW1, SW2

13.3.1.3 Test sequence No 3

Initial Conditions
DUT is powered on in operational mode and the NFC is enabled
The following configuration is loaded into the UICC:

- PKCS#15 ADF with a DODF present and valid
- an ACMF is present and valid
- an ACRF is present and valid and contains a rule for all other AIDs and a path for one ACCF containing an SP1 hash condition

⇨ SP1 has full access to all AID
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		Put the DUT in Radio OFF state (e.g. Flight mode, Airplane Mode, etc.)	None
2		Enable NFC in the DUT	NFC is enabled in the DUT
3		Launch **GSMA_Mobile_App_SP2_signed**	None
4		Call “Select AID1” function	No SELECT_BY_DF_NAME with AID1 is sent. Call is unsuccessful, returning an error indicating that the access control rights are not granted.

13.3.2 Enabled / Disabled states

Test Purpose
Verify that the device provides the current status on NFC i.e. Enabled / Disabled

Referenced requirement
- TS26_NFC_REQ_022

Method of Test

Initial Conditions
- **ReferenceApplication.cap** managing the reference transaction with AID1 selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.

Procedure

13.3.2.1 Test sequence No 1

Initial Conditions
NFC is disabled on the DUT

Step	Direction	Sequence	Expected Result
1		Enable NFC on the DUT	None
2		Check in the Wireless Settings option if it sets the current state of NFC to "Enabled"	“NFC enabled” indication is present
3		Perform the reference transaction using a contactless reader	Reference transaction is performed successfully
4		Try to read a NFC tag	NFC Tag is read successfully
5		Check if the DUT allows ensuring NFC is enabled (for example a "N" logo).	DUT indicates NFC is enabled

Step	Direction	Sequence	Expected Result
6		Disable NFC on the DUT	None
7		Check in the Wireless Settings option if the DUT changes the current state of NFC to "Disabled"	“NFC enabled” indication is absent
8		Perform the reference transaction using a contactless reader	Reference transaction is not performed
9		Try to read a NFC tag	NFC Tag is not read
10		Check if the DUT no longer indicates NFC is enabled	DUT no longer indicates NFC is enabled
11		Enable NFC on the DUT	None
12		Perform the reference transaction using a contactless reader	Reference transaction is performed successfully
		Try to read a NFC tag	NFC Tag is read successfully

14 VOID

15 Android specific test cases

15.1 General overview

This chapter addresses test cases which are related to Android specific requirements

The list of conformance requirements tested within this section is listed in the table in section 15.2.

15.2 Conformance requirements

<table>
<tr><td>TS26_NFC_REQ_093</td><td>There SHALL be APIs (as described below) to check if the NFC Controller is enabled and to ask the system to enable the NFC functionality. User input SHALL be required to enable NFC. This UI dialogue SHALL be generated by the OS and not by the calling application.
Note: Refer to the Javadoc linked to this document for more details.
<table><tr><th colspan="2">“com.gsma.services.utils.NfcController”</th></tr><tr><th>Classes</th><th>Methods</th></tr><tr><td>Nested classes (NfcController.Callbacks)</td><td>All methods except “onCardEmulationMode“</td></tr><tr><td>NfcController</td><td>isEnabled, enableNfcController</td></tr></table></td></tr>
<tr><td>TS26_NFC_REQ_098</td><td>The Device SHALL support for Multicast mode for “EVT_TRANSACTION”</td></tr>
<tr><td>TS26_NFC_REQ_099</td><td>The Device SHALL support an API to switch from Unicast mode to Multicast mode for “EVT_TRANSACTION” management as described below.
<table><tr><th colspan="2">“com.gsma.services.utils.Handset”</th></tr><tr><th>Classes</th><th>Methods</th></tr><tr><td>Handset</td><td>enableMultiEvt_transactionReception</td></tr></table>
This is considered a sensitive API</td></tr>
<tr><td>TS26_NFC_REQ_123</td><td>All GSMA APIs SHALL be included in the following jar file:
“com.gsma.services.nfc.jar”
Note: Library SHALL be built as a shared library in order to be used with the following tag:
<uses-library android:name="com.gsma.services.nfc” android:required="false"/></td></tr>
<tr><td>TS26_NFC_REQ_127</td><td>As per TS26_NFC_REQ_055 devices SHALL implement classes and methods as described below.
Note: Refer to the Javadoc linked to this document for more details.
<table><tr><th colspan="2">“com.gsma.services.nfc.NfcController”</th></tr><tr><th>Classes</th><th>Methods</th></tr><tr><td>nested classes (NfcController.Callbacks)</td><td>All methods except “onCardEmulationMode”</td></tr></table></td></tr>
</table>

	NfcController	All methods except “disableCardEmulationMode, enableCardEmulationMode, isCardEmulationEnabled”

“com.gsma.services.nfc.AIDGroup”	
Classes	**Methods**
AIDGroup	All methods

“com.gsma.services.nfc.OffHostService”	
Classes	**Methods**
OffHostService	All methods

15.3 NFC Features

15.3.1 General overview

This section provides test cases for checking Android specific core NFC features like:

- Availability of the GSMA APIs in the device
- APIs used to handle the NFC Controller

The list of conformance requirements tested within this section is listed in the table in section 15.3.2.

15.3.2 Conformance requirements

TS26_NFC_REQ_093	There SHALL be APIs (as described below) to check if the NFC Controller is enabled and to ask the system to enable the NFC functionality. User input SHALL be required to enable NFC. This UI dialogue SHALL be generated by the OS and not by the calling application. Note: Refer to the Javadoc linked to this document for more details.

“com.gsma.services.utils.NfcController”	
Classes	**Methods**
Nested classes (NfcController.Callbacks)	All methods except “onCardEmulationMode“
NfcController	isEnabled, enableNfcController

TS26_NFC_REQ_123	All GSMA APIs SHALL be included in the following jar file: “com.gsma.services.nfc.jar” Note: Library SHALL be built as a shared library in order to be used with the following tag: <uses-library android:name="com.gsma.services.nfc” android:required="false"/>

15.3.3 Test Cases

15.3.3.1 Shared library

Test Purpose
Ensure DUT contains all GSMA APIs as decribed in the GSMA NFC Handset Requirements and its Javadoc.

Referenced requirement
- TS26_NFC_REQ_123

15.3.3.1.1 Test sequence No 1

Initial Conditions
- N/A

Step	Direction	Sequence	Expected Result
1	Application →DUT	Using reflection ("Class.forName"), application is checking if all the requested GSMA classes are available in the DUT. List of GSMA classes: • com.gsma.services.utils.Handset •com.gsma.services.nfc.NfcController •com.gsma.services.nfc.NfcController$Callbacks •com.gsma.services.nfc.OffHostService •com.gsma.services.nfc.AidGroup •com.gsma.services.utils. •InsufficientResourcesException •com.gsma.services.nfc.SEController •com.gsma.services.nfc.SEController$Callbacks	A valid Class object shall be returned for every call
2	Application →DUT	Using reflection, application is checking com.gsma.services.utils.Handset implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • enableMultiEvt_transactionReception • getProperty • getVersion	Constructor has no parameter Modifiers of the constructor are compliant All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared

Step	Direction	Sequence	Expected Result
3	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.NfcController implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • defineOffHostService • deleteOffHostService • disableCardEmulationMode • enableCardEmulationMode • enableNfcController • getDefaultController • getDefaultOffHostService • getOffHostServices • isCardEmulationEnabled • isEnabled	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared
4	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.NfcController$Callbacks implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • onCardEmulationMode • onEnableNfcController • onGetDefaultController	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared
5	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.OffHostService implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • commit • defineAidGroup • deleteAidGroup • getAidGroups • getBanner • getDescription • getLocation • setBanner	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared

Step	Direction	Sequence	Expected Result
6	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.AidGroup implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • addNewAid • getCategory • getDescription • removeAid	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared
7	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.SEController implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • getActiveSecureElement • getDefaultController • setActiveSecureElement	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared
8	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.SEController$Callbacks implements the following methods and their definition is fully compliant to GSMA requirements. List of methods: • onGetDefaultController	Any constructor is not accessible All methods are implemented For each of them, Modifiers are compliant Return type is compliant Number of parameters and associated types are compliant Requested checked exception has been declared
9	Application →DUT	Using reflection, application is checking com.gsma.services.utils.Handset defines the following constants and their type/value is fully compliant to GSMA requirements. List of constants: • BATTERY_LOW_MODE • BATTERY_POWER_OFF_MODE • FELICA • HCI_SWP • MIFARE_CLASSIC • MIFARE_DESFIRE • MULTIPLE_ACTIVE_CEE • NFC_FORUM_TYPE3 • OMAPI	All constants are defined For each of them, Modifiers are compliant Type is compliant Associated value is equal to the value defined by GSMA

Step	Direction	Sequence	Expected Result
10	Application →DUT	Using reflection, application is checking com.gsma.services.nfc.NfcController$Callbacks defines the following constants and their type/value is fully compliant to GSMA requirements. List of constants: CARD_EMULATION_DISABLED CARD_EMULATION_ENABLED CARD_EMULATION_ERROR	All constants are defined For each of them, Modifiers are compliant Type is compliant Associated value is equal to the value defined by GSMA

15.3.3.2 Getting an instance of the "NFController" class

Test Purpose
Ensure DUT can provide asynchronously an instance of the "NFCController" class as defined in the javadoc linked to the GSMA NFC Requirements.

Referenced requirement
- TS26_NFC_REQ_123
- TS26_NFC_REQ_093

15.3.3.2.1 Test sequence No 1

Initial Conditions
Application SHALL define the following variables
- [var01] refers to the "Context" of the Android application
- [var02] contains an instance of the "NfcController$Callbacks" class and is not *null*

Step	Direction	Sequence	Expected Result
1		"getDefaultController" static method is called with the following parameters: • context: [var01] • cb: [var02]	• "onGetDefaultController" method, member of the instance refered by [var02] is called • "onGetDefaultController" parameter is not *null*
2		"isEnabled" method is called from the parameter returned by the "onGetDefaultController" method	No exception thrown by the system

15.3.3.2.2 Test sequence No 2

Initial Conditions
Application SHALL define the following variable
- [var01] refers to the "Context" of the Android application

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		"getDefaultController" static method is called with the following parameters: • context: [var01] • cb: *null*	*3 options:* • No exception thrown by the system • An "IllegalArgumentException" is thrown by the system • An "NullPointerException" is thrown by the system

15.3.3.3 Getting the status the NFC Controller

Test Purpose
Ensure DUT can provide the status of the NFC Controller from APIs as decribed in the GSMA NFC Handset Requirements and its Javadoc.

Referenced requirement
- TS26_NFC_REQ_093

15.3.3.3.1 Test sequence No 1

Initial Conditions
- NFC is activated on the DUT
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "isEnabled" method is called	Method SHALL return *true*

15.3.3.3.2 Test sequence No 2

Initial Conditions
- NFC is deactivated on the DUT
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "isEnabled" method is called	Method SHALL return *false*

15.3.3.4 Enabling the NFC Controller

Test Purpose
Ensure DUT can enable the NFC Controller from APIs as described in the GSMA NFC Handset Requirements and its Javadoc.

Referenced requirement
- TS26_NFC_REQ_093

15.3.3.4.1 Test sequence No 1

Initial Conditions
- NFC is deactivated on the DUT

- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "enableNfcController" method is called	A dialog box asking confirmation to the user is displayed. This dialog box SHALL provide an option to cancel or confirm the action
2		Action is cancelled	• Dialog box SHALL be withdrawn • An icon representing enablement of NFC SHALL not be displayed in the menu bar
3		From the "NfcController" instance, the "isEnabled" method is called	Method SHALL return *false*
4		Check that NFC is indicated as disabled in the settings	NFC Status SHALL be "disabled"

15.3.3.4.2 Test sequence No 2

Initial Conditions

- NFC is deactivated on the DUT
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "enableNfcController" method is called	A dialog box asking confirmation to the user is displayed. This dialog box SHALL provide an option to cancel or confirm the action
2		Action is confirmed	• Dialog box SHALL be withdrawn • An icon representing enablement of NFC SHALL be displayed in the menu bar *Note:* On some devices, icon may not be available
3		From the "NfcController" instance, the "isEnabled" method is called	Method SHALL return *true*
4		In the "Setting" menu, NFC status is checked	NFC status SHALL be "enabled"

15.3.3.4.3 Test sequence No 3

Initial Conditions

- NFC is activated on the DUT
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "enableNfcController" method is called	*2 options:* • A dialog box asking confirmation to the user is displayed. This dialog box SHALL provide an option to cancel or confirm the action → Execute Step 2a & 2b • Nothing appears → Go to step 3
2a		Action is cancelled	• Dialog box SHALL be withdrawn • Icon representing enablement of NFC SHALL not be updated in the menu bar *Note:* On some devices, icon may not be available → Go to step 3
2b		Action is confirmed	• Dialog box SHALL be withdrawn • Icon representing enablement of NFC SHALL not be updated in the menu bar *Note:* On some devices, icon may not be available →Continue to Step 3
3		From the "NfcController" instance, the "isEnabled" method is called	Method SHALL return *true*
4		In the "Setting" menu, NFC status is checked	NFC status SHALL be "enabled"

15.4 VOID

15.5 Secure Element Access Control

15.5.1 General overview

This section provides test cases for checking reception of NFC Transaction events.

The list of conformance requirements tested within this section is listed in the table in section 15.5.2.

15.5.2 Conformance requirements

TS26_NFC_REQ_098	The Device SHALL support for Multicast mode for "EVT_TRANSACTION"
TS26_NFC_REQ_099	The Device SHALL support an API to switch from Unicast mode to Multicast mode for "EVT_TRANSACTION" management as described below. "com.gsma.services.utils.Handset" **Classes** / **Methods** Handset / enableMultiEvt_transactionReception This is considered a sensitive API

15.5.3 Test Cases

15.5.3.1 Switching to"Multicast" mode

Test Purpose
Ensure DUT implements correctly defined API for switching to "Multicast" mode and bradcast accordingly "EVT_Transaction" to several applications.

Referenced requirement
- TS26_NFC_REQ_098
- TS26_NFC_REQ_099

15.5.3.1.1 Test sequence No 1

Initial Conditions
- PKCS#15 structure of the UICC contains only a certificate [cert01]
  EF_CDF is referencing it / EF_ODF contains a reference to EF_CDF
- Applications [app01] is signed with certificate which is not [cert01]
- An instance of the "Handset" class has been retrieved successfully by the application and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "Handset" instance, "enableMultiEvt_transactionReception" method is called	"SecurityException" is thrown by the system

15.5.3.1.2 Test sequence No 2

Initial Conditions

- PKCS#15 structure of the UICC contains a certificate [cert01]
  EF_CDF is referencing it / EF_ODF contains a reference to EF_CDF
- Application [app01]
  - Defines an "intent" which listens a "com.gsma.services.nfc.action.TRANSACTION_EVENT" action
- Applications [app02] and [app03]
  - Signed with certificate [cert01]
  - An instance of the "Handset" class has been retrieved successfully and is not *null*
  - Define a "BroadcastReceiver" which listens "com.gsma.services.nfc.action.TRANSACTION_EVENT" action
- Access Controll is allowing communication from any applets in the UICC to [app01], [app02] and [app03]

Step	Direction	Sequence	Expected Result
1		From the "Handset" instance, "enableMultiEvt_transactionReception" method is called by [app02]	No exception thrown by the system
2		A transaction event is generated from an applet in the UICC	• [app02] and [app03] are receiving the event • [app01] is not receiving the event
3		DUT is restarted	
4		A transaction event is generated from an applet in the UICC	• [app01] is receiving the event • [app02] and [app03] are not receiving the event

15.6 VOID

15.7 Multiple Card Emulation Environment

15.7.1 General overview

This section provides test cases for checking features linked to Dynamic AIDs Registration.

The list of conformance requirements tested within this section is listed in the table in section 15.7.2.

15.7.2 Conformance requirements

<table>
<tr><td>TS26_NFC_REQ_127</td><td>As per TS26_NFC_REQ_055 devices SHALL implement classes and methods as described below.
Note: Refer to the Javadoc linked to this document for more details.

<table>
<tr><th colspan="2">“com.gsma.services.nfc.NfcController”</th></tr>
<tr><th>Classes</th><th>Methods</th></tr>
<tr><td>nested classes (NfcController.Callbacks)</td><td>All methods except “onCardEmulationMode”</td></tr>
<tr><td>NfcController</td><td>All methods except “disableCardEmulationMode, enableCardEmulationMode, isCardEmulationEnabled”</td></tr>
</table>

<table>
<tr><th colspan="2">“com.gsma.services.nfc.AIDGroup”</th></tr>
<tr><th>Classes</th><th>Methods</th></tr>
<tr><td>AIDGroup</td><td>All methods</td></tr>
</table>

<table>
<tr><th colspan="2">“com.gsma.services.nfc.OffHostService”</th></tr>
<tr><th>Classes</th><th>Methods</th></tr>
<tr><td>OffHostService</td><td>All methods</td></tr>
</table>
</td></tr>
</table>

15.7.3 Test Cases

15.7.3.1 Dynamic AIDs Registration APIs when Multiple Active CEEs model is supported

Test Purpose
Ensure DUT implements correctly all APIs related to Dynamic AIDs Registration when device support Multiple Actice CEEs (as decribed in the GSMA NFC Handset Requirements and its Javadoc).

Referenced requirement

- TS26_NFC_REQ_127

15.7.3.1.1 Test sequence No 1

Initial Conditions

- A Secure Element (UICC)exists and is stored in (SEName)
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "defineOffHostService" method is called with the following parameters: • description: "*Description"* • SEName: (SEName)	Method SHALL return an instance of the "OffHostService" class
2		From the "OffHostService" instance returned by the "defineOffHostService" method, "getDescription" method is called	• No exception thrown by the system • Method SHALL return the string "Description"

15.7.3.1.2 Test sequence No 2

Initial Conditions

- A Secure Element (UICC)exists and is stored in (SEName)
- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "defineOffHostService" method is called with the following parameters: • description: *null* • SEName: (SEName)	• No exception thrown by the system • Method SHALL return an instance of the "OffHostService"
2		From the "OffHostService" instance returned by the "defineOffHostService" method, "getDescription" method is called	• No exception thrown by the system • Method SHALL return *null*

15.7.3.1.3 Test sequence No 3

Initial Conditions

- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "defineOffHostService" method is called with the following parameters: • description: *Description* • SEName: *SIMXX*	*2 Options:* • Method is returning an instance of the "OffHostService" →Go to Step 2 • An "IllegalArgumentException" is thrown by the system →Test Completed

Step	Direction	Sequence	Expected Result
2		From the "OffHostService" instance returned by the "defineOffHostService" method, "getDescription" method is called	• No exception thrown by the system • Method SHALL return the string "Description"

15.7.3.1.4 Test sequence No 4

Initial Conditions

- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "defineOffHostService" method is called with the following parameters: • description: *null* • SEName: *SIMXX*	*2 Options:* • Method is returning an instance of the "OffHostService" →Go to Step 2 • An "IllegalArgumentException" is thrown by the system →Test Completed

Step	Direction	Sequence	Expected Result
2		From the "OffHostService" instance returned by the "defineOffHostService" method, "getDescription" method is called	• No exception thrown by the system • Method SHALL return *null*

15.7.3.2 APIs for handling Dynamic AID Registration

Test Purpose

Ensure DUT implements correctly all APIs related to Dynamic AIDs Registration as decribed in the GSMA NFC Handset Requirements and its Javadoc.

Referenced requirement

- TS26_NFC_REQ_127

15.7.3.2.1 Test sequence No 1

Initial Conditions

- An instance of the "NfcController" class has been retrieved successfully and is not *null*
- An instance of "OffHostService" class (previously created dynamically with GSMA APIs) is available in variable [var01]

Step	Direction	Sequence	Expected Result
1		From the "NfcController" instance, the "deleteOffHostService" method is called with the following parameter: • service: [var01]	
2		From the "NfcController" instance, the "getOffHostServices" method is called	Service referred in the variable [var01] does not appear in the list returned by the method
3		From the "Setting" menu, the "Tap&Pay" entry is opened	Service referred in the variable [var01] is not present in the list of available services

15.7.3.2.2 Test sequence No 2

Initial Conditions

- An instance of the "NfcController" class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1		From the "Setting" menu, the "Tap&Pay" entry is opened	
2		From the "NfcController" instance, the "deleteOffHostService" method is called with the following parameter: • service: *null*	*2 Options:* • No exception is thrown • An "IllegalArgumentException" is thrown by the system • A "NullPointerException" is thrown by the system In all cases, no service is deleted
3		From the "Setting" menu, the "Tap&Pay" entry is opened	List of service in Step 1 should be the same

16 BlackBerry specific test cases

17 Windows Phone specific test cases

18 Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.2	10/09/13	First published version	GSM Association	GSMA NFC Project
2.0	15/01/14	Updated in accordance with the NFC Handset Requirements version 4.0 •	TSG	Paul Gosden/ GSMA
3.0	25/04/14	Updated Introduction, Scope, Abbreviations, Terms of Definitions, References An enhanced document structure with a new test case and section numbering. A new test case layout with aligned structure for all test cases. Addition of tables for recommended Test Case Applicability and a list of optional device features. Improvements to the definition of the Test Environment Improvements of existing Test Cases Addition of new Test Cases or deletion of Test Cases (e.g. if covered by referenced specifications or other Test Cases) Tables in the Annex with a complete list of test cases and an option and applicability table.	TSG	P Gosden GSMA
4.0	10/10/14	The changes to the TS.27 NFC Handset Test Book V4.0 include the following: • Alignment with TS.26 NFC Handset Requirement V6.0. • Test descriptions improvements within many sections. • Adoption of Global Platform SEAC Test Plan V1.06, section 5 (169 tests). • New test cases added in sections 3, 7, 8, 11, 12 and 15 (60 tests). • Reference to GCF WI-190 included to align with GCF(5 tests). • Applicability table updated. • Removal of SCWS. • Tables in the Annex with a complete list of test cases, Option and Applicability table.	TSG	Kay Fritz, Donna Mackay.

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.

Annex A Reference Application

The following Annex provides clarification on the application to be used to complete the reference transaction.

A.1 Description

The applet simulates an internal file structure described in paragraph A.3.

The operations permitted are the file selection described in section A.4, the file reading described in section A.4.2 and the file update that is described in paragraph A.4.3.

The applet also implements the External Authenticate command described in paragraph A.4.4.

A.2 AID

- Package A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 50
- Applet A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41

A.3 Structure FileGil

The structure file of the applet test is as follows:

5F 00 (DR) Folder

1F 00 (EF) First file in the folder initialized to FF

1F 01 (EF) Second file in the folder initialized to 00

The file size is 128 byte.

A.4 Commands Permitted

A.4.1 SELECT

This command is used to select the applet, the directory (5F 00) or files (1F 00, 1F 01)

Code	Value	Meaning
CLA	00	
INS	A4	
P1	04 o 00	04 when you select the applet 00 when you select the directory or files
P2	00	
Lc	Data Length	
Data	Data	Applet AID or Directory AID or files AID

Table 10: Select command details

A.4.2 READ BINARY

This command is used to read the contents of the selected file

Code	Value	Meaning
CLA	00	
INS	B0	
P1	00	
P2	00	
Le	80	

Table 11: Read Binary command details

A.4.3 UPDATE BINARY

This command is used to update the contents of the selected file

Code	Value	Meaning
CLA	00	
INS	D6	
P1	00	
P2	00	
Lc	80	
Data	Data to be updated	

Table 12: Update Binary command details

A.4.4 EXTERNAL AUTHENTICATE

This command is used to verify the input data encrypted, to be equal to the applet's data decrypted.

The input data correspond to the string "00 01 02 03 04 05 06 07" encrypted 3DES with 3 keys (K1 = A0 A1 A2 A3 A4 A5 A6 A7, K2 = B0 B1 B2 B3 B4 B5 B6 B7, K3 = C0 C1 C2 C3 C4 C5 C6 C7) and CBC (ICV = D0 D1 D2 D3 D4 D5 D6 D7).

The applet decrypted input data, if the data correspond to the string in clear (00 01 02 03 04 05 06 07) the applet will respond with 90 00, otherwise with 69 84.

Code	Value	Meaning
CLA	04	
INS	82	

P1	00	
P2	00	
Lc	08	
Data	9E EA C0 F9 4D 60 53 34	

Table 13: External Authenticate command details

A.5 Source Code (Java)

```
---------------------------------------------------------------------------------------------------
/*
* Copyright 2014 Telecom Italia S.p.A.
* Licensed under the Apache License, Version 2.0 (the “License”);
* you may not use this file except in compliance with the License.
* You may obtain a copy of the License at

* http://www.apache.org/licenses/License-2.0

* Unless required by applicable law or agreed to in writing, software
* distributed under the License is distributed on an “AS IS” BASIS,
* WITHOUT WARRANTIES OR CONDITIONS OR ANY KIND, either express or implied.
* See the License for the specific language governing permissions and
* limitations under the License.
*
*/

package com.gsma.test.nfc;

// None Imported packages
// None specific import for Javacard API access
import javacard.framework.APDU;
import javacard.framework.Applet;
import javacard.framework.ISO7816;
import javacard.framework.ISOException;
import javacard.framework.JCSystem;
import javacard.framework.Util;
import javacard.security.DESKey;
import javacard.security.KeyBuilder;
import javacardx.crypto.Cipher;
import uicc.access.FileView;
import uicc.access.UICCException;

public class Test extends Applet {

          private FileView mFileView;
          static final short FID_MASTER_FILE = 0x5F00;
          static final short FID_FILE_1F00 = (short) 0x1F00;
          static final short FID_FILE_1F01 = (short) 0x1F01;
          static final short FILE_SIZE = 128;
          private byte[] mTmp = new byte[FILE_SIZE];
          static final byte INS_UPDATE_BINARY = (byte) 0xD6;
          static final byte INS_SELECT_FILE = (byte) 0xA4;
          static final byte INS_READ_BINARY = (byte) 0xB0;
          static final byte INS_EXTERNAL_AUTHENTICATE = (byte) 0x82;
          private byte[] mBufferIn = new byte[(short) 128 + 5];
          private byte[] mBufferOut = new byte[(short) 128];
          static final short SW_GENERIC_ERROR = 0x6500;
          static final short SW_CMD_INCOMPATIBLE_WITH_FILE_STRUCTURE = (short) 0x6981;

          private Cipher desCipher;
          private DESKey key;

          private final static byte[] KEY_BYTES = new byte[] { (byte) 0xa0,
                              (byte) 0xa1, (byte) 0xa2, (byte) 0xa3, (byte) 0xa4, (byte) 0xa5,
                              (byte) 0xa6, (byte) 0xa7, (byte) 0xb0, (byte) 0xb1, (byte) 0xb2,
                              (byte) 0xb3, (byte) 0xb4, (byte) 0xb5, (byte) 0xb6, (byte) 0xb7,
                              (byte) 0xc0, (byte) 0xc1, (byte) 0xc2, (byte) 0xc3, (byte) 0xc4,
                              (byte) 0xc5, (byte) 0xc6, (byte) 0xc7 };

          private final static byte[] ICV = new byte[] { (byte) 0xd0,
                    (byte) 0xd1, (byte) 0xd2, (byte) 0xd3, (byte) 0xd4, (byte) 0xd5,
                    (byte) 0xd6, (byte) 0xd7};

          private final static byte[] RISULTATO = new byte[] { (byte) 0x00,
                    (byte) 0x01, (byte) 0x02, (byte) 0x03, (byte) 0x04, (byte) 0x05,
                    (byte) 0x06, (byte) 0x07};

          protected Test(byte[] baBuffer, short sOffset, byte bLength) {

                    register(baBuffer, (short) (sOffset + 1), (byte) baBuffer[sOffset]);

                    desCipher = Cipher.getInstance(Cipher.ALG_DES_CBC_NOPAD, false);
                    key = (DESKey) KeyBuilder.buildKey(KeyBuilder.TYPE_DES,
                                        KeyBuilder.LENGTH_DES3_3KEY, false);
                    key.setKey(KEY_BYTES, (short) 0);

                    mFileView = new FakeFileView();

                    clearFs();
                    ((FakeFileView) mFileView).reset();

          }

          public static void install(byte[] baBuffer, short sOffset, byte bLength) {
                    // None applet instance creation
                    new Test(baBuffer, sOffset, bLength);

          }

          public void process(APDU apdu) throws ISOException {
                    if (selectingApplet())
                              return;

                    byte[] buffer = apdu.getBuffer();
```

```
                Util.arrayCopyNonAtomic(buffer, (short) 0, mBufferIn, (short) 0,
                                ISO7816.OFFSET_CDATA);
                if (buffer[ISO7816.OFFSET_INS] != INS_READ_BINARY) {
                        short lc = (short) (buffer[ISO7816.OFFSET_LC] & 0xFF);
                                if (lc != 0 && lc <=128) {
                                        apdu.setIncomingAndReceive();
                                        Util.arrayCopyNonAtomic(buffer, (short) 0, mBufferIn, (short) 0, (short)
                                        (ISO7816.OFFSET_CDATA + lc));
                                } else {
                                ISOException.throwIt(ISO7816.SW_WRONG_LENGTH);
                        }
                }

                short len = exchangeAPDU(mBufferIn, mBufferOut);
                Util.arrayFillNonAtomic(mBufferIn, (short) 0, (short) mBufferIn.length,
                                (byte) 0);
                if (len > 0) {
                        apdu.setOutgoing();
                        apdu.setOutgoingLength(len);
                        apdu.sendBytesLong(mBufferOut, (short) 0, len);
                        Util.arrayFillNonAtomic(mBufferOut, (short) 0, (short) mBufferOut.length, (byte) 0);
                }

        }

        private void clearFs() {
                mFileView.select(FID_MASTER_FILE);

                mFileView.select(FID_FILE_1F00);
                fillFile(mFileView, FILE_SIZE, (byte) 0xFF);
                mFileView.select(FID_FILE_1F01);
                fillFile(mFileView, FILE_SIZE, (byte) 0x00);

        }

        private void fillFile(final FileView fileView, final short fileSize,
                        final byte filler) throws UICCException {
                Util.arrayFillNonAtomic(mTmp, (short) 0, fileSize, filler);
                short offset = 0;
                for (short fs = fileSize; fs > 0;) {
                        short xfer = (short) (fs > mTmp.length ? mTmp.length : fs);
                        fileView.updateBinary(offset, mTmp, (short) 0, xfer);
                        fs -= xfer;
                        offset += xfer;
                }
        }

        protected short exchangeAPDU(byte[] buffer, byte[] outBuf) {
                //None HUGE ASSUMPTION: buffer and outBuf are different!

                //None TODO: this check will need yo be modified the day we get UICC's with
                //None support for logical channels 04-19
                byte cla = (byte) (buffer[ISO7816.OFFSET_CLA]);
                //None High-order nibble should be zero
                if ((cla & 0xF0) != 0) {
                        ISOException.throwIt(ISO7816.SW_CLA_NOT_SUPPORTED);
                }

                byte ins = buffer[ISO7816.OFFSET_INS];
                byte p1 = buffer[ISO7816.OFFSET_P1];
                byte p2 = buffer[ISO7816.OFFSET_P2];
                short lc = (short) (buffer[ISO7816.OFFSET_LC] & 0x00FF);
                short le = (short) 0;

                switch (ins) {

                case INS_SELECT_FILE: //None OK
                {
                        if (p1 != (byte) 0 || p2 != (byte) 0) {
                                ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
                        }
                        if (lc != (byte) 2) {
                                ISOException.throwIt(ISO7816.SW_WRONG_LENGTH);
                        }
                        try {
                                mFileView.select(Util.makeShort(buffer[ISO7816.OFFSET_CDATA],
                                                buffer[ISO7816.OFFSET_CDATA + 1]));
                        } catch (UICCException e) {
                                uicc2isoException(e);
                        }
                        break;
                }

                case INS_UPDATE_BINARY: //None OK
                {
                        if (p2 == (byte) 0x00) {
                                if (p1 >= (byte) 0x00 && p1 <= (byte) 0x7F) {
                                        if (lc == 0 || lc > 128) {
                                        ISOException.throwIt(ISO7816.SW_WRONG_LENGTH);
                        }
                        short offset = (short) (buffer[ISO7816.OFFSET_P1] & 0x00FF);

                        JCSystem.beginTransaction();
                        try {
                                mFileView.updateBinary(offset, buffer, ISO7816.OFFSET_CDATA, lc);
                        } catch (UICCException e) {
                                uicc2isoException(e);
                        }
                        JCSystem.commitTransaction();
                        }else{
        ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
}
```

```
} else {
ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
}


                    }
                              break;

                    case INS_READ_BINARY: //None OK
                    {
if (p2 == (byte) 0x00) {
if (p1 >= (byte) 0x00 && p1 <= (byte) 0x7F) {
if (lc == 0 || lc > 128) {
ISOException.throwIt(ISO7816.SW_WRONG_LENGTH);
} else {
le = lc;
}
short offset = (short) (buffer[ISO7816.OFFSET_P1] & 0x00FF);
                              try {
                                        mFileView.readBinary(offset, outBuf, (short) 0, le);
                              } catch (UICCException e) {
                                        uicc2isoException(e);
                              }
          }else{
ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
}
} else {
ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
}

                    }
                              break;
                    case INS_EXTERNAL_AUTHENTICATE:{
if (p1 == (byte) 0x00 && p2 == (byte) 0x00) {
if(lc == 8){

                              desCipher.init(key, Cipher.MODE_DECRYPT, ICV, (short)0, (short) ICV.length);
                              le = desCipher.doFinal(buffer, ISO7816.OFFSET_CDATA, lc, outBuf, (short) 0);
                              if(Util.arrayCompare(RISULTATO, (short) 0, outBuf, (short) 0, le)!= 0){
                                        ISOException.throwIt(ISO7816.SW_DATA_INVALID);
                              }
                              le=0;
}else{
ISOException.throwIt(ISO7816.SW_WRONG_LENGTH);
}
} else {
ISOException.throwIt(ISO7816.SW_INCORRECT_P1P2);
}

                    }break;
                    default: {
                              ISOException.throwIt(ISO7816.SW_INS_NOT_SUPPORTED);
                    }
                    }
                    return le;
          }

          static private void uicc2isoException(UICCException e) {
                    short sw = SW_GENERIC_ERROR;
                    switch (e.getReason()) {
                    case UICCException.COMMAND_INCOMPATIBLE:
                              sw = SW_CMD_INCOMPATIBLE_WITH_FILE_STRUCTURE;
                              break;

                    case UICCException.FILE_NOT_FOUND:
                              sw = ISO7816.SW_FILE_NOT_FOUND;
                              break;

                    case UICCException.INVALID_MODE:
                              sw = ISO7816.SW_INCORRECT_P1P2;
                              break;

                    case UICCException.NO_EF_SELECTED:
                              sw = ISO7816.SW_COMMAND_NOT_ALLOWED;
                              break;

                    case UICCException.OUT_OF_FILE_BOUNDARIES:
                              sw = ISO7816.SW_WRONG_LENGTH;
                              break;

                    case UICCException.OUT_OF_RECORD_BOUNDARIES:
                              sw = ISO7816.SW_INCORRECT_P1P2;
                              break;

                    case UICCException.RECORD_NOT_FOUND:
                              sw = ISO7816.SW_RECORD_NOT_FOUND;
                              break;
                    }
                    ISOException.throwIt(sw);
          }
}
```

```
/*
*         Copyright 2014 Telecom Italia S.p.A.

 * Licensed under the Apache License, Version 2.0 (the "License");
 * you may not use this file except in compliance with the License.
 * You may obtain a copy of the License at

 *  http://www.apache.org/licenses/LICENSE-2.0

 * Unless required by applicable law or agreed to in writing, software
 * distributed under the License is distributed on an "AS IS" BASIS,
 * WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied.
 * See the License for the specific language governing permissions and
 * limitations under the License.
 *
 */

package com.gsma.test.nfc;

import javacard.framework.ISOException;
import javacard.framework.Util;
import uicc.access.FileView;
import uicc.access.UICCException;

public class FakeFileView implements FileView {

          private static final short FILETYPE_TRANSPARENT = 1;
          private short mSelFid;
          private byte[] mSelFile;
          private short mFileType;
          private short mRecSize;

          static final short FILE_SIZE = 128;
          final private byte[] file_1F00 = new byte[FILE_SIZE];
          final private byte[] file_1F01 = new byte[FILE_SIZE];
          static final short FID_MASTER_FILE = 0x5F00;
          static final short FID_FILE_1F00 = (short) 0x1F00;
          static final short FID_FILE_1F01 = (short) 0x1F01;

          static final short SW_BUG = (short) 0x98FF;

          private void unsupported() {
                    ISOException.throwIt(SW_BUG);
          }

          public void activateFile() throws UICCException {
                    unsupported();
          }

          public void deactivateFile() throws UICCException {
                    unsupported();
          }

          public short increase(byte[] arg0, short arg1, short arg2, byte[] arg3,
                              short arg4) throws NullPointerException,
                              ArrayIndexOutOfBoundsException, UICCException {
                    unsupported();
                    return 0;
          }

          public short searchRecord(byte arg0, short arg1, short arg2, byte[] arg3,
                              short arg4, short arg5, short[] arg6, short arg7, short arg8)
                              throws NullPointerException, ArrayIndexOutOfBoundsException,
                              UICCException {
                    unsupported();
                    return 0;
          }

          public void select(byte arg0) throws UICCException {
                    unsupported();
          }

          public short select(short arg0, byte[] arg1, short arg2, short arg3)
                              throws NullPointerException, ArrayIndexOutOfBoundsException,
                              UICCException {
                    unsupported();
                    return 0;
          }

          public short status(byte[] arg0, short arg1, short arg2)
                              throws NullPointerException, ArrayIndexOutOfBoundsException,
                              UICCException {
                    unsupported();
                    return 0;
          }

          public short readRecord(short recNumber, byte mode, short recOffset,
                              byte[] resp, short respOffset, short respLength)
                              throws NullPointerException, ArrayIndexOutOfBoundsException,
                              UICCException {
                    unsupported();
                    return 0;
          }

          public void updateRecord(short recNumber, byte mode, short recOffset,
                              byte[] data, short dataOffset, short dataLength)
                              throws NullPointerException, ArrayIndexOutOfBoundsException,
                              UICCException {
                    unsupported();
          }
```

```
        // ##########################################################################
        // S U P P O R T E D C O M M A N D S
        // ##########################################################################

        private byte[] getFileArray(short fid) {
                switch (fid) {
                case FID_FILE_1F00:
                        return file_1F00;
                case FID_FILE_1F01:
                        return file_1F01;
                default:
                        return null;
                }
        }

        private short getFileType(short fid) {
                switch (fid) {
                case FID_FILE_1F00:
                case FID_FILE_1F01:
                        return FILETYPE_TRANSPARENT;
                default:
                        unsupported();
                        return -1;
                }
        }

        public void select(short fid) throws UICCException {
                if (fid == FID_MASTER_FILE) {
                        mSelFid = fid;
                } else {
                        switch (mSelFid) {
                        case FID_MASTER_FILE:
                        case FID_FILE_1F00:
                        case FID_FILE_1F01:
                                if (fid == FID_FILE_1F00 || fid == FID_FILE_1F01) {
                                        mSelFid = fid;
                                } else {
                                        UICCException.throwIt(UICCException.FILE_NOT_FOUND);
                                }
                                break;
                        default:
                                UICCException.throwIt(UICCException.FILE_NOT_FOUND);
                        }
                }
                mSelFile = getFileArray(mSelFid);
                if (mSelFile != null) {
                        mFileType = getFileType(mSelFid);
                        mRecSize = (short) (mFileType >>> 8);
                        mRecSize = (mRecSize == 0) ? 256 : mRecSize;
                        mFileType &= 0xFF;
                }
        }

        public short readBinary(short fileOffset, byte[] resp, short respOffset,
                        short respLength) throws NullPointerException,
                        ArrayIndexOutOfBoundsException, UICCException {
                checkEFSelected();
                checkCommandCompatible(FILETYPE_TRANSPARENT);

                if (fileOffset < 0 || fileOffset + respLength > mSelFile.length) {
                        UICCException.throwIt(UICCException.OUT_OF_FILE_BOUNDARIES);
                }
                Util.arrayCopy(mSelFile, fileOffset, resp, respOffset, respLength);
                return (short) (respOffset + respLength);
        }

        public void updateBinary(short fileOffset, byte[] data, short dataOffset,
                        short dataLength) throws NullPointerException,
                        ArrayIndexOutOfBoundsException, UICCException {

                checkEFSelected();
                checkCommandCompatible(FILETYPE_TRANSPARENT);

                if (fileOffset + dataLength > mSelFile.length) {
                        UICCException.throwIt(UICCException.OUT_OF_FILE_BOUNDARIES);
                }
                Util.arrayCopy(data, dataOffset, mSelFile, fileOffset, dataLength);

        }

        private void checkEFSelected() {
                if (mSelFile == null) {
                        UICCException.throwIt(UICCException.NO_EF_SELECTED);
                }
        }

        private void checkCommandCompatible(short type) {
                if (mFileType != type) {
                        UICCException.throwIt(UICCException.COMMAND_INCOMPATIBLE);
                }
        }

        public void reset() {
                mSelFid = 0;
                mSelFile = null;
        }

}
```

Annex B Reference to other test plan

The GSMA NFC Handset Test Book refers to test specification developed by other organisations (EMVCo, ISO, ETSI, 3GPP, OMA). These organisations defined their own requirements for test benches, test applicability and pass criteria's.

B.1 SIMalliance

Reference test Specification: The test book refers to "SIMalliance Open Mobile API test specification for Transport API [5]

> "SIMalliance Open Mobile API test specification for Transport API" specifies a number of optional features for the device. The following table lists which optional features are mandatory according to GSMA requirements:

Options	Name	GSMA Status
access to the basic channel is blocked by the DUT	OP-002	Mandatory
access to the basic channel is allowed by the DUT	OP-003	SHALL not be supported
access to the default applet is blocked by the DUT	OP-011	Mandatory
access to the default applet is allowed by the DUT	OP-010	SHALL not be supported

The following test cases are applicable according to the applicability table of the referred SIMalliance test specification:

TS.27 Numbering	SIMalliance Clause [6]	Test case number and description
6.3.1.6.1.1	6.1.1	Constructor: SEService(Context context, SEService.CallBack listener)
6.3.1.6.1.2	6.1.2	Method: Reader[] getReaders()
6.3.1.6.1.3	6.1.3	Method: boolean isConnected()
6.3.1.6.1.4	6.1.4	Method: void shutdown()
6.3.1.6.1.5	6.1.5	Method: String getVersion()
6.3.1.6.2.1	6.2.1	Method: void serviceConnected(SEService service)
6.3.1.6.3.1	6.3.1	Method: String getName()
6.3.1.6.3.2	6.3.2	Method SEService getSEService()
6.3.1.6.3.3	6.3.3	Method: boolean isSecureElementPresent()
6.3.1.6.3.4	6.3.4	Method: Session openSession()
6.3.1.6.3.5	6.3.5	Method: void closeSessions()
6.3.1.6.4.1	6.4.1	Method: Reader getReader()
6.3.1.6.4.2	6.4.2	Method: byte[] getATR()
6.3.1.6.4.3	6.4.3	Method: void close()
6.3.1.6.4.4	6.4.4	Method: boolean isClosed()

TS.27 Numbering	SIMalliance Clause [6]	Test case number and description
6.3.1.6.4.5	6.4.5	Method: void closeChannels()
6.3.1.6.4.6	6.4.6	Method: Channel openBasicChannel() ID7
6.3.1.6.4.7	6.4.7	Method: Channel openLogicalChannel() ID1, ID2, ID3b, ID5 - ID17
6.3.1.6.5.1	6.5.1	Method: void close() ID1, ID3, ID4
6.3.1.6.5.2	6.5.2	Method: boolean isBasicChannel() ID2
6.3.1.6.5.3	6.5.3	Method: boolean isClosed()
6.3.1.6.5.4	6.5.4	Method: byte[] getSelectResponse()ID1, ID2, ID4 – ID8
6.3.1.6.5.5	6.5.5	Method: Session getSession()
6.3.1.6.5.6	6.5.6	Method: byte[] transmit(byte[] command) ID2 – ID21
6.3.1.6.5.7	6.5.7	Method: Boolean[] selectNext()

Table 14: SIMalliance test case

B.2 EMVCo

The GSMA requires device manufacturer to pass the whole EMVCo test plan in the scope of a device evaluation. This applies for Analog and Digital testing.

Note: In GCF, this is covered by GCF-CC Annex F.3.7.

B.3 ISO 10373-6

The GSMA limits the scope of this testing to the load modulation as per the ISO 10373-6 v2011. The GSMA agrees on the method nevertheless, the ISO 14443 pass criteria for these tests are modified for field weaker than 2.5A/m

The modified values can be found in the AFIMB implementation specifications for ISO14443 [4]: Requirement [Rdr5]. Those values are intended for contactless readers in public transport.

B.4 ETSI TS 102 613 SWP

Reference test Specification: ETSI TS 102 694-1 v9.5.0

ETSI TS 102 694-1 specifies a number of optional features for the device. The following table lists which optional features from ETSI TS 102 694-1 are mandatory (M) or recommended (R) according to GSMA requirements:

Item	Option	Mnemonic	GSMA Status
1	Class B	O_CLASS_B	R
2	Class C full power mode	O_CLASS_C_FULL	M

3	Class C low power mode	O_CLASS_C_LOW	C001
6	Terminal supports DEACTIVATED followed by subsequent SWP interface activation in full power mode	O_DEAC_SUBACT_FULL	M
7	Window size of 3	O_WS_3	R
8	Window size of 4	O_WS_4	R
9	HCI as per TS 102 622 [4]	O_102_622	M
10	CLT, ISO/IEC 14443-3 [5] Type A	O_CLT_A	M
11	CLT, ISO/IEC 18092 [8] Note this mandatory requirement will become effective from the 1st August 2015.	O_CLT_F	R Up to 31st July 2015 M – from 1st August 2015
18	Card Emulation, ISO/IEC 14443-4 [6] type A	O_CE_A	M
19	Card Emulation, ISO/IEC 14443-4 [6] type B	O_CE_B	M
C001: IF O_BAT_OFF THEN M ELSE O			

The following test cases are applicable:

1) Test cases verified by GCF WI 133 are listed in Table 15. These test cases are validated by GCF.

Index	TC Title
5.3.2.2.2	Test case 1: activation of SWP additionally to other interfaces
5.3.2.2.3	Test case 2: activation of SWP in low power mode
5.3.2.3.2	Test case 1: SWP initial activation in full power mode – normal procedure
5.3.2.3.4	Test case 3: SWP initial activation in full power mode – corrupted ACT_SYNC frame (repeat the last frame)
5.3.2.3.5	Test case 4: SWP initial activation in full power mode – no ACT_SYNC frame (repeat the last frame)
5.3.2.3.7	Test case 6: SWP initial activation failed in full power mode – no ACT_SYNC frame (multiple)
5.3.2.3.9	Test case 8: SWP Initial activation in full power mode – no ACT_READY frame (repeat last frame)
5.3.2.3.10	Test case 9: SWP initial activation failed in full power mode – corrupted ACT_READY frame (multiple)
5.3.2.3.12	Test case 11: SWP initial activation in low power mode
5.3.2.3.13	Test case 12:SWP initial activation in low power mode – corrupted ACT_SYNC frame (repeat the last frame)
5.3.2.3.14	Test case 13: SWP initial activation in low power mode – no ACT_SYNC frame (repeat the last frame)
5.3.2.3.15	Test case 14: SWP initial activation failed in low power mode – corrupted

Index	TC Title
	ACT_SYNC frame (multiple)
5.3.2.3.16	Test case 15: SWP initial activation failed in low power mode – no ACT_SYNC frame (multiple)
5.3.2.3.17	Test case 16: SWP subsequent activation in full power mode
5.4.1.3.2	Test case 1: current provided in low power mode, no spikes
5.4.1.3.3	Test case 2: current provided in low power mode, with spikes
5.4.1.4.2	Test case 1: communication with S2 variation in full power mode
5.4.1.4.3	Test case 2: communication with S2 variation in low power mode
5.4.1.5.2.2	Test case 1: communication with S2 variation in full power mode
5.4.1.5.2.3	Test case 2: communication with S2 variation in low power mode
5.5.1.2	Test case 1: S1 waveforms, default bit duration
5.5.1.3	Test case 2: S1 waveforms, extended bit durations
5.5.3.2	Test case 1: SWP states and transitions, communication
5.5.4.2	Test case 1: power provided in full power mode
5.5.4.3	Test case 2: switching from full to low power mode
5.5.4.4	Test case 3: switching from low to full power mode
5.6.2.2.2	Test case 1: interpretation of incorrectly formed frames – SHDLC RSET frames
5.6.2.2.3	Test case 2: interpretation of incorrectly formed frames – SHDLC I-frames
5.6.2.3.2	Test case 1: behaviour of CLF with bit stuffing in frame
5.6.3.2.2	Test case 1: ignore ACT LLC frame reception after the SHDLC link establishment
5.6.3.2.3	Test case 2: ignore ACT LLC frame reception in CLT session
5.6.3.2.5	Test case 4: closing condition of CLT session whereas SHDLC link has been established before CLT session
5.6.4.2.2	Test case 1: not matching SYNC_ID verification in low power mode
5.7.1.2	Test Case 1: data passed up to the next layer
5.7.1.3	Test Case 2: error management – corrupted I-frame
5.7.1.4	Test Case 3: error management – corrupted RR frame
5.7.6.4.2	Test case 1: initial state at link reset – reset by the UICC
5.7.7.3.2	Test Case 1: link establishment by the UICC
5.7.7.3.3	Test case 2: Link establishment and connection time out
5.7.7.3.4	Test Case 3: requesting unsupported window size and/or SREJ support - link establishment by UICC
5.7.7.3.5	Test Case 4: forcing lower window size and SREJ not used – link establishment by the T
5.7.7.5.2	Test case 1: I-frame transmission
5.7.7.5.3	Test case 2: I-frame reception - single I-Frame reception
5.7.7.5.4	Test case 3: I-frame reception - multiple I-Frame reception
5.7.7.6.2	Test case 1: REJ transmission – multiple I-frames received
5.7.7.6.3	Test case 2: REJ reception
5.7.7.7.2	Test Case 1: retransmission of multiple frames

Index	TC Title
5.7.7.8.2	Test case 1: RNR reception
5.8.5.2	Test case 1: ISO/IEC14443-3 Type A, no administrative command
5.8.6.3.1.2	Test case 1: opening a CLT session with CL_PROTO_INF(A)
5.9.2.1.2	Test case 1: CLF processing time - Type A aligned communication, with RF response
5.9.2.1.3	Test case 2: CLF processing time, no RF response
5.9.2.2.2	Test case 1: CLF processing time, Request Guard Time - Type A state transition
5.9.2.2.3	Test case 2: CLF processing time, Request Guard Time from HALT state- Type A state transition

Table 15: List of applicable test cases from GCF WI 133

2) Test cases verified by GCF WI 190 are listed in Table 16. These test cases are validated by GCF.

Index	TC Title
5.3.2.3.6	Test case 5: SWP initial activation failed in full power mode – corrupted ACT_SYNC frame (multiple)
5.3.2.3.8	Test case 7: SWP Initial activation in full power mode – corrupted ACT_READY frame (repeat last frame)
5.3.2.3.11	Test case 10: SWP initial activation failed in full power mode – no ACT_READY frame (multiple)
5.3.2.3.19	Test case 18: SWP initial activation in full power mode – send ACT frames in wrong order, ACT_READY frame after activation (repeat the last frame)
5.5.3.3	SWP resume after upper layer indication that the UICC requires no more activity on this interface
5.7.7.8.3	Test case 2: Empty I-frame transmission
5.8.6.3.2.2	Opening a CLT session with CL_PROTO_INF(F)
5.8.6.3.2.3	Empty CLT(F) Frame
5.8.6.3.2.4	RF off during CLT session not expecting Empty CLT
5.8.6.3.2.5	RF off during CLT session expecting Empty CLT
5.9.1.2.2	Transceiving non-chained data over RF in Card Emulation

Table 16: List of additional test cases

B.5 ETSI TS 102 622 HCI

Reference test Specification: ETSI TS 102 695-1

ETSI TS 102 695-1 specifies a number of optional features for the device. The following table lists which optional features from ETSI TS 102 695-1 are mandatory (M) or recommended (R) according to GSMA requirements:

Item	Option	Mnemonic	GSMA Status
1	Data link layer specified in TS 102 613 is used	O_102_613	M
2	Card RF gate for technology A is	O_CE_TypeA	M

Item	Option	Mnemonic	GSMA Status
	supported		
3	Card RF gate for technology B is supported	O_CE_TypeB	M
4	Reader RF gate for technology A is supported	O_Reader_TypeA	M
5	Reader RF gate for technology B is supported	O_Reader_TypeB	M
6	Card RF gate for technology F is supported	O_CE_TypeF	R
7	Low power mode is supported	O_Low_Power_Mode	C001
8	Item 2 and item 3 are supported.	O_CE_TypeA AND O_CE_TypeB	M
9	Item 6 and either item 2 or item 3 is supported	(O_CE_TypeA OR O_CE_TypeB) AND O_CE_TypeF	R
10	CLT for Type A as specified in TS 102 613 [**Error! Reference source not found.**] is supported.	O_CE_CLT_TypeA	M
11	Item 10 and item 3 are supported.	O_CE_CLT_TypeA AND O_CE_TypeB	M
12	Connectivity gate is supported in the terminal host	O_Conn	M
C001: IF O_BAT_OFF THEN M ELSE O			

The following test cases shall be verified:

1) Test cases verified by GCF WI 133 are listed in Table 17. These test cases are validated by GCF.

   All the test cases listed by work item 133 shall be run.

Index	TC Title
5.1.3.2	TC 1: existence of gates
5.1.4.2	TC 1: static pipe deletion
5.1.4.3	TC 2: initial pipe state and persistence of pipe state and registry value
5.1.5.2	TC 1: registry deletion
5.2.2.2	TC 1: commands/events on pipe which is not open
5.3.1.2.3.2	TC 1: ANY_OPEN_PIPE reception
5.3.1.2.4.2	TC 1: ANY_CLOSE_PIPE reception
5.3.2.2	TC 1: response to unknown command
5.3.3.2	TC 1: reception of unknown events
5.4.2.1.1.2	TC 1: SESSION_IDENTITY

Index	TC Title
5.4.2.1.1.3	TC 2: MAX_PIPE
5.4.2.1.1.4	TC 3: WHITELIST
5.4.2.1.1.5	TC 4: HOST_LIST
5.4.2.3.1.2	TC 1: registry parameters
5.5.1.2.2	TC 1: valid pipe deletion from host to host controller
5.5.1.3.2	TC 1: identity reference data when TS 102 613 is used
5.5.1.3.3	TC 2: reception of ADM_CLEAR_ALL_PIPE – static pipes, dynamic pipes to host
5.5.4.2	TC 1: inhibited state
5.5.4.3	TC 2: inhibited state, followed by subsequent successful identity check
5.5.5.2	TC 1: processing of EVT_POST_DATA
5.6.1.2	TC 1: RF gate of type A
5.6.1.3	TC 2: RF gate of type B
5.6.3.3.4.2.2	TC 1: UID_REG - default
5.6.3.3.4.2.3	TC 2: SAK
5.6.3.3.4.2.4	TC 3: ATS – default parameters
5.6.3.3.4.2.5	TC 4: APPLICATION_DATA
5.6.3.3.4.2.6	TC 5: DATARATE_MAX
5.6.3.3.4.3.2	TC 1: PUPI_REG - default
5.6.3.3.4.3.3	TC 2: ATQB – verify the different parameter
5.6.3.3.4.3.4	TC 3: HIGHER_LAYER_RESPONSE
5.6.4.1.2	TC 1: ISO/IEC14443-3 Type A – Full Power Mode
5.6.4.1.3	TC 2: ISO/IEC14443-3 Type B

Table 17: List of applicable test cases from GCF WI 133

2) Test cases verified by GCF WI 190 are listed in Table 18. These test cases are validated by GCF.

Index	TC Title
5.6.1.4	Test case x: RF gate of type F
5.6.4.4.2	Test case 1: ISO/IEC 18092 Type F
5.6.4.4.3	Test case y: RF off during ISO/IEC 18092 [4] Type F commands handling
5.7.2.3.1.2	Test case 1: ISO/IEC 14443-4 [7] compliant type A
5.7.2.3.2.2	Test case 1: ISO/IEC 14443-4 [7] compliant type B
5.6.4.1.4	Test case 3: Routing EVT_FIELD_ON and EVT_FIELD_OFF to RF Gate with lowest GID
5.6.4.2.2	Test case 1: None ISO/IEC 14443-4 type A
5.8.2.3.5.2	Test case 1: EVT_TRANSACTION

Table 18: List of additional test cases

3) Additional applicable test cases from TS 102 695-1 are listed in Table 29

Index	TC Title
5.6.4.2.3	Test case 2: Routing EVT_FIELD_ON and EVT_FIELD_OFF to RF Gate with lowest GID
5.6.4.4.4	Test case 3: Routing EVT_FIELD_ON and EVT_FIELD_OFF to RF Gate with lowest GID

Table 29: List of additional test cases

.

B.6 ETSI TS 102.384, 3GPP 31.124

Reference test Specification: ETSI TS 102 384 v10.0.0 and 3GPP TS 31.124 v10.0.0

The test cases in Table 19 are applicable to verify TS26_NFC_REQ_078 as following:

1) Applicable test cases verified by GCF WI 035 are listed in Table 19. These test cases are validated by GCF.

Ref Spec	Index	TC Title	Sequence Name
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, no local address, no alpha identifier, no network access name
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment GPRS, no alpha identifier, with network access name
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with alpha identifier
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with null alpha identifier
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, command performed with modifications (buffer size)

Ref Spec	Index	TC Title	Sequence Name
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
3GPP TS 31.124	27.22.4.28.1	CLOSE CHANNEL(normal)	CLOSE CHANNEL, successful
3GPP TS 31.124	27.22.4.28.1	CLOSE CHANNEL(normal)	CLOSE CHANNEL, with an invalid channel identifier
3GPP TS 31.124	27.22.4.28.1	CLOSE CHANNEL(normal)	CLOSE CHANNEL, on an already closed channel
3GPP TS 31.124	27.22.4.29.1	RECEIVE DATA (NORMAL)	RECEIVE DATA, already opened channel, GPRS
3GPP TS 31.124	27.22.4.30.1	SEND DATA (normal)	SEND DATA, immediate mode, GPRS
3GPP TS 31.124	27.22.4.30.1	SEND DATA (normal)	SEND DATA, Store mode, GPRS
3GPP TS 31.124	27.22.4.30.1	SEND DATA (normal)	SEND DATA, Store mode, Tx buffer fully used, GPRS
3GPP TS 31.124	27.22.4.30.1	SEND DATA (normal)	SEND DATA, 2 consecutive SEND DATA Store mode, GPRS
3GPP TS 31.124	27.22.4.30.1	SEND DATA (normal)	SEND DATA, immediate mode with a bad channel identifier, GPRS
3GPP TS 31.124	27.22.4.31	GET CHANNEL STATUS	GET STATUS, with a BIP channel currently opened, GPRS
3GPP TS 31.124	27.22.4.31	GET CHANNEL STATUS	GET STATUS, after a link dropped, GPRS
3GPP TS 31.124	27.22.7.10	Data available event	EVENT DOWNLOAD – Data available, GPRS
3GPP TS 31.124	27.22.7.11	Channel Status event	EVENT DOWNLOAD – Channel Status on a link dropped

Table 19: List of applicable test cases from GCF WI 035

The applicable test cases to verify TS26_NFC_REQ_079:

1) The applicable test case from 3GPP TS 31.124 is listed in Table 20.

• Ref Spec	Index	TC Title	Sequence Name
3GPP TS 31.124	27.22.4.27.2	Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, no alpha identifier, with network access name

Table 20: applicable test cases from GCF WI 035

2) Additional test cases in Table 21shall be verified for all combinations of the following parameters:

- **Bearer Type** '02' = *GPRS / UTRAN packet service / E-UTRAN*
- **Bearer Type** '03' = default bearer for requested transport layer; (OPEN CHANNEL related to Default (network) Bearer)
- **UICC/terminal interface transport level (Transport protocol type + port number)** '01': UDP, UICC in client mode, remote connection
- **UICC/terminal interface transport level (Transport protocol type + port number)** '02': TCP, UICC in client mode, remote connection;

1. Note: these parameters are not tested in 3GPP 31.124.

TC Title	Sequence name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, no local address, no alpha identifier, no network access name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment GPRS, no alpha identifier, with network access name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with alpha identifier
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with null alpha identifier
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, command performed with modifications (buffer size)
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
CLOSE CHANNEL(normal)	CLOSE CHANNEL, successful
CLOSE CHANNEL(normal)	CLOSE CHANNEL, with an invalid channel identifier
CLOSE CHANNEL(normal)	CLOSE CHANNEL, on an already closed channel

TC Title	Sequence name
RECEIVE DATA (NORMAL)	RECEIVE DATA, already opened channel, GPRS
SEND DATA (normal)	SEND DATA, immediate mode, GPRS
SEND DATA (normal)	SEND DATA, Store mode, GPRS
SEND DATA (normal)	SEND DATA, Store mode, Tx buffer fully used, GPRS
SEND DATA (normal)	SEND DATA, 2 consecutive SEND DATA Store mode, GPRS
SEND DATA (normal)	SEND DATA, immediate mode with a bad channel identifier, GPRS
GET CHANNEL STATUS	GET STATUS, with a BIP channel currently opened, GPRS
GET CHANNEL STATUS	GET STATUS, after a link dropped, GPRS
Data available event	EVENT DOWNLOAD – Data available, GPRS
Channel Status event	EVENT DOWNLOAD – Channel Status on a link dropped

Table 21: Applicable test cases

The test cases in Table 22 are applicable to verify TS26_NFC_REQ_080 with all combinations of the following parameters:

- **Bearer Type** '02' = *GPRS / UTRAN packet service / E-UTRAN*
- **Bearer Type** '03' = default bearer for requested transport layer; (OPEN CHANNEL related to Default (network) Bearer)
- **UICC/terminal interface transport level (Transport protocol type + port number)** '01': UDP, UICC in client mode, remote connection;
- **UICC/terminal interface transport level (Transport protocol type + port number)** '02': TCP, UICC in client mode, remote connection

Note: these parameters are not tested in 3GPP 31.124.

TC Title	Sequence name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, no alpha identifier, with network access name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, no local address, no alpha identifier, no network access name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment GPRS, no alpha identifier, with network access name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with alpha identifier
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, with null alpha identifier
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, command performed with modifications (buffer size)

TC Title	Sequence name
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
Open Channel (related to GPRS)	OPEN CHANNEL, immediate link establishment, GPRS, open command with alpha identifier, User did not accept the proactive command
CLOSE CHANNEL(normal)	CLOSE CHANNEL, successful
CLOSE CHANNEL(normal)	CLOSE CHANNEL, with an invalid channel identifier
CLOSE CHANNEL(normal)	CLOSE CHANNEL, on an already closed channel
RECEIVE DATA (NORMAL)	RECEIVE DATA, already opened channel, GPRS
SEND DATA (normal)	SEND DATA, immediate mode, GPRS
SEND DATA (normal)	SEND DATA, Store mode, GPRS
SEND DATA (normal)	SEND DATA, Store mode, Tx buffer fully used, GPRS
SEND DATA (normal)	SEND DATA, 2 consecutive SEND DATA Store mode, GPRS
SEND DATA (normal)	SEND DATA, immediate mode with a bad channel identifier, GPRS
GET CHANNEL STATUS	GET STATUS, with a BIP channel currently opened, GPRS
GET CHANNEL STATUS	GET STATUS, after a link dropped, GPRS
Data available event	EVENT DOWNLOAD – Data available, GPRS
Channel Status event	EVENT DOWNLOAD – Channel Status on a link dropped

Table 22: Applicable test cases

The test cases are applicable to verify TS26_NFC_REQ_081 as following:

2) The test case verified by GCF WI 035 listed in Table 23

Ref Spec	Index	TC Title	Sequence name
3GPP TS 31.124	27.22.5.1	SMS-PP Data Download	Seq 1.9: SMS-PP Data Download over CS/PS, UTRAN/GERAN

Table 23: Applicable test cases

The test cases are applicable to verify ***Annex B*** as following:

1) The test case verified by GCF WI 035 listed in Table 24

Ref Spec	Index	TC Title	Sequence name
3GPP TS 31.124	27.22.4.26.1	LAUNCH BROWSER (No session already launched)	LAUNCH BROWSER, connect to the default URL
3GPP TS 31.124	27.22.4.26.1	LAUNCH BROWSER (No session already launched)	LAUNCH BROWSER, connect to the specified URL, alpha identifier length=0
3GPP TS 31.124	27.22.4.26.1	LAUNCH BROWSER (No session already launched)	LAUNCH BROWSER, Browser identity, no alpha identifier
3GPP TS 31.124	27.22.4.26.1	LAUNCH BROWSER (No session already launched)	LAUNCH BROWSER, one bearer specified and gateway/proxy identity
3GPP TS 31.124	27.22.4.26.2	LAUNCH BROWSER (Interaction with current session)	LAUNCH BROWSER, use the existing browser, connect to the default URL
3GPP TS 31.124	27.22.4.26.2	LAUNCH BROWSER (Interaction with current session)	LAUNCH BROWSER, close the existing browser session and launch new browser session, connect to the default URL
3GPP TS 31.124	27.22.4.26.2	LAUNCH BROWSER (Interaction with current session)	LAUNCH BROWSER, if not already launched
3GPP TS 31.124	27.22.7.9.1	Browser termination (normal)	EVENT DOWNLOAD - Browser termination

Table 24: Applicable test cases

The test cases in Table 25 are applicable to verify Annex B

Ref Spec	Index	TC Title	Sequence name
ETSI TS 102.384	27.22.7.18	HCI connectivity event	HCI connectivity event (normal)
ETSI TS 102.384	27.22.4.32	ACTIVATE	ACTIVATE

Table 25: List of additional test cases

The applicable test cases in Table 26 are to verify *TS26_NFC_REQ_086*.

Ref Spec	Index	TC Title

Ref Spec	Index	TC Title
TS 102 384	27.22.4.27.6	OPEN CHANNEL, TCP in LISTEN state, successful
TS 102 384	27.22.4.27.6	OPEN CHANNEL, TCP in LISTEN state, command performed with modification
TS 102 384	27.22.4.28.3	CLOSE CHANNEL, go to "TCP in LISTEN state", successful
TS 102 384	27.22.4.28.3	CLOSE CHANNEL, go to "TCP in CLOSED state", successful
TS 102 384	27.22.4.31.2	GET CHANNEL STATUS, in LISTEN state
TS 102 384	27.22.4.31.2	GET CHANNEL STATUS, in ESTABLISHED state
TS 102 384	27.22.7.10.2	EVENT DOWNLOAD - Data available, successful
TS 102 384	27.22.7.11.2	EVENT DOWNLOAD - Channel Status, TCP in LISTEN state
TS 102 384	27.22.7.11.2	EVENT DOWNLOAD - Channel Status, TCP in ESTABLISHED state

Table 26: List of applicable test cases from GCF WI 126

B.7 Void

B.8 GP Secure Element Access Control

Reference test Specification: The test book refers to “GlobalPlatform - SEAC DeviceSide Test Plan v1.0.6” specification.

The following table indicates which test cases are included in the current version of the Test Book :

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.1.1	ACE_DETECT_CORRUPTED_RULES_IN_ARF	1	ACCESS_DENIED_CORRUPTED_ARF_ERROR_00_No_EF_DIR (0b-1c-c7)	No
5.4.1.2		2	ACCESS_DENIED_CORRUPTED_ARF_ERROR_01_No_PKCS15_Referenced_in_EF_DIR (0b-20-bd)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.1.3		3	ACCESS_DENIED_CORRUPTED_ARF_ERROR_02_ODF_Bad_Padding (0b-8a-03)	Yes
5.4.1.4		4	ACCESS_DENIED_CORRUPTED_ARF_ERROR_03_DODF_Without_OID (0b-5b-03)	Yes
5.4.1.5		5	ACCESS_DENIED_CORRUPTED_ARF_ERROR_04_DODF_With_BadLength_BadOffset (0b-86-a3)	Yes
5.4.1.6		6	ACCESS_DENIED_CORRUPTED_ARF_ERROR_05_ACMF_Not_Found (0b-81-11)	Yes
5.4.1.7		7	ACCESS_DENIED_CORRUPTED_ARF_ERROR_06_ACMF_Zero_Length (0b-d1-10)	No
5.4.1.8		8	ACCESS_DENIED_CORRUPTED_ARF_ERROR_07_ACMF_Bad_Padding (0b-df-cf)	No
5.4.1.9		9	ACCESS_DENIED_CORRUPTED_ARF_ERROR_08_ACRF_Not_Found (0b-d2-a1)	Yes
5.4.1.10		10	ACCESS_DENIED_CORRUPTED_ARF_ERROR_09_ACRF_Zero_Length (0b-15-c1)	Yes
5.4.1.11		11	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0A_ACRF_Bad_Padding (0b-19-6f)	Yes
5.4.1.12		12	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0B_ACRF_Without_Any_Rule (0b-dc-4f)	Yes
5.4.1.13		13	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0C_ACCF_Not_Found (0b-ff-4d)	Yes
5.4.1.14		14	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0D_ACCF_Zero_Length (0b-fb-57)	Yes
5.4.1.15		15	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0E_ACCF_Bad_Padding (0b-f8-05)	No
5.4.1.16		16	ACCESS_DENIED_CORRUPTED_ARF_ERROR_0F_ACCF_Wrong_Certificate_Length (0b-9c-9a)	Yes
5.4.2.1	ACE_DETECT_CORRUPTED_RULES	1	ACCESS_DENIED_ERROR_APDU_AR_DO_Bad_Length (c0-c4-e9)	No
5.4.2.2		2	ACCESS_DENIED_ERROR_APDU_AR_DO_Bad_value (c0-62-b5)	No
5.4.2.3		3	ACCESS_DENIED_ERROR_NFC_AR_DO_Bad_Length (c0-90-88)	No
5.4.2.4		4	ACCESS_DENIED_ERROR_NFC_AR_DO_Bad_Value (c0-d7-66)	No
5.4.3.1	ACCESS_DENIED	1	ACCESS_DENIED_SUCCESS__ARA_M_locked (c0-36-1d)	Yes
5.4.3.2		2	ACCESS_DENIED_SUCCESS__ARA_M_not_present (c0-36-1e)	Yes
5.4.3.3		3	ACCESS_DENIED_SUCCESS__ARA_M_not_selectable (c0-f6-26)	Yes
5.4.4.1	ALGORITHM_SPECIFIC_DEVICE_APP_AND_SPECIFIC_SE_APP	1	ALGORITHM_A_SUCCESS__R1_SEApp1_DevApp1__R2_All_All__request_DevApp2_SEApp1 (c0-ee-09)	No
5.4.4.2		2	ALGORITHM_A_SUCCESS__R1_SEApp1_DevApp1__R2_All_All__request_DevApp2_SEApp1 (ff-ee-09)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.5.1	ALGORITHM_SPECIFIC_DEVICE_APP_AND_GENERIC_SE_APP	1	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1__R2_All_All__request_DevApp2_SEApp1 (c0-d4-fb)	No
5.4.5.2		2	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1__R2_All_All__request_DevApp2_SEApp1 (ff-d4-fb)	No
5.4.5.3		3	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_ALWAYS__R2_All_SEApp_DevApp2_NEVER (c0-40-30)	No
5.4.5.4		4	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_ALWAYS__R2_All_SEApp_DevApp2_NEVER (ff-40-30)	No
5.4.5.5		5	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_NEVER__R2_All_SEApp_DevApp2_ALWAYS (c0-80-26)	No
5.4.5.6		6	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_NEVER__R2_All_SEApp_DevApp2_ALWAYS (ff-80-26)	No
5.4.6.1	ALGORITHM_GENERIC_DEVICE_APP_AND_GENERIC_SE_APP	1	ALGORITHM_D_SUCCESS__All_SEApp_All_DevApp_APDU_access_ALWAYS (c0-2e-b0)	No
5.4.6.2		2	ALGORITHM_D_SUCCESS__All_SEApp_All_DevApp_APDU_access_ALWAYS (ff-2e-b0)	No
5.4.6.3		3	ALGORITHM_D_SUCCESS__No_Rule (c0-cf-16)	No
5.4.6.4		4	ALGORITHM_D_SUCCESS__No_Rule (ff-cf-16)	No
5.4.7.1	ANNEX_D_EXAMPLE_01	1	ANNEX_D_SUCCESS__example_01 (c0-38-cb)	No
5.4.7.2		2	ANNEX_D_SUCCESS__example_01 (ff-38-cb)	No
5.4.8.1	ANNEX_D_EXAMPLE_02	1	ANNEX_D_SUCCESS__example_02 (c0-a1-46)	No
5.4.8.2		2	ANNEX_D_SUCCESS__example_02 (ff-a1-46)	No
5.4.9.1	ANNEX_D_EXAMPLE_03	1	ANNEX_D_SUCCESS__example_03 (c0-39-49)	No
5.4.9.2		2	ANNEX_D_SUCCESS__example_03 (ff-39-49)	No
5.4.10.1	ANNEX_D_EXAMPLE_04	1	ANNEX_D_SUCCESS__example_04 (c0-2f-62)	No
5.4.10.2		2	ANNEX_D_SUCCESS__example_04 (ff-2f-62)	No
5.4.11.1	ANNEX_D_EXAMPLE_05	1	ANNEX_D_SUCCESS__example_05 (c0-d2-97)	No
5.4.11.2		2	ANNEX_D_SUCCESS__example_05 (ff-d2-97)	No
5.4.12.1	ANNEX_D_EXAMPLE_06	1	ANNEX_D_SUCCESS__example_06 (c0-51-08)	No
5.4.12.2		2	ANNEX_D_SUCCESS__example_06 (ff-51-08)	No
5.4.13.1	ANNEX_D_EXAMPLE_07	1	ANNEX_D_SUCCESS__example_07 (c0-e0-83)	No
5.4.13.2		2	ANNEX_D_SUCCESS__example_07 (ff-e0-83)	No
5.4.14.1	ANNEX_D_EXAMPLE_09	1	ANNEX_D_SUCCESS__example_09 (c0-4c-cb)	No
5.4.14.2		2	ANNEX_D_SUCCESS__example_09 (ff-4c-cb)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.15.1	ANNEX_D_EXAMPLE_10	1	ANNEX_D_SUCCESS__example_10 (c0-ad-7d)	No
5.4.15.2		2	ANNEX_D_SUCCESS__example_10 (ff-ad-7d)	No
5.4.16.1	ANNEX_D_EXAMPLE_11	1	ANNEX_D_SUCCESS__example_11 (c0-f4-52)	No
5.4.16.2		2	ANNEX_D_SUCCESS__example_11 (ff-f4-52)	No
5.4.17.1	ANNEX_D_EXAMPLE_12	1	ANNEX_D_SUCCESS__example_12 (c0-24-84)	No
5.4.17.2		2	ANNEX_D_SUCCESS__example_12 (ff-24-84)	No
5.4.18.1	ANNEX_D_EXAMPLE_13	1	ANNEX_D_SUCCESS__example_13 (c0-29-86)	No
5.4.18.2		2	ANNEX_D_SUCCESS__example_13 (ff-29-86)	No
5.4.19.1	ANNEX_D_EXAMPLE_14	1	ANNEX_D_SUCCESS__example_14 (c0-72-b7)	No
5.4.19.2		2	ANNEX_D_SUCCESS__example_14 (ff-72-b7)	No
5.4.20.1	ANNEX_D_EXAMPLE_15	1	ANNEX_D_SUCCESS__example_15 (c0-db-cb)	No
5.4.20.2		2	ANNEX_D_SUCCESS__example_15 (ff-db-cb)	No
5.4.21.1	ANNEX_D_EXAMPLE_16	1	ANNEX_D_SUCCESS__example_16 (c0-cf-17)	No
5.4.21.2		2	ANNEX_D_SUCCESS__example_16 (ff-cf-17)	No
5.4.22.1	ANNEX_D_EXAMPLE_17	1	ANNEX_D_SUCCESS__example_17 (c0-0b-29)	No
5.4.22.2		2	ANNEX_D_SUCCESS__example_17 (ff-0b-29)	No
5.4.23.1	ANNEX_D_EXAMPLE_18	1	ANNEX_D_SUCCESS__example_18 (c0-71-16)	No
5.4.23.2		2	ANNEX_D_SUCCESS__example_18 (ff-71-16)	No
5.4.24.1	APDU_FILTER_DEFINITION	1	APDU_FILTER_DEFINITION_SUCCESS__ALWAYS__true (c0-01-e8)	Yes
5.4.24.2		2	APDU_FILTER_DEFINITION_SUCCESS__ALWAYS__true (ff-01-e8)	Yes
5.4.24.3		3	APDU_FILTER_DEFINITION_SUCCESS__FILTER__1_filter__true (c0-dc-08)	Yes
5.4.24.4		4	APDU_FILTER_DEFINITION_SUCCESS__FILTER__1_filter__true (ff-dc-08)	Yes
5.4.24.5		5	APDU_FILTER_DEFINITION_SUCCESS__FILTER__2_filters__true (c0-6c-24)	No
5.4.24.6		6	APDU_FILTER_DEFINITION_SUCCESS__FILTER__2_filters__true (ff-6c-24)	No
5.4.24.7		7	APDU_FILTER_DEFINITION_SUCCESS__FILTER__3_filters__true (c0-44-7d)	No
5.4.24.8		8	APDU_FILTER_DEFINITION_SUCCESS__FILTER__3_filters__true (ff-44-7d)	No
5.4.24.9		9	APDU_FILTER_DEFINITION_SUCCESS__NEVER__false (c0-82-ed)	Yes
5.4.24.10		10	APDU_FILTER_DEFINITION_SUCCESS__NEVER__false (ff-82-ed)	Yes
5.4.25.1	ACE_MANAGE_BIG_RULES	1	BIG_RULES_MANAGEMENT_One_Big_Rule (c0-64-2b)	No
5.4.25.2		2	BIG_RULES_MANAGEMENT_One_Big_Rule (ff-64-2b)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.1	RULE_CONFLICT_RESOLUTION_MORE_RESTRICTIVE	1	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (c0-68-d7)	Yes
5.4.26.2		2	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (ff-68-d7)	Yes
5.4.26.3		3	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (c0-fb-a4)	Yes
5.4.26.4		4	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (ff-fb-a4)	Yes
5.4.26.5		5	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (c0-87-cd)	Yes
5.4.26.6		6	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (ff-87-cd)	Yes
5.4.26.7		7	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (c0-d7-15)	Yes
5.4.26.8		8	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (ff-d7-15)	Yes
5.4.26.9		9	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (c0-c9-58)	Yes
5.4.26.10		10	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (ff-c9-58)	Yes
5.4.26.11		11	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (c0-2d-54)	Yes
5.4.26.12		12	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (ff-2d-54)	Yes
5.4.26.13		13	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (c0-e1-ed)	Yes
5.4.26.14		14	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (ff-e1-ed)	Yes
5.4.26.15		15	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (c0-9b-af)	Yes
5.4.26.16		16	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (ff-9b-af)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.17		17	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_always (c0-73-51)	Yes
5.4.26.18		18	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_always (ff-73-51)	Yes
5.4.26.19		19	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_never (c0-42-e5)	Yes
5.4.26.20		20	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_never (ff-42-e5)	Yes
5.4.26.21		21	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_filter (c0-55-e2)	Yes
5.4.26.22		22	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_filter (ff-55-e2)	Yes
5.4.26.23		23	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_never (c0-78-2b)	Yes
5.4.26.24		24	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_never (ff-78-2b)	Yes
5.4.26.25		25	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_always (c0-b4-23)	Yes
5.4.26.26		26	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_always (ff-b4-23)	Yes
5.4.26.27		27	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_filter (c0-21-c2)	Yes
5.4.26.28		28	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_filter (ff-21-c2)	Yes
5.4.26.29		29	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_never (c0-e1-e4)	Yes
5.4.26.30		30	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_never__R2_apdu_never (ff-e1-e4)	Yes
5.4.26.31		31	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (c0-58-e2)	Yes
5.4.26.32		32	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (ff-58-e2)	Yes
5.4.26.33		33	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (c0-0c-1d)	Yes
5.4.26.34		34	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (ff-0c-1d)	Yes
5.4.26.35		35	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (c0-c4-0e)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.36		36	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (ff-c4-0e)	Yes
5.4.26.37		37	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (c0-29-e9)	Yes
5.4.26.38		38	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (ff-29-e9)	Yes
5.4.26.39		39	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (c0-c8-fb)	Yes
5.4.26.40		40	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (ff-c8-fb)	Yes
5.4.26.41		41	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (c0-49-26)	Yes
5.4.26.42		42	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (ff-49-26)	Yes
5.4.26.43		43	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (c0-00-3a)	Yes
5.4.26.44		44	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (ff-00-3a)	Yes
5.4.26.45		45	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (c0-c2-82)	Yes
5.4.26.46		46	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (ff-c2-82)	Yes
5.4.26.47		47	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (c0-41-a5)	Yes
5.4.26.48		48	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (ff-41-a5)	Yes
5.4.26.49		49	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (c0-16-b9)	Yes
5.4.26.50		50	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (ff-16-b9)	Yes
5.4.26.51		51	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (c0-03-40)	Yes
5.4.26.52		52	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (ff-03-40)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.53		53	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (c0-d7-89)	Yes
5.4.26.54		54	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (ff-d7-89)	Yes
5.4.26.55		55	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (c0-2d-17)	Yes
5.4.26.56		56	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (ff-2d-17)	Yes
5.4.26.57		57	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (c0-50-39)	Yes
5.4.26.58		58	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (ff-50-39)	Yes
5.4.26.59		59	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (c0-29-95)	Yes
5.4.26.60		60	CONFLICT_SUCCESS__1_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (ff-29-95)	Yes
5.4.26.61		61	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (c0-9c-1e)	Yes
5.4.26.62		62	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (ff-9c-1e)	Yes
5.4.26.63		63	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (c0-91-2e)	Yes
5.4.26.64		64	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (ff-91-2e)	Yes
5.4.26.65		65	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (c0-e8-cf)	Yes
5.4.26.66		66	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (ff-e8-cf)	Yes
5.4.26.67		67	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (c0-b3-b0)	Yes
5.4.26.68		68	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (ff-b3-b0)	Yes
5.4.26.69		69	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (c0-dd-11)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.70		70	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (ff-dd-11)	Yes
5.4.26.71		71	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (c0-28-6e)	Yes
5.4.26.72		72	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (ff-28-6e)	Yes
5.4.26.73		73	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (c0-73-cf)	Yes
5.4.26.74		74	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (ff-73-cf)	Yes
5.4.26.75		75	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (c0-db-29)	Yes
5.4.26.76		76	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (ff-db-29)	Yes
5.4.26.77		77	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_always (c0-3c-2b)	Yes
5.4.26.78		78	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_always (ff-3c-2b)	Yes
5.4.26.79		79	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_never (c0-99-f5)	Yes
5.4.26.80		80	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_always__R2_apdu_never (ff-99-f5)	Yes
5.4.26.81		81	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_filter (c0-81-35)	Yes
5.4.26.82		82	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_filter (ff-81-35)	Yes
5.4.26.83		83	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_never (c0-5c-7f)	Yes
5.4.26.84		84	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_filter__R2_apdu_never (ff-5c-7f)	Yes
5.4.26.85		85	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_always (c0-f2-08)	Yes
5.4.26.86		86	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_always (ff-f2-08)	Yes
5.4.26.87		87	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_filter (c0-bf-dc)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.88		88	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_filter (ff-bf-dc)	Yes
5.4.26.89		89	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_never___false (c0-7a-bc)	Yes
5.4.26.90		90	CONFLICT_SUCCESS__All_SEApp_1_DevApp__R1_apdu_never__R2_apdu_never___false (ff-7a-bc)	Yes
5.4.26.91		91	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (c0-b0-96)	No
5.4.26.92		92	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (ff-b0-96)	No
5.4.26.93		93	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (c0-52-54)	No
5.4.26.94		94	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (ff-52-54)	No
5.4.26.95		95	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (c0-70-6c)	No
5.4.26.96		96	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (ff-70-6c)	No
5.4.26.97		97	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (c0-58-13)	No
5.4.26.98		98	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (ff-58-13)	No
5.4.26.99		99	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (c0-4b-98)	No
5.4.26.100		100	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (ff-4b-98)	No
5.4.26.101		101	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (c0-38-d9)	No
5.4.26.102		102	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (ff-38-d9)	No
5.4.26.103		103	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (c0-85-6b)	No
5.4.26.104		104	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (ff-85-6b)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.105		105	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (c0-3c-9f)	No
5.4.26.106		106	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (ff-3c-9f)	No
5.4.26.107		107	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (c0-06-0b)	No
5.4.26.108		108	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (ff-06-0b)	No
5.4.26.109		109	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (c0-c8-59)	No
5.4.26.110		110	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (ff-c8-59)	No
5.4.26.111		111	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (c0-76-bc)	No
5.4.26.112		112	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (ff-76-bc)	No
5.4.26.113		113	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (c0-b0-d6)	No
5.4.26.114		114	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (ff-b0-d6)	No
5.4.26.115		115	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (c0-09-ca)	No
5.4.26.116		116	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (ff-09-ca)	No
5.4.26.117		117	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (c0-b1-4f)	No
5.4.26.118		118	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (ff-b1-4f)	No
5.4.26.119		119	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (c0-74-d7)	No
5.4.26.120		120	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (ff-74-d7)	No
5.4.26.121		121	CONFLICT_SUCCESS__R1_1_SEApp_1_DevApp_ALWAYS__R2_All_SEApp_1_DevApp_NEVER (c0-61-bf)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.122		122	CONFLICT_SUCCESS__R1_1_SEApp_1_DevApp_ALWAYS__R2_All_SEApp_1_DevApp_NEVER (ff-61-bf)	Yes
5.4.26.123		123	CONFLICT_SUCCESS__R1_1_SEApp_1_DevApp_NEVER__R2_All_SEApp_1_DevApp_ALWAYS (c0-38-79)	Yes
5.4.26.124		124	CONFLICT_SUCCESS__R1_1_SEApp_1_DevApp_NEVER__R2_All_SEApp_1_DevApp_ALWAYS (ff-38-79)	Yes
5.4.27.1	RULES_CACHED_IN_DEVICE_ACCESS_GRANTED_OR_NOT	1	RULES_CACHED_IN_DEVICE_ACCESS_GRANTED_SUCCESS__access_granted (c0-39-94)	Yes
5.4.27.2		2	RULES_CACHED_IN_DEVICE_ACCESS_GRANTED_SUCCESS__access_granted (ff-39-94)	Yes
5.4.27.3		3	RULES_CACHED_IN_DEVICE_ACCESS_GRANTED_SUCCESS__access_not_granted (c0-40-f5)	Yes
5.4.27.4		4	RULES_CACHED_IN_DEVICE_ACCESS_GRANTED_SUCCESS__access_not_granted (ff-40-f5)	Yes
5.4.27.5		5	RULES_CACHED_IN_DEVICE_FILTER_APDU_SUCCESS__filter_not_passed (c0-5a-ed)	Yes
5.4.27.6		6	RULES_CACHED_IN_DEVICE_FILTER_APDU_SUCCESS__filter_not_passed (ff-5a-ed)	Yes
5.4.27.7		7	RULES_CACHED_IN_DEVICE_FILTER_APDU_SUCCESS__filter_passed (c0-27-ed)	Yes
5.4.27.8		8	RULES_CACHED_IN_DEVICE_FILTER_APDU_SUCCESS__filter_passed (ff-27-ed)	Yes
5.4.28.1	RULE_SPECIFIC_EXCLUDES_GENERIC_ONE	1	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__R1_SEApp1_DevApp1_ALWAYS__R2_SEApp1_DevApp2_NEVER (c0-25-7c)	Yes
5.4.28.2		2	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__R1_SEApp1_DevApp1_ALWAYS__R2_SEApp1_DevApp2_NEVER (ff-61-94)	Yes
5.4.28.3		3	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__R1_SEApp1_DevApp1_NEVER__R2_SEApp1_DevApp2_ALWAYS (c0-1d-e7)	Yes
5.4.28.4		4	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__R1_SEApp1_DevApp1_NEVER__R2_SEApp1_DevApp2_ALWAYS (ff-1d-e7)	Yes
5.4.28.5		5	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__rule_SEApp1_DevApp1__no_rule_SEApp1_DevApp2 (c0-a3-3d)	Yes
5.4.28.6		6	RULES_SPECIFIC_EXCLUDE_GENERIC_SUCCESS__rule_SEApp1_DevApp1__no_rule_SEApp1_DevApp2 (ff-a3-3d)	Yes
5.4.29.1	RULES_TARGET	1	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_ALWAYS (c0-0f-f2)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.29.2		2	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_ALWAYS (ff-0f-f2)	Yes
5.4.29.3		3	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_FILTER (c0-ca-f7)	Yes
5.4.29.4		4	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_FILTER (ff-ca-f7)	Yes
5.4.29.5		5	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_NEVER__NFC_Access_NEVER (c0-c5-18)	Yes
5.4.29.6		6	RULES_TARGET_SUCCESS__1_SEApp__1_DevApp__APDU_Access_NEVER__NFC_Access_NEVER (ff-c5-18)	Yes
5.4.29.7		7	RULES_TARGET_SUCCESS__1_SEApp__2_DevApp__APDU_Access_ALWAYS (c0-9c-6f)	Yes
5.4.29.8		8	RULES_TARGET_SUCCESS__1_SEApp__2_DevApp__APDU_Access_ALWAYS (ff-9c-6f)	Yes
5.4.29.9		9	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_ALWAYS (c0-3c-d7)	Yes
5.4.29.10		10	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_ALWAYS (ff-3c-d7)	Yes
5.4.29.11		11	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_FILTER (c0-8b-13)	Yes
5.4.29.12		12	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_FILTER (ff-8b-13)	Yes
5.4.29.13		13	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_NEVER (c0-68-75)	Yes
5.4.29.14		14	RULES_TARGET_SUCCESS__1_SEApp__All_DevApp__APDU_Access_NEVER (ff-68-75)	Yes
5.4.29.15		15	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_ALWAYS (c0-64-b0)	Yes
5.4.29.16		16	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_ALWAYS (ff-64-b0)	Yes
5.4.29.17		17	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_FILTER (c0-9c-4f)	Yes
5.4.29.18		18	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_FILTER (ff-9c-4f)	Yes
5.4.29.19		19	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_NEVER (c0-88-b4)	Yes
5.4.29.20		20	RULES_TARGET_SUCCESS__All_SEApp__1_DevApp__APDU_Access_NEVER (ff-88-b4)	Yes
5.4.29.21		21	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_ALWAYS (c0-1a-31)	Yes
5.4.29.22		22	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_ALWAYS (ff-1a-31)	Yes
5.4.29.23		23	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_FILTER (c0-59-59)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.29.24		24	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_FILTER (ff-59-59)	Yes
5.4.29.25		25	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_NEVER (c0-55-81)	Yes
5.4.29.26		26	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__APDU_Access_NEVER (ff-55-81)	Yes
5.4.29.27		27	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__NFC_Access_ALWAYS (c0-b1-9a)	Yes
5.4.29.28		28	RULES_TARGET_SUCCESS__All_SEApp__All_DevApp__NFC_Access_ALWAYS (ff-b1-9a)	Yes
5.4.30.1	RULES_CACHED_IN_DEVICE_REFRESH_TAG_DEVICE_SIDE	1	RULES_UPDATED_SUCCESS__All_rules_deleted (c0-53-95)	No
5.4.30.2		2	RULES_UPDATED_SUCCESS__All_rules_deleted (ff-53-95)	No
5.4.30.3		3	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_ALWAYS__New_All_SEApp_All_DevApp_ALWAYS (c0-60-75)	No
5.4.30.4		4	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_ALWAYS__New_All_SEApp_All_DevApp_ALWAYS (ff-60-75)	No
5.4.30.5		5	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_NEVER__New_All_SEApp_DevApp2_ALWAYS (c0-d9-54)	No
5.4.30.6		6	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_NEVER__New_All_SEApp_DevApp2_ALWAYS (ff-d9-54)	No
5.4.30.7		7	RULES_UPDATED_SUCCESS__Rule_All_SEApp_DevApp1_NEVER__Delete_All_SEApp_All_DevApp (c0-df-ce)	No
5.4.30.8		8	RULES_UPDATED_SUCCESS__Rule_All_SEApp_DevApp1_NEVER__Delete_SEApp1_DevApp1_ALWAYS (c0-07-84)	No
5.4.30.9		9	RULES_UPDATED_SUCCESS__rule_equally_restrictive_added__filters_merged (c0-e5-98)	No
5.4.30.10		10	RULES_UPDATED_SUCCESS__rule_equally_restrictive_added__filters_merged (ff-e5-98)	No
5.4.30.11		11	RULES_UPDATED_SUCCESS__rule_less_restrictive_added (c0-e8-d9)	No
5.4.30.12		12	RULES_UPDATED_SUCCESS__rule_less_restrictive_added (ff-e8-d9)	No
5.4.30.13		13	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_ALWAYS_to_NEVER (c0-06-c3)	No
5.4.30.14		14	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_ALWAYS_to_NEVER (ff-06-c3)	No
5.4.30.15		15	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_NEVER_to_ALWAYS (c0-90-64)	No

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.30.16		16	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_NEVER_to_ALWAYS (ff-90-64)	No
5.4.30.17		17	RULES_UPDATED_SUCCESS__rule_more_restrictive_added (c0-90-c9)	No
5.4.30.18		18	RULES_UPDATED_SUCCESS__rule_more_restrictive_added (ff-90-c9)	No
5.4.30.19		19	RULES_UPDATED_SUCCESS__rule_with_higher_priority_added (c0-a2-fd)	No
5.4.30.20		20	RULES_UPDATED_SUCCESS__rule_with_higher_priority_added (ff-a2-fd)	No
5.4.30.21		21	RULES_UPDATED_SUCCESS__rule_with_lower_priority_added (c0-93-41)	No
5.4.30.22		22	RULES_UPDATED_SUCCESS__rule_with_lower_priority_added (ff-93-41)	No
5.4.31.1	SPECIFIC_CASE_ARF	1	SPECIFIC_CASE_ARF_ACCF_with_many_dummy_hashes (0b-a7-07)	Yes
5.4.31.2		2	SPECIFIC_CASE_ARF_ACRF_with_many_dummy_se_aids (0b-b9-07)	Yes
5.4.31.3		3	SPECIFIC_CASE_ARF_one_ACCF_with_2_hashes (0b-85-07)	Yes
5.4.31.4		4	SPECIFIC_CASE_ARF_shared_ACCF_for_2_different_rules (0b-11-07)	No
5.4.32.1	SPECIFIC_RULES_HAVE_PRIORITY	1	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_1_DevApp__R2_1_SEApp_All_DevApp (c0-3f-5e)	Yes
5.4.32.2		2	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_1_DevApp__R2_1_SEApp_All_DevApp (ff-3f-5e)	Yes
5.4.32.3		3	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_1_DevApp__R2_All_SEApp_1_DevApp (c0-e2-eb)	Yes
5.4.32.4		4	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_1_DevApp__R2_All_SEApp_1_DevApp (ff-e2-eb)	Yes
5.4.32.5		5	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_All_DevApp__R2_All_SEApp_1_DevApp (c0-9e-3b)	Yes
5.4.32.6		6	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_All_DevApp__R2_All_SEApp_1_DevApp (ff-9e-3b)	Yes
5.4.32.7		7	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_All_DevApp__R2_All_SEApp_All_DevApp (c0-db-bd)	Yes
5.4.32.8		8	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_1_SEApp_All_DevApp__R2_All_SEApp_All_DevApp (ff-db-bd)	Yes
5.4.32.9		9	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_All_SEApp_1_DevApp__R2_All_SEApp_All_DevApp (c0-9b-05)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.32.10		10	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_All_SEApp_1_DevApp__R2_All_SEApp_All_DevApp (ff-9b-05)	Yes

Annex C **Reference Tags - Real NFC Tags**

- Topaz512 43x43mm (Type 1, Broadcom) NDEF Formatted
- NTAG203F 42mm dia. (Type 2, NXP) NDEF Formatted
- FeliCa Lite 43x43mm (Type 3, Sony Corp.) NDEF Formatted
- Mifare DESFire 2k 80x50mm (Type 4, NXP) NDEF Formatted
- RF430CL330H 3K 5mmx4.4 mm (Type 4B, Texas Instruments) NDEF Formatted

Annex D Tables from the Test Book

D.1 List of test cases

TB V4.0 Test number	Test Book V4.0 Test case name	Status
3	NFC Features	
3.3	Reader / Writer mode	
3.3.3.1	NFC Forum Tag Type 1 – Read NFC Tag	
3.3.3.1.1	Test sequence No 1	
3.3.3.2	NFC Forum Tag Type 2 – Read NFC Tag	
3.3.3.2.1	Test sequence No 1	
3.3.3.3	NFC Forum Tag Type 3 – Read NFC Tag	
3.3.3.3.1	Test sequence No 1	
3.3.3.4	NFC Forum Tag Type 4 – Read NFC Tag	
3.3.3.4.1	Test sequence No 1	
3.3.3.4.2	Test sequence No 2	
3.3.3.5	NFC Forum Tag Type 1 – Write NFC Tag	
3.3.3.5.1	Test sequence No 1	
3.3.3.5.2	Test sequence No 2	
3.3.3.6	NFC Forum Tag Type 2 – Write NFC Tag	
3.3.3.6.1	Test sequence No 1	
3.3.3.6.2	Test sequence No 2	
3.3.3.7	NFC Forum Tag Type 3 – Write NFC Tag	
3.3.3.7.1	Test sequence No 1	
3.3.3.8	NFC Forum Tag Type 4 – Write NFC Tag	
3.3.3.8.1	Test sequence No 1	
3.3.3.8.2	Test sequence No 2	
3.3.3.9	Distance for NFC Tag Type 1 reading	
3.3.3.9.1	Test sequence No 1: Distance for NFC Tag Type 1 Reading - 0,0cm	
3.3.3.9.2	Test sequence No 2: Distance for NFC Tag Type 1 Reading - 0,5cm	
3.3.3.9.3	Test sequence No 3: Distance for NFC Tag Type 1 Reading - 1,0cm	
3.3.3.9.4	Test sequence No 4: Distance for NFC Tag Type 1 Reading - 2,0cm	
3.3.3.9.5	Test sequence No 5: Distance for NFC Tag Type 1 Reading - 3,0cm	
3.3.3.9.6	Test sequence No 6: Distance for NFC Tag Type 1 Reading - 4,0cm	New
3.3.3.10	Distance for NFC Tag Type 2 reading	
3.3.3.10.1	Test sequence No 1: Distance for NFC Tag Type 2 Reading - 0,0cm	
3.3.3.10.2	Test sequence No 2: Distance for NFC Tag Type 2 Reading - 0,5cm	
3.3.3.10.3	Test sequence No 3: Distance for NFC Tag Type 2 Reading - 1,0cm	
3.3.3.10.4	Test sequence No 4: Distance for NFC Tag Type 2 Reading - 2,0cm	
3.3.3.10.5	Test sequence No 5: Distance for NFC Tag Type 2 Reading - 3,0cm	
3.3.3.10.6	Test sequence No 6: Distance for NFC Tag Type 2 Reading - 4,0cm	New
3.3.3.11	Distance for NFC Tag Type 3 reading	
3.3.3.11.1	Test sequence No 1: Distance for NFC Tag Type 3 Reading - 0,0cm	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
3.3.3.11.2	Test sequence No 2: Distance for NFC Tag Type 3 Reading - 0,5cm	
3.3.3.11.3	Test sequence No 3: Distance for NFC Tag Type 3 Reading - 1,0cm	
3.3.3.11.4	Test sequence No 4: Distance for NFC Tag Type 3 Reading - 2,0cm	
3.3.3.11.5	Test sequence No 5: Distance for NFC Tag Type 3 Reading - 3,0cm	
3.3.3.11.6	Test sequence No 6: Distance for NFC Tag Type 3 Reading - 4,0cm	New
3.3.3.12	Distance for NFC Tag Type 4A reading	
3.3.3.12.1	Test sequence No 1: Distance for NFC Tag Type 4A Reading - 0,0cm	
3.3.3.12.2	Test sequence No 2: Distance for NFC Tag Type 4A Reading - 0,5cm	
3.3.3.12.3	Test sequence No 3: Distance for NFC Tag Type 4A Reading - 1,0cm	
3.3.3.12.4	Test sequence No 4: Distance for NFC Tag Type 4A Reading - 2,0cm	
3.3.3.12.5	Test sequence No 5: Distance for NFC Tag Type 4A Reading – 3,0cm	
3.3.3.12.6	Test sequence No 6: Distance for NFC Tag Type 4A Reading - 4,0cm	New
3.3.3.13	Distance for NFC Tag Type 4B reading	
3.3.3.13.1	Test sequence No 1: Distance for NFC Tag Type 4B Reading - 0,0cm	
3.3.3.13.2	Test sequence No 2: Distance for NFC Tag Type 4B Reading - 0,5cm	
3.3.3.13.3	Test sequence No 3: Distance for NFC Tag Type 4B Reading - 1,0cm	
3.3.3.13.4	Test sequence No 4: Distance for NFC Tag Type 4B Reading - 2,0cm	
3.3.3.13.5	Test sequence No 5: Distance for NFC Tag Type 4B Reading - 3,0cm	
3.3.3.13.6	Test sequence No 6: Distance for NFC Tag Type 4B Reading - 4,0cm	New
3.3.3.14	NFC Tag type 1 reading performance	
3.3.3.14.1	Test sequence No 1	
3.3.3.15	NFC Tag type 2 reading performance	
3.3.3.15.1	Test sequence No 1	
3.3.3.16	NFC Tag type 3 reading performance	
3.3.3.16.1	Test sequence No 1	
3.3.3.17	NFC Tag type 4A reading performance	
3.3.3.17.1	Test sequence No 1	
3.3.3.18	NFC Tag type 4B reading performance	
3.3.3.18.1	Test sequence No 1	
3.3.3.19	NFC Tag handling during an active data transfer	
3.3.3.19.1	Test sequence No 1	
3.3.3.20	Reader mode via UICC	FFS
3.3.3.21	Reader mode via Application Processor	FFS
3.3.3.22	Reader mode events routing	FFS
3.3.3.23	NFC Forum RTD signature	FFS
3.4	Card emulation mode	
3.4.3.1	Card Emulation enabled as soon as NFC hardware is on	
3.4.3.1.1	Test sequence No 1	
3.4.3.1.2	Test sequence No 2	
3.4.3.2	NFC during Standby time	
3.4.3.2.1	Test sequence No 1	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
3.4.3.3	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes	
3.4.3.3.1	Test sequence No 1: Card Emulation Mode Type A in Battery Power Off mode	
3.4.3.3.2	Test sequence No 2: Card Emulation Mode Type B in Battery Power Off mode	
3.4.3.3.3	Test sequence No 3:CLT (A) in Battery Power Off mode	FFS
3.4.3.3.4	Test sequence No 4: Card Emulation Mode Type A in Battery Low Mode	
3.4.3.3.5	Test sequence No 5: Card Emulation Mode Type B in Battery Low Mode	
3.4.3.3.6	Test sequence No 6: CLT (A) in Battery Low Mode	FFS
3.4.3.4	Distance for card emulation	
3.4.3.5	Distance for card emulation in Battery Power-off Mode(0cm)	
3.4.3.5.1	Test sequence No 1	
3.4.3.6	Distance for card emulation in Battery Power-off Mode (0.5cm)	
3.4.3.6.1	Test sequence No 1	
3.4.3.7	Distance for card emulation in Battery Power-off Mode (1cm)	
3.4.3.7.1	Test sequence No 1	
3.4.3.8	Distance for card emulation in Battery Power-off Mode (1.5cm)	
3.4.3.8.1	Test sequence No 1	
3.4.3.9	Distance for card emulation in Battery Power-off Mode (2cm)	
3.4.3.9.1	Test sequence No 1	
3.4.3.10	Distance for card emulation in Battery Power-low Mode (0cm)	
3.4.3.10.1	Test sequence No 1	New
3.4.3.11	Distance for card emulation in Battery Power-low Mode (0,5cm)	
3.4.3.11.1	Test sequence No 1	New
3.4.3.12	Distance for card emulation in Battery Power-low Mode (1cm)	
3.4.3.12.1	Test sequence No 1	New
3.4.3.13	Distance for card emulation in Battery Power-low Mode (1,5cm)	
3.4.3.13.1	Test sequence No 1	New
3.4.3.14	Distance for card emulation in Battery Power-low Mode (2cm)	
3.4.3.14.1	Test sequence No 1	New
3.4.3.15	Distance for card emulation in Battery Power-operational Mode (0cm)	
3.4.3.15.1	Test sequence No 1	New
3.4.3.16	Distance for card emulation in Battery Power- operational Mode (0,5cm)	
3.4.3.16.1	Test sequence No 1	New
3.4.3.17	Distance for card emulation in Battery Power- operational Mode (1cm)	
3.4.3.17.1	Test sequence No 1	New
3.4.3.18	Distance for card emulation in Battery Power- operational Mode (1,5cm)	
3.4.3.18.1	Test sequence No 1	New

TB V4.0 Test number	Test Book V4.0 Test case name	Status
3.4.3.19	Distance for card emulation in Battery Power- operational Mode (2cm)	
3.4.3.19.1	Test sequence No 1	New
3.5	Core and common features	
3.5.3.1	SWP Compliance testing	
3.5.3.2	HCI Compliance testing	
3.5.3.3	SWP Stress test	
3.5.3.3.1	Test sequence No 1	
3.5.3.4	Switch mode	
3.5.3.4.1	Test sequence No 1	
3.5.3.5	RF Protocol compliance	
3.5.3.6	VOID	
4	VOID	
5	Secure Element Access Control	
5.3.1	GP SE Access Control	
5.3.1.1	Test sequence No 1	
5.3.1.2	Test sequence No 2	
5.3.1.3	Test sequence No 3	
5.3.1.4	Test sequence No 4	
5.3.1.5	Test sequence No 5	
5.3.1.6	Test sequence No 6	
5.3.1.7	Test sequence No 7	
5.3.1.8	Test sequence No 8	
5.3.1.9	Test sequence No 9	
5.3.2	GP SE Access Control - Refresh tag	
5.3.2.1	Test sequence No 1	
5.3.2.2	Test sequence No 2	
5.3.3	GP SE Access Control – ADF_PKCS#15 and DF PKCS#15	
5.3.3.1	Test sequence No 1	
5.3.4	GP SE Access Control – PKCS#15 selection via EF_DIR	
5.3.4.1	Test sequence No 1	
5.3.5	GP SE Access Control – Configuration limits	
5.3.5.1	Test sequence No 1	
5.3.5.2	Test sequence No 2	
5.3.6	GP SE Access Control – No access	
5.3.6.1	Test sequence No 1	
5.3.6.2	Test sequence No 2	
5.3.6.3	Test sequence No 3	
5.3.6.4	Test sequence No 4	
5.3.6.5	Test sequence No 5	
5.4	GP SE Access Control – GP Test Plan	New
6	Secure Element Access API	
6.3.1	SIMalliance APIs	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
6.3.2	Prevent access to basic channel	
6.3.3	VOID	
6.3.4	VOID	
6.3.5	VOID	
7	Multiple Card Emulation Environment	
7.3.1	SE Availabilities from mobile applications	FFS
7.3.2	SE selection	
7.3.2.1	Test sequence No 1	
7.3.3	UICC as default SE	FFS
7.3.4	SE Selection API	FFS
7.3.5	SE Toggle switching API	FFS
7.3.6	Switch for Card Emulation Mode	
7.3.6.1	Test sequence No 1	New
7.3.7	Multiple CE Environments	
7.3.7.1	Test sequence No 1	New
7.3.7.2	Test sequence No 2	New
7.3.7.3	Test sequence No 3	New
7.3.7.4	Test sequence No 4	New
7.3.7.5	Test sequence No 5	New
7.3.7.6	Test sequence No 6	New
7.3.7.7	Test sequence No 7	New
7.3.7.8	Test sequence No 8	New
7.3.7.9	Test sequence No 9	New
7.3.7.10	Test sequence No 10	New
7.3.8	Active Card Emulation in Multiple CE Environments / Card Emulation	
7.3.8.1	Test sequence No 1	New
7.3.8.2	Test sequence No 2	New
7.3.8.3	Test sequence No 3	New
7.3.8.4	Test sequence No 4	New
7.3.8.5	Test sequence No 5	New
7.3.8.6	Test sequence No 6	New
8	UI Application triggering	
8.3.1	EVT_TRANSACTION	
8.3.2	Permissions	
8.3.2.1	Test sequence No 1	
8.3.2.2	Test sequence No 2	
8.3.2.3	Test sequence No 3	
8.3.3	Intent management	
8.3.3.1	Test sequence No 1	
8.3.3.2	Test sequence No 2	
8.3.4	Application’s triggering order	
8.3.4.1	Test sequence No 1	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
8.3.4.2	Test sequence No 2	
8.3.4.3	Test sequence No 3	
8.3.5	Triggering on HCI event EVT_CARD_DEACTIVATED	
8.3.5.1	Test sequence No 1	
8.3.6	Triggering on HCI event EVT_FIELD_OFF	
8.3.6.1	Test sequence No 1	
9	VOID	
10	Void	
10.3.1	void	
11	Mobile Device APN management	
11.3.1	OPEN CHANNEL	
11.3.1.1	Test Sequence No 1: (OPEN CHANNEL - Default APN Always-ON - Multiple APN supported - with different APN)	
11.3.1.2	Test Sequence No 2: (OPEN CHANNEL - Default APN Always-ON - Only Single APN supported - with different APN)	
11.3.1.3	Test Sequence No 3: (OPEN CHANNEL - Default APN Always-ON - APN empty)	
11.3.1.4	Test Sequence No 4: (OPEN CHANNEL – Default APN Always-ON – APN empty – Default Bearer Type used)	New
11.3.2	CLOSE CHANNEL	
11.3.2.1	Test Sequence No 1: (CLOSE CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN- SUCCESSFUL)	
11.3.2.2	Test Sequence No 2: (CLOSE CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN- SUCCESSFUL)	
11.3.2.3	Test Sequence No 3: (CLOSE CHANNEL – Default APN Always-ON – APN empty- SUCCESSFUL)	
11.3.2.4	Test Sequence No 4: (CLOSE CHANNEL – Default APN Always-ON – APN empty - SUCCESSFUL– Default Bearer Type used)	New
11.3.3	RECEIVE DATA	
11.3.3.1	Test Sequence No 1: (RECEIVE DATA – Default APN Always-ON – Multiple APN supported – with different APN)	
11.3.3.2	Test Sequence No 2: (RECEIVE DATA – Default APN Always-ON – Only Single APN supported – with different APN)	
11.3.3.3	Test Sequence No 3: (RECEIVE DATA – Default APN Always-ON – APN empty)	
11.3.3.4	Test Sequence No 4: (RECEIVE DATA – Default APN Always-ON – APN empty – Default Bearer Type used)	New
11.3.4	SEND DATA	
11.3.4.1	Test Sequence No 1: (SEND DATA - Default APN Always-ON - Multiple APN supported -with different APN- BUFFER FULLY USED)	
11.3.4.2	Test Sequence No 2: (SEND DATA - Default APN Always-ON - Only Single APN supported - with different APN- BUFFER FULLY USED)	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
11.3.4.3	Test Sequence No 3: (SEND DATA - Default APN Always-ON - APN empty- BUFFER FULLY USED)	
11.3.4.4	Test Sequence No 4: (SEND DATA – Default APN Always-ON – APN empty BUFFER FULLY USED – Default Bearer Type used)	New
12	Remote Management of NFC Services	
12.3	Basic remote Management	
12.3.3.1	Remote management in BIP	
12.3.3.2	OPEN CHANNEL	
12.3.3.2.1	Test Sequence No 1: (OPEN CHANNEL, No APN, immediate link establishment, Default Bearer for requested transport layer, No local address, no alpha identifier)	
12.3.3.2.2	Test sequence No 2: (OPEN CHANNEL, with APN, immediate link establishment, Default Bearer for requested transport layer, no alpha identifier	
12.3.3.2.3	Test Sequence No 3: (OPEN CHANNEL, with alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	
12.3.3.2.4	Test Sequence No 4: (OPEN CHANNEL, with null alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	
12.3.3.2.5	Test Sequence No 5: (OPEN CHANNEL, command performed with modifications (buffer size), immediate link establishment, Default Bearer for requested transport layer)	
12.3.3.2.6	Test Sequence No 6A: (OPEN CHANNEL, user rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier,)	
12.3.3.2.7	Test Sequence No 6B: (OPEN CHANNEL, User rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier)	
12.3.3.3	CLOSE CHANNEL	
12.3.3.3.1	Test Sequence No 1: (CLOSE CHANNEL, successful)	
12.3.3.3.2	Test Sequence No 2: (CLOSE CHANNEL, with an invalid channel identifier) Initial Conditions	
12.3.3.3.3	Test Sequence No 3: (CLOSE CHANNEL, on an already closed channel) Initial Conditions	
12.3.3.4	RECEIVE DATA	
12.3.3.4.1	Test Sequence No 1: (RECEIVE DATA)	
12.3.3.5	SEND DATA	
12.3.3.5.1	Test sequence No 1 (SEND DATA, immediate mode)	
12.3.3.5.2	Test sequence No 2 (SEND DATA, Store mode)	
12.3.3.5.3	Test Sequence No 3: (SEND DATA, Tx buffer fully used , Store mode)	
12.3.3.5.4	Test Sequence No 4: (SEND DATA, 2 consecutive SEND DATA Store mode)	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
12.3.3.5.5	Test Sequence No 5: (SEND DATA, immediate mode with a bad channel identifier)	
12.3.3.6	GET CHANNEL STATUS	
12.3.3.6.1	Test Sequence No 1: (GET STATUS, without any BIP channel opened)	
12.3.3.6.2	Test Sequence No 2: (GET STATUS, with a BIP channel currently opened)	
12.3.3.6.3	Test sequence No 3: (GET STATUS, after a link dropped)	
12.3.3.7	Data available event	
12.3.3.7.1	Test Sequence No 1: (EVENT DOWNLOAD - Data available)	
12.3.3.8	Channel Status event	
12.3.3.8.1	Test Sequence No 1: (EVENT DOWNLOAD - Channel Status on a link dropped)	
12.3.3.9	SMS-PP Data Download	
12.3.3.9.1	Test Sequence No 1: (SMS-PP - followed by Open channel - Send/Receive data)	FFS
12.3.3.9.2	Test Sequence No 2: (SMS-PP - Send SM -followed by Open channel - Send/Receive data)	FFS
12.3.3.9.3	Test Sequence No 3: (SMS-PP - Send SM -followed by Open channel - Send/Receive data with timer management)	FFS
12.3.3.10	Concurrent BIP channels	
12.3.3.10.1	Test Sequence No 1	
12.3.3.11	Contents of the TERMINAL PROFILE	
12.3.3.11.1	Test Sequence No 1: (TERMINAL PROFILE – command for BIP in UDP, client mode)	New
12.4	Remote Management use cases	
12.4.3.1	Contactless transaction during BIP session	
12.4.3.1.1	Test Sequence No 1 – Receiving and send a SMS during BIP data transfer	
12.4.3.2	OTA Data Loading during a phone call	
12.4.3.2.1	Test Sequence No 1 : <Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network with 2G/3G fallback >	
12.4.3.2.2	Test Sequence No 2 <Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network only>	
12.4.3.2.3	Test Sequence No 3: < Voice Call made from the device during BIP CAT-TP session on LTE network with 2G/3G fallback >	
12.4.3.2.4	Test Sequence No 4: < Voice Call made from the device during BIP CAT-TP session on LTE network only >	
12.4.3.2.5	Test Sequence No 5 < BIP CAT-TP data transfer during a Voice Call is established on LTE network with 2G/3G fallback >	
12.4.3.2.6	Test Sequence No 6 < BIP CAT-TP data transfer during a Voice Call is established on LTE network only >	
12.4.3.3	OTA Data Loading with and withour proof of Receipt (PoR)	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
12.4.3.3.1	Test Sequence No 1 : OTA data Loading without PoR requested by OTA server	New
12.4.3.3.2	Test Sequence No 2 : OTA data Loading with PoR requested by OTA server	New
12.4.3.3.3	Test Sequence No 3: OTA data Loading with PoR requested by OTA server when the device is booting	New
12.4.3.3.4	Test Sequence No 4: OTA data Loading with PoR requested by OTA server after the device comes back from airplane mode	New
12.4.3.3.5	Test Sequence No 5 OTA data Loading with PoR requested by OTA server only on error	New
12.4.3.4	Secure Element Access during BIP session	
12.4.3.4.1	Test Sequence No 1	New
12.4.3.5	SMS and Internet Connection during OTA data Loading	
12.4.3.5.1	Test Sequence No 1: Receiving and send a SMS during BIP data transfer	New
13	General Device Support	
13.3.1	Secure Element Access API and Access control in Radio OFF State	
13.3.1.1	Test sequence No 1	
13.3.1.2	Test sequence No 2	
13.3.1.3	Test sequence No 3	
13.3.2	Enabled / Disabled states	
13.3.2.1	Test sequence No 1	
15	Android Operating System	
15.3.3.1	Shared library	
15.3.3.1.1	Test sequence No 1	New
15.3.3.2	Getting an instance of the NFC controller class	
15.3.3.2.1	Test sequence No 1	New
15.3.3.2.2	Test sequence No 2	New
15.3.3.3	Getting the status of the NFC Controller	
15.3.3.3.1	Test sequence No 1	New
15.3.3.3.2	Test sequence No 2	New
15.3.3.4	Enabling the NFC Controller	
15.3.3.4.1	Test sequence No 1	New
15.3.3.4.2	Test sequence No 2	New
15.3.3.4.3	Test sequence No 3	New
15.5.3.1	Switching to Multicast mode	
15.5.3.1.1	Test sequence No 1	New
15.5.3.1.2	Test sequence No 2	New
15.7.3.1	Dynamic AIDs Registration APIs when Mulitple Active CEEs model is supported	

TB V4.0 Test number	Test Book V4.0 Test case name	Status
15.7.3.1.1	Test sequence No 1	New
15.7.3.1.2	Test sequence No 2	New
15.7.3.1.3	Test sequence No 3	New
15.7.3.1.4	Test sequence No 4	New
15.7.3.2	API for handling Dynamic AID Registration	
15.7.3.2.1	Test sequence No 1	New
15.7.3.2.2	Test sequence No 2	New

Table 27: List of test cases

D.2 Table of optional features

Item	Option	Status	Support	Mnemonic
1	void			
2	Support of LTE/IMS	O		O_LTE/IMS
3	Support of LTE with fallback to 2G/3G	O		O_LTE/2G-3G
4	Support of read/write NFC Tag at distance > 1,0cm and ≤ 2,0cm	O		O_TAG_DISTANCE 2CM
5	Support of read/write NFC Tag at distance > 2,0cm and ≤ 3,0cm, see note 5	O		O_TAG_DISTANCE_3CM
6	Support of read/write NFC Tag at distance > 3,0cm and ≤ 4,0cm, see note 4	O		O_TAG_DISTANCE_4CM
7	Support of battery low mode, see note 2	O		O_BAT_LOW
8	Support of battery off mode, see notes 2 and 3	O		O_BAT_OFF
9	Support of multiple SE	O		O_MUL_SE
10	Support of Multiple APN	O		O_MULTI_APN
11	Terminal executes User confirmation phase before sending PDP context activation request	O		O_User_Confirm_Before_PDP_Context_Request
12	Support of Multiple Active CEEs model in Battery Operational Mode	O		O_MULTI_CEE_ON
13	Single Active Card Emulation Environment is supported	O		O_Single_Active_CEE
14	ISO_DEP route selection mechanism is provided by the DUT	O		O_ISO_DEP_ROUTE_SELECTION _
15	The NFC status is persistent across DUT power off and power on	O		O_NFC _PERSISTENCE

Note 1: In order to reflect current industry implementation, test cases with read/write distance > 1cm are optional for this version

Note 2: For options O_BAT_LOW and O_BAT_OFF the DUT shall support at least one of these options or both.

Note 3: If the device supports O_BAT_OFF, the device manufacturer must supply details of how to enter this state

Note 4: If option O_TAG_DISTANCE_4CM is supported, then O_TAG_DISTANCE_2CM and O_TAG_DISTANCE_3CM must be supported.

Note 5: If option O_TAG_DISTANCE_3CM is supported, then O_TAG_DISTANCE_2CM must be supported

Table 4: Options

D.3 Applicability table

Test case	Description	Test case applicability			Support
		Android	Windows	BlackBerry	
3.3.3.1	NFC Forum Tag Type 1 – Read NFC Tag	M	M	M	
3.3.3.2	NFC Forum Tag Type 2 – Read NFC Tag	M	M	M	
3.3.3.3	NFC Forum Tag Type 3 – Read NFC Tag	M	M	M	
3.3.3.4	NFC Forum Tag Type 4 – Read NFC Tag	M	M	M	
3.3.3.5	NFC Forum Tag Type 1 – Write NFC Tag	M	M	M	
3.3.3.6	NFC Forum Tag Type 2 – Write NFC Tag	M	M	M	
3.3.3.7	NFC Forum Tag Type 3 – Write NFC Tag	M	M	M	
3.3.3.8	NFC Forum Tag Type 4 – Write NFC Tag	M	M	M	
3.3.3.9.1	Distance for NFC Tag Type 1 reading Test Sequence No 1: Distance for NFC Tag Type 1 Reading – 0,0cm	M	M	M	
3.3.3.9.2	Distance for NFC Tag Type 1 reading Test Sequence No 2: Distance for NFC Tag Type 1 Reading – 0,5cm	M	M	M	
3.3.3.9.3	Distance for NFC Tag Type 1 reading Test Sequence No 3: Distance for NFC Tag Type 1 reading – 1,0cm	M	M	M	
3.3.3.9.4	Distance for NFC Tag Type 1 reading Test Sequence No 4: Distance for NFC Tag Type 1 Reading – 2,0cm	C015	C015	C015	
3.3.3.9.5	Distance for NFC Tag Type 1 reading Test Sequence No 5: Distance for NFC Tag Type 1 Reading – 3,0cm	C016	C016	C016	

Test case	Description	Test case applicability			Support
3.3.3.9.6	Distance for NFC Tag Type 1 reading Test Sequence No 6: Distance for NFC Tag Type 1 Reading – 4,0cm	C017	C017	C017	
3.3.3.10.1	Distance for NFC Tag Type 2 reading Test Sequence No 1: Distance for NFC Tag Type 2 Reading – 0,0cm	M	M	M	
3.3.3.10.2	Distance for NFC Tag Type 2 reading Test Sequence No 2: Distance for NFC Tag Type 2 Reading – 0,5cm	M	M	M	
3.3.3.10.3	Distance for NFC Tag Type 2 reading Test Sequence No 3: Distance for NFC Tag Type 2 reading – 1,0cm	M	M	M	
3.3.3.10.4	Distance for NFC Tag Type 2 reading Test Sequence No 4: Distance for NFC Tag Type 2 Reading – 2,0cm	C015	C015	C015	
3.3.3.10.5	Distance for NFC Tag Type 2 reading Test Sequence No 5: Distance for NFC Tag Type 2 Reading – 3,0cm	C016	C016	C016	
3.3.3.10.6	Distance for NFC Tag Type 2 reading Test Sequence No 6: Distance for NFC Tag Type 2 Reading – 4,0cm	C017	C017	C017	
3.3.3.11.1	Distance for NFC Tag Type 3 reading Test Sequence No 1: Distance for NFC Tag Type 3 Reading – 0,0cm	M	M	M	
3.3.3.11.2	Distance for NFC Tag Type 3 reading Test Sequence No 2: Distance for NFC Tag Type 3 Reading – 0,5cm	M	M	M	
3.3.3.11.3	Distance for NFC Tag Type 3 reading Test Sequence No 3: Distance for NFC Tag Type 3 reading – 1,0cm	M	M	M	
3.3.3.11.4	Distance for NFC Tag Type 3 reading Test Sequence No 4: Distance for NFC Tag Type 3 Reading – 2,0cm	C015	C015	C015	
3.3.3.11.5	Distance for NFC Tag Type 3 reading Test Sequence No 5: Distance for NFC Tag Type 3 Reading – 3,0cm	C016	C016	C016	
3.3.3.11.6	Distance for NFC Tag Type 3 reading Test Sequence No 6: Distance for NFC Tag Type 3 Reading – 4,0cm	C017	C017	C017	

Test case	Description	Test case applicability			Support
3.3.3.12.1	Distance for NFC Tag Type 4A reading Test Sequence No 1: Distance for NFC Tag Type 4A Reading – 0,0cm	M	M	M	
3.3.3.12.2	Distance for NFC Tag Type 4A reading Test Sequence No 2: Distance for NFC Tag Type 4A Reading – 0,5cm	M	M	M	
3.3.3.12.3	Distance for NFC Tag Type 4A reading Test Sequence No 3: Distance for NFC Tag Type 4A reading – 1,0cm	M	M	M	
3.3.3.12.4	Distance for NFC Tag Type 4A reading Test Sequence No 4: Distance for NFC Tag Type 4A Reading – 2,0cm	C015	C015	C015	
3.3.3.12.5	Distance for NFC Tag Type 4A reading Test Sequence No 5: Distance for NFC Tag Type 4A Reading – 3,0cm	C016	C016	C016	
3.3.3.12.6	Distance for NFC Tag Type 4A reading Test Sequence No 6: Distance for NFC Tag Type 4A Reading – 4,0cm	C017	C017	C017	
3.3.3.13.1	Distance for NFC Tag Type 4B reading Test Sequence No 1: Distance for NFC Tag Type 4B Reading – 0,0cm	M	M	M	
3.3.3.13.2	Distance for NFC Tag Type 4B reading Test Sequence No 2: Distance for NFC Tag Type 4B Reading – 0,5cm	M	M	M	
3.3.3.13.3	Distance for NFC Tag Type 4B reading Test Sequence No 3: Distance for NFC Tag Type 4B reading – 1,0cm	M	M	M	
3.3.3.13.4	Distance for NFC Tag Type 4B reading Test Sequence No 4: Distance for NFC Tag Type 4B Reading – 2,0cm	C015	C015	C015	

Test case	Description	Test case applicability			Support
3.3.3.13.5	Distance for NFC Tag Type 4B reading Test Sequence No 5: Distance for NFC Tag Type 4B Reading – 3,0cm	C016	C016	C016	
3.3.3.13.6	Distance for NFC Tag Type 4B reading Test Sequence No 5: Distance for NFC Tag Type 4B Reading – 4,0cm	C017	C017	C017	
3.3.3.14	NFC Tag type 1 reading performance	M	M	M	
3.3.3.15	NFC Tag type 2 reading performance	M	M	M	
3.3.3.16	NFC Tag type 3 reading performance	M	M	M	
3.3.3.17	NFC Tag type 4A reading performance	M	M	M	
3.3.3.18	NFC Tag type 4B reading performance	M	M	M	
3.3.3.19	NFC Tag handling during an active data transfer.	M	M	M	
3.4.3.1.1	Card Emulation enabled as soon as NFC hardware is on Test Sequence No.1	M	M	M	
3.4.3.1.2	Card Emulation enabled as soon as NFC hardware is on Test sequence No 2	C014	C014	C014	
3.4.3.2	NFC during Standby time	C005	C005	C005	
3.4.3.3.1	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 1: Card Emulation Mode Type A in Battery Power Off mode	C006	C006	C006	

Test case	Description	Test case applicability			Support
3.4.3.3.2	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 2: Card Emulation Mode Type B in Battery Power Off mode	C006	C006	C006	
3.4.3.3.4	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 4: Card Emulation Mode Type A in Battery Low Mode	C005	C005	C005	
3.4.3.3.5	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction either in Battery Power Off or Battery Low modes Test sequence No 5: Card Emulation Mode Type B in Battery Low Mode	C005	C005	C005	
3.4.3.4	Distance for card emulation	M	M	M	
3.4.3.5	Distance for card emulation in Battery Power-off Mode(0cm)	C006	C006	C006	
3.4.3.6	Distance for card emulation in Battery Power-off Mode (0.5cm)	C006	C006	C006	
3.4.3.7	Distance for card emulation in Battery Power-off Mode (1cm)	C006	C006	C006	
3.4.3.8	Distance for card emulation in Battery Power-off Mode (1.5cm)	C006	C006	C006	
3.4.3.9	Distance for card emulation in Battery Power-off Mode (2cm)	C006	C006	C006	
3.4.3.10	Distance for card emulation in Battery Power-low Mode(0cm)	C005	C005	C005	
3.4.3.11	Distance for card emulation in Battery Power-low Mode (0.5cm)	C005	C005	C005	
3.4.3.12	Distance for card emulation in Battery Power-low Mode (1cm)	C005	C005	C005	

Test case	Description	Test case applicability			Support
3.4.3.13	Distance for card emulation in Battery Power-low Mode (1.5cm)	C005	C005	C005	
3.4.3.14	Distance for card emulation in Battery Power-low Mode (2cm)	C005	C005	C005	
3.4.3.15	Distance for card emulation in Battery Power- operational Mode(0cm)	M	M	M	
3.4.3.16	Distance for card emulation in Battery Power- operational Mode (0.5cm)	M	M	M	
3.4.3.17	Distance for card emulation in Battery Power- operational Mode (1cm)	M	M	M	
3.4.3.18	Distance for card emulation in Battery Power- operational Mode (1.5cm)	M	M	M	
3.4.3.19	Distance for card emulation in Battery Power- operational Mode (2cm)	M	M	M	
3.5.3.1	SWP Compliance testing	M	M	M	
3.5.3.2	HCI Compliance testing	M	M	M	
3.5.3.3	SWP Stress test	M	M	M	
3.5.3.4	Switch mode	M	M	M	
3.5.3.5	RF Protocol compliance	M	M	M	
5.3.1	GP SE Access Control	M	TNR	TNR	
5.3.2	GP SE Access Control–- Refresh tag	M	TNR	TNR	
5.3.3	GP SE Access Control – ADF_PKCS#15 and DF PKCS#15	M	TNR	TNR	
5.3.4	GP SE Access Control – PKCS#15 selection via EF_DIR	M	TNR	TNR	
5.3.5	GP SE Access Control – Configuration limits	M	TNR	TNR	
5.3.6	GP SE Access Control – No access	M	TNR	TNR	
5.4	GP SE Access Control – GP Test Plan	M	M	M	
6.3.1	SIMalliance APIs	M	N/A	TNR	
7.3.2	SE selection	C007	C007	C007	
7.3.6	Switch for Card Emulation Mode	C012	TNR	TNR	
7.3.7	Multiple CE Environments	C019	N/A	N/A	

Test case	Description	Test case applicability			Support
7.3.8.1	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 1	C018	C018	C018	
7.3.8.2	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 2	C018	C018	C018	
7.3.8.3	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 3	C019	C019	C019	
7.3.8.4	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 4	C019	C019	C019	
7.3.8.5	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 5	C019	C019	C019	
7.3.8.6	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 6	C019	C019	C019	
8.3.1	EVT_TRANSACTION	M	M	M	
8.3.2	Permissions	M	N/A	N/A	
8.3.3	Intent management	M	N/A	N/A	
8.3.4	Application's triggering order	TNR	TNR	TNR	
8.3.5	Triggering on HCI event EVT_CARD_DEACTIVATED	M	TNR	TNR	
8.3.6	Triggering on HCI event EVT_FIELD_OFF	M	TNR	TNR	
11.3.1.1	OPEN CHANNEL Test Sequence 1: (OPEN CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN)	C008	C008	C008	
11.3.1.2	OPEN CHANNEL 11.3.1.2 Test Sequence 2: (OPEN CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN)	C009	C009	C009	
11.3.1.3	OPEN CHANNEL Test Sequence 3: (OPEN CHANNEL – Default APN Always-ON – APN empty)	M	M	M	
11.3.1.4	OPEN CHANNEL Test Sequence No 4: (OPEN CHANNEL – Default APN Always-ON – APN empty- Default Bearer Type used)	M	M	M	

Test case	Description	Test case applicability			Support
11.3.2.1	CLOSE CHANNEL Test Sequence 1: (CLOSE CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN- SUCCESSFUL)	C008	C008	C008	
11.3.2.2	CLOSE CHANNEL Test Sequence 2: (CLOSE CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN- SUCCESSFUL)	C009	C009	C009	
11.3.2.3	CLOSE CHANNEL Test Sequence 3: (CLOSE CHANNEL – Default APN Always-ON – APN empty- SUCCESSFUL)	M	M	M	
11.3.2.4	CLOSE CHANNEL Test Sequence No 4: (CLOSE CHANNEL – Default APN Always-ON – APN empty- SUCCESSFUL- Default Bearer Type Used)	M	M	M	
11.3.3.1	RECEIVE DATA Test Sequence 1: (RECEIVE DATA – Default APN Always-ON – Multiple APN supported – with different APN)	C008	C008	C008	
11.3.3.2	RECEIVE DATA Test Sequence 2: (RECEIVE DATA – Default APN Always-ON – Only Single APN supported – with different APN)	C009	C009	C009	
11.3.3.3	RECEIVE DATA Test Sequence 3: (RECEIVE DATA – Default APN Always-ON – APN empty)	M	M	M	
11.3.3.4	RECEIVE DATA Test Sequence No 4: (RECEIVE DATA –Default APN Always-ON – APN empty-Default Bearer Type used)	M	M	M	
11.3.4.1	SEND DATA Test Sequence 1: (SEND DATA – Default APN Always-ON – Multiple APN supported –with different APN- BUFFER FULLY USED)	C008	C008	C008	
11.3.4.2	SEND DATA Test Sequence 2: (SEND DATA – Default APN Always-ON – Only Single APN supported – with different APN- BUFFER FULLY USED)	C009	C009	C009	

Test case	Description	Test case applicability			Support
11.3.4.3	SEND DATA Test Sequence 3: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED)	M	M	M	
11.3.4.4	SEND DATA Test Sequence No 4: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED- Default Bearer Type used)	M	M	M	
12.3.3.1	Remote management in BIP	M	M	M	
12.3.3.2.1	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, No APN, immediate link establishment, Default Bearer for requested transport layer, No local address, no alpha identifier)	M	M	M	
12.3.3.2.2	OPEN CHANNEL Test sequence No : (OPEN CHANNEL, with APN, immediate link establishment, Default Bearer for requested transport layer, no alpha identifier)	M	M	M	
12.3.3.2.3	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, with alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.4	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, with null alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.5	OPEN CHANNEL Test Sequence No : (OPEN CHANNEL, command performed with modifications (buffer size), immediate link establishment, Default Bearer for requested transport layer)	M	M	M	
12.3.3.2.6	OPEN CHANNEL Test Sequence No 6A : (OPEN CHANNEL, user rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier,)	C010	C010	C010	

Test case	Description	Test case applicability			Support
12.3.3.2.7	OPEN CHANNEL Test Sequence No 6B : (OPEN CHANNEL, User rejection, immediate link establishment, Default Bearer for requested transport layer, open command with alpha identifier)	C011	C011	C011	
12.3.3.3	CLOSE CHANNEL	M	M	M	
12.3.3.4	RECEIVE DATA	M	M	M	
12.3.3.5	SEND DATA	M	M	M	
12.3.3.6	GET CHANNEL STATUS	M	M	M	
12.3.3.7	Data available event	M	M	M	
12.3.3.8	Channel Status event	M	M	M	
12.3.3.10	Concurrent BIP channels	M	M	M	
12.3.3.11	Contents of the TERMINAL PROFILE	M	M	M	
12.4.3.1	Contactless transaction during BIP session	M	M	M	
12.4.3.2.1	Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.2	Receiving and accepting a voice call during BIP CAT-TP data transfer on LTE network only	C002	C002	C002	
12.4.3.2.3	Voice Call made from the device during BIP CAT-TP session on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.4	Voice Call made from the device during BIP CAT-TP session on LTE network only	C002	C002	C002	
12.4.3.2.5	BIP CAT-TP data transfer during a Voice Call is established on LTE network with 2G/3G fallback	C003	C003	C003	
12.4.3.2.6	BIP CAT-TP data transfer during a Voice Call is established on LTE network only	C002	C002	C002	
12.4.3.3.1	Test Sequence No 1 : OTA data Loading without PoR requested by OTA server	M	M	M	

Test case	Description	Test case applicability			Support
12.4.3.3.2	Test Sequence No 2 : OTA data Loading with PoR requested by OTA server	M	M	M	
12.4.3.3.3	Test Sequence No 3 : OTA data Loading with PoR requested by OTA server when the device is booting	M	M	M	
12.4.3.3.4	Test Sequence No 4 : OTA data Loading with PoR requested by OTA server after the device comes back from airplane mode.	M	M	M	
12.4.3.3.5	Test Sequence No 5 : OTA data Loading with PoR requested by OTA server only on error	M	M	M	
12.4.3.4	Secure Element Access during BIP session	M	M	M	
12.4.3.5.	SMS and Internet Connection during OTA data Loading	M	M	M	
13.3.1	Secure Element Access API & Access control in Radio Off State	M	TNR	TNR	
13.3.2	Enabled / Disabled states	M	M	M	
15.3.3.1	Shared library	M	N/A	N/A	
15.3.3.2	Getting an instance of the “NFC controller” class	M	N/A	N/A	
15.3.3.3	Getting the status of the NFC Controller	M	N/A	N/A	
15.3.3.4	Enabling the NFC Controller	M	N/A	N/A	
15.5.3.1	Switching to “Multicast” Mode	M	N/A	N/A	
15.7.3.1	Dynamic AIDs Registration APIs when Multiple Active CEEs model is supported	C018	N/A	N/A	
15.7.3.2	APIs for handling Dynamic AID Registration	M	N/A	N/A	

Table 5: Applicability of tests

Annex E Reference PKCS#15 structures

Logically

The reference PKCS#15 structures are using the following AID-s:
AID1 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41’

AID2 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42’

AID3 = ‘A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 43’

The hash values used for the sake of simplicity are:

Hash of certificate SP1 = '11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11'

Hash of certificate SP2 = '22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22'

E.1 PKCS#15 application for test case 5.3.1.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 11
30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42 30 04 04 02 43 11
30 08 82 00 30 04 04 02 43 10

ACConditions1:
30 05 A1 03 08 01 01

ACConditions2:
30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.2 PKCS#15 application for test case 5.3.1.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 00

E.3 PKCS#15 application for test case 5.3.1.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.4 PKCS#15 application for test case 5.3.1.4

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.5 PKCS#15 application for test case 5.3.1.5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 11

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions2:

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.6 PKCS#15 application for test case 5.3.1.6

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.7 PKCS#15 application for test case 5.3.1.7

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

30 08 82 00 30 04 04 02 43 11

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions2:

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.8 PKCS#15 application for test case 5.3.1.8

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.9 PKCS#15 application for test case 5.3.1.9

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 11

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions2:

30 00

E.10 PKCS#15 application for test case 5.3.2.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 00

E.11 PKCS#15 application for test case 5.3.2.1 Step5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 08 04 06 11 11 11 11 11 11 <!-- corrupted Hash -->

E.12 PKCS#15 application for test case 5.3.2.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 00

E.13 PKCS#15 application for test case 5.3.3.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.14 PKCS#15 file system for test case 5.3.3.1

PKCS#15 file system

MF (3F00)

|- EF DIR (2F00) --> shall reference PKCS-15

|

|- DF PKCS-15 (7F50)

  |

  |- ODF (5031) --> shall reference DODF

  |- DODF (5207) --> shall reference EF ACMain

  |- EF ACMain (4200) --> shall reference EF ACRules

  |- EF ACRules (4300) --> shall reference EF ACConditions files

  |- EF ACConditions1 (4310)

  |- EF ACConditions2 (4311)

  |- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42 30 04 04 02 43 10

ACConditions1:

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.15 PKCS#15 file system for test case 5.3.4.1

PKCS#15 file system

MF (3F00)

|- EF DIR (2F00) --> shall reference PKCS-15

|

|- DF PKCS-15 (7F50)

  |

  |- ODF (5031) --> shall reference DODF

  |- DODF (5207) --> shall reference EF ACMain

  |- EF ACMain (4200) --> shall reference EF ACRules

  |- EF ACRules (4300) --> shall reference EF ACConditions files

  |- EF ACConditions1 (4310)

  |- EF ACConditions2 (4311)

  |- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.16 PKCS#15 application for test case 5.3.5.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42 30 04 04 02 43 11

ACConditions1:

30 16 04 14 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01

30 16 04 14 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02

30 16 04 14 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03

30 16 04 14 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04

30 16 04 14 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05

30 16 04 14 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06

30 16 04 14 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07

30 16 04 14 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08

30 16 04 14 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09
30 16 04 14 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A
30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions2:
30 16 04 14 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01
30 16 04 14 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02 02
30 16 04 14 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03
30 16 04 14 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04
30 16 04 14 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05 05
30 16 04 14 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06 06
30 16 04 14 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07 07
30 16 04 14 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08 08
30 16 04 14 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09 09
30 16 04 14 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A 0A
30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.17 PKCS#15 application for test case 5.3.5.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 42 30 04 04 02 43 11

30 1A A0 12 04 10 A0 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 01 30 04 04 02 43 12

30 1A A0 12 04 10 A0 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 02 30 04 04 02 43 12

...

<!--44 rules with dummy AID-->

...

30 1A A0 12 04 10 A0 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 47 30 04 04 02 43 12

30 1A A0 12 04 10 A0 04 00 00 00 00 00 00 00 00 00 00 00 00 00 00 48 30 04 04 02 43 12

ACConditions1:

30 16 04 14 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions2:

30 16 04 14 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01 01

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

ACConditions3:

30 16 04 14 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03 03

30 16 04 14 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04 04

E.18 PKCS#15 application for test case 5.3.6.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

<!-- ACRF is absend -->

ACConditions:

<!-- ACCF is absend -->

E.19 PKCS#15 application for test case 5.3.6.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

<!-- ACRF is present but empty -->

ACConditions:

<!-- ACCF is absend -->

E.20 PKCS#15 application for test case 5.3.6.4

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 12 04 10 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.21 PKCS#15 application for test case 5.3.6.5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 1A A0 12 04 10 A0 00 00 00 FF 50 00 00 00 00 00 00 52 41 44 41 30 04 04 02 43 10

ACConditions1:

30 18 04 16 F5 75 8A C7 F3 1C 1C F7 7F 45 1D 37 E3 15 CA 03 F9 89 59 2A 00 00

E.22 PKCS#15 application for test case 8.3.4.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.23 PKCS#15 application for test case 8.3.4.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.24 PKCS#15 application for test case 8.3.4.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 12

ACConditions1:

30 05 A1 03 08 01 01

ACConditions2:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

ACConditions3:

30 16 04 14 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22 22

E.25 PKCS#15 application for test case 13.3.1.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF

|- DODF (5207) --> shall reference EF ACMain

|- EF ACMain (4200) --> shall reference EF ACRules

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

|- EF ACConditions2 (4311)

|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.26 PKCS#15 application for test case 13.3.1.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11

E.27 PKCS#15 application for test case 13.3.1.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- ODF (5031) --> shall reference DODF
|- DODF (5207) --> shall reference EF ACMain
|- EF ACMain (4200) --> shall reference EF ACRules
|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

ODF:

A7 06 30 04 04 02 52 07

DODF:

A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:

30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 16 04 14 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11 11