NFC Handset Test Book
Version 14.1
30 January 2019

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

Table of Contents

1 Introduction

1.1 Overview

The main aim of the GSMA NFC activities is to accelerate the commercial launch of SE (Secure Element) based NFC services in a number of markets by ensuring interoperability of services.

It may not be possible to perform all the test cases currently defined in TS.27 using an eUICC or an eSE (Embedded Secure Element).

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 primarily the SE 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].

The NFC Test Book contains test cases for the following versions of TS.26:

- GSMA TS.26 NFC Handset Requirements V5.0 [1a]
- GSMA TS.26 NFC Handset Requirements V6.0 [1b]
- GSMA TS.26 NFC Handset Requirements V7.0 [1c]
- GSMA TS.26 NFC Handset Requirements V8.0 [1d]
- GSMA TS.26 NFC Handset Requirements V9.0 [1e]
- GSMA TS.26 NFC Handset Requirements V10.0 [1f]
- GSMA TS.26 NFC Handset Requirements V11.0 [1g]
- GSMA TS.26 NFC Handset Requirements V12.0 [1h]
- GSMA TS.26 NFC Handset Requirements V13.0 [1i]
- GSMA TS.26 NFC Handset Requirements V14.0 [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.

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 SE based NFC services (i.e. devices implementing SWP protocol). 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 refer 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 SE NFC services.

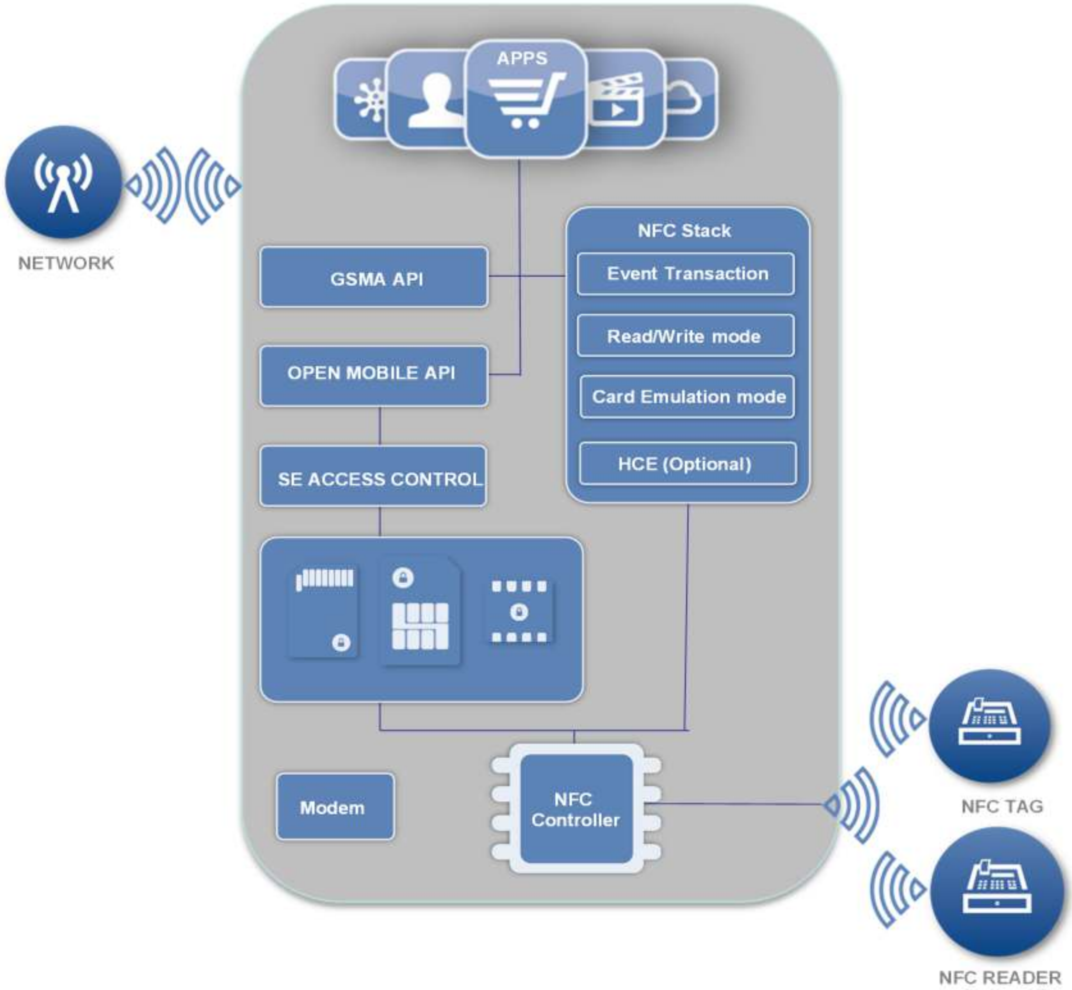


Figure 1.1: Reference architecture for a NFC enabled device with SE NFC services

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. VOID
17. VOID
18. VOID
19. Other OS specific test cases

Annexes

Other OS specific test cases can be added based on contributions.

1.3 Definition of Terms

Term	Description
Active UICC Profile	When the physical UICC is a standard UICC: the UICC itself. When the physical UICC is an eUICC: the combination of the Enabled Profile and the eUICC onto which the Profile has been provisioned.
Card Emulation Environment	A Card Emulation Environment is an execution environment used together with a NFC controller to manage a Card Emulation transaction. It can be a Secure Element (e.g. UICC, embedded Secure Element or micro-SD) or an application running in a device host.
Embedded UICC	A removable or non-removable UICC which enables the remote and/or local management of Profiles in a secure way. NOTE: The term originates from “embedded UICC”.
Default Route	The route to be used by the NFC controller for APDUs transmission when an AID is not found in the routing table.

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.
Distance	This refers to the distance from the back of the device to Point of Sale NFC antenna or to Tag surface.
Factory Reset	The action of the user to perform "device reset" to restore the factory configuration of the device. Note: The information on how to perform a factory reset shall be provided by the device manufacturer.
Issuer Security Domain	According to GlobalPlatform Card Specification: "The primary on-card entity providing support for the control, security, and communication requirements of the card administrator (typically the Card Issuer)"
Multiple Active CEEs model	A model where the device can activate several CEE at the same time. RF traffic can be provided to a CEE based on routing mechanisms. Note: an implementation may support Multiple Active CEEs model in Battery Operational Mode and Single Active CEE model in Battery Low or Power-Off Mode.
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.
Powered Off	The device was turned OFF by the end-user or the device is in battery low mode or the device is in battery power-off mode.
Screen Lock	The device functionality can only be accessed via a user intervention.
Screen OFF	The battery of the device is in Battery Operational Mode and the screen of the device was turned off either by the end-user or automatically by the device after a timeout.
Screen ON	The battery of the device is in Battery Operational Mode and the screen of the device was turned on by the end-user (i.e. the screen is active).
Secure Element	A SE is a tamper-resistant component which is used to provide security, confidentiality, and multiple application environments required to support various business models. In TS.27, the term SE includes UICC, eUICC and eSE.
Sensitive API	An API which shall be protected from malicious use.
Single Active CEE model	A model where the device only activates one CEE at a time. Other CEEs, if available, are not active.
Switched OFF	The device was turned OFF by the end-user or the device is in battery low mode or the device is in battery power-off mode.

Term	Description
Test Book	Document describing the test cases that allow testing the requirements listed in the GSMA TS.26 NFC Handset Requirements [1]
Test Lab	This refers to a test lab which will run the test cases according to the Test Book for testing NFC Devices.
Vendor	Device manufacturer

Table 1.1: Definition of Terms

1.4 Abbreviations

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
APDU	Application Protocol Data Unit
API	Application Programming Interface
APN	Access Point Network
BIP	Bearer Independent Protocol
C-APDU	Command APDU
CE	Card Emulation
CEE	Card Emulation Environment
CEN	European Committee for Standardization
CLF	Contactless Frontend
CS	Circuit Switched
DODF	Data Object Directory File
DUT	Device Under Test
EMV	EMV specifications and related testing processes are managed by EMVCo. (Europay, MasterCard, Visa)
eSE	Embedded Secure Element
ETSI	European Telecommunication Standards Institute
eUICC	A removable or non-removable UICC which enables the remote and/or local management of Profiles in a secure way. NOTE: The term originates from "embedded UICC".
EVT	Event
FFS	For Future Study

Acronyms	Description
GCF	Global Certification Scheme
HCE	Host Card Emulation
HCI	Host Controller Interface
IEC	International Electrotechnical Commission
ISD	Issuer Security Domain
ISO	International Organization for Standardization
JCP	Java Community Process
JSR	Java Specification Request
JVM	Java Virtual Machine
ME	Mobile Equipment
MIDP	Mobile Information Device Profile
MNO	Mobile Network Operator
MO	Mobile Originated
MT	Mobile Terminated
NFC	Near Field Communication
ODM	Original Device Manufacturer
OEM	Original Equipment Manufacturer
OS	Operating System
PCD	Proximity Coupling Device
PC/SC	PC SmartCard reader
PKCS	Public Key Cryptographic Standard
PoR	Proof of Receipt
PoS	Point of Sale
PS	Packet Switched
R-APDU	Response APDU
RIL	Radio Interface Layer
RTD	Record Type Definition
RTS	Routing Table Size
SCWS	Smart Card Web Server
SE	Secure Element
SIM	Subscriber Identity Module
SP	Service Provider
STA	Smart Ticketing Alliance
SW	Status Word

Acronyms	Description
SWP	Single Wire Protocol
UI	User Interface
UICC	Universal Integrated Circuit Card (USIM)
USS	UMTS System Simulator

Table 1.2: Abbreviations

1.4.1 Power mode definition

This section gives the definition for different battery modes for the support NFC services as shown in Figure 1.2.

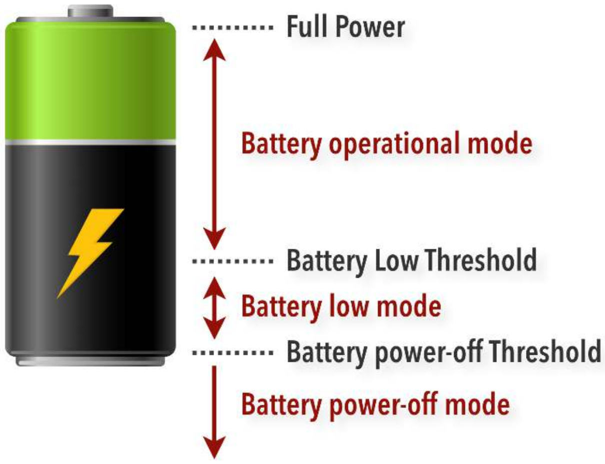


Figure 1.2: Battery power levels within the NFC mobile devices

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 the 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. powered by the field).

Table 1.3: Battery Power Levels

1.5 Document Cross-References

Ref	Title
[1]	GSMA TS.26 NFC Handset Requirements v14.0

Ref	Title
[1a]	GSMA TS.26 NFC Handset Requirements v5.0
[1b]	GSMA TS.26 NFC Handset Requirements v6.0
[1c]	GSMA TS.26 NFC Handset Requirements v7.0
[1d]	GSMA TS.26 NFC Handset Requirements v8.0
[1e]	GSMA TS.26 NFC Handset Requirements v9.0
[1f]	GSMA TS.26 NFC Handset Requirements v10.0
[1g]	GSMA TS.26 NFC Handset Requirements v11.0
[1h]	GSMA TS.26 NFC Handset Requirements v12.0
[1i]	GSMA TS.26 NFC Handset Requirements V13.0
[2]	VOID
[3]	VOID
[4]	VOID
[5]	GlobalPlatform OMAPI Transport API Test Specification V3.3
[6]	SIMalliance – Open Mobile API specification V3.2 and Second Errata, or later (backwards compatible)
[6a]	GlobalPlatform Open Mobile API Specification v3.3 or later
[7]	GlobalPlatform – Secure Element Access Control V1.0
[8]	ETSI TS 102 221 (V13.1.0 or later) – UICC-Terminal interface – Physical and logical characteristics
[9]	ETSI TS 102 613 (V11.0.0 or later) – UICC – Contactless Front-end (CLF) Interface – Part 1: Physical and data link layer characteristics
[10]	ETSI TS 102 622 (V13.0.0 or later) – UICC – Contactless Front-end (CLF) Interface – Host Controller Interface (HCI)
[11]	ETSI TS 102 694-1 (V10.2.0 or later) – Test specification for the Single Wire Protocol (SWP) interface; Part 1: Terminal features
[12]	ETSI TS 102 695-1 (V12.1.0 or later) - Test specification for the Host Controller Interface (HCI); Part 1: Terminal features
[13]	ETSI TS 102 384 (V10.3.0 or later) – Card Application Toolkit (CAT) conformance specification
[14]	VOID
[15]	GCF WI – 35 – USAT Testing
[16]	GCF WI – 133 – SWP/HCI
[17]	VOID
[18]	VOID
[19]	NFC Forum-TS-Analog NFC Forum-TS-Digital

Ref	Title
	NFC Forum-TS-Activity NFCForum-TS-T2T NFCForum-TS-T3T NFCForum-TS-T4T NFC Forum-TS-NDEF NFC Forum-TS-NCI The versions of each referenced Specification, are defined in the NFC Forum Technical Specification Release 2017 (or later release)
[20]	3GPP TS 31.121 (V13.4.0 or later) – UICC-terminal interface; Universal Subscriber Identity Module (USIM) application test specification
[21]	3GPP TS 31.124 (V13.5.0 or later) – Mobile Equipment (ME) conformance test specification; Universal Subscriber Identity Module Application Toolkit (USAT) conformance test specification
[22]	ETSI TS 102 223 (V12.0.0 or later) – Smart Cards; Card Application Toolkit (CAT)
[23]	ETSI TS 102 226 (V13.0.0 or later) – Smart Cards ;Remote APDU structure for UICC based applications
[24]	ETSI TS 102 127 (V6.13.0 or later) – Smart Cards; Transport protocol for CAT applications; Stage 2
[25]	3GPP TS 34.108 (V13.0.0 or later) – 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)
[29]	X.509 Certificate is published as ITU recommendation ITU-T X.509 (formerly CCITT X.509) and ISO/IEC/ITU 9594-8. It defines a standard certificate format for public key certificates and certification validation.
[30]	VOID
[31]	VOID
[32]	VOID
[33]	VOID
[34]	3GPP TS 31.116 (Release V14.0.0 or later) Remote APDU (Application Protocol Data Unit) Structure for (Universal) Subscriber Identity Module (U)SIM Toolkit applications
[35]	ISO/IEC 7816-3: 2006 or later "Identification cards - Integrated circuit cards - Part 3: Cards with contacts - Electrical interface and transmission protocols".
[36]	3GPP TS 36.508 (V13.1.0 or later) – LTE; Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Packet Core (EPC); Common test environments for User Equipment (UE) conformance testing

Ref	Title
[37]	VOID
[38]	EMV Contactless Communication Protocol Specification, Book D, Version 2.6 (or later)
[39]	EMV Mobile Product Level 1 Type Approval, Interoperability Testing Requirements, Version 1.0 (or later)
[40]	VOID
[41]	ETSI TS 102 230-1 (V11.0.0 or later) - Smart Cards ; UICC-Terminal interface Physical, electrical and logical test specification Test specification of [8]
[42]	GSMA: SGP12 v2.0 NFC Multi Protocol for Interoperability
[43]	Test Applets for GlobalPlatform OMAPI Test Specification for Transport API v3.3 are available at: https://github.com/GlobalPlatform/OMAPI-applets in “Test-Applets” repository.
[44]	GitHubGlobalPlatform sample ARA applet available at https://github.com/GlobalPlatform/OMAPI-applets in “ARA-Applet” repository.
[45]	Android OMAPI documentation : https://developer.android.com/reference/android/se/omapi/package-summary.html
[46]	NFC Forum Test Cases For Type 1 Tag Operation NFC Forum Test Cases For Type 2 Tag and Type 2 Tag Operation NFC Forum Test Cases For Type 3 Tag and Type 3 Tag Operation NFC Forum Test Cases For Type 4 Tag and Type 4 Tag Operation NFC Forum Test Cases for Analog NFC Forum Devices Requirements The versions of each referenced Test Specification and the Devices Requirements document above, are defined in the NFC Forum Certification Release 11 (or later)
[47]	GSMA TS.27 NFC Handset Requirements V13.0

Table 1.4: Document Cross-References

1.6 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

Table 1.5: Conventions

2 Test environment

2.1 Applicability

The purpose of this section is to confirm whether a test case as defined in the TS.27 is applicable.

For test cases defined in referenced specifications, the corresponding applicability is defined in the referenced specifications.

The applicability depends on the features supported in the device and/or on the Operating System.

This section consists of 6 tables which are the normative tables:

Table 2.4, 2.5 and 2.7 are to be completed by device supplier and test house respectively:

- Table 2.4: "Optional features": This is a template with features (device characteristics) optional for the device to support. This table should be completed by the supplier of the device. The completed template can be input for the compilation of list of applicable test cases from table 2.5.
- Table 2.5: "Applicability Table": This is a template which can be used to establish the list of applicable test cases depending on the supported features and the Operating System. The table provide a "Support" Column which should be used to state the established applicability complied from the conditional expressions.
- Table 2.7: "Device default configuration". Additional device information used for the testing.

Table 2.1, 2.2, 2.3 and 2.6 explain the format and content of Table 4 and 5.

- Table 2.1, 2.2 and 2.3: These tables explain the columns, the format and status notifications used in Table 4 and 5.
- Table 2.6: "Conditional Items": This is a list of conditional (Boolean) expressions to be evaluated by the test house. The expressions are evaluated based on Table 4 Optional Features and used to establish the complete list of applicable test cases for the device to be tested.

The format and usage of applicability definition follow the description within ETSI specifications e.g. ETSI TS 102 694-1 [11], but simplified to only cover the scope of TS.27.

2.1.1 Format of the table of optional features

The columns in table of optional features have the following meaning:

Column	Meaning
Item	Unique numbering of each optional feature
Optional Feature	The name of the optional feature supported or not supported by the device implementation.

Column	Meaning
Support	The support columns are to be filled in by the supplier of the device. The following common notations can be used: Y The feature is supported by the device. N The feature is not supported by the device.
Mnemonic	The mnemonic column contains mnemonic identifiers for each item which is a short name for the optional feature.

Table 2.1: Format of the table of optional features

2.1.2 Format of the applicability table

The format of the Applicability table is defined in the table below.

The columns in Table 2.5 have the following meaning:

Column	Meaning
Test case	The “Test case” column gives a unique reference to the test case.
Test Case Title	The “Test Case Title” column gives the title of the test case.
TS.26 Versions	The "TS.26 versions" column gives the item in the "Test Case title" column the applicable requirements version: - *Version x.y* “onwards”: if the requirement is applicable from this TS.26 version and for the later versions “Up to” *Version x.y:* if the requirement is applicable until this TS.26 version and it has been changed or replaced for the later TS.26 versions
Test case applicability	The “Test case applicability” column indicates which test cases are applicable per given Device Operating System. Several different status notifications can be used in this column. They are defined in the table in section 2.1.3.
Applicable	This column can be used to state the final applicability for each test case derived from the conditional expressions. Completion of this column shall be either “Yes” or “No”.

Table 2.2: Format of the applicability table

The Applicability Table does not include test cases in the status FFS. The FFS test cases are only included in the complete list of test cases in Annex D.1.

2.1.3 Status and Notations of the Applicability Table

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 test case is mandatory for a device implementation using the given Operating System. If the test case refers to an external specification, there might be several additional test cases required. This means the specific applicability of each underlying test cases has to be evaluated according to the applicability within the external specification. For example if an "M" is stated in the TS.27, it does not necessarily mean that all the underlying test cases are applicable.
FFS	See section 2.2.4
N/A	Not Applicable – the test case is not applicable for a device using the given Operating System, i.e. the test is not required. N/A is considered as a permanent "Not Applicable" test case compared to TNR, see below.
TNR	Test Not Ready – the test case is not available in this version of TS.27 for a device using the given Operating System. This means in a future version of TS.27, the test case is expected to be updated to support the specific Operating System or a new test case will be defined.
Ci	Conditional – the requirement on the capability ("M", "O" or "N/A") depends on the support of other optional or conditional items. "i" is an integer identifying a 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 2.3: Status and Notations

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	Optional Feature	Support	Mnemonic (short name for the optional feature)
1	VOID		
2	Support of LTE/IMS		O_LTE/IMS
3	Support of LTE with fallback to 2G/3G		O_LTE/2G-3G
4	Support of read/write NFC Tag at distance > 1,0cm and ≤ 2,0cm		O_TAG_DISTANCE 2CM
5	Support of read/write NFC Tag at distance > 2,0cm and ⩽ 3,0cm, see note 5		O_TAG_DISTANCE_3CM

Item	Optional Feature	Support	Mnemonic (short name for the optional feature)
6	Support of read/write NFC Tag at distance > 3,0cm and ≤ 4,0cm, see note 4		O_TAG_DISTANCE_4CM
7	Support of battery low mode, see note 2		O_BAT_LOW
8	Support of battery off mode, see notes 2 and 3		O_BAT_OFF
9	VOID		
10	Support of Multiple APN		O_MULTI_APN
11	Terminal executes User confirmation phase before sending PDP context activation request		O_User_Confirm_Before_PDP_Context_Request
12	Support of Multiple Active CEEs model in Battery Operational Mode		O_MULTI_CEE_ON
13	VOID		
14	Default route selection user menu is provided by the DUT		O_DEFAULT_ROUTE_SELECTION_BY_USER_MENU
15	The NFC status is persistent across DUT power off and power on		O_NFC _PERSISTENCE
16	VOID		
17	Support of Card Emulation for FeliCa on UICC (see note 7)		O_CARD_EM_FELICA_UICC
18	Support of MIFARE Classic Reader/Writer		O_MIFARE_CLASSIC_RW
19	Support of MIFARE DESFire Reader/Writer		O_MIFARE_DESFIRE_RW
20	Terminal supports Short Message Service (SMS) MT over CS (see note 8 and 10)		pc_SMS_CS_MT
21	Terminal supports Short Message Service (SMS) MO over CS (see note 9 and 10)		pc_SMS_CS_MO
22	Terminal supports Short Message Service (SMS) MT over PS (see note 8 and 10)		pc_SMS_PS_MT
23	Terminal supports Short Message Service (SMS) MO over PS (see note 9 and 10)		pc_SMS_PS_MO

Item	Optional Feature	Support	Mnemonic (short name for the optional feature)
24	Preferred buffer size supported by the terminal for Open Channel command is greater than 0 byte and less than 65535 bytes		O_BUFFER_SIZE
25	VOID		
26	Support of REQ_143 (see Note 11)		O_REQ_143
27	DUT implements Android versions before OS 6.0 Marshmallow (Marshmallow is not included)		O_BEFORE_ANDROID_MARSHMALLOW
28	Support of complete functionality of the GSMA Android NFC API as defined in TS.26 v8.0 or later (see note 12)		O_ANDROID_API_V8_OR_LATER
29	Support of TS26 version 9.0 (see note 14)		O_TS26_VERSION_9
30	Support of TS26 version 10.0 (see note 14)		O_TS26_VERSION_10
31	DUT implements Android versions before OS 7.0 Nougat (Nougat is not included)		O_BEFORE_ANDROID_NOUGAT
32	DUT implements GSMA Android NFC API (See note 15)		O_GSMA_API
33	DUT contains eSE (see note 16)		O_eSE
34	DUT implements Android versions before P (P is not included)		O_BEFORE_ANDROID_P
35	Support of REQ_167.1		O_REQ_167.1

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

Note 2: Up to TS.26 v13.0, for options O_BAT_LOW and O_BAT_OFF the DUT shall support at least one of these options or both. From TS.26 v14.0, O_BAT_LOW SHALL be supported, and O_BAT_OFF SHALL not be supported.

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

Note 6: VOID

Note 7: As per TS26_NFC_REQ_009, TS26_NFC_REQ_009.1, TS26_NFC_REQ_025 and TS26_NFC_REQ_025.1: O_CARD_EM_FELICA_UICC was optional until 31st July 2015; from 1st August 2015, it is now mandatory.

Note 8: IF pc_SMS_PS_MT is supported, then pc_SMS_CS_MT is optional, ELSE pc_SMS_CS_MT is mandatory

Note 9: IF pc_SMS_PS_MO is supported, then pc_SMS_CS_MO is optional, ELSE pc_SMS_CS_MO

Item	Optional Feature	Support	Mnemonic (short name for the optional feature)
is mandatory Note 10: The options pc_SMS_CS_MT, pc_SMS_CS_MO, pc_SMS_PS_MT and pc_SMS_PS_MO are related to the test cases in Chapter 12.3.3.9. Note 11: O_REQ_143 is used in section 2.6.1. For devices supporting TS.26 v8.0 or later and O_MULTI_CEE_ON, it is mandatory that the device supports this option, as O_REQ_143 is mandatory from TS.26 v8.0. For devices supporting versions of TS.26 before v8.0, support of O_REQ_143 is optional. Note 12: O_ANDROID_API_V8_OR_LATER is used in test cases in section 15. For devices supporting TS.26 v8.0 , v9.0 or v10.0, support of O_ANDROID_API_V8_OR_LATER is mandatory. For devices supporting versions of TS.26 before v8.0 or after v10.0, support of O_ANDROID_API_V8_OR_LATER is optional. If a device indicates support of the GSMA Android NFC API as defined in a particular version of TS.26, it shall support the complete functionality of the API as defined in that version. Note 13: VOID. Note 14: • For devices supporting TS.26 v9.0 the support of O_TS26_VERSION_9 is mandatory. • For devices supporting TS.26 v10.0 the support of O_TS26_VERSION_10 is mandatory. Note 15: For devices supporting TS.26 v5.0 to v10.0, the support of O_GSMA_API is mandatory. For devices supporting TS.26 v11.0 or later, the support of O_GSMA_API is optional. Note 16: Devices containing eSE shall be configured according to Annex F.			

Table 2.4: Optional Features

2.1.5 Applicability Table

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

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
			Android	Others	
3.3.3.1	NFC Forum Type 1 Tag – Read NFC Tag	5.0 to 13.0	M	FFS	
3.3.3.2	NFC Forum Type 2 Tag – Read NFC Tag	5.0 onwards	M	FFS	
3.3.3.3	NFC Forum Type 3 Tag – Read NFC Tag	5.0 onwards	M	FFS	
3.3.3.4	NFC Forum Type 4 Tag – Read NFC Tag	5.0 onwards	M	FFS	
3.3.3.5.1	NFC Forum Type 1 Tag – Write NFC Tag – Test Sequence No 1	5.0 to 13.0	M	FFS	
3.3.3.5.2	NFC Forum Type 1 Tag – Write NFC Tag – Test Sequence No 2	5.0 to 13.0	FFS	FFS	
3.3.3.6	NFC Forum Type 2 Tag – Write NFC Tag	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.3.3.7	NFC Forum Type 3 Tag – Write NFC Tag	5.0 onwards	M	FFS	
3.3.3.8	NFC Forum Type 4 Tag – Write NFC Tag	5.0 onwards	M	FFS	
3.3.3.9.1	Distance for NFC Type 1 Tag reading Test Sequence No 1: Distance for NFC Type 1 Tag Reading – 0,0cm	5.0 to 13.0	M	FFS	
3.3.3.9.2	Distance for NFC Type 1 Tag reading Test Sequence No 2: Distance for NFC Type 1 Tag Reading – 0,5cm	5.0 to 13.0	M	FFS	
3.3.3.9.3	Distance for NFC Type 1 Tag reading Test Sequence No 3: Distance for NFC Type 1 Tag reading – 1,0cm	5.0 to 13.0	M	FFS	
3.3.3.9.4	Distance for NFC Type 1 Tag reading Test Sequence No 4: Distance for NFC Type 1 Tag Reading – 2,0cm	5.0 to 13.0	C015	FFS	
3.3.3.9.5	Distance for NFC Type 1 Tag reading Test Sequence No 5: Distance for NFC Type 1 Tag Reading – 3,0cm	6.0 to 13.0	C016	FFS	
3.3.3.9.6	Distance for NFC Type 1 Tag reading Test Sequence No 6: Distance for NFC Type 1 Tag Reading – 4,0cm	6.0 to 13.0	C017	FFS	
3.3.3.10.1	Distance for NFC Type 2 Tag reading Test Sequence No 1: Distance for NFC Type 2 Tag Reading – 0,0cm	5.0 onwards	M	FFS	
3.3.3.10.2	Distance for NFC Type 2 Tag reading Test Sequence No 2: Distance for NFC Type 2 Tag Reading – 0,5cm	5.0 onwards	M	FFS	
3.3.3.10.3	Distance for NFC Type 2 Tag reading Test Sequence No 3: Distance for NFC Type 2 Tag reading – 1,0cm	5.0 onwards	M	FFS	
3.3.3.10.4	Distance for NFC Type 2 Tag reading Test Sequence No 4: Distance for NFC Type 2 Tag Reading – 2,0cm	5.0 onwards	C015	FFS	
3.3.3.10.5	Distance for NFC Type 2 Tag reading Test Sequence No 5: Distance for NFC Type 2 Tag Reading – 3,0cm	6.0 onwards	C016	FFS	
3.3.3.10.6	Distance for NFC Type 2 Tag reading Test Sequence No 6: Distance for NFC Type 2 Tag Reading – 4,0cm	6.0 onwards	C017	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.3.3.11.1	Distance for NFC Type 3 Tag reading Test Sequence No 1: Distance for NFC Type 3 Tag Reading – 0,0cm	5.0 onwards	M	FFS	
3.3.3.11.2	Distance for NFC Type 3 Tag reading Test Sequence No 2: Distance for NFC Type 3 Tag Reading – 0,5cm	5.0 onwards	M	FFS	
3.3.3.11.3	Distance for NFC Type 3 Tag reading Test Sequence No 3: Distance for NFC Type 3 Tag reading – 1,0cm	5.0 onwards	M	FFS	
3.3.3.11.4	Distance for NFC Type 3 Tag reading Test Sequence No 4: Distance for NFC Type 3 Tag Reading – 2,0cm	5.0 onwards	C015	FFS	
3.3.3.11.5	Distance for NFC Type 3 Tag reading Test Sequence No 5: Distance for NFC Type 3 Tag Reading – 3,0cm	6.0 onwards	C016	FFS	
3.3.3.11.6	Distance for NFC Type 3 Tag reading Test Sequence No 6: Distance for NFC Type 3 Tag Reading – 4,0cm	6.0 onwards	C017	FFS	
3.3.3.12.1	Distance for NFC Type 4A Tag reading Test Sequence No 1: Distance for NFC Type 4 TagA Reading – 0,0cm	5.0 onwards	M	FFS	
3.3.3.12.2	Distance for NFC Type 4A Tag reading Test Sequence No 2: Distance for NFC Type 4 TagA Reading – 0,5cm	5.0 onwards	M	FFS	
3.3.3.12.3	Distance for NFC Type 4A Tag reading Test Sequence No 3: Distance for NFC Type 4 TagA reading – 1,0cm	5.0 onwards	M	FFS	
3.3.3.12.4	Distance for NFC Type 4A Tag reading Test Sequence No 4: Distance for NFC Type 4 TagA Reading – 2,0cm	5.0 onwards	C015	FFS	
3.3.3.12.5	Distance for NFC Type 4A Tag reading Test Sequence No 5: Distance for NFC Type 4 TagA Reading – 3,0cm	6.0 onwards	C016	FFS	
3.3.3.12.6	Distance for NFC Type 4A Tag reading Test Sequence No 6: Distance for NFC Type 4 TagA Reading – 4,0cm	6.0 onwards	C017	FFS	
3.3.3.13.1	Distance for NFC Type 4B Tag reading Test Sequence No 1: Distance for NFC Type 4 TagB Reading – 0,0cm	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.3.3.13.2	Distance for NFC Type 4B Tag reading Test Sequence No 2: Distance for NFC Type 4 TagB Reading – 0,5cm	5.0 onwards	M	FFS	
3.3.3.13.3	Distance for NFC Type 4B Tag reading Test Sequence No 3: Distance for NFC Type 4 TagB reading – 1,0cm	5.0 onwards	M	FFS	
3.3.3.13.4	Distance for NFC Type 4B Tag reading Test Sequence No 4: Distance for NFC Type 4 TagB Reading – 2,0cm	5.0 onwards	C015	FFS	
3.3.3.13.5	Distance for NFC Type 4B Tag reading Test Sequence No 5: Distance for NFC Type 4 TagB Reading – 3,0cm	6.0 onwards	C016	FFS	
3.3.3.13.6	Distance for NFC Type 4B Tag reading Test Sequence No 5: Distance for NFC Type 4 TagB Reading – 4,0cm	6.0 onwards	C017	FFS	
3.3.3.14	NFC Type 1 Tag reading performance	5.0 to 13.0	M	FFS	
3.3.3.15	NFC Type 2 Tag reading performance	5.0 onwards	M	FFS	
3.3.3.16	NFC Type 3 Tag reading performance	5.0 onwards	M	FFS	
3.3.3.17	NFC Type 4A Tag reading performance	5.0 onwards	M	FFS	
3.3.3.18	NFC Type 4B Tag reading performance	5.0 onwards	M	FFS	
3.3.3.19	NFC Tag handling during an active data transfer.	5.0 onwards	M	FFS	
3.3.3.24.1	NFC Forum Type 1 Tag Operations Test Cases	6.0 to 13.0	M	FFS	
3.3.3.24.2	NFC Forum Type 2 Tag Operations Test Cases	6.0 onwards	M	FFS	
3.3.3.24.3	NFC Forum Type 3 Tag Operations Test Cases	6.0 onwards	M	FFS	
3.3.3.24.4	NFC Forum Type 4 Tag Operations Test Cases	6.0 onwards	M	FFS	
3.3.3.25	NFC Forum Test Cases for Analog (all valid versions)	8.0 onwards	M	FFS	
3.3.3.27	NFC Forum Test Cases for Analog 2.0 only	10.0. onwards	M	FFS	
3.3.3.28	Extended Length APDU handling	11.0. onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.4.3.1.1	Card Emulation enabled as soon as NFC hardware is on Test Sequence No.1	5.0 onwards	M	FFS	
3.4.3.1.2	Card Emulation enabled as soon as NFC hardware is on Test sequence No 2	5.0 onwards	C014	FFS	
3.4.3.1.3	Card emulation in device on but in screen locked	12.0 onwards	M	FFS	
3.4.3.1.4	Card emulation in device on but screen off	12.0 onwards	M	FFS	
3.4.3.2	NFC during Standby time	5.0 onwards	C005	FFS	
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	5.0 to 13.0	C006	FFS	
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	5.0 to 13.0	C006	FFS	
3.4.3.3.4	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction in Battery Low modes Test sequence No 4: Card Emulation Mode Type A in Battery Low Mode	5.0 onwards	C005	FFS	
3.4.3.3.5	Verify that device is able to perform Card Emulation Mode A, Card Emulation Mode B and CLT A transaction in Battery Low modes Test sequence No 5: Card Emulation Mode Type B in Battery Low Mode	5.0 onwards	C005	FFS	
3.4.3.4	Distance for card emulation	5.0 onwards	M	FFS	
3.4.3.5	Distance for card emulation in Battery Power-off Mode (0cm)	5.0 to 13.0	C006	FFS	
3.4.3.6	Distance for card emulation in Battery Power-off Mode (0.5cm)	5.0 to 13.0	C006	FFS	
3.4.3.7	Distance for card emulation in Battery Power-off Mode (1cm)	5.0 to 13.0	C006	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.4.3.8	Distance for card emulation in Battery Power-off Mode (1.5cm)	5.0 to 13.0	C006	FFS	
3.4.3.9	Distance for card emulation in Battery Power-off Mode (2cm)	5.0 onwards	C006	FFS	
3.4.3.10	Distance for card emulation in Battery Power-low Mode (0cm)	5.0 onwards	C005	FFS	
3.4.3.11	Distance for card emulation in Battery Power-low Mode (0.5cm)	5.0 onwards	C005	FFS	
3.4.3.12	Distance for card emulation in Battery Power-low Mode (1cm)	5.0 onwards	C005	FFS	
3.4.3.13	Distance for card emulation in Battery Power-low Mode (1.5cm)	5.0 onwards	C005	FFS	
3.4.3.14	Distance for card emulation in Battery Power-low Mode (2cm)	5.0 onwards	C005	FFS	
3.4.3.15	Distance for card emulation in Battery Power- operational Mode (0cm)	5.0 onwards	M	FFS	
3.4.3.16	Distance for card emulation in Battery Power- operational Mode (0.5cm)	5.0 onwards	M	FFS	
3.4.3.17	Distance for card emulation in Battery Power- operational Mode (1cm)	5.0 onwards	M	FFS	
3.4.3.18	Distance for card emulation in Battery Power- operational Mode (1.5cm)	5.0 onwards	M	FFS	
3.4.3.19	Distance for card emulation in Battery Power- operational Mode (2cm)	5.0 onwards	M	FFS	
3.4.3.20.1	Card emulation with switched off device (0cm)	11.0 onwards	M	FFS	
3.4.3.20.2	Card emulation with switched off device (0.5cm)	11.0 onwards	M	FFS	
3.4.3.20.3	Card emulation with switched off device (1cm)	11.0 onwards	M	FFS	
3.4.3.20.4	Card emulation with switched off device (1.5cm)	11.0 onwards	M	FFS	
3.4.3.20.5	Card emulation with switched off device (2cm)	11.0 onwards	M	FFS	
3.4.3.21	Extended Length APDU handling	11.0 onwards	M	FFS	
3.5.3.1	SWP Compliance testing	5.0 onwards	M	FFS	
3.5.3.2	HCI Compliance testing	5.0 onwards	M	FFS	
3.5.3.3	SWP Stress test	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
3.5.3.4	Switch mode	5.0 onwards	M	FFS	
3.5.3.5	RF Analog Protocol compliance	5.0 onwards	M	FFS	
3.5.3.7	RF Digital Protocol compliance	11 onwards	M	FFS	
5.3.1.1	GP SE Access Control – Test Sequence 1	5.0 onwards	M	FFS	
5.3.1.2	GP SE Access Control – Test Sequence 2	5.0 onwards	M	FFS	
5.3.1.3	GP SE Access Control – Test Sequence 3	5.0 onwards	M	FFS	
5.3.1.4	GP SE Access Control – Test Sequence 4	5.0 onwards	M	FFS	
5.3.1.5	GP SE Access Control – Test Sequence 5	5.0 onwards	C029	FFS	
5.3.1.6	GP SE Access Control – Test Sequence 6	5.0 onwards	C029	FFS	
5.3.1.7	GP SE Access Control – Test Sequence 7	5.0 onwards	C029	FFS	
5.3.1.8	GP SE Access Control – Test Sequence 8	5.0 onwards	M	FFS	
5.3.1.9	GP SE Access Control – Test Sequence 9	5.0 onwards	M	FFS	
5.3.2	GP SE Access Control - Refresh tag	5.0 onwards	M	FFS	
5.3.3	GP SE Access Control – ADF_PKCS#15 and DF PKCS#15	5.0 onwards	M	FFS	
5.3.4	GP SE Access Control – PKCS#15 selection via EF_DIR	5.0 onwards	M	FFS	
5.3.5	GP SE Access Control – Configuration limits	5.0 onwards	M	FFS	
5.3.6	GP SE Access Control – No access	5.0 onwards	M	FFS	
5.4	GP SE Access Control – GP Test Plan See Note 1 at the end of the table.	5.0 onwards	M	FFS	
6.3.1	GlobalPlatform OMAPI	See Annex B.1	M	FFS	
6.3.7	GlobalPlatform OMAPI for eSE	See Annex B.1	C028	FFS	
7.3.7.1	Multiple CE Environments Test Sequence No 1: Default route UICC, contactless session with unregistered AID	6.0 onwards	C019	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
7.3.7.2	Multiple CE Environments Test Sequence No 2: Default route HCE, contactless session with unregistered AID	6.0 onwards	C018	FFS	
7.3.7.3	Multiple CE Environments Test Sequence No 3: Default route UICC, off-host AID	6.0 onwards	C019	FFS	
7.3.7.4	Multiple CE Environments Test Sequence No 4: Default route HCE, off-host AID	6.0 onwards	C018	FFS	
7.3.7.5	Multiple CE Environments Test Sequence No 5: Default route UICC, AID conflict, off-host service selected	6.0 onwards	C019	FFS	
7.3.7.6	Multiple CE Environments Test Sequence No 6: Default route HCE, AID conflict, off-host service selected	6.0 onwards	C018	FFS	
7.3.7.7	Multiple CE Environments Test Sequence No 7: Default route UICC, off-host service selected in Tap&Pay	6.0 onwards	C019	FFS	
7.3.7.8	Multiple CE Environments Test Sequence No 8: Default route HCE, off-host service selected in Tap&Pay	6.0 onwards	C018	FFS	
7.3.7.9	Multiple CE Environments Test Sequence No 9: Default route UICC, HCE service selected in Tap&Pay	6.0 onwards	C019	FFS	
7.3.7.10	Multiple CE Environments Test Sequence No 10: Default route HCE, HCE service selected in Tap&Pay	6.0 onwards	C018	FFS	
7.3.8.2	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 2: REQ_065 for NFCA	6.0 onwards	C019	FFS	
7.3.8.3	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 3: REQ_118.2 for NFCA	6.0 onwards	C019	FFS	
7.3.8.4	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 4: REQ_118.2 for NFCB	6.0 onwards	C019	FFS	
7.3.8.5	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 5: REQ_118.1 and REQ_162.1 for NFCA	6.0 onwards	C019	FFS	
7.3.8.6	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 6: REQ_065 for NFCB	6.0 onwards	C019	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
7.3.8.9	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 9: REQ_118.2 and REQ_162.1 for NFCA	9.0 onwards	C019	FFS	
7.3.8.10	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 10: REQ_118.2 and REQ_162.1 for NFCB	9.0 onwards	C019	FFS	
7.3.8.11	Active Card Emulation in Multiple CE Environments / Card Emulation Sequence No 11: REQ_177 for NFCA	11.0 onwards	C019	FFS	
7.3.9	Size of the CLF AID Routing table	10.0 onwards	M	FFS	
8.3.1	EVT_TRANSACTION	5.0 onwards	M	FFS	
8.3.2	Permissions	5.0 onwards	C022	FFS	
8.3.3	Intent management	5.0 onwards	M	FFS	
8.3.4.1	Application's triggering order – Test Sequence 1	5.0 onwards	C029	FFS	
8.3.4.2	Application's triggering order – Test Sequence 2	5.0 onwards	C029	FFS	
8.3.4.3	Application's triggering order – Test Sequence 3	5.0 onwards	C029	FFS	
8.3.5	Triggering on HCI event EVT_CARD_DEACTIVATED	5.0 onwards	M	FFS	
8.3.6	Triggering on HCI event EVT_FIELD_OFF	5.0 onwards	M	FFS	
11.3.1.1	OPEN CHANNEL Test Sequence 1: (OPEN CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN)	5.0 onwards	C008	FFS	
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)	5.0 onwards	C009	FFS	
11.3.1.3	OPEN CHANNEL Test Sequence 3: (OPEN CHANNEL – Default APN Always-ON – APN empty)	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
11.3.1.4	OPEN CHANNEL Test Sequence No 4: (OPEN CHANNEL – Default APN Always-ON – APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	
11.3.2.1	CLOSE CHANNEL Test Sequence 1: (CLOSE CHANNEL – Default APN Always-ON – Multiple APN supported – with different APN-SUCCESSFUL)	5.0 onwards	C008	FFS	
11.3.2.2	CLOSE CHANNEL Test Sequence 2: (CLOSE CHANNEL – Default APN Always-ON – Only Single APN supported – with different APN-SUCCESSFUL)	5.0 onwards	C009	FFS	
11.3.2.3	CLOSE CHANNEL Test Sequence 3: (CLOSE CHANNEL – Default APN Always-ON – APN empty-SUCCESSFUL)	5.0 onwards	M	FFS	
11.3.2.4	CLOSE CHANNEL Test Sequence No 4: (CLOSE CHANNEL – Default APN Always-ON – APN empty-SUCCESSFUL- Default Bearer Type Used)	5.0 onwards	M	FFS	
11.3.3.1	RECEIVE DATA Test Sequence 1: (RECEIVE DATA – Default APN Always-ON – Multiple APN supported – with different APN)	5.0 onwards	C008	FFS	
11.3.3.2	RECEIVE DATA Test Sequence 2: (RECEIVE DATA – Default APN Always-ON – Only Single APN supported – with different APN)	5.0 onwards	C009	FFS	
11.3.3.3	RECEIVE DATA Test Sequence 3: (RECEIVE DATA – Default APN Always-ON – APN empty)	5.0 onwards	M	FFS	
11.3.3.4	RECEIVE DATA Test Sequence 4: (RECEIVE DATA – Default APN Always-ON – APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
11.3.4.1	SEND DATA Test Sequence 1: (SEND DATA – Default APN Always-ON – Multiple APN supported –with different APN- BUFFER FULLY USED)	5.0 onwards	C008	FFS	
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)	5.0 onwards	C009	FFS	
11.3.4.3	SEND DATA Test Sequence 3: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED)	5.0 onwards	M	FFS	
11.3.4.4	SEND DATA Test Sequence 4: (SEND DATA – Default APN Always-ON – APN empty- BUFFER FULLY USED- Default Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.1	Remote management in BIP	5.0 onwards	M	FFS	
12.3.3.2.1	OPEN CHANNEL Test Sequence No 1: (OPEN CHANNEL, No APN, immediate link establishment, Default Bearer for requested transport layer, No local address, no alpha identifier)	5.0 onwards	M	FFS	
12.3.3.2.2	OPEN CHANNEL Test sequence No 2: (OPEN CHANNEL, with APN, immediate link establishment, Default Bearer for requested transport layer, no alpha identifier)	5.0 onwards	M	FFS	
12.3.3.2.3	OPEN CHANNEL Test Sequence No 3: (OPEN CHANNEL, with alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	5.0 onwards	M	FFS	
12.3.3.2.4	OPEN CHANNEL Test Sequence No 4: (OPEN CHANNEL, with null alpha identifier, immediate link establishment, Default Bearer for requested transport layer)	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
12.3.3.2.5	OPEN CHANNEL Test Sequence No 5: (OPEN CHANNEL, command performed with modifications (buffer size), immediate link establishment, Default Bearer for requested transport layer)	5.0 onwards	C020	FFS	
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,)	5.0 onwards	C010	FFS	
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)	5.0 onwards	C011	FFS	
12.3.3.3	CLOSE CHANNEL	5.0 onwards	M	FFS	
12.3.3.4	RECEIVE DATA	5.0 onwards	M	FFS	
12.3.3.5	SEND DATA	5.0 onwards	M	FFS	
12.3.3.6	GET CHANNEL STATUS	5.0 onwards	M	FFS	
12.3.3.7	Data available event	5.0 onwards	M	FFS	
12.3.3.8	Channel Status event	5.0 onwards	M	FFS	
12.3.3.9.1	**SMS-PP Data Download** Test Sequence No 1: (SMS-PP – followed by Open channel – Send/Receive data)	6.0 onwards	M	FFS	
12.3.3.9.2	SMS-PP Data Download Test Sequence No 2: (SMS-PP – Send SM – followed by Open channel – Send/Receive data)	6.0 onwards	M	FFS	
12.3.3.9.3	SMS-PP Data Download Test Sequence No 3: (SMS-PP – Send SM – followed by Open channel – Send/Receive data with timer management)	6.0 onwards	M	FFS	
12.3.3.9.5	Test Sequence No 4: (SMS-PP - Open channel - Send/Receive data - Send SM with More Time)	9.0 onwards	M	FFS	
12.3.3.9.6	Test Sequence No 5: (SMS-PP - Open channel - Send/Receive data - Send SM without More Time)	9.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
12.3.3.10	Concurrent BIP channels	5.0 onwards	M	FFS	
12.3.3.11	Contents of the TERMINAL PROFILE	5.0 onwards	M	FFS	
12.3.3.12.1	OPEN CHANNEL – Terminal connected to Wi-Fi Test Sequence No 1: (OPEN CHANNEL, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.12.2	OPEN CHANNEL – Terminal connected to Wi-Fi Test Sequence No 2: (OPEN CHANNEL, Terminal connected to Wi-Fi-APN empty-GPRS Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.13.1	CLOSE CHANNEL – Terminal connected to Wi-Fi Test Sequence No 1: (CLOSE CHANNEL, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.13.2	CLOSE CHANNEL – Terminal connected to Wi-Fi Test Sequence No 2: (CLOSE CHANNEL, Terminal connected to Wi-Fi-APN empty-GPRS Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.14	RECEIVE DATA – Terminal connected to Wi-Fi Test Sequence No 1: (RECEIVE DATA, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	
12.3.3.15	SEND DATA – Terminal connected to Wi-Fi Test Sequence No 1: (SEND DATA, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)	5.0 onwards	M	FFS	
12.4.3.1	Contactless transaction during BIP session	5.0 onwards	M	FFS	
12.4.3.2.1	Receiving and accepting a voice call during BIP CAT-TP data transfer	5.0 onwards	M	FFS	
12.4.3.2.3	Voice Call made from the device during BIP CAT-TP session	5.0 onwards	M	FFS	
12.4.3.2.5	BIP CAT-TP data transfer during a Voice Call is established	5.0 onwards	M	FFS	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
12.4.3.3.1	Test Sequence No 1: OTA data Loading without PoR requested by OTA server	5.0 onwards	M	FFS	
12.4.3.3.2	Test Sequence No 2: OTA data Loading with PoR requested by OTA server	5.0 onwards	M	FFS	
12.4.3.3.5	Test Sequence No 5: OTA data Loading with PoR requested by OTA server only on error	5.0 onwards	M	FFS	
12.4.3.4	Secure Element Access during BIP session	5.0 onwards	M	FFS	
12.4.3.5	SMS and Internet Connection during OTA data Loading	5.0 onwards	M	FFS	
13.3.1	Secure Element Access API in Radio Off State	5.0 onwards	M	FFS	
13.3.2	Enabled / Disabled states	5.0 onwards	M	FFS	
13.3.3	Modem and UICC over APDU exchange	5.0 onwards	M	FFS	
13.3.4	Modem retrieves the response data to the SELECT command	8.0 onwards	M	FFS	
13.3.5	Modem supports 19 logical channels	8.0 onwards	M	FFS	
13.3.6.1	Get Response APDU segmented from UICC (Case2 Command)	9.0 onwards	M	FFS	
13.3.6.2	Get Response APDU segmented from UICC (Case4 Command)	9.0 onwards	M	FFS	
13.3.6.3	Long APDU answer from UICC (Case2 Command)	11.0 onwards	M	FFS	
13.3.6.4	Long APDU command + answer from UICC (Case4 Command)	11.0 onwards	M	FFS	
13.3.7	Terminal Capability TAG 82	9.0 onwards	M	FFS	
13.3.8	Reselect previously non-existing applet	10.0 onwards	M	FFS	
13.3.9	Retrieve CIN and IIN from eSE ISD by mobile application	12.0 onwards	C028	FFS	
15.3.3.1	Shared library	6.0 to 13.0	C023	N/A	
15.3.3.2	Getting an instance of the “NFC controller” class	5.0 to 13.0	C023	N/A	
15.3.3.3	Getting the status of the NFC Controller	5.0 to 13.0	C023	N/A	
15.3.3.4	Enabling the NFC Controller	6.0 to 13.0	C023	N/A	
15.4.3.1	Open Mobile Service layer API	6.0 onwards	M	N/A	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
15.4.3.2	GlobalPlatform OMAPI & GP access Control just after device boot	6.0 onwards	M	N/A	
15.4.3.3	Getting the list of available Secure Elements	8.0 to 13.0	C023	N/A	
15.4.3.4.1	Usage of identical SE Names across device components	8.0 to 13.0	C024	N/A	
15.4.3.4.2	Usage of identical SE Names across device components (without using GSMA APIs)	11.0 onwards	C018	N/A	
15.5.3.1	Switching to “Multicast” Mode	6.0 to 13.0	C030	N/A	
15.5.3.2	Receiving EVT Transaction from “BroadcastReceiver”	6.0 to 13.0	C030	N/A	
15.5.3.3	Checking EVT Transaction data sent through “BroadcastReceiver”	6.0 to 13.0	C030	N/A	
15.5.3.4	Test EVT Transaction event shall be handled only by the appropriate application	5.0 onwards	C029	N/A	
15.5.3.5	Application Permission for Open Mobile API	6.0 onwards	C029	N/A	
15.5.3.6	EVT Transaction is received only by the appropriate application based on priority scheme	8.0 onwards	C029	N/A	
15.7.3.1	Dynamic AIDs Registration APIs – “com.gsma.services.nfc.NfcController” class	6.0 to 13.0	C024	N/A	
15.7.3.2	Dynamic AIDs Registration APIs – “com.gsma.services.nfc.OffHostService” class	6.0 to 13.0	C024	N/A	
15.7.3.3	Dynamic AIDs Registration APIs – “com.gsma.services.nfc.AidGroup” class	6.0 to 13.0	C024	N/A	
15.7.3.4.1	“Long press” on “Tap&Pay” menu entries – Test Sequence No 1	6.0 to 13.0	C024	N/A	
15.7.3.4.2	“Long press” on “Tap&Pay” menu entries - Test Sequence No 2	6.0 to 13.0	C024	N/A	
15.7.3.5	Routing in Multiple CEE model	6.0 to 13.0	C024	N/A	
15.7.3.6.1	AID Conflict Resolution mechanism – Test Sequence No 1	6.0 to 13.0	C024	N/A	
15.7.3.6.2	AID Conflict Resolution mechanism - Test Sequence No 2	11.0 onwards	C018	N/A	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
15.7.3.7.1	Test Sequence No 1: Application uninstalled	6.0 to 13.0	C024	N/A	
15.7.3.7.2	Test Sequence No 2: Application disabled and re-enabled	9.0 onwards	C018	N/A	
15.7.3.7.3	Test Sequence No 3: Application uninstalled (without using GSMA API)	11.0 onwards	C018	N/A	
15.7.3.8.1	Routing update when Application is updated / upgraded in Multiple CEE model - Test Sequence No 1	6.0 to 13.0	C024	N/A	
15.7.3.8.2	Routing update when Application is updated / upgraded in Multiple CEE model – Test Sequence No 2	6.0 onwards	C018	N/A	
15.7.3.8.3	Routing update when Application is updated / upgraded in Multiple CEE model - Test Sequence No 3	11.0 onwards	C018	N/A	
15.7.3.9.1	NFC Controller routing table – Test Sequence No 1	6.0 to 8.0	C018	N/A	
15.7.3.9.3	NFC Controller routing table – Test Sequence No 3	6.0 to 13.0	C024	N/A	
15.7.3.9.4	NFC Controller routing table – Test Sequence No 4	6.0 to 13.0	C024	N/A	
15.7.3.9.5	NFC Controller routing table – Test Sequence No 5	11.0 onwards	C031	N/A	
15.7.3.9.6	NFC Controller routing table – Test Sequence No 6	11.0 onwards	C031	N/A	
15.7.3.10.1	Tap&Pay menu – routing of APDUs for payment services – Test Sequence No 1	8.0 onwards	C018	N/A	
15.7.3.10.2	Tap&Pay menu – routing of APDUs for payment services – Test Sequence No 2	8.0 onwards	C018	N/A	
15.7.3.10.3	Tap&Pay menu – routing of APDUs for payment services – Test Sequence No 3	11.0 onwards	C018	N/A	
15.7.3.11.1	Dynamic & Automatic switch of AID default route – Test Sequence No 1	8.0 onwards	C018	N/A	
15.7.3.11.2	Dynamic & Automatic switch of AID default route - Test Sequence No 2	8.0 to 13.0	C024	N/A	
15.7.3.11.3	Dynamic & Automatic switch of AID default route – Test Sequence No 3	11.0 onwards	C031	N/A	
15.7.3.12.1	Routing in Multiple CEE model without using GSMA API – Test Sequence No 1	6.0 onwards	C018	N/A	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
15.7.3.12.2	Routing in Multiple CEE model without using GSMA API – Test Sequence No 2	11.0 onwards	C018	N/A	
15.7.3.12.3	Routing in Multiple CEE model without using GSMA API – Test Sequence No 3	11.0 onwards	C018	N/A	
15.7.3.12.4	Routing in Multiple CEE model without using GSMA API – Test Sequence No 4	11.0 onwards	C018	N/A	
15.7.3.12.5	Routing in Multiple CEE model without using GSMA API – Test Sequence No 5	11.0 onwards	C018	N/A	
15.7.3.12.6	Routing in Multiple CEE model without using GSMA API – Test Sequence No 6	11.0 onwards	C018	N/A	
15.7.3.12.7	Routing in Multiple CEE model without using GSMA API – Test Sequence No 7	11.0 onwards	C018	N/A	
15.7.3.12.8	Routing in Multiple CEE model without using GSMA API – Test Sequence No 8	11.0 onwards	C018	N/A	
15.7.3.12.9	Routing in Multiple CEE model without using GSMA API – Test Sequence No 9	11.0 onwards	C018	N/A	
15.7.3.12.10	Routing in Multiple CEE model without using GSMA API – Test Sequence No 10	11.0 onwards	C018	N/A	
15.7.3.13	Routing in Multiple CEE model with eSE	11.0 onwards	C027	N/A	
15.7.3.14	Routing in Multiple CEE model with eSE in Battery Low Mode	11.0 onwards	C027	N/A	
15.7.3.15	nonAID based services registration and conflict management	11.0 onwards	C027	N/A	
15.8.3.1	"getVersion" API	6.0 to 13.0	C023	N/A	
15.8.3.2.1	"getProperty" API – Test Sequence No 1: OMAPI	6.0 to 13.0	C023	N/A	
15.8.3.2.2	"getProperty" API – Test Sequence No 2: MULTIPLE_ACTIVE_CEE	6.0 to 13.0	C023	N/A	
15.8.3.2.3	"getProperty" API – Test Sequence No 3: HCI_SWP, BATTERY_LOW_MODE	6.0 to 13.0	C023	N/A	
15.8.3.2.4	"getProperty" API - Test Sequence No 4: BATTERY_POWER_OFF_MODE, FELICA, MIFARE_CLASSIC, MIFARE_DESFIRE	6.0 to 8.0	C023	N/A	
15.8.3.2.5	"getProperty" API - Test Sequence No 5: Invalid argument	6.0 to 13.0	C023	N/A	

Test Case	Test Case Title	TS.26 versions	Test Case Applicability		Applicable Yes / No
15.8.3.2.6	"getProperty" API - Test Sequence No 6: BATTERY_POWER_OFF_MODE, FELICA, MIFARE_CLASSIC, MIFARE_DESFIRE	9.0 to 13.0	C023	N/A	
15.9.3.1	Permissions	6.0 onwards	M	N/A	
15.9.3.2	APDU Logs	9.0 onwards	M	N/A	

Table 2.5: Applicability of tests

Conditional item	Condition
C001	VOID
C002	VOID
C003	VOID
C004	VOID
C005	IF (O_BAT_LOW) THEN M ELSE N/A
C006	IF (O_BAT_OFF) THEN M ELSE N/A
C007	VOID
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	VOID
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_DEFAULT_ROUTE_SELECTION_BY_USER_MENU OR O_REQ_143)) THEN M ELSE N/A
C020	IF (O_BUFFER_SIZE) THEN M ELSE N/A
C021	VOID
C022	IF (O_BEFORE_ANDROID_MARSHMALLOW) THEN M ELSE N/A

Conditional item	Condition
C023	IF (O_GSMA_API) THEN M ELSE N/A
C024	IF (O_MULTI_CEE_ON AND O_GSMA_API) THEN M ELSE N/A
C025	VOID
C026	VOID
C027	IF (O_MULTI_CEE_ON AND O_eSE) THEN M ELSE N/A
C028	IF (O_eSE) THEN M ELSE N/A
C029	IF (O_BEFORE_ANDROID_P) THEN M ELSE N/A
C030	IF (O_GSMA_API AND O_BEFORE_ANDROID_P) THEN M ELSE N/A
C031	IF (O_MULTI_CEE_ON AND NOT O_REQ_167.1) THEN M ELSE N/A

Table 2.6: Conditional items referenced by Table 2.5

2.1.6 Information to be provided by the Vendor

The Vendor shall provide information with respect to Device default configuration.

Item	Description	Value	Status
1	Preferred buffer size supported by the terminal for Open Channel command		C
2	The value of the Issuer Identification Number of the eSE as personalized in the ISD		C
3	The value of the Card Image Number of the eSE as personalized in the ISD		C
Note: Conditional values shall be provided if the corresponding option is supported in Table 4: Options			

Table 2.7: Device default configuration

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 application. If the DUT supports O_MULTI_CEE_ON and unless otherwise specified in the Test Case, the UICC shall stay accessible by declaring all required UICC AIDs in the “other” category of an OffHostService.

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, GlobalPlatform and NFC Forum). These organisations defined their own requirements for test benches, test applicability and pass criteria's.

The GSMA fully relies on these test specifications 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.

When determining the applicability of the test cases for the DUT in each of these external test specifications, those device options with GSMA Status set to M in the relevant sub-section of Annex B should be set to Supported in the device options in the external test specification.

2.2.2 VOID

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.

2.2.5 Test Cases "Direction"

Test cases includes a "Direction" column. Different test platform elements (mobile application, NFC tags, UICC,) are involved in the test cases execution. This information is provided to clarify the test platform elements between which a test step is performed.

These elements or "actors" used over this document are listed in the table below:

Actor	Description
DUT	Represents the Device Under Test according to the definition of Device provided in section 1.3
ME	Represents the Mobile Equipment as defined in section 1.3. This is a synonym for the DUT, used in certain test cases for consistency with external specifications.
User	Represents the User as defined in section 1.3
Tag	Represents an NFC Tag according to section 2.5.4 "Tag testing"

Actor	Description
PCD	Represents the contactless reader equipment. It follows requirements in section 2.5.6 “Reader equipment”
UICC	Represents the UICC as defined in section 2.5.1 of this test book
App	Represents the software application installed on the DUT to interact as the applicative level and check the capabilities of the DUT according to the Operating System
USS	Represents a system simulating the mobile network
Server	Represents the OTA server able to send data over the air. This should be part of the test environment defined in section 2.5.8

Table 2.9: Definition of Test Case “Direction”

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 to 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 follows:

The transaction always starts with putting DUT into reader RF field. Then the reader establishes the contactless connection with the DUT. Afterwards the following APDUs will be exchanged. For each command, the test tool shall check that the expected response is returned by the DUT.

Command	Expected response
Select by AID A0000005595000000000000052414441	SW: '90 00'
Select by File ID (5F00)	SW: '90 00'
Select by File ID (1F00)	SW: '90 00'
Read Binary	Response data: 128 bytes with value '00' SW: '90 00'
Update Binary (with 128 bytes with value 0xFF)	SW: '90 00'

Command	Expected response
Read Binary	Response data: 128 bytes with value 'FF' SW: '90 00'
Update Binary (with 128 bytes with value 0x00)	SW: '90 00'
External Authenticate	SW: '90 00'

Table 2.10: List of expected responses by the DUT

The transaction always ends with a DESELECT 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 case 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 provide description of the UICC behaviour, which can be either simulated or a 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/eUICC must be used. For most of the test sequences described in this document the UICC has an 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 a 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 are 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 ETSI TS 102 613 [9], TS 102 221 [8] and ETSI TS 102 622 [10]). The accuracy of the UICC simulator's settings shall be taken into account when ensuring this.

The UICC (simulated or real) 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 ETSI TS 102622 [10].
- Shall support card emulation in both full power mode and low power mode, as specified in ETSI TS 102 613 [9] and ETSI TS 102 622 [10] for Type A, Type B and Type F.
- 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 3GPP TS 31.124 [21]
- Shall provide all necessary information (Specification, ADM codes) to manage the card content and the file system

In addition to the above listed requirements the UICC simulator shall implement the following functionalities:

- Shall fulfil the requirements for SWP/HCI as specified in ETSI TS 102 694-1 [11] clause 4.4, and ETSI TS 102 695-1 [12] clause 4.4
- Shall fulfil the requirements for Remote Management of NFC Services and for Mobile Device APN as specified in 3GPP TS 31.121 [20] clause 4.1 and in 3GPP TS 31.124 [21] in 27.22.2A, 27.22.2B and 27.22.2C.
- Shall implement the behaviour for the device interface commands in the scope of the Secure Element Access Control related device tests (e.g.: GET_DATA [all], GET_DATA [specific] for ARA)

- For the case 4 APDU exchanges utilizing T=0 transmission protocol and originating from the Secure Element Access API when the UICC is required to return an R-APDU with response data and with SW='62 XX' or SW='63 XX' in response to a case 4 C-APDU, the UICC simulator shall be able to implement both the ISO and ETSI behaviour:

  - Behaviour recommended by ISO: send first a "61 XX" and then - after receiving GET RESPONSE command from the device - the data with the warning status word
  - Behaviour recommended by ETSI: send first SW warning instead of 61 XX and follow the procedure as described in Annex C of [8].

  Note: Unless otherwise specified the ISO behaviour is used.

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 Basic Remote Management of NFC Services (section 12.3) and Mobile Device APN Management (section 11) test execution, the test equipment shall fulfil the requirements specified in 3GPP TS 34.108 [25] clause 4. Network simulator set up for other sections is defined in the relevant chapters.

2.5.3 Common applications

The following applications are common to different test packages.

2.5.3.1 UICC Applications

- **ReferenceApplication.cap**: A UICC application according to the description in Annex A, which can be used to run the reference transaction. The source code of this application is available at:

  https://github.com/GSMATerminals/NFC-Test-Book-Public

- **APDU_TestApplication.cap**: Based on the ReferenceApplication.cap, this application allows managing different APDU answers. The application sends EVT_TRANSACTION on the EVT_FIELD_OFF event. The application implements the sequence used by the MobileApplication (defined in Chapter 2.5.3.2):
- On APDU Case 1 => 0x0001[P1]00

  - returns SW1-SW2

- On APDU Case 2 => 0x0002[P1]00[L_e]

  - returns [Data field L_e bytes long] only if SW1 = 0x62 or 0x63 or 0x90 + SW1-SW2

- On APDU Case 3 => 0x0003[P1]00[L_c][Data field L_c bytes long]

  - returns SW1-SW2

- On APDU Case 4 => 0x0004[P1] 00[L_c] [Data field L_c bytes long] [L_e]
    - returns [Data field L_e 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
0x00	0x9000
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

[P1]	SW1-SW2
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
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 2.11: Status Word

- **APDU_TestApplication_card_deactivated.cap:** a modified version of the APDU_TestApplication.cap. This application sends EVT_TRANSACTION only on the EVT_CARD_DEACTIVATED event.

2.5.3.2 Device Applications

- **MobileApplication:** A device application allowing the following access to the UICC:
  - Open Logical Channel via Select AID
    - SELECT_BY_DF_name on AID01
  - 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

MobileApplication is considered as launched if it is selected and started by the User.

On Android Devices supporting Multiple Card Environment the AIDs of the instances of ReferenceApplication.cap shall be registered to UICC with “Other” category for each test case where the ReferenceApplication.cap (or derivative) is used.

NOTE: The AID registration does not apply to test cases in section 15.7.

On Android Devices supporting Multiple Card Environment the AIDs of the instances of APDU_TestApplication.cap shall be registered to UICC with “other” category for each test case where the APDU_TestApplication.cap (or derivative) is used.

NOTE: The AID registration does not apply to test cases in section 15.7.

On Android Devices supporting Multiple Card Environment the AIDs of the instances of APDU_TestApplication_card_deactivated.cap shall be registered to UICC with “other” category for each test case where the APDU_TestApplication_card_deactivated.cap (or derivative) is used.

NOTE: The AID registration does not apply to test cases in section 15.7.

2.5.3.2.1 Android OS version

For devices based on Android version prior to Android 9 the relevant Device Application:

- shall use the SIMalliance Open Mobile API; and
- shall apply the TS.26 requirements for transaction events and permissions which are applicable for “before Android 9”.

For devices based on Android 9 or following Android releases the relevant Device Application:

- shall use “android.se.omapi” package. For details see [45]
- shall apply the TS.26 requirements for transaction events and permissions which are applicable for “Android 9 onwards”.

Unless stated otherwise it is allowed to use the same Device Application for both devices before Android 9 and devices based on Android 9 or following Android releases.

2.5.3.3 Other Applications

- **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.

2.5.3.4 Logically

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

AID_REF = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 41’

AID01 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31’

AID02 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32’

AID03 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 33’

2.5.3.5 eSE Applications

Note: these applications are mandatory only if the DUT supports O_eSE. See Annex F for the configuration of a device with eSE.

- **Applet1**: SE application returning “4f 46 46 48 4f 53 54” + SW90 00 for the SELECT by AID command. This application is available at:

  https://github.com/GSMATerminals/NFC-Test-Book-Public/

  Under eSE TestApplet/build
- **Applet2**: SE application returning “48 43 45” + SW90 00 for the SELECT by AID command. This application is available at:

  https://github.com/GSMATerminals/NFC-Test-Book-Public/

  Under eSE TestApplet/build
- **Applet3**: SE application returning “65 53 45” + SW90 00 for the SELECT by AID command. This application is available at:

  https://github.com/GSMATerminals/NFC-Test-Book-Public/

  Under eSE TestApplet/build

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 a physically available Tag provided by Tag 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 specified by the 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. If the device includes an indication to the user of the position of the NFC antenna (see TS26_NFC_REQ_107), the position as indicated to the user shall be used. Otherwise, this point shall be provided by the device manufacturer for testing purposes; 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 taking into account the form factor of the DUT.
- 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:

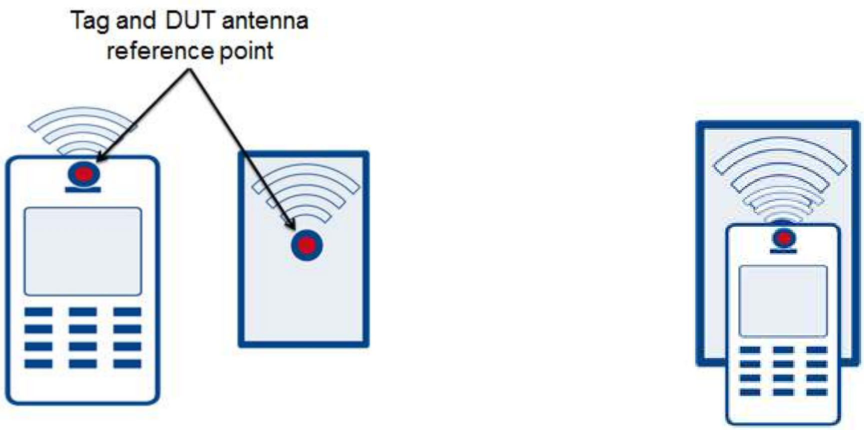


Figure 2.1: Tag and DUT antenna reference point

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

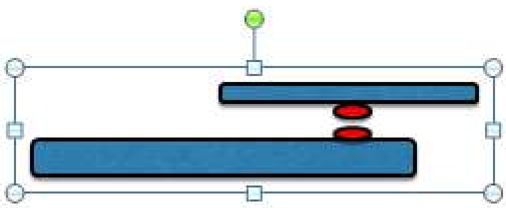


Figure 2.2: Antenna positioning

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 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

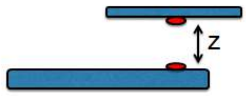


Figure 2.3: "z" distance

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 is z = 0,0cm between DUT and Tag antenna reference point.

2.5.4.3 Tag requirements

NFC Forum Type 2 Tag:

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

NFC Forum Type 3 Tag:

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

NFC Forum Type 4A Tag:

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

NFC Forum Type 4B Tag:

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

2.5.4.4 Tag Read/Write Applications

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

NFC Tag application: An external tag reader and writer with application for tag content read verification and for tag writing of reference tags. The tag reader/writer shall support NFC Forum Type 1-4 tags, as specified in NFC Forum Tag Operation Specifications [19].

NFC Tag mobile application: A mobile application based on the operating system standardized APIs for tag reading and writing. This application is typically provided by the device Vendor or by the test tool manufacturer.

Reference NFC Tags: A set of reference NFC tags (Type 1, 2, 3 and 4) as specified in Annex C.

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"	Type: "text/x-vCard" BEGIN:VCARD VERSION:2.1 N:Smith;John;;; FN:John Smith TEL;CELL: 332312345678 END:VCARD
"URI"	Type: "U" file://test
"Text"	Type: "T" Encoding: UTF-8 Lang: "en-US" "Hello, world!"
"SmartPoster" (launch browser)	Type: "Sp" Text Type: "T" Encoding: UTF-8 Lang: "en-US" Test: "GSMA Website" URI Type: "U" http://www.gsma.com
"SmartPoster" (SMS Sending)	Type: "Sp" URI Type: "U" sms:332312345678?body=Hello, world!

Reference	NFC Tag Content
“SmartPoster” (phone call)	Type: “Sp” Text Type: “T” Encoding: UTF-8 Lang: “en-US” Test: “John Smith” URI Type: “U” Tel: 442312345678
“SmartPoster” (email)	Type: “Sp” URI Type: “U” mailto:john.smith@gsma.com?subject=email subject&body=email content Text Type: “T” Encoding: UTF-8 Lang: “en-US” Test: “email title”

Table 2.12: NFC Tags content

NOTE 1: For NFC Type 2 Tag, these tag contents represent either static or dynamic memory layouts.

2.5.4.6 NFC Forum Analog Tests

Support of the GSMA Transport requirements using NFC protocol requires the inclusion of NFC Forum’s Analog Specification to ensure interoperability. References to the NFC Forum Digital Protocol and Activity Specifications are added for completeness as devices need to be compliant to all three technical specifications in order to support the transport testing requirements covered by the suite of NFC Forum Analog test cases.

2.5.5 Reader equipment

The contactless reader shall support the NFC Forum type A and B functionality.

2.5.6 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.

See section 2.5.3.2.1 for further requirements for Android applications for transaction events.

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 a future version of TS.26)

Unless otherwise specified, when a card emulation session is present within a test procedure, the test shall be carried out with Card emulation Type A as specified in [9] and [10].

2.5.7 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.

The communication network shall be LTE only as specified in 3GPP TS 36.508 [36] clause 4 or with 3G/2G fallback according to the capability of the DUT.

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.

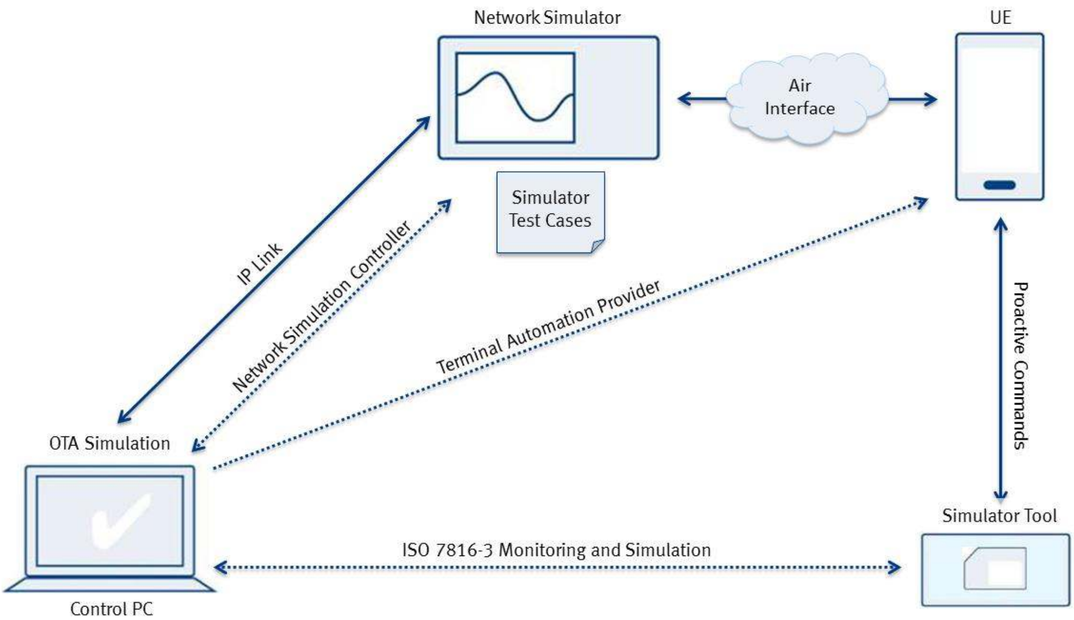


Figure 2.4: Test Environment

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.

2.5.8 Card emulation testing

2.5.8.1 Common positioning of Reader and Device

The provisions of section 2.5.4.1 apply with the tag and tag antenna reference point being replaced by the reader and reader antenna reference point.

2.5.8.2 Distance specific positioning

The provisions of section 2.5.4.2 apply with the tag and tag antenna reference point being replaced by the reader and reader antenna reference point and only with distances up to 2.0cm.

2.6 Common procedures

2.6.1 Setting the default AID route

This section applies only to devices which support O_MULTI_CEE_ON

Various test cases indicate that the default AID route should be set to HCE or to UICC. This section addresses how to achieve that condition for devices with different attributes according to the following logic:

If the DUT supports O_DEFAULT_ROUTE_SELECTION_BY_USER_MENU then the default AID route shall be set using the user menu.

If the DUT supports O REQ143 then the default AID route shall be set using the procedures defined in Sections 2.6.1.1, 2.6.1.2.

If the DUT uses Android OS and supports O_MULTI_CEE_ON, but does not support either O_DEFAULT_ROUTE_SELECTION_BY USER_MENU or O_REQ_143 then the default AID route is expected to be HCE by default.

2.6.1.1 Procedure to ensure the default AID route is HCE with O_REQ_143

The aim of this procedure is to provide a method in order to ensure that the default AID route on the DUT is set to HCE.

This procedure is intended to be executed as part of a referencing test case.

When this procedure has been successfully completed, Dynamic Other Host will be installed, 255 AIDs (TestAIDHCE xx) will be registered, and the default route will be set to HCE.

Note: This procedure shall be run even if the default AID route of the device is already HCE, in order to fill up the routing table with AIDs.

Initial conditions:

- DUT is powered ON and device is unlocked and the screen ON

Applications needed:

- Dynamic_Other_Host: An application able to register a configurable list non-payment AID on the HOST (HCE) using the dynamic registration API.
- AIDs generated by the application SHALL be AIDs of 16 byte matching the following template:
- The 1st byte of the TestAIDHCE xx shall be increased by one bit for each consecutive AID starting from 0x01.
- The 16th byte of the TestAIDHCE xx shall be set to “0x01”
- The other bytes of the AID shall be set according to the table below:

AID byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
value	0x01 for TestAIDHCE01 0x02 for TestAIDHCE02 … 0x64 for TestAIDHCE100 ….. 0xFF for TestAIDHCE255	0x 02	0x 03	0x 04	0x 05	0x 06	0x 07	0x 08	0x 09	0x 10	0x 11	0x 12	0x 13	0x 14	0x 15	0x 01

Step	Direction	Sequence	Expected Result
1	User →DUT	Install Application Dynamic_Other_Host	Installation successful
2	User →DUT	Use Dynamic_Other_Host and register 255 Host AIDs	No error while registering the AIDs

Table 2.13: Procedure to ensure the default AID route is HCE

2.6.1.2 Procedure to ensure the default AID route is UICC with O_REQ_143

The aim of this procedure is to provide a method in order to ensure that the default AID route on the DUT is set to UICC.

This procedure is intended to be executed as part of a referencing test case.

When this procedure has been successfully completed, Dynamic_Other_OffHost will be installed, 255 AIDs (TestAIDUICC xx) will be registered, and the default route will be set to UICC.

Note: This procedure shall be run even if the default AID route of the device is already UICC, in order to fill up the routing table with AIDs.

Initial conditions:

- DUT is powered ON and device is unlocked and the screen is ON

Applications needed:

Dynamic_Other_OffHost: An application able to register a configurable list of non-payment AID on the OffHost UICC using the registerAIDsForService() method of Android API. It defines an “OffHost” other service in its Manifest.

- AIDs generated by the application SHALL be AIDs of 16 byte matching the following template:
- The 1st byte of the TestAIDUICC xx shall be increased by one bit for each consecutive AID starting from 0x01.
- The 16th byte of the TestAIDUICC xx shall be set to “0x02”
- The other bytes of the AID shall be set according to the table below:

AID byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
value	0x01 for TestAIDUICC01 0x02 for TestAIDUICC02 … 0x64 for TestAIDUICC100 ….. 0xFF for TestAIDUICC255	0x 02	0x 03	0x 04	0x 05	0x 06	0x 07	0x 08	0x 09	0x 10	0x 11	0x 12	0x 13	0x 14	0x 15	0x 02

Step	Direction	Sequence	Expected Result
1	User →DUT	Install Application Dynamic_Other_OffHost	Installation successful
2	User →DUT	Use Dynamic_Other_OffHost and register 255 OffHost AIDs using the registerAIDsForService method	registerAidsForService method returns a boolean for success

Table 2.14: Procedure to ensure the default AID route is UICC

2.6.2 Procedure to identify the size of the AID routing table of a DUT

The purpose of this procedure is to provide a method in order to know the maximum number of 16 bytes AID that can be inserted in the AID routing table of a DUT before reaching an AID routing overflow.

This procedure implies that TS26_NFC_REQ_143 is implemented on the DUT in order to work. So it is advised to ensure this requirement is implemented before applying the procedure.

This method is applicable at any time on the device as long as initial conditions are met.

This procedure is intended to be executed independently of any test case.

Initial conditions:

- The UICC contains a cardlet with a known AID [referred as AID01].
- AID01 is different from the AIDs generated by any application installed and only available on the UICC
- AID01 is not registered to the Host using a device application (neither in manifest nor dynamically)
- Device is powered ON and device is unlocked and the screen is on
- All NFC applications on the device are uninstalled except applications that are preinstalled

The following three initial conditions need to be executed in this order:

- Set the default AID route to HCE (See section 2.6.1.1)
- Unregister all AIDs
- Run the procedure 2.6.4 to determine if the UICC is accessible:
    - If the UICC is accessible the run 2.6.2.1
    - Otherwise run 2.6.2.2

2.6.2.1 Default AID Route is UICC after unregistering of all AIDs

Step	Direction	Sequence	Expected Result
1	User →DUT	Define a counter N Define a counter S (Start)=1 Define a counter E (End)=255	
2	User →DUT	Install Application Dynamic_Other_Host	Installation successful
3	App→DUT	Use “Dynamic_Other_Host” to register E AIDs	
4	User → DUT	Run the procedure 2.6.4 on AID01	If the UICC is accessible • Exit this procedure and consider the calling test case as failed
5	App→DUT	Unregister all AIDs. Set back the default AID route to UICC (see section 2.6.1.2) Unregister all AIDs	The default route is UICC
6	App→DUT	Set N=[(S+E)/2] (only the integer value is to be taken into account) and Use "Dynamic_Other_Host" to register N AIDs	

Step	Direction	Sequence	Expected Result

7	User→ DUT	Run the procedure 2.6.4 on AID01	IF the UICC is accessible then
			• set S=N+1
			• Go to step 9
			ELSE
			• set E=N-1

8	App→DUT	Unregister all AIDs. Set back the default AID route to UICC (see section 2.6.1.2) Unregister all AIDs	The default route is UICC
9	Loop	Repeat steps 6 to 8 while S<E	
10	App→DUT	Unregister all AIDs. Use "Dynamic_Other_Host" to register S AIDs	

11	User →DUT	Run the procedure 2.6.4 on AID01	IF the UICC is accessible then
			• RTS = S
			ELSE
			• RTS = S-1

12	User →DUT	Uninstall “Dynamic_Other_Host” application	

Table 2.15: Procedure to identify the size of the AID routing table of a DUT when initial Default Route is UICC

2.6.2.2 Default Route is HCE after unregistering of all AIDs

Applications needed:

- Dynamic_Other_OffHost: as described in 2.6.1.2.

Step	Direction	Sequence	Expected Result
1	User →DUT	Define a counter N, Define a counter S (Start)=1, Define a counter E (End)=255.	
2	User →DUT	Install Application Dynamic_Other_OffHost	Installation successful
3	App→DUT	use “Dynamic_Other_OffHost” to register E AIDs using registerAidsForService method	registerAidsForService method returns a boolean for success
4	User → DUT	Run the procedure 2.6.4 on AID01	If the UICC is not accessible • Exit this procedure and consider the calling test case as failed

Step	Direction	Sequence	Expected Result
5	App→DUT	Unregister all AIDs, set back the default AID route to HCE (see section 2.6.1.1). Unregister all AIDs	The default route is HCE
6	App→DUT	Set N=[(S+E)/2] (only the integer value is to be taken into account), unregister all AIDs, use "Dynamic_Other_OffHost" to register N AIDs using registerAidsForService method	registerAidsForService method returns a boolean for success

7	User→DUT	Run the procedure 2.6.4 on AID01	IF the UICC is not accessible then
			• set S=N+1
			• Go to step 9
			ELSE
			• set E=N-1

8	App→DUT	Unregister all AIDs, set back the default AID route to HCE (see section 2.6.1.1). Unregister all AIDs	The default route is HCE
9	Loop	Repeat steps 6 to 8 while S<E	
10	App→DUT	Unregister all AIDs, use "Dynamic_Other_OffHost" to register S AIDs using registerAidsForService method	registerAidsForService method returns a boolean for success

11	User →DUT	Run the procedure 2.6.4 on AID01	IF the UICC is not accessible then
			• RTS = S
			ELSE
			• RTS = S-1

12	User →DUT	Uninstall “Dynamic_Other_OffHost” application	

Table 2.16: Procedure to identify the size of the AID routing table of a DUT when initial Default Route is HCE

NOTE: RTS = the number of 16 bytes AIDs that can be contained in the NFC AID Routing table of the DUT

2.6.3 Procedure to send a transaction event

Various test cases require the sending of a transaction event (EVT Transaction). Depending on the approach and for sake of clarity, sending a transaction event is considered as a single test step. Nevertheless, each time this step applies in a TC, the following procedure must be executed.

This procedure is intended to be executed as part of a referencing test case.

Direction	Sequence	Expected Result
PCD	Power on RF field	
PCD → DUT	Perform RF protocol initialisation	
PCD → DUT	Using the **APDU application**, send a SELECT command with *[AIDxx]*	**APDU Application** receives Status Word 90 00
PCD	Power off RF field	
DUT → UICC	Send EVT_FIELD_OFF	
	The **card application** sends EVT_TRANSACTION to the **mobile application**	*[Expected result]*

Table 2.16: Procedure to send transaction event

- *[AIDxx]* has to be replaced by the AID from the step calling this procedure.
- *[Expected result]* is the expected result detailed in the test case as expected result of the step calling this procedure.
- On Android Devices supporting Multiple Card Emulation Environment the *[AIDxx]* needs to be registered to the UICC with “other” category so that the event transaction procedure can be successfully performed. Note: This AID registration does not apply to test cases in section 15.7.

2.6.4 Procedure to check if the UICC is accessible

Various test cases require the sending of a select command to check that the UICC is accessible on the contactless interface.
For sake of clarity, this check is considered as a single test step. Nevertheless, each time this step applies in a TC, the following procedure must be executed using a specific AID parameter.

This procedure is intended to be executed as part of a referencing TC.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	

3	PCD → DUT	Send “SELECT APDU” command with *[AIDxx]* as parameter	IF SW = 9000 then
			• The UICC is accessible
			ELSE
			• The UICC is not accessible

4	User → PCD	Power off the field	

Table 2.17: Procedure to check if the UICC is accessible

- *[AIDxx]* has to be replaced by the AID from the step calling this procedure.
- The UICC contains a cardlet with a known AID referred as *[AIDxx]*.

[AIDxx] is not available on any host service.

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 Read/Write Mode section, Card Emulation Mode testing and NFC core functions including the SWP/HCI testing.

3.2 Conformance requirements

The Requirements tested are referenced in each test case.

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 testing on application level in sections 3.3.3.1 – 3.3.3.8 and testing lower level functionality in section 3.3.3.24. A limited set of distances between device and NFC Tag is covered in section 3.3.3.9 – 3.3.3.13. Reading performance and general reader mode testing are covered in sections 3.3.3.14 – 3.3.3.23.

3.3.2 Conformance requirements

The Requirements tested are referenced in each test case.

3.3.3 Test Cases

3.3.3.1 NFC Forum Type 1 Tag – Read NFC Tag

For further test details, see TS.27 v13.0 [47]

3.3.3.1.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

3.3.3.2 NFC Forum Type 2 Tag – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Type 2 Tag 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.

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The following tag content should be configured to perform the test:
  - NFC Type 2 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).

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 → App	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

3.3.3.3 NFC Forum Type 3 Tag – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Type 3 Tag 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.

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 Type 3 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).

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 → App	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 Type 4 Tag – Read NFC Tag

Test Purpose

To ensure the DUT allows reading of NFC Forum Type 4A Tag 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.

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- 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).

3.3.3.4.1 Test Sequence No 1: Type 4A Tag

Initial Conditions

The tag content should be configured as below:

NFC Type 4A Tag - NFC Tag is personalized with a “SmartPoster” (phone call)

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 → App	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

3.3.3.4.2 Test Sequence No 2: Type 4B Tag

Initial Conditions

The tag content should be configured as below:

NFC Type 4B Tag - NFC Tag is personalized with a “SmartPoster” (email)

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 → App	The DUT to read the tag content	The tag content is correctly received by the application.

Step	Direction	Sequence	Expected Result
3	User → DUT	Remove the DUT from the read range	None

3.3.3.5 NFC Forum Type 1 Tag – Write NFC Tag

For further test details, see TS.27 v13.0 [47]

3.3.3.5.1 Test Sequence No 1: Dynamic

For further test details, see TS.27 v13.0 [47]

3.3.3.5.2 Test Sequence No 2: Static

For further test details, see TS.27 v13.0 [47]

3.3.3.6 NFC Forum Type 2 Tag – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Type 2 Tag 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 the 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 tag contents shall be configured to perform the test as following:
    - Initial conditions for Test Sequence No #1: Type 2 Tag empty is initialized with Dynamic memory layout
    - Initial conditions for Test Sequence No #2: Type 2 Tag empty is initialized with Static memory layout
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

3.3.3.6.1 Test Sequence No 1: Dynamic

Initial Conditions

Write the following tag content:

NFC Type 2 Tag - NFC Tag will be personalized with a “SmartPoster” (SMS)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to write to NFC Type 2 Tag	The DUT writes to NFC Type 2 Tag.
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: Static

Initial Conditions

Write the following tag content:

NFC Type 2 Tag - NFC Tag will be personalized with a “SmartPoster” (SMS)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to write to NFC Type 2 Tag	The DUT writes to NFC Type 2 Tag.
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 Forum Type 3 Tag – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Type 3 Tag 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

Initial Conditions

- The DUT is powered on
- NFC is enabled in the DUT
- The Tag should be in initialized state and shall not bear any NDEF message
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

3.3.3.7.1 Test Sequence No 1

Initial Conditions

Write the following tag content:

NFC Type 3 Tag - 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	App → DUT DUT → Tag	Use the application to write to NFC Type 3 Tag	The DUT writes to NFC Type 3 Tag.
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 Forum Type 4 Tag – Write NFC Tag

Test Purpose

To ensure the DUT allows writing of NFC Forum Type 4A Tag 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 No 1: Empty initialized Type 4A Tag
    - Initial conditions for Test Sequence No 2: Empty initialized Type 4B Tag
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

3.3.3.8.1 Test Sequence No 1: Type 4A Tag

Initial Conditions

Write the following tag content:

For NFC Type 4A Tag - NFC Tag is blank and will be personalized with a “SmartPoster” (Browser)

	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to write to NFC Type 4A Tag.	The DUT writes to NFC Type 4A Tag.
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: Type 4B Tag

Initial Conditions

Write the following tag content:

For NFC Type 4B Tag - NFC Tag is blank and will be personalized with a "SmartPoster" (Phone Call)

Step	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to write to NFC Type 4B Tag	The DUT writes to NFC Type 4B Tag.
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

3.3.3.9 Distance for NFC Type 1 Tag reading

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

3.3.3.10 Distance for NFC Type 2 Tag reading

Test Purpose

This test case verifies the correct interpretation of NFC Type 2 Tag 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

Initial Conditions

- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 2 with RTD “SmartPoster” (launch browser) is available

3.3.3.10.1 Test Sequence No 1: Distance for NFC Type 2 Tag 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
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 Type 2 Tag 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 Type 2 Tag 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
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 Type 2 Tag 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 Type 2 Tag 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

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

3.3.3.10.6 Test Sequence No 6: Distance for NFC Type 2 Tag 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 Type 3 Tag reading

Test Purpose

This test case verifies the correct interpretation of NFC Type 3 Tag 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

Initial Conditions

- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 3 with RTD “SmartPoster” (launch browser) is available

3.3.3.11.1 Test Sequence No 1: Distance for NFC Type 3 Tag 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

3.3.3.11.2 Test Sequence No 2: Distance for NFC Type 3 Tag 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 Type 3 Tag 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
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 Type 3 Tag 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 Type 3 Tag 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 Type 3 Tag 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

Step	Direction	Sequence	Expected Result
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 Type 4A Tag reading

Test Purpose

This test case verifies the correct interpretation of NFC Type 4A Tag 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

Initial Conditions

- Antenna reference point may be marked on the outside of the DUT
- NFC Type 4A Tag with RTD “SmartPoster” (launch browser) is available

3.3.3.12.1 Test Sequence No 1: Distance for NFC Type 4A Tag 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
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 Type 4A Tag 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 Type 4A Tag 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
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 Type 4A Tag 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

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

3.3.3.12.5 Test Sequence No 5: Distance for NFC Type 4A Tag 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 Type 4A Tag 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 Type 4B Tag reading

Test Purpose

This test case verifies the correct interpretation of NFC Type 4B Tag 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

Initial Conditions

- Antenna reference point may be marked on the outside of the DUT
- NFC Tags Type 4B with RTD “SmartPoster” (launch browser) is available

3.3.3.13.1 Test Sequence No 1: Distance for NFC Type 4B Tag 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
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 Type 4B Tag 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 Type 4B Tag 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
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 Type 4B Tag 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 Type 4B Tag 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

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

3.3.3.13.6 Test Sequence No 6: Distance for NFC Type 4B Tag 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 Type 1 Tag reading performance

For further test details, see TS.27 v13.0 [47]

3.3.3.14.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

3.3.3.15 NFC Type 2 Tag reading performance

Test Purpose

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

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.

3.3.3.15.1 Test Sequence No 1

Initial Conditions

NFC Type 2 Tag is personalized with RTD "SmartPoster" (launch browser)

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

3.3.3.16 NFC Type 3 Tag reading performance

Test Purpose

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

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.

3.3.3.16.1 Test Sequence No 1

Initial Conditions

NFC Type 3 Tag is personalized with RTD "SmartPoster" (launch browser)

Step	Direction	Sequence	Expected Result
1		Start the RF spy	None
2		Read a NFC Type 3 Tag	NFC Tag content is read

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

3.3.3.17 NFC Type 4A Tag reading performance

Test Purpose

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

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.

3.3.3.17.1 Test Sequence No 1

Initial Conditions

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

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

3.3.3.18 NFC Type 4B Tag reading performance

Test Purpose

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

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.

3.3.3.18.1 Test Sequence No 1

Initial Conditions

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

Step	Direction	Sequence	Expected Result
1		Start the RF spy	None
2		Read a NFC Type 4B Tag	NFC Tag content is read
3		As soon as the DUT prompts the end user, stop 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_035

Initial Conditions

- NFC Forum Type 2 Tag with content as described in Section 2.5.4.5 is available for testing (i.e. vCard, URI or Text).

Set up a network simulator for supported network technology as defined in chapter 2.5.8.

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

Immediate link establishment,

Bearer Type 03 (Default Bearer for requested transport layer)

No alpha identifier

- Test data with a size of 60k Bytes to induce OTA Load duration in CAT-TP

- Also, the DUT with a test phone number which can be called and permits to maintain the call for several minutes is necessary.
- Prior to this test the device shall have been powered ON and ISO7816 initialization has been completed.
  - Tests shall be made based on the capability of the DUT (Example: For LTE device, test shall use LTE; otherwise, use 3G).

3.3.3.19.1 Test Sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Perform Push SMS procedure as defined in section 12.4.3.7.1	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1 The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of 0xFF)	
5	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the start of the expected data from the server. 91 XX
6		Read a NFC tag	NFC Tag is read
7		Repeat steps 8 to 9 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD – Data Available commands may be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	

Step	Direction	Sequence	Expected Result
8	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
9	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the remainder of the expected data from the server. 91 XX
10	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
11	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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.

3.3.3.20 VOID

3.3.3.21 VOID

3.3.3.22 VOID

3.3.3.23 VOID

3.3.3.24 NFC Forum Tag Operation Test Cases

This chapter addresses the inclusion of selected NFC Forum Tag test cases to ensure a device is able to Read and Write to any of the Tags called out in TS.27. Incorporation of these additional NFC Forum test cases improves the depth of coverage for test cases involving reading and writing to tags, checking for supported payload with tags, error redundancy with tags, timing parameters and that a device may work with future Tags used in the current test scenarios.

Test Purpose

To ensure the DUT follows the NFC Forum Specifications for reading and writing to any of the required Tag types.

These tests should be performed prior to Test Cases 3.3.3.1 through 3.3.3.19, which test only the application level of a device’s read and write operation.

Referenced requirement

- TS26_NFC_REQ_035
- TS26_NFC_REQ_036
- TS26_NFC_REQ_037

Related Specs/Docs:

NFCForum Test Cases For Type 2 Tag and Type 2 Tag Operation [46]
NFCForum Test Cases For Type 3 Tag and Type 3 Tag Operation [46]
NFCForum Test Cases For Type 4 Tag and Type 4 Tag Operation [46]

Test Procedure

The DUT shall pass the Test Cases with ID REQ from the NFC Forum related specs/docs above. The set of applicable test cases is referenced in Table B.9.1, Table B.9.2, Table B.9.3, and Table B.9.4.

3.3.3.25 NFC Forum Test Cases for Analog (all valid versions)

This chapter addresses the inclusion of NFC Forum Test Cases for Analog. Incorporation of the NFC Forum Analog test cases establishes an appropriate test coverage for NFC-A, NFC-B and NFC-F technologies in polling and listening modes. The associated test cases cover Test Cases for Analog test specification versions.

Referenced requirement

- TS26_NFC_REQ_042

Related Specs/Docs:

- NFCForum-TS-Analog [19]
- NFC Forum Test Cases for Analog [46]
- NFC Forum Devices Requirements [46]

Test Procedure

The DUT shall pass the Test Cases with ID REQ from the NFC Forum related specs/docs above. The set of applicable test cases is referenced in Table B.9.5.

3.3.3.26 VOID

3.3.3.27 NFC Forum Test Cases for Analog V2.0 only

This chapter addresses the inclusion of the specific NFC Forum Test Cases for Analog V2.0. With this version of the Analog Test Cases, interoperability of NFC mobile devices with transport fare management infrastructures according to ISO/IEC14443 and ISO/IEC18092 will be supported.

Referenced requirement

- TS26_NFC_REQ_042

Related Specs/Docs:

- NFC Forum-TS-Analog [19]
- NFC Forum Test Cases for Analog [46]
- NFC Forum Devices Requirements [46]

Test Procedure

The DUT shall pass the Test Cases with ID REQ from the NFC Forum related specs/docs above. The set of applicable test cases is referenced in Table B.9.7.

3.3.3.28 Extended Length APDU handling

Test Purpose

To ensure the DUT allows writing and reading of NFC Forum Type 4A Tag resulting in communication using extended length APDUs.

This test only test the Tag to 2048 bytes as there are currently no commercial Tags available which support 32767 bytes.

Referenced requirement

- TS26_NFC_REQ_160

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 Tag is an empty initialized Type 4A Tag
- The DUT is not placed in the Write Range (more than 50cm from the Tag).

3.3.3.28.1 Test Sequence No 1: Write 2048 bytes to Type 4A Tag

Initial Conditions

Write the following tag content:

For NFC Type 4A Tag - NFC Tag is blank and will be personalized with a payload resulting in an APDU containing 2048 command data payload.

	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to write to NFC Type 4A Tag using an extended length APDU	The DUT writes to NFC Type 4A Tag.
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

3.3.3.28.2 Test Sequence No 2: Read 2048 bytes from Type 4A Tag

Initial Conditions

Write the following tag content:

For NFC Type 4A Tag - NFC Tag contains a NDEF with a payload resulting in an APDU containing 2048 response data.

	Direction	Sequence	Expected Result
1	User → DUT	Place DUT in the NFC write range	
2	App → DUT DUT → Tag	Use the application to read from NFC Type 4A Tag using an extended length APDU	The DUT reads from NFC Type 4A Tag.
3	User → DUT	Remove the DUT from the write range	None
4	User	Verify that tag content was read correctly by comparing it to the personalization	The tag content is correctly read by the DUT.

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 as under different battery modes and distances.

3.4.2 Conformance requirements

The Requirements tested are referenced in each test case.

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

Initial Conditions

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

3.4.3.1.1 Test Sequence No 1: Card emulation available after boot

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, place the DUT in the area where the field will be powered on.	None
4	User → PCD	Power on the field	None

Step	Direction	Sequence	Expected Result
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: Card emulation available after reboot

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
6	User → DUT	While the field is off, place the DUT 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

3.4.3.1.3 Test Sequence No 3: Card emulation when device is on but in screen locked

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	(if not enabled) Enable the NFC on the DUT	None
3	User-→DUT	Lock the screen of the device and ensure that the screen is on	lockscreen is shown on the DUT
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	None
5	User → PCD	Power on the field	None
6	PCD → DUT DUT → UICC	Perform the reference transaction using a contactless reader	The reference transaction is performed successfully

3.4.3.1.4 Test Sequence No 4: Card emulation when device on but screen off

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	(if not enabled) Enable the NFC on the DUT	None
3	User-→DUT	Switch off the screen of the DUT	The screen is switched off.
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	None
5	User → PCD	Power on the field	None
6	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 powered off due to a low battery level.

DUT SHALL accept 15 correct reference transactions.

Referenced requirement

- TS26_NFC_REQ_020

Initial Conditions

- **ReferenceApplication.cap** managing the reference transaction with AID_REF selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.
- NFC is enabled on the DUT

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 in Battery Low mode

Test Purpose

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

Referenced requirement

- TS26_NFC_REQ_021

Initial Conditions

- **ReferenceApplication.cap** managing the reference transaction with AID_REF selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.
- NFC is enabled on the DUT

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

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

Initial Conditions

- The DUT is in Battery Low 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

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

Initial Conditions

- The DUT 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
- TS26_NFC_REQ_157

Initial Conditions

None

Test 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)

For further test details, see TS.27 v13.0 [47]

-

3.4.3.5.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

-

3.4.3.6.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

-

3.4.3.7.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

-

3.4.3.8.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

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

For further test details, see TS.27 v13.0 [47]

3.4.3.9.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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 successfully

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

Step	Direction	Sequence	Expected Result
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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- While the field is off, the DUT is set to 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.

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

3.4.3.20 Card emulation with switched off device

Test Purpose

To ensure that card emulation mode is working when the device is switched off

Referenced requirement

- TS26_NFC_REQ_020
- TS26_NFC_REQ_174

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 AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.

3.4.3.20.1 Test Sequence No 1: Distance 0 cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power off the device	the device is powered off
2	User → DUT	While the field is off, place the DUT at 0cm of area where the field will be powered on.	None
3	User → PCD	Power on the field	None
4	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully
5	PCD → DUT DUT → UICC	Repeat Step 4 2 times	**Reference transaction** is performed successfully 2 times

3.4.3.20.2 Test Sequence No 2: Distance 0.5 cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power off the device	the device is powered off
2	User → DUT	While the field is off, place the DUT at 0.5cm of area where the field will be powered on.	None
3	User → PCD	Power on the field	None
4	PCD → DUT DUT → UICC	Execute the reference transaction	Reference transaction is performed successfully
5	PCD → DUT DUT → UICC	Repeat Step 4 2 times	Reference transaction is performed successfully 2 times

3.4.3.20.3 Test Sequence No 3: Distance 1 cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power off the device	the device is powered off
2	User → DUT	While the field is off, place the DUT at 1cm of area where the field will be powered on.	None
3	User → PCD	Power on the field	None
4	PCD → DUT DUT → UICC	Execute the reference transaction	Reference transaction is performed successfully
5	PCD → DUT DUT → UICC	Repeat Step 4 2 times	Reference transaction is performed successfully 2 times

3.4.3.20.4 Test Sequence No 4: Distance 1.5 cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power off the device	the device is powered off
2	User → DUT	While the field is off, place the DUT at 1.5cm of area where the field will be powered on.	None
3	User → PCD	Power on the field	None
4	PCD → DUT DUT → UICC	Execute the reference transaction	Reference transaction is performed successfully
5	PCD → DUT DUT → UICC	Repeat Step 4 2 times	Reference transaction is performed successfully 2 times

3.4.3.20.5 Test Sequence No 5: Distance 2 cm

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → PCD	Power off the device	the device is powered off
2	User → DUT	While the field is off, place the DUT at 2cm of area where the field will be powered on.	None
3	User → PCD	Power on the field	None
4	PCD → DUT DUT → UICC	Execute the reference transaction	Reference transaction is performed successfully
5	PCD → DUT DUT → UICC	Repeat Step 4 2 times	Reference transaction is performed successfully 2 times

3.4.3.21 Extended Length APDU handling

Test Purpose

To ensure correct handling of extended length encoded APDUs when working in card emulation mode.

Referenced requirement

- TS26_NFC_REQ_158

Initial Conditions

- An instance of the UICC application **APDU_TestApplication.cap** with AID01 is selectable.
- The **APDU application** defined in 2.5.3.3 is used to send APDU commands.
- In the NFC Controller the default AID route is set to UICC (see section 2.6.1)
- NFC is enabled in the DUT
- Card emulation is enabled in the DUT.
- The UICC used for testing SHALL support extended length APDU.

3.4.3.21.1 Test Sequence No 1: Get Response APDU with 2048 byte data field (Case 2)

Step	Direction	Sequence	Expected Result
1	User -> DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User -> PCD	Power on the field	
3	PCD -> DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
4	PCD -> DUT	**APDU application** sends an extended case 2 command with L_e = 0x0800	**APDU Application** receives data field containing 2048 bytes and a SW:9000

3.4.3.21.2 Test Sequence No 2: Send Command APDU with 2048 byte data field (Case 4)

Step	Direction	Sequence	Expected Result
1	User -> DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User -> PCD	Power on the field	
3	PCD -> DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
4	PCD -> DUT	**APDU application** sends an extended case 4 command with L_c = 0x000800 and L_e = 0x0800 and 2048 bytes of command data	**APDU Application** receives data field containing 2048 bytes and a SW:9000

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.

3.5.2 Conformance requirements

The Requirements tested are referenced in each test case.

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_009.1
- TS26_NFC_REQ_010
- TS26_NFC_REQ_011
- TS26_NFC_REQ_014
- TS26_NFC_REQ_015

Method of Test

Related Specs/Docs: ETSI TS 102.613 |9]

Test Procedure

The DUT shall pass all applicable test cases referenced in Table B.4.2 and Table B.4.3.

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

Related Specs/Docs: ETSI TS 102 622 [10]

Test Procedure

The DUT shall pass all applicable test cases referenced in Table B.5.2 and Table B.5.3.

3.5.3.3 SWP Stress test

Test Purpose

To ensure the DUT manages 100 transactions consecutively

Referenced requirement

- TS26_NFC_REQ_006

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 AID_REF selectable into the reference UICC.
- **APDU Application** to send APDUs according to the reference transaction.

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, place the DUT 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 continuously switch between card emulation mode and reader emulation mode.

Referenced requirement

- TS26_NFC_REQ_041

Initial Conditions

- The DUT is on
- HCI initialization has been correctly performed

- UICC application with AID01 selectable
- A Type 2 Tag with the RTD "Text" content
- The Tag and the reader are separated by at least 50cm
- The NFC is enabled

3.5.3.4.1 Test Sequence No 1

Initial Conditions

Backlight is on. DUT 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 AID01	**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 AID01	**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 AID01	**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 AID01	**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 AID01	**APDU application** receives Status word 90 00

3.5.3.5 RF Analog Protocol compliance

Test Purpose

To ensure that a mobile device is compliant with NFCForum-TS-Analog [19] specifications for card and reader emulation modes.

Referenced requirement

- TS26_NFC_REQ_025

- TS26_NFC_REQ_033

Related Specs/Docs: NFC Forum-TS-Analog [19]

Test Procedure

The DUT shall pass all the test cases referenced in Table B.9.2.1, Table B.9.2.2 and Table B.9.2.3.

3.5.3.6 VOID

3.5.3.7 RF Digital Protocol compliance

Test Purpose

To ensure that a mobile device is compliant with NFCForum-TS-Digital Protocol [19] and NFCForum TS Activity [19] specifications for card and reader emulation modes.

Referenced requirement

- TS26_NFC_REQ_025
- TS26_NFC_REQ_033

Related Specs/Docs: NFC Forum-TS-Digital Protocol [19]; NFC Forum Activity [19]

Test Procedure

The RF Digital Protocol compliance is tested by the test cases referenced in Annex B.9.3.

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 Secure Element Access Control [7] standard. It will grant or refuse the communication to/from applets stored in the UICC SE.

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

5.2 Conformance requirements

The Requirements tested are referenced in each test case.

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 AID01 and AID02 are selectable.

For that purpose, **MobileApplication** is registered for EVT_TRANSACTION handling from AID01 and AID02 and implements the functions “Select AID01” and “Select AID02” 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

For devices based on Android version prior to Android 9 GSMA_AC_Mobile_App_SP1_signed is installed first.

For devices based on Android 9 or following Android releases TS26_NFC_REQ_150 and TS26_NFC_REQ_150.1 are not applicable. It results that the installation order is not considered by the device when mobile applications are triggered. The Test Tool shall not check the triggering order. In test cases 5.3.1.2 and 5.3.1.3 the EVT TRANSACTION triggers both mobile applications. In step 15 the test tool shall close both mobile applications triggered after the first EVT TRANSACTION .

Note1: 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: Power off RF field 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 GlobalPlatform Specification GPD_SE_Access_Control for:

Secure Element Access API
NFC Event

Referenced requirement

- TS26_NFC_REQ_082
- TS26_NFC_REQ_083
- TS26_NFC_REQ_084
- TS26_NFC_REQ_150.1
- TS26_NFC_REQ_152
- TS26_NFC_REQ_152.2

5.3.1.1 Test Sequence No 1: Single app access to all AIDs

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 SP1 hash condition

  ⇨ SP1 has full access to all AIDs

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.16, 5.4.29.2 and 5.4.27.1 provide test steps that are now similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	SELECT command is successful and call to "Select AID01" function returns successfully
3		Call "Select AID02" function	SELECT command is successful and call to "Select AID02" function returns successfully

Step	Direction	Sequence	Expected Result
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.2 Test Sequence No 2: All apps access to single AID

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 AID01 and a path for one ACCF. The ACCF is present and contains no hash condition (access allowed for mobile apps)

  ⇨ AID01 is always accessible, no access allowed for any other AID

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.10 provides test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is successful
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	No application is triggered

5.3.1.3 Test Sequence No 3: All apps access to all AIDs

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 AIDs

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.22 provides test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID01" function	Call is successful
7		Call "Select AID02" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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	

Step	Direction	Sequence	Expected Result
16		repeat steps 9 to 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.4 Test Sequence No 4: Single app access to single AID

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 AID01 and a path for one ACCF containing SP1 hash condition

⇨ only access to AID01 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.2 and 5.4.27.1 provide test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	

Step	Direction	Sequence	Expected Result
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	No application is triggered

5.3.1.5 Test Sequence No 5: Multiple apps access to single AID, single ACCF, installation order

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 AID01 and a path for one ACCF containing SP1 hash condition
    - one specific target rule for AID01 and a path for one ACCF containing SP2 hash condition

⇨ only access to AID01 by SP1 and SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.8 and 5.4.27.1 provide test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful

Step	Direction	Sequence	Expected Result
3		Call "Select AID02" 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 AID01" function	Call is successful
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 (first installed application)
15		Close GSMA_AC_Mobile_App_SP1_signed	
16		repeat steps 9 to 15 with AID02 instead of AID01	No application is triggered

5.3.1.6 Test Sequence No 6: Multiple apps access to single AID, separate ACCFs, installation order

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 AID01 and a path for one ACCF containing SP1 hash condition and SP2 hash condition.

⇨ SP1 and SP2 can access AID01 only

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.31.3 provides test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in a later version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is successful
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 (first installed application)
15		Close GSMA_AC_Mobile_App_SP1_signed	
16		Repeat steps 9 to 15 with AID02 instead of AID01	No application is triggered

5.3.1.7 Test Sequence No 7: Multiple apps access to all AIDs, installation order

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 have full access to all AIDs

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.29.10 provides test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID01" function	Call is successful
7		Call "Select AID02" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 (first installed application)

Step	Direction	Sequence	Expected Result
15		Close GSMA_AC_Mobile_App_SP1_signed	
16		Repeat steps 9 to 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.8 Test Sequence No 8: Single app access to multiple AIDs

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 AID01 and a path for one ACCF containing SP1 hash condition
    - one specific target rule for AID02 and a path for the same ACCF

⇨ SP1 has access to AID01 and AID02

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 AID01" function	Call is successful
3		Call "Select AID02" function	Call is successful
4		Close GSMA_AC_Mobile_App_SP1_signed	
5		Launch GSMA_AC_Mobile_App_SP2_signed	
6		Call "Select AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	

Step	Direction	Sequence	Expected Result
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP1_signed** is launched

5.3.1.9 Test Sequence No 9: Single app access to single AID, further empty ACCF rule

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 AID01 and a path for one ACCF containing SP1 hash condition
    - one specific target rule for AID01 and a path for one ACCF. The ACCF contains no hash condition (access allowed for mobile apps)

⇨ only access to AID01 by SP1 is allowed

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.32.2 provides test steps that are similar to steps 1 to 8 of this test case. Redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful

Step	Direction	Sequence	Expected Result
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	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
- TS26_NFC_REQ_122

- TS26_NFC_REQ_122.2

Initial Conditions

- An instance of the UICC application **APDU_TestApplication.cap** with AID01 is selectable.
- **MobileApplication** is installed on the DUT and implements a function “Select AID01”.
- The application is signed with test certificate SP1 (**GSMA_Mobile_App_SP1_signed).**

5.3.2.1 Test Sequence No 1: Refresh tag not updated, refresh tag updated

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 AID01 and a path for one ACCF containing an empty hash condition

⇨ only access to AID01 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Using the **MobileApplication**, select AID01	Call is successful
2		Start the ISO7816 spy	
3		Using the **MobileApplication**, select AID01	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 AID01 and a path for one ACCF	

Step	Direction	Sequence	Expected Result
		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 AID01	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: Device rebooted

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 AID01 and a path for one ACCF containing an empty hash condition

⇨ only access to AID01 is allowed

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1		Using the **MobileApplication**, select AID01	Call is successful
2		Power off the DUT	
3		Start the ISO7816 spy	
4		Power on the DUT	
5		Using the **MobileApplication**, select AID01	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

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

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 AID01 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 AID02 and a path for one ACCF containing a SP2 hash condition

⇨ only access to AID01 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 AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	

Step	Direction	Sequence	Expected Result
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	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, if 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

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

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 AID01 and a path for one ACCF containing a SP1 hash condition

⇨ only access to AID01 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 AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" 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	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	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

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**

5.3.5.1 Test Sequence No 1: Many hash conditions

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 AID01 and a path for one ACCF containing 10 dummy hash conditions and a SP1 hash condition
    - one specific target rule for AID02 and a path for one ACCF containing 10 dummy hash conditions and a SP2 hash condition

⇨ access to AID01 by SP1 is allowed – access to AID02 by SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.31.1 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" function	Call is successful

Step	Direction	Sequence	Expected Result
8		Close GSMA_AC_Mobile_App_SP2_signed	
9	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP2_signed** is launched

5.3.5.2 Test Sequence No 2: Many rules

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 AID01 and a path for one ACCF containing 1 dummy hash condition and a SP1 hash condition
- one specific target rule for AID02 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 AID01 by SP1 is allowed – access to AID02 by SP2 is allowed

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform – SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.31.2 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch GSMA_AC_Mobile_App_SP1_signed	
2		Call "Select AID01" function	Call is successful
3		Call "Select AID02" 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 AID01" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted
7		Call "Select AID02" function	Call is successful
8		Close GSMA_AC_Mobile_App_SP2_signed	
9	PCD	Power on RF field	
10	PCD→ DUT	Perform RF protocol initialisation	
11	PCD→ DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
12	PCD	Power off RF field	
13	DUT→ UICC	Send 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 15 with AID02 instead of AID01	**GSMA_AC_Mobile_App_SP2_signed** is launched

5.3.6 GP SE Access Control – No access

Test Purpose

To ensure the DUT denies the access to

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

Referenced requirement

- TS26_NFC_REQ_083

Initial Conditions

An instance of the UICC application **APDU_TestApplication.cap** with AID01 is selectable.

For that purpose, **MobileApplication** is registered for EVT_TRANSACTION handling from AID01 and implements a function “Select AID01”.

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

5.3.6.1 Test Sequence No 1: PKCS#15 ADF absent

Initial Conditions

The following configuration is loaded into the UICC:

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

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.1.2 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID01" 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	PCD	Power on RF field	
5	PCD→ DUT	Perform RF protocol initialisation	
6	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
7	PCD	Power off RF field	
8	DUT→ UICC	Send EVT_FIELD_OFF	

Step	Direction	Sequence	Expected Result
9		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: ACRF absent

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.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.1.9 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		**Launch GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID01" 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	PCD	Power on RF field	
5	PCD→ DUT	Perform RF protocol initialisation	
6	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
7	PCD	Power off RF field	
8	DUT→ UICC	Send EVT_FIELD_OFF	
9		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: ACRF empty

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.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.1.14 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID01" 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	PCD	Power on RF field	
5	PCD→ DUT	Perform RF protocol initialisation	
6	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
7	PCD	Power off RF field	
8	DUT→ UICC	Send EVT_FIELD_OFF	
9		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: Corrupted certificate, wrong length

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 AID01 and a path for one ACCF containing an entry with a corrupted certificate (wrong length)

The reference PKCS#15 structure is in Annex E.

Note: Annex B.8 of this document lists the test cases from the GlobalPlatform - SEAC DeviceSide Test Plan [27]. Test cases referenced as 5.4.1.16 provide test steps that are similar to steps 1 to 8 of this test case. For a sake of clarity, redundancies will be handled in the next version of this Test Book.

Step	Direction	Sequence	Expected Result
1		Launch **GSMA_AC_Mobile_App_SP1_signed**	
2		Call "Select AID01" 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	PCD	Power on RF field	
5	PCD→ DUT	Perform RF protocol initialisation	
6	PCD→ DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
7	PCD	Power off RF field	
8	DUT→ UICC	Send EVT_FIELD_OFF	
9		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: Corrupted certificate, invalid content

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 AID01 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 AID01" 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	PCD	Power on RF field	
5	PCD→ DUT	Perform RF protocol initialisation	
6	PCD→ DUT	Start **APDU application** by sending a SELECT command with AID01	**APDU Application** receives Status Word 90 00
7	PCD	Power off RF field	
8	DUT→ UICC	Send EVT_FIELD_OFF	
9		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 GlobalPlatform Specification GPD_SE_Access_Control

Referenced requirement

- TS26_NFC_REQ_082
- TS26_NFC_REQ_083

Related Specs/Docs: GlobalPlatform - SEAC DeviceSide Test Plan [27]

The DUT shall pass the Test Cases with ID REQ from GlobalPlatform - SEAC DeviceSide Test Plan [27], the set of applicable test cases is referenced in Table B.8.1.

6 Secure Element Access API

6.1 General overview

This chapter addresses the implementation of the Mobile Device APIs according to the GlobalPlatform 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 and eSEs.

6.2 Conformance requirements

The Requirements tested are referenced in each test case.

6.3 Test Cases

6.3.1 GlobalPlatform OMAPI

The SIMalliance group has published the "Open Mobile API" specification until version 3.2. The ownership of the specifications has for the following versions moved to GlobalPlatform.

Test Purpose

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

Referenced requirement

- TS26_NFC_REQ_045.1
- TS26_NFC_REQ_047
- TS26_NFC_REQ_047.1
- TS26_NFC_REQ_047.3
- TS26_NFC_REQ_069
- TS26_NFC_REQ_114
- TS26_NFC_REQ_155
- TS26_NFC_REQ_186

Related Specs/Docs: GlobalPlatform - Open Mobile API specification [6]

The DUT shall pass the test cases referenced in Table B1.2.

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 Table B1.2.

6.3.3 VOID

6.3.4 VOID

6.3.5 VOID

6.3.6 VOID

6.3.7 GlobalPlatform APIs for eSE

Test Purpose

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

Referenced requirement

- TS26_NFC_REQ_047
- TS26_NFC_REQ_047.1
- TS26_NFC_REQ_070
- TS26_NFC_REQ_186

Related Specs/Docs: GlobalPlatform - Open Mobile API specification [6a]

The DUT shall pass the following test cases referenced in Table B1.2:

- 6.3.1.6.3.1eSE
- 6.3.1.6.3.3eSE
- 6.3.1.6.4.7eSE
- 6.3.1.6.5.6eSE
- 6.3.1.6.5.7eSE

The column “ISO Command Expectation” is out of the scope, because the test tool has no direct physical access to the eSE and it is not possible to verify the APDU communication with the eSE.

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.

7.2 Conformance requirements

The Requirements tested are referenced in each test case.

7.3 Test Cases

7.3.1 VOID

7.3.2 VOID

7.3.3 VOID

7.3.4 VOID

7.3.5 VOID

7.3.6 VOID

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_068
- TS26_NFC_REQ_068.01
- TS26_NFC_REQ_117
- TS26_NFC_REQ_162

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully
- NFC is enabled in the DUT
- No applications should be started manually on the DUT
- ReferenceApplication.cap for managing the reference transaction with AID_REF is installed and selectable on the UICC
- APDU Application to send APDUs according to the reference transaction.
- No off_host_apdu_service and/or host_apdu_service shall be registered with AID_REF in the CLF routing table.

7.3.7.1 Test Sequence No 1: Default route UICC, contactless session with unregistered AID

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- In the NFC Controller the default AID route is set to UICC (see section 2.6.1)
- The AID_REF is not registered.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT DUT → UICC	Execute the reference transaction	Reference transaction is performed successfully with UICC as CEE.

7.3.7.2 Test Sequence No 2: Default route HCE, contactless session with unregistered AID

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- In the NFC Controller the default AID route is set to HCE (see section 2.6.1)
- The AID_REF is not registered.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT DUT → UICC	Execute the **reference transaction**	The DUT returns SW indicating error code on the select AID command. No APDU shall be forwarded to the UICC.

7.3.7.3 Test Sequence No 3: Default route UICC, off-host AID

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as "other".

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to UICC (see section 2.6.1). .	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
4	PCD	Power on RF field	
5	PCD → DUT	Perform RF protocol initialisation	
6	PCD → DUT DUT → UICC	Execute the reference transaction For AID_REF	Reference transaction is performed successfully with UICC as CEE.

7.3.7.4 Test Sequence No 4: Default route HCE, off-host AID

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as "other".

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to HCE. (see section 2.6.1)	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
4	PCD	Power on RF field	

Step	Direction	Sequence	Expected Result
5	PCD → DUT	Perform RF protocol initialisation	
6	PCD → DUT DUT → UICC	Execute the reference transaction for AID_REF.	Reference transaction is performed successfully with UICC as CEE.

7.3.7.5 Test Sequence No 5: Default route UICC, AID conflict, off-host service selected

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “other”.
- App02: an android application which registers in its Manifest a host_apdu_service (HCE) for AID_REF and specifies the category as “other”. This App manages the reference transaction.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to UICC. (see section 2.6.1)	
3	User → DUT	Install App02	The application is installed successfully
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	None
5	PCD	Power on RF field	
6	PCD → DUT	Perform RF protocol initialisation	
7	PCD → DUT DUT → UICC	Execute the **reference transaction**	• Reference transaction fails • the DUT shall prompt the user with a pop-up, asking to select the desired application (Conflict of AIDs as the same AID is registered towards both UICC and HCE)
8	User → DUT	Select App01	
9	PCD	Power off RF field	

Step	Direction	Sequence	Expected Result
10	PCD	Power on RF field	
11	PCD → DUT	Perform RF protocol initialisation	
12	PCD → DUT DUT → UICC	Execute the **reference transaction** for AID.	Reference transaction is performed successfully with UICC as CEE.

7.3.7.6 Test Sequence No 6: Default route HCE, AID conflict, off-host service selected

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “other”.
- App02: an android application which registers in its Manifest a host_apdu_service (HCE) for AID_REF and specifies the category as “other”. This App manages the reference transaction.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to HCE. (see section 2.6.1)	
3	User → DUT	Install App02	The application is installed successfully
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
5	PCD	Power on RF field	
6	PCD → DUT DUT → UICC	Perform RF protocol initialisation	
7	PCD → DUT DUT → UICC	Execute the **reference transaction**	• Reference transaction fails • the DUT shall prompt the user with a pop-up, asking to select the desired application (Conflict of AIDs as the same AID is registered towards both UICC and HCE)

Step	Direction	Sequence	Expected Result
8	User → DUT	Select App01	
9	User → PCD	Power off the field	
10	User → PCD	Power on the field	
11	PCD → DUT DUT → UICC	Start card emulation session	Contactless Session is started
12	PCD → DUT DUT → UICC	Execute the **reference transaction** for AID_REF	**Reference transaction** is performed successfully with UICC as CEE.

7.3.7.7 Test Sequence No 7: Default route UICC, off-host service selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “payment”.
- App02: an android application which registers in its Manifest at the host_apdu_service (HCE) for AID_REF and specifies the category as “payment”. This App manages the reference transaction.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01.	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to UICC. (see section 2.6.1)	
3	User → DUT	Install App02	The application is installed successfully
4	User → DUT	Select App01 in the Tap&Pay menu	App01 is selected as Tap&Pay.
5	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
6	PCD	Power on RF field	

Step	Direction	Sequence	Expected Result
7	PCD → DUT	Perform RF protocol initialisation	
8	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully with UICC as CEE. The DUT shall NOT prompt the user with a pop-up.

7.3.7.8 Test Sequence No 8: Default route HCE, off-host service selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “payment”.
- App02: an android application which registers in its Manifest a host_apdu_service (HCE) for AID_REF and specifies the category as “payment”. This App manages the reference transaction.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to HCE. (see section 2.6.1)	
3	User → DUT	Install App02	The application is installed successfully
4	User → DUT	Select the App01 in the Tap&Pay menu	App01 is selected as Tap&Pay.
5	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
6	PCD	Power on RF field	
7	PCD → DUT	Perform RF protocol initialisation	
8	PCD → DUT DUT → UICC	Execute the **reference transaction**	**Reference transaction** is performed successfully with UICC as CEE. DUT shall NOT prompt the user with a pop-up.

7.3.7.9 Test Sequence No 9: Default route UICC, HCE service selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “payment”.
- App02: an android application which registers in its Manifest a host_apdu_service (HCE) for AID_REF and specifies the category as “payment”. This App manages the reference transaction

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to UICC. (see section 2.6.1)	
3	User → DUT	Install App02.	The application is installed successfully
4	User → DUT	Select App02 in the Tap&Pay menu	App02 is selected as Tap&Pay.
5	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
6	PCD	Power on RF field	
7	PCD → DUT	Perform RF protocol initialisation	
8	PCD → DUT DUT → UICC	Execute the **reference transaction** For AID_REF	**Reference transaction** is performed successfully with HCE as CEE. The DUT shall 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: Default route HCE, HCE service selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- App01: an android application which registers in its Manifest an off_host_apdu_service for AID_REF and specifies the category as “payment”.
- App02: an Android application which registers in its Manifest a host_apdu_service (HCE) for AID_REF and specifies the category as “payment”. This App manages the reference transaction

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	In the NFC Controller set the default AID route to HCE. (see section 2.6.1)	The default Card Emulation Environment is now HCE.
3	User → DUT	Install App02.	The application is installed successfully
4	User → DUT	Select App02 in the Tap&Pay menu	App02 is selected as Tap&Pay.
5	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
6	PCD	Power on RF field	
7	PCD → DUT	Perform RF protocol initialisation	
8	PCD → DUT DUT → UICC	Execute the reference transaction For AID_REF	Reference transaction is performed successfully with HCE as CEE. The DUT shall 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 after initial power up or factory reset NFC communication is routed to the UICC by default and RF parameters are properly set by the device.

Referenced requirement

- TS26_NFC_REQ_065
- TS26_NFC_REQ_118.1
- TS26_NFC_REQ_118.2
- TS26_NFC_REQ_162.1
- TS26_NFC_REQ_177

7.3.8.1 VOID

7.3.8.2 Test Sequence No 2: REQ_065 for NFCA

Initial Conditions

- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- The default AID route is set to HCE (see section 2.6.1)
- The routing table of the CLF contains an entry for an Applet identified by [AID01] and route for AID01 is set to UICC

- Install an Applet with [AID01] on the UICC implementing External Authenticate according to Annex A.4.4.

Step	Direction	Sequence	Expected Result
1		Use a contactless reader to explicitly select this Applet by AID01	Status Word 90 00 is received by the contactless reader
2		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using the contactless reader	Status Word 90 00 is received by the contactless reader

7.3.8.3 Test Sequence No 3: REQ_118.2 for NFCA

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- Install an applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCA. Note: The reader shall not explicitly select the Applet by AID
- The default AID route is set to UICC (see section 2.6.1)

Step	Direction	Sequence	Expected Result
1		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using a contactless reader Note: The reader shall access the applet without explicitly selecting it by AID.	Status Word 90 00 is received by the contactless reader

7.3.8.4 Test Sequence No 4: REQ_118.2 for NFCB

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type B only or provide a mechanism to make sure the NFC transaction will be performed using RF type B.
- Install an applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCB. Note: The reader shall not explicitly select the Applet by AID
- The default AID route is set to UICC (see section 2.6.1)

Step	Direction	Sequence	Expected Result
1		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using a contactless reader Note: The reader shall access the applet without explicitly selecting it by AID.	Status Word 90 00 is received by the contactless reader

7.3.8.5 Test Sequence No 5: REQ_118.1 and REQ_162.1 for NFCA

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- The NFC reader is establishing an ISO 14443-3 communication over type A.
- Install an Applet on the UICC, to handle CLT=A mode or use an intrinsic UICC mechanism (e.g. MIFARE Classic)
- The default AID route is set to HCE (see section 2.6.1)

Step	Direction	Sequence	Expected Result
1		Use a contactless reader to exchange command with this applet while remaining at ISO 14443-3 communication level (e.g. a MIFARE classic reader).	Status Word 90 00 is returned

7.3.8.6 Test Sequence No 6: REQ_065 for NFCB

Initial Conditions

- The NFC reader is polling in type B only or provide a mechanism to make sure the NFC transaction will be performed using RF type B.
- Install an Applet with [AID01] on the UICC implementing External Authenticate according to Annex A.4.4
- The default AID route is set to HCE (see section 2.6.1)
- The routing table of the CLF contains an entry with [AID01] and route for AID01 is set to UICC

Step	Direction	Sequence	Expected Result
1		Use a contactless reader to explicitly select this Applet by its AID	Status Word 90 00 is received by the contactless reader
2		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using a contactless reader	Status Word 90 00 is received by the contactless reader

7.3.8.7 VOID

7.3.8.8 VOID

7.3.8.9 Test Sequence No 9: REQ_118.2 and REQ_162.1 for NFCA

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before this test.
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- Install an Applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCA. Note: The reader shall not explicitly select the Applet by AID.
- The default AID route is set to HCE (see section 2.6.1.)

Step	Direction	Sequence	Expected Result
1		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using a contactless reader Note: The reader shall access the applet without explicitly selecting it by AID.	Status Word 90 00 is received by the contactless reader

7.3.8.10 Test Sequence No 10: REQ_118.2 and REQ_162.1 for NFCB

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before this test.
- The NFC reader is polling in type B only or provide a mechanism to make sure the NFC transaction will be performed using RF type B.
- Install an Applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCB. Note: The reader shall not explicitly select the Applet by AID.
- The default AID route is set to HCE (see section 2.6.1.)

Step	Direction	Sequence	Expected Result
1		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using a contactless reader Note: The reader shall access the applet without explicitly selecting it by AID.	Status Word 90 00 is received by the contactless reader

7.3.8.11 Test Sequence No 11: REQ_177 for NFCA

Initial Conditions

- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- The default AID route is set to HCE (see section 2.6.1)
- The routing table of the CLF contains an entry for an Applet identified by [AID01] and route for AID01 is set to UICC
- Install an Applet with [AID01] on the UICC implementing External Authenticate according to Annex A.4.4. When activated the Applet requests the Contactless parameters according to “Basic profile” in Table 2 of GSMA SGP12 [42]

Step	Direction	Sequence	Expected Result
1		Use a contactless reader to explicitly select this Applet by AID01	Status Word 90 00 is received by the contactless reader
2		Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) using the contactless reader	Status Word 90 00 is received by the contactless reader
3	PCD → DUT	The test tool verifies the following contactless protocol parameters: GP Tag ‘80’ – UID (LV) GP Tag ‘81’ - SAK GP Tag ‘82’ - ATQA GP Tag ‘83’ – ATS (LV) GP Tag ’84 - FWI/SFGI GP Tag ‘85’ – CID support GP Tag ‘86’ - Data_Rate Max	The values of these parameters are matching the values of profile 1 as defined in Table 3 of GSMA SGP12 [42]

7.3.9 Size of the CLF AID Routing table

Test Purpose

Ensure that the device supports at least 16 AIDs of 16 bytes inside the AID routing table of the CLF as specified in TS26

Referenced requirement

- TS26_NFC_REQ_167

7.3.9.1 Test Sequence No 1: Size of the CLF AID Routing

Step	Direction	Sequence	Expected Result
1		Apply procedure described in Section 2.6.2	RTS value SHALL be greater than OR equal to 16

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.

8.2 Conformance requirements

The Requirements tested are referenced in each test case.

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 [10] specification

Referenced requirement

- TS26_NFC_REQ_071

Initial Conditions

Related Specs/Docs: ETSI TS 102 622 [10]

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_131

Initial Conditions

The application is registered for EVT_TRANSACTION handling from AID01 with *com.gsma.services.nfc.action.TRANSACTION_EVENT*

8.3.2.1 Test Sequence No 1: NFC and TRANSACTION_EVENT declared

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	User → DUT	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: NFC not declared, TRANSACTION_EVENT declared

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	User → DUT	Install the Android application	DUT does not require the user to authorize the NFC permission

8.3.2.3 Test Sequence No 3: NFC declared, TRANSACTION_EVENT not declared

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	User → DUT	Install the Android application	DUT shall prompt the user to authorize the application installation indicating 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_069
- TS26_NFC_REQ_096
- TS26_NFC_REQ_129
- TS26_NFC_REQ_187

- TS26_NFC_REQ_188

Initial Conditions

- The DUT is powered on
- HCI initialization has been performed successfully
- NFC is enabled in the DUT
- Three instances of the UICC application APDU_TestApplication.cap with AID01, AID02 and AID03 are selectable.
- Different mobile applications must be used for devices before Android 9 and for devices based on Android 9 or following Android releases:
  - For devices before Android 9 the mobile application registers in its manifest for EVT_TRANSACTION handling from AID01 and AID02 **only** with com.gsma.services.nfc.action.TRANSACTION_EVENT
  - For devices based on Android 9 or following Android releases the mobile application registers in its manifest for EVT_TRANSACTION handling from AID01 and AID02 **only** with android.nfc.action.TRANSACTION_DETECTED.
- No applications should be started manually on the DUT

8.3.3.1 Test Sequence No 1: EVT_TRANSACTION, no data

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT -> UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	Application is triggered and receives the transaction event and URI format is the following: • nfc://secure:0/SE_Name/AID with • AID equals to AID01 • SE_Name according to GlobalPlatform open mobile API specification

Step	Direction	Sequence	Expected Result
8	App → DUT	Open OMAPI session with the reader named “SE_NAME” returned in step 7	Session is opened successfully
9	DUT→ UICC	Call the “Select AID01” function	Application with AID01 is selected
10	DUT → UICC	Send **APDU Case 4** P1 = 0x00 ’XX 04 00 00 FF’ <Data field of 255 bytes> FF	Expected response returned from “SE_Name” is: R-APDU – data field of 255 bytes, SW1, SW2

8.3.3.2 Test Sequence No 2: EVT_TRANSACTION, with data

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID02	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT -> UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID02 with data (0x20 bytes long)	Application is triggered and receives the transaction event with additional data (0x20 bytes long). The URI format is the following: • nfc://secure:0/SE_Name/AID with • AID equals to AID02 SE_Name according to GlobalPlatform open mobile API specification • On devices before Android 9 the received data is retrieved from com.gsma.services.nfc.extra.DATA • On devices with Android 9 or later the received data is retrieved from android.nfc.extra.DATA

Step	Direction	Sequence	Expected Result
8	App → DUT	Open OMAPI session with the reader named “SE_NAME” returned in step 7	Session is opened successfully
9	DUT→ UICC	Call the “Select AID02” function	Application with AID02 is selected and SW 90 00 is returned by SE_Name
10	DUT → UICC	Send **APDU Case 4** P1 = 0x00 ’XX 04 00 00 FF’ <Data field of 255 bytes> FF	Expected response returned from reader “SE_Name” is: R-APDU – data field of 255 bytes, SW1, SW2

8.3.3.3 Test Sequence No 3: EVT_TRANSACTION, application not registered for AID

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID03	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT -> UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID03	Application doesn’t receive the transaction event

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_150
- TS26_NFC_REQ_150.1

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 AID01 and AID02

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 AID02

Three instances of the **APDU_TestApplication.cap** (respectively with AID01, AID02 and AID03).

- No applications should be started manually on the DUT

8.3.4.1 Test Sequence No 1: First installed application triggered

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 AIDs
The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT → UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	**GSMA_Mobile_App_UIA_#1** is launched

Step	Direction	Sequence	Expected Result
8	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
9	User → DUT	Repeat steps 2 – 6 with AID02 instead of AID01	
10	UICC → DUT	UICC sends EVT_TRANSACTION with AID02	**GSMA_Mobile_App_UIA_#1** is launched
11	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
12	User → DUT	Repeat steps 2 – 6 with AID03 instead of AID01	
13	UICC → DUT	UICC sends EVT_TRANSACTION with AID03	No mobile application is launched

8.3.4.2 Test Sequence No 2: Different AID

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 AIDs

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, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT→ UICC	Send EVT_FIELD_OFF	

Step	Direction	Sequence	Expected Result
7		UICC sends EVT_TRANSACTION with AID01	**GSMA_Mobile_App_UIA_#1** is launched
8	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
9	User → DUT	Repeat steps 2 – 6 with AID02 instead of AID01	
10	UICC → DUT	UICC sends EVT_TRANSACTION with AID02	**GSMA_Mobile_App_UIA_#1**is launched
11	User → DUT	Close the application **GSMA_Mobile_App_UIA_#1**	Application is closed
12	User → DUT	Repeat steps 2 – 6 with AID03 instead of AID01	
13	UICC → DUT	UICC sends EVT_TRANSACTION with AID03	**GSMA_Mobile_App_UIA_#3** is launched

8.3.4.3 Test Sequence No 3: Single application has SEAC access

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 SP2 hash condition

⇨ SP2 has full access to all AIDs

The reference PKCS#15 structure is in Annex E.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT → UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	No mobile application is launched

Step	Direction	Sequence	Expected Result
8	User → DUT	Repeat steps 2 – 6 with AID02 instead of AID01	
9	UICC → DUT	UICC sends EVT TRANSACTION with AID02	**GSMA_Mobile_App_UIA_#2** is launched
10	User → DUT	Close the application **GSMA_Mobile_App_UIA_#2**	Application is closed
11	User → DUT	Repeat steps 2 – 6 with AID03 instead of AID01	
12	UICC → DUT	UICC sends EVT TRANSACTION with AID03	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_072

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 AID01
- **MobileApplication** is installed on the DUT and is launched on EVT_TRANSACTION from AID01
- No applications should be started manually on the DUT.

8.3.5.1 Test Sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	None
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	

Step	Direction	Sequence	Expected Result
4	PCD → DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
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 AID01

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_072

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 AID01
- MobileApplication is installed on the DUT and is launched on EVT_TRANSACTION from AID01
- No applications should be started manually on the DUT.
- APDU_TestApplication is not selected on UICC.

8.3.6.1 Test Sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	None
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00

Step	Direction	Sequence	Expected Result
5	PCD	Power off RF field	
6	DUT → UICC	Send EVT_ FIELD_OFF	
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 AID01

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.

11.2 Conformance requirements

The Requirements tested are referenced in each test case.

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

Initial Conditions

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

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 size 1024

11.3.1.2 Test 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	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	
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

Initial Conditions

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

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	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: 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

Initial Conditions

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

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]
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	

Step	Direction	Sequence	Expected Result
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
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 buffer

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.

Step	Direction	Sequence	Expected Result
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	
15	ME → USS	Transfer 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 1024 Bytes of data to the ME 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	

Step	Direction	Sequence	Expected Result
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	

11.3.3.3 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.

Step	Direction	Sequence	Expected Result
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 11.3.1	See OPEN CHANNEL 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 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 1024 Bytes of data to the ME 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	

Step	Direction	Sequence	Expected Result
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	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.

Step	Direction	Sequence	Expected Result
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 CHANNEL 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 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 1024 Bytes of data to the ME 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	

Step	Direction	Sequence	Expected Result
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

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]
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	

Step	Direction	Sequence	Expected Result
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 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

Step	Direction	Sequence	Expected Result
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 packets 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	[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	

Step	Direction	Sequence	Expected Result
25	ME→UICC	FETCH	
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[200 Bytes]
27	ME → USS	Transfer 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 packets 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]

Step	Direction	Sequence	Expected Result
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 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	

Step	Direction	Sequence	Expected Result
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 packets 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	

Step	Direction	Sequence	Expected Result
26	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 11.1.5	[204 Bytes]
27	ME → USS	Transfer 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 GSMA 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.

12.2 Conformance requirements

The Requirements tested are referenced in each test case.

12.3 Basic Remote Management

12.3.1 General overview

This section addresses the testing of the Bearer Independent Protocol (BIP) used in remote management of NFC services.

12.3.2 Conformance requirements

The Requirements tested are referenced in each test case.

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

Related Specs/Docs: ETSI TS 102 223 [22]

Test Procedure

The DUT shall pass all test cases referenced in Table B.6.1 and Table B.6.2.

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

Initial Conditions

- All TCs are defined by making use of Bearer Type ‘03’= default bearer for requested transport layer.
- The DUT is registered in idle mode and is configured to not establish a PDN connection triggered by the OS itself

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]

<u>PROACTIVE COMMAND: OPEN CHANNEL 1.1</u>

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: ME

Bearer description

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

<u>TERMINAL RESPONSE: OPEN CHANNEL 1.1</u>

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	

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 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 description

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	Confirm	
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 description

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)

Initial Conditions

None

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	Confirm	[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]

<u>PROACTIVE COMMAND: OPEN CHANNEL 4.1</u>

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 description

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 description

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 6a/1 "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	Reject	
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 description

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	Reject	
8	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 6.1	[User did not accept the proactive command]

<u>PROACTIVE COMMAND: OPEN CHANNEL 6.1</u>

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 description

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

Initial Conditions

- All TCs are defined by making use of Bearer Type '03'= default bearer for requested transport layer.
- The DUT is registered in idle mode and is configured to not establish a PDN connection triggered by the OS itself

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	
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	

Step	Direction	Sequence	Expected Result
10	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 2.1	
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	

Step	Direction	Sequence	Expected Result
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]
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]

<u>TERMINAL RESPONSE: CLOSE CHANNEL 3.1A</u>

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

<u>TERMINAL RESPONSE: CLOSE CHANNEL 3.1B</u>

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

Initial Conditions

- All TCs are defined by making use of Bearer Type ‘03’= default bearer for requested transport layer.
- The DUT is registered in idle mode and is configured to not establish a PDN connection triggered by the OS itself

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	

Step	Direction	Sequence	Expected Result
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 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]
17	USS → ME	Transfer 1000 Bytes of data to the ME 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	

Step	Direction	Sequence	Expected Result
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	
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)

<u>TERMINAL RESPONSE: SEND DATA 1.1</u>

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

<u>ENVELOPE: EVENT DOWNLOAD - Data available 1.1</u>

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)

<u>PROACTIVE COMMAND: RECEIVE DATA 1.1</u>

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

<u>TERMINAL RESPONSE: RECEIVE DATA 1.1</u>

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

<u>TERMINAL RESPONSE: RECEIVE DATA 1.2</u>

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

<u>TERMINAL RESPONSE: RECEIVE DATA 1.3</u>

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

Initial Conditions

- All TCs are defined by making use of Bearer Type ‘03’ = default bearer for requested transport layer.

- The DUT is registered in idle mode and is configured to not establish a PDN connection triggered by the OS itself

12.3.3.5.1 Test Sequence No 1: (SEND DATA, immediate mode)

Initial Conditions

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 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

<u>TERMINAL RESPONSE: OPEN CHANNEL 1.1</u>

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

<u>PROACTIVE COMMAND: SEND DATA 1.1</u>

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)

<u>TERMINAL RESPONSE: SEND DATA 1.1</u>

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]
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]

Step	Direction	Sequence	Expected Result
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 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
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 packets 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]

Step	Direction	Sequence	Expected Result
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 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 packets of 200 Bytes

Step	Direction	Sequence	Expected Result
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]
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 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 packets of 200 Bytes
32	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 3.1	[Command performed successfully]

Step	Direction	Sequence	Expected Result
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]
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 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

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 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]

<u>PROACTIVE COMMAND: SEND DATA 5.1</u>

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)

<u>TERMINAL RESPONSE: SEND DATA 5.1</u>

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

Initial Conditions

All TCs are defined by making use of Bearer Type ‘03’= default bearer for requested transport layer.

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]

<u>PROACTIVE COMMAND: GET STATUS 1.1</u>

Command details

Command number: 1

Command type: GET STATUS

Command qualifier: RFU

Device identities

Source device: UICC

Destination device: ME

<u>TERMINAL RESPONSE: GET STATUS 1.1A</u>

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	
10	ME → UICC	TERMINAL RESPONSE GET STATUS 2.1A Or TERMINAL RESPONSE: GET STATUS 2.1B	[Command performed successfully]

<u>PROACTIVE COMMAND: OPEN CHANNEL 1.1</u>

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 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 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	

Step	Direction	Sequence	Expected Result
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 → UICC	TERMINAL RESPONSE: OPEN CHANNEL 1.1	[Command performed successfully]
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]

<u>TERMINAL RESPONSE: GET STATUS 3.1A</u>

Same as TERMINAL RESPONSE: GET STATUS 1.1A

<u>TERMINAL RESPONSE: GET STATUS 3.1B</u>

Same as TERMINAL RESPONSE: GET STATUS 1.1B

<u>TERMINAL RESPONSE: GET STATUS 3.1C</u>

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 EVENT 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 EVENT 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

Initial Conditions

All TCs are defined by making use of Bearer Type ‘03’ = default bearer for requested transport layer.

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: 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	
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	[Command performed successfully]
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	
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 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]
17	USS → ME	Send data through the BIP channel using the ME's port number, which was retrieved in step 11	
18	ME → UICC	ENVELOPE 1.1 (Event-Data Available)	

PROACTIVE COMMAND: SET UP EVENT LIST 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 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: 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

Initial Conditions

All TCs are defined by making use of Bearer Type ‘03’= default bearer for requested transport layer.

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	

Step	Direction	Sequence	Expected Result
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	USS → ME	Drop Link	
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 EVENT 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 EVENT 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 - Channel Status 1.1

Event list

Event: Channel Status

Device identities

Source 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

Initial Conditions

All TCs are defined by making use of Bearer Type ‘02’= GPRS bearer for requested transport layer.

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 12.3.3.9.4.8
2		Test Procedure Open Channel	as specified in 12.3.3.9.4.1
3		Test Procedure Send Data	as specified in 12.3.3.9.4.2
4		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
5		Test Procedure Send Data	as specified in 12.3.3.9.4.2
6		Test Procedure Send Data	as specified in 12.3.3.9.4.2
7		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
8		Test Procedure Send Data	as specified in 12.3.3.9.4.2
9		Test Procedure Send Data	as specified in 12.3.3.9.4.2
10		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
11		Test Procedure Send Data	as specified in 12.3.3.9.4.2
12		Test Procedure Send Data	as specified in 12.3.3.9.4.2
13		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
14		Test Procedure Send Data	as specified in 12.3.3.9.4.2
15		Test Procedure Send Data	as specified in 12.3.3.9.4.2
16		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
17		Test Procedure Send Data	as specified in 12.3.3.9.4.2
18		Test Procedure Send Data	as specified in 12.3.3.9.4.2
19		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4

Step	Direction	Sequence	Expected Result
20		Test Procedure Send Data	as specified in 12.3.3.9.4.2
21		Test Procedure Send Data	as specified in 12.3.3.9.4.2
22		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
23		Test Procedure Send Data	as specified in 12.3.3.9.4.2
24		Test Procedure Send Data	as specified in 12.3.3.9.4.2
25		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
26		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3

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 12.3.3.9.4.8
2		Test Procedure Send Short Message	as specified in 12.3.3.9.4.7
3		Test Procedure Open Channel	as specified in 12.3.3.9.4.1
4		Test Procedure Send Data	as specified in 12.3.3.9.4.2
5		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
6		Test Procedure Send Data	as specified in 12.3.3.9.4.2
7		Test Procedure Send Data	as specified in 12.3.3.9.4.2
8		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
9		Test Procedure Send Data	as specified in 12.3.3.9.4.2
10		Test Procedure Send Data	as specified in 12.3.3.9.4.2
11		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
12		Test Procedure Send Data	as specified in 12.3.3.9.4.2
13		Test Procedure Send Data	as specified in 12.3.3.9.4.2
14		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
15		Test Procedure Send Data	as specified in 12.3.3.9.4.2
16		Test Procedure Send Data	as specified in 12.3.3.9.4.2
17		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
18		Test Procedure Send Data	as specified in 12.3.3.9.4.2

Step	Direction	Sequence	Expected Result
19		Test Procedure Send Data	as specified in 12.3.3.9.4.2
20		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
21		Test Procedure Send Data	as specified in 12.3.3.9.4.2
22		Test Procedure Send Data	as specified in 12.3.3.9.4.2
23		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
24		Test Procedure Send Data	as specified in 12.3.3.9.4.2
25		Test Procedure Send Data	as specified in 12.3.3.9.4.2
26		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
27		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3

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 12.3.3.9.4.8
2		Test Procedure Send Short Message	as specified in 12.3.3.9.4.7
3		Test Procedure Open Channel	as specified in 12.3.3.9.4.1
4		Test Procedure Send Data	as specified in 12.3.3.9.4.2
5		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
6		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
7		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
8		Test Procedure Send Data	as specified in 12.3.3.9.4.2
9		Test Procedure Send Data	as specified in 12.3.3.9.4.2
10		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
11		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
12		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
13		Test Procedure Send Data	as specified in 12.3.3.9.4.2
14		Test Procedure Send Data	as specified in 12.3.3.9.4.2

Step	Direction	Sequence	Expected Result
15		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
16		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
17		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
18		Test Procedure Send Data	as specified in 12.3.3.9.4.2
19		Test Procedure Send Data	as specified in 12.3.3.9.4.2
20		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
21		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
22		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
23		Test Procedure Send Data	as specified in 12.3.3.9.4.2
24		Test Procedure Send Data	as specified in 12.3.3.9.4.2
25		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
26		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
27		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
28		Test Procedure Send Data	as specified in 12.3.3.9.4.2
29		Test Procedure Send Data	as specified in 12.3.3.9.4.2
30		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
31		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
32		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
33		Test Procedure Send Data	as specified in 12.3.3.9.4.2
34		Test Procedure Send Data	as specified in 12.3.3.9.4.2
35		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
36		Test Procedure Receive Data 2	as specified in 12.3.3.9.4.4
37		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
38		Test Procedure Send Data	as specified in 12.3.3.9.4.2
39		Test Procedure Send Data	as specified in 12.3.3.9.4.2

Step	Direction	Sequence	Expected Result
40		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
41		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
42		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
43		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6

12.3.3.9.4 Reference Test Procedures

12.3.3.9.4.1 Test Procedure Open Channel (OPEN CHANNEL, immediate link establishment, GPRS, no local address)

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: SET UP EVENT LIST	
2	ME→UICC	FETCH	
3	UICC→ ME	PROACTIVE COMMAND: SET UP EVENT LIST	
4	ME→ UICC	TERMINAL RESPONSE: SET UP EVENT LIST	
5	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL	
6	ME → UICC	FETCH	
7	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL	
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	[Command performed successfully]

PROACTIVE COMMAND: SET UP EVENT LIST

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

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: OPEN CHANNEL

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
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: 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

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

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

12.3.3.9.4.2 Test Procedure Send Data (SEND DATA, immediate mode)

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA	
2	ME→UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate)	
4	ME → USS	Transfer 40 Bytes of data to the USS through channel 1	
5	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate)	[Command performed successfully]

PROACTIVE COMMAND: SEND DATA

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 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13 14 15 16 17 18 19 1A 1B 1C 1D 1E 1F 20 21 22 23 24 25 26 27 (40 Bytes of data)

TERMINAL RESPONSE: SEND DATA

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

12.3.3.9.4.3 Test Procedure Receive Data 1 (RECEIVE DATA)

Step	Direction	Sequence	Expected Result
1	USS → ME	Transfer 20 Bytes of data to the ME through channel 1	
2	ME → UICC	ENVELOPE: EVENT DOWNLOAD - Data available 1	(20 Bytes of data in the ME buffer)1
3	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 1	
4	ME→UICC	FETCH	
5	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 1	20 Bytes
6	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 1	

ENVELOPE: EVENT DOWNLOAD - Data available 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: 20

PROACTIVE COMMAND: RECEIVE DATA 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: 20

TERMINAL RESPONSE: RECEIVE DATA 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 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 10 11 12 13(20 Bytes of data)

Channel data length: 00

12.3.3.9.4.4 Test Procedure Receive Data 2 (RECEIVE DATA)

Step	Direction	MESSAGE / Action	Comments
1	USS → ME	Transfer 1022 Bytes of data to the ME through channel 1	
2	ME → UICC	ENVELOPE: EVENT DOWNLOAD - Data available 2	(FF Bytes of data in the ME buffer) 2
3	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 2.1	
4	ME→UICC	FETCH	
5	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 2.1	FF Bytes

Step	Direction	MESSAGE / Action	Comments
6	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 2.1	
7	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 2.2	
8	ME→UICC	FETCH	
9	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 2.2	FF Bytes
10	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 2.2	
11	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 2.3	
12	ME→UICC	FETCH	
13	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 2.3	FF Bytes
14	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 2.3	
15	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 2.4	
16	ME→UICC	FETCH	
17	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 2.4	FF Bytes
18	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 2.4	
19	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 2.5	
20	ME→UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 2.5	74 Bytes
22	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 2.5	

ENVELOPE: EVENT DOWNLOAD - Data available 2

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 2.1

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: FF

PROACTIVE COMMAND: RECEIVE DATA 2.2

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: FF

PROACTIVE COMMAND: RECEIVE DATA 2.3

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: FF

PROACTIVE COMMAND: RECEIVE DATA 2.4

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: FF

PROACTIVE COMMAND: RECEIVE DATA 2.5

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: 74

TERMINAL RESPONSE: RECEIVE DATA 2.1

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…….EC (237 Bytes)

Channel data length: FF Bytes

TERMINAL RESPONSE: RECEIVE DATA 2.2

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: ED EE EF……D9 (237 Bytes)

Channel data length: FF

<u>TERMINAL RESPONSE: RECEIVE DATA 2.3</u>

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: DA DB…..C6(237 Bytes)

Channel data Length: FF

<u>TERMINAL RESPONSE: RECEIVE DATA 2.4</u>

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: C7 C9…..B3(237 Bytes)

Channel data Length: 74

<u>TERMINAL RESPONSE: RECEIVE DATA 2.5</u>

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: B4…..FD (74 Bytes)

Channel data length: 00

12.3.3.9.4.5 Test Procedure Timer Management (Start Timer)

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: TIMER MANAGEMENT	
2	ME→UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: TIMER MANAGEMENT	Start timer 1
4	ME → UICC	TERMINAL RESPONSE: TIMER MANAGEMENT	Command performed successfully.

<u>PROACTIVE COMMAND: TIMER MANAGEMENT (Start timer)</u>

Command details

Command number: 1

Command type: TIMER MANAGEMENT

Command qualifier: start the Timer

Device identities

Source device: UICC

Destination device: ME

Timer identifier

Identifier of timer: 1

Timer value

Value of timer: 00:02:00

<u>TERMINAL RESPONSE: TIMER MANAGEMENT</u>

Command details

Command number: 1

Command type: TIMER MANAGEMENT

Command qualifier: start the Timer

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Timer identifier

Identifier of timer: 1

12.3.3.9.4.6 Test Procedure Timer Management (Deactivate Timer)

 Non-confidential

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: TIMER MANAGEMENT	
2	ME→UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: TIMER MANAGEMENT	Deactivate timer 1
4	ME → UICC	TERMINAL RESPONSE: TIMER MANAGEMENT	Command performed successfully.

PROACTIVE COMMAND: TIMER MANAGEMENT (Deactivate Timer)

Command details

Command number: 1

Command type: TIMER MANAGEMENT

Command qualifier: deactivate the Timer

Device identities

Source device: UICC

Destination device: ME

Timer identifier

Identifier of timer: 1

TERMINAL RESPONSE: TIMER MANAGEMENT

Command details

Command number: 1

Command type: TIMER MANAGEMENT

Command qualifier: deactivate the Timer

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

Timer identifier

Identifier of timer: 1

Timer value

Value of timer: not checked

12.3.3.9.4.7 Test Procedure Send Short Message

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: SEND SHORT MESSAGE	
2	ME→UICC	FETCH	

Step	Direction	Sequence	Expected Result
3	UICC → ME	PROACTIVE COMMAND: SEND SHORT MESSAGE	[packing not required,8 bit data]
4	ME → USS	Send RP-DATA containing SMS-PP (SEND SHORT MESSAGE) Message	CS or PS domain is used to send and receive short messages
5	USS→ME	RP-ACK	
6	ME → UICC	TERMINAL RESPONSE: SEND SHORT MESSAGE	[Command performed successfully]

PROACTIVE COMMAND: SEND SHORT MESSAGE

Command details

Command number: 1

Command type: SEND SHORT MESSAGE

Command qualifier: packing not required

Device identities

Source device: UICC

Destination device: Network

Address

TON: International number

NPI: "ISDN / telephone numbering plan"

Dialling number string "491720354333"

SMS TPDU

TP-MTI: SMS-SUBMIT (in the direction MS to SC)

TP-RD: Instruct the SC to accept an SMS-SUBMIT for a SM

TP-VPF: TP-VP field not present

TP-RP: TP-Reply-Path is not set in this SMS-SUBMIT

TP-UDHI: The beginning of the TP-UD field contains a header in addition to the short message

TP-SRR: A status report is not requested

TP-MR: "00"

TP-DA

TON: Unknown

NPI: "ISDN / telephone numbering plan"

Address value: "10001"

TP-PID: no interworking, but SME to SME protocol

TP-DCS: 8-bit data, Class 2 SIM-specific Message

TP-UDL: 19

Information-Element-Ident: RFU

Data: "A@@...."

TP-UD: 02 71 00 00 0E 0A C0 00 00 00 00 00 04 31 00 00 01 6A 88

SMS-PP (SEND SHORT MESSAGE) Message

TP-MTI: SMS-SUBMIT (in the direction MS to SC)

TP-RD: Instruct the SC to accept an SMS-SUBMIT for a SM

TP-VPF: TP-VP field not present

TP-RP: TP-Reply-Path is not set in this SMS-SUBMIT

TP-UDHI: The beginning of the TP-UD field contains a header in addition to the short message

TP-SRR: A status report is not requested

TP-MR: "01"

TP-DA

TON: Unknown

NPI: "ISDN / telephone numbering plan"

Address value: "10001"

TP-PID: no interworking, but SME to SME protocol

TP-DCS: 8-bit data, Class 2 SIM-specific Message

TP-UDL: 19

Information-Element-Ident: RFU

Data: "A@@...."

TP-UD: 02 71 00 00 0E 0A C0 00 00 00 00 00 04 31 00 00 01 6A 88

TERMINAL RESPONSE: SEND SHORT MESSAGE

Command details

Command number: 1

Command type: SEND SHORT MESSAGE

Command qualifier: packing not required

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

12.3.3.9.4.8 Test Procedure SMS-PP Data Download

Step	Direction	Sequence	Expected Result
1	User → ME	Power the ME on	ME will perform Profile Download and USIM initialisation
2	ME → USS	ME performs CS/PS or PS registration.	
3	USS → ME	SMS-PP Data Download Message	See Note 1.
4	ME → User	The ME shall not display the message or alert the user of a short message waiting.	
5	ME → UICC	ENVELOPE: SMS-PP DOWNLOAD	
6	UICC → ME	SMS-PP Data Download UICC Acknowledgement	[SW1 / SW2 of '90 00']
7	ME → USS	SMS-PP Data Download UICC Acknowledgement in the TP-User-Data element of the RP-ACK message. The values of protocol identifier and data coding scheme in RP-ACK shall be as in the original message.	
Note 1: CS or PS domain is used to send and receive short messages			

<u>SMS-PP (Data Download) Message</u>

SMS TPDU

TP-MTI: SMS-DELIVER

TP-MMS: No more messages waiting for the MS in this SC

TP-RP: TP-Reply-Path is not set in this SMS-DELIVER

TP-UDHI: The beginning of the TP-UD field contains a header in addition to the short message

TP-SRI: A status report will be returned to the SME

TP-OA

TON Unknown

NPI "ISDN / telephone numbering plan"

Address value "10001"

TP-PID (U): U SIM Data download

TP-DCS

Coding Group General Data Coding

Compression Text is uncompressed

Message Class: Class 2 USIM Specific Message

Alphabet 8 bit data

TP-SCTS: 01/01/98 00:00:00 +0

TP-UDL : 109

TP-UD 02 70 00 00 68 15 16 21 19 19 C0 00 00 4F F5 A4 61 BE 1E E9 C0 6A 62 44 15 23 47 DA 22 24 B8 87 27 CC F7 0B 32 38 B2 6D D2 E0 7F 18 33 5A 06 4E 5F C5 C1 44 F7 0E 17 68 51 41 09 D9 28 43 79 B3 65 16 F4 E0 6F E3 10 0A 04 C2 18 0B 64 D7 F8 7C 88 6D BB F1 D9 EC 39 0C 02 67 24 BB DC 7B 50 06 9A 22 15 6F FC 3F 04 1B EE E1 C7 04 33

ENVELOPE: SMS-PP DOWNLOAD

SMS-PP Download

Device identities

Source device: Network

Destination device: UICC

Address

TON: International number

NPI: "ISDN / telephone numbering plan"

Dialling number string: "491720354333"

SMS TPDU

TP-MTI: SMS-DELIVER

TP-MMS: No more messages waiting for the MS in this SC

TP-RP: TP-Reply-Path is not set in this SMS-DELIVER

TP-UDHI: The beginning of the TP-UD field contains a header in addition to the short message

TP-SRI: A status report will be returned to the SME

TP-OA

TON: Unknown

NPI: "ISDN / telephone numbering plan"

Address value: "10001"

TP-PID: USIM Data download

TP-DCS

Coding Group General Data Coding

Compression Text is uncompressed

Message Class: Class 2 (U)SIM Specific Message

Alphabet : 8 bit data

TP-SCTS: 01/01/98 00:00:00 +0

TP-UDL 109

TP-UD: 02 70 00 00 68 15 16 21 19 19 C0 00 00 4F F5 A4 61 BE 1E E9 C0 6A 62 44 15 23 47 DA 22 24 B8 87 27 CC F7 0B 32 38 B2 6D D2 E0 7F 18 33 5A 06 4E 5F C5 C1 44 F7 0E 17 68 51 41 09 D9 28 43 79 B3 65 16 F4 E0 6F E3 10 0A 04 C2 18 0B 64 D7 F8 7C 88 6D BB F1 D9 EC 39 0C 02 67 24 BB DC 7B 50 06 9A 22 15 6F FC 3F 04 1B EE E1 C7 04 33

SMS-PP Data Download UICC Acknowledgement

12.3.3.9.4.9 Test Procedure More Time

Step	Direction	Sequence	Expected Result
1	UICC → ME	PROACTIVE COMMAND PENDING: MORE TIME	
2	ME→UICC	FETCH	
3	UICC → ME	PROACTIVE COMMAND: MORE TIME	
4	ME→UICC	TERMINAL RESPONSE: MORE TIME	Command performed successfully.

<u>PROACTIVE COMMAND: MORE TIME</u>

Command details

Command number: 1

Command type: MoreTime

Command qualifier: RFU

Device identities

Source device: UICC

Destination device: ME

<u>TERMINAL RESPONSE: MORE TIME</u>

Command details

Command number: 1

Command type: MoreTime

Command qualifier: RFU

Device identities

Source device: ME

Destination device: UICC

Result

General Result: Command performed successfully

12.3.3.9.5 Test Sequence No 4: (SMS-PP - Open channel - Send/Receive data - Send SM with More Time)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Test Procedure SMS-PP Data Download	as specified in 12.3.3.9.4.8
2		Test Procedure Open Channel	as specified in 12.3.3.9.4.1
3		Test Procedure Send Data	as specified in 12.3.3.9.4.2

Step	Direction	Sequence	Expected Result
4		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
5		Test Procedure More Time	as specified in 12.3.3.9.4.9
6		Test Procedure More Time	as specified in 12.3.3.9.4.9
7		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
8		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
9		Test Procedure Send Data	as specified in 12.3.3.9.4.2
10		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
11		Test Procedure Send SMS	as specified in 12.3.3.9.4.7
12		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
13		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6

12.3.3.9.6 Test Sequence No 5: (SMS-PP - Open channel - Send/Receive data - Send SM without More Time)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1		Test Procedure SMS-PP Data Download	as specified in 12.3.3.9.4.8
2		Test Procedure Open Channel	as specified in 12.3.3.9.4.1
3		Test Procedure Send Data	as specified in 12.3.3.9.4.2
4		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
5		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
6		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6
7		Test Procedure Send Data	as specified in 12.3.3.9.4.2
8		Test Procedure Timer Management (Start Timer)	as specified in 12.3.3.9.4.5
9		Test Procedure Send SMS	as specified in 12.3.3.9.4.7
10		Test Procedure Receive Data 1	as specified in 12.3.3.9.4.3
11		Test Procedure Timer Management (Deactivate Timer)	as specified in 12.3.3.9.4.6

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

Initial Conditions

None

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.	The Channel is correctly opened

12.3.3.11 Contents of the TERMINAL PROFILE

Direction: terminal to UICC.

Test Purpose

To verify the content 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.

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
1	User → ME	Power on the ME.	
2	ME → UICC	Send 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 (Minimum = 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)

12.3.3.12 OPEN CHANNEL - Terminal connected to Wi-Fi

Test Purpose

To verify OPEN CHANNEL for terminal connected to Wi-Fi, UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ 078

Initial Conditions

For Wi-Fi the test platform has to assure exclusive SSID which does not allow access except the DUT, same for login and password.

The DUT has to be connected to Wi-Fi

12.3.3.12.1 Test Sequence No 1: (OPEN CHANNEL, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)

Initial Conditions

Use Bearer Type '03'= default bearer for requested transport layer.

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for "Always on connection" (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration
3	ME	Disconnect ME from the first APN for "Always on connection" (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.3.3.12.1	
5	ME → UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.12.1	
7	ME → User	The ME may display channel opening information	
8	ME → USS	PDP context activation request on the cellular network	
9	USS → ME	PDP context activation accept on the cellular network	
10	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.12.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.12.1

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 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 12.3.3.12.1

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 Type

Buffer

Buffer size: 1024

12.3.3.12.2 Test Sequence No 2: (OPEN CHANNEL, Terminal connected to Wi-Fi-APN empty-GPRS Bearer Type used)

Initial Conditions

Use **GPRS Bearer Type** for requested transport layer.

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for “Always on connection” (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration
3	ME	Disconnect ME from the first APN for “Always on connection” (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.3.3.12.2	
5	ME→UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.12.2	
7	ME → User	The ME may display channel opening information	
8	ME → USS	PDP context activation request	
9	USS → ME	PDP context activation accept	
10	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.12.2	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.12.2

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 size: 1024

UICC/ME interface transport level

Transport format: UDP

Port number: 44444

Data destination address: 01.01.01.01

TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.12.2

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

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

12.3.3.13 CLOSE CHANNEL – Terminal connected to Wi-Fi

Test Purpose

To verify CLOSE CHANNEL for terminal connected to Wi-Fi, UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Initial Conditions

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

For Wi-Fi the test platform has to assure exclusive SSID which does not allow access except the DUT, same for login and password.

12.3.3.13.1 Test Sequence No 1: (CLOSE CHANNEL, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)

Initial Conditions

Use Bearer Type ‘03’ = default bearer for requested transport layer.

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for “Always on connection” (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration
3	ME	Disconnect ME from the first APN for “Always on connection” (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL12.3.3.13.1	
5	ME→UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.13.1	
7	ME → User	The ME may display channel opening information	
8	ME → USS	PDP context activation request	
9	USS → ME	PDP context activation accept	
10	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.13.1	[Command performed successfully]
11	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.3.3.13.1	
12	ME→UICC	FETCH	
13	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 12.3.3.13.1	
14	ME → USS	PDP context deactivation request	
15	USS → ME	PDP context deactivation accept	
16	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 12.3.3.13.1	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.13.1

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 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 12.3.3.13.1

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 Type

Buffer

Buffer size: 1024

PROACTIVE COMMAND: CLOSE CHANNEL 12.3.3.13.1

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 12.3.3.13.1

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

12.3.3.13.2 Test Sequence No 2: (CLOSE CHANNEL, Terminal connected to Wi-Fi-APN empty-GPRS Bearer Type used)

Initial Conditions

Use **GPRS Bearer Type** for requested transport layer.

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for “Always on connection” (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration
3	ME	Disconnect ME from the first APN for “Always on connection” (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.3.3.13.2	
5	ME→UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.13.2	
7	ME → User	The ME may display channel opening information	
8	ME → USS	PDP context activation request	
9	USS → ME	PDP context activation accept	
10	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.13.2	[Command performed successfully]

Step	Direction	Sequence	Expected Result
11	UICC → ME	PROACTIVE COMMAND PENDING: CLOSE CHANNEL 12.3.3.13.2	
12	ME→UICC	FETCH	
13	UICC → ME	PROACTIVE COMMAND: CLOSE CHANNEL 12.3.3.13.2	
14	ME → USS	PDP context deactivation request	
15	USS → ME	PDP context deactivation accept	
16	ME → UICC	TERMINAL RESPONSE CLOSE CHANNEL 12.3.3.13.2	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.13.2

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 12.3.3.13.2

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

PROACTIVE COMMAND: CLOSE CHANNEL 12.3.3.13.2

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 12.3.3.13.2

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

12.3.3.14 RECEIVE DATA – Terminal connected to Wi-Fi

Test Purpose

To verify RECEIVE DATA related to Default (network) Bearer, for terminal connected to Wi-Fi, UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Initial Conditions

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

For the Wi-Fi the test platform has to assure exclusive SSID which does not allow access except the DUT, same for login and password.

12.3.3.14.1 Test Sequence No 1: (RECEIVE DATA, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for “Always on connection” (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration
3	ME	Disconnect ME from the first APN for “Always on connection” (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST 12.3.3.14.1 PENDING	

Step	Direction	Sequence	Expected Result
5	ME→UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: SET UP EVENT LIST12.3.3.14.1	
7	ME → UICC	TERMINAL RESPONSE: SET UP EVENT LIST 12.3.3.14.1	
8	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.3.3.14.1	
9	ME→UICC	FETCH	
10	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.14.1	
11	ME → User	The ME may display channel opening information	
12	ME → USS	PDP context activation request	
13	USS → ME	PDP context activation accept	
14	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.14.1	[Command performed successfully]
15	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 12.3.3.14.1	
16	ME→UICC	FETCH	
17	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 12.3.3.14.1	
18	ME → USS	Transfer 8 Bytes of data to the USS through channel 1	[To retrieve ME's port number]
19	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 12.3.3.14.1	[Command performed successfully]
20	USS → ME	Transfer 1024 Bytes of data to the ME through channel 1 using the ME's port number, which was retrieved in step 18	
21	ME → UICC	ENVELOPE: EVENT DOWNLOAD - Data available 12.3.3.14.1	(1024 Bytes of data in the ME buffer)
22	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 12.3.3.14.1	
23	ME→UICC	FETCH	
24	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.1	205 Bytes
25	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.1	

Step	Direction	Sequence	Expected Result
26	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 12.3.3.14.2	
27	ME→UICC	FETCH	
28	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.2	205 Bytes
29	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.2	
30	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 12.3.3.14.3	
31	ME→UICC	FETCH	
32	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.3	205 Bytes
33	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.3.3.12.3.3	
34	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 12.3.3.14.4	
35	ME→UICC	FETCH	205 Bytes
36	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.4	
37	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.4	
38	UICC → ME	PROACTIVE COMMAND PENDING: RECEIVE DATA 12.3.3.14.5	
39	ME→UICC	FETCH	204 Bytes
40	UICC → ME	PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.5	
41	ME → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.5	

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.14.1

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 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 12.3.3.14.1

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 Type

Buffer

Buffer size: 1024

PROACTIVE COMMAND: SET UP EVENT LIST 12.3.3.14.1

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 12.3.3.14.1

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: SEND DATA 12.3.3.14.1

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 12.3.3.14.1

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 12.3.3.14.1

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 12.3.3.14.1

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: 205

PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.2

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: 205

PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.3

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: 205

PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.4

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: 205

PROACTIVE COMMAND: RECEIVE DATA 12.3.3.14.5

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: 204

TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.1

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 .. CC (205 Bytes of data)

Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.2

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

Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.3

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

Channel data length: FF

TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.4

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

Channel data length: CC

TERMINAL RESPONSE: RECEIVE DATA 12.3.3.14.5

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

Channel data length: 00

12.3.3.15 SEND DATA - Terminal connected to Wi-Fi

Test Purpose

To verify SEND DATA related to Default (network) Bearer, for terminal connected to Wi-Fi, UICC in client mode for UDP

Referenced requirement

- TS26_NFC_REQ_078

Initial Conditions

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

For the Wi-Fi the test platform has to assure exclusive SSID which does not allow access except the DUT, same for login and password.

12.3.3.15.1 Test Sequence No 1: (SEND DATA, Terminal connected to Wi-Fi-APN empty-Default Bearer Type used)

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	ME	Connect ME to the USS and establish the first PDN to the APN for “Always on connection” (web.network.com).	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	ME	Connect ME to the local Wi-Fi hot spot	Wi-Fi needs to be turned ON after first PDN registration

Step	Direction	Sequence	Expected Result
3	ME	Disconnect ME from the first APN for “Always on connection” (web.network.com)	
4	UICC → ME	PROACTIVE COMMAND PENDING: OPEN CHANNEL 12.3.3.15.1	
5	ME→UICC	FETCH	
6	UICC → ME	PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.15.1	
7	ME → UICC	The ME may display channel opening information	
8	ME → USS	PDP context activation request	
9	USS → ME	PDP context activation accept	
10	ME → UICC	TERMINAL RESPONSE: OPEN CHANNEL 12.3.3.15.1	[Command performed successfully]
11	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 12.3.3.15.1	
12	ME→UICC	FETCH	
13	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 12.3.3.15.1	Send 1024 Bytes of data by packets of 205 Bytes
14	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 12.3.3.15.1	[Command performed successfully]
15	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 12.3.3.15.2	
16	ME→UICC	FETCH	
17	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 12.3.3.15.2	[205 Bytes]
18	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 12.3.3.15.2	[Command performed successfully]
19	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 12.3.3.15.3	
20	ME→UICC	FETCH	
21	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 12.3.3.15.3	[205 Bytes]
22	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 12.3.3.15.3	[Command performed successfully]
23	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA 12.3.3.15.4	
24	ME→UICC	FETCH	

Step	Direction	Sequence	Expected Result
25	UICC → ME	PROACTIVE COMMAND: SEND DATA (store mode) 12.3.3.15.4	[205 Bytes]
26	ME → UICC	TERMINAL RESPONSE: SEND DATA (store mode) 12.3.3.15.4	[Command performed successfully]
27	UICC → ME	PROACTIVE COMMAND PENDING: SEND DATA12.3.3.15.5	
28	ME→UICC	FETCH	
29	UICC → ME	PROACTIVE COMMAND: SEND DATA (immediate) 12.3.3.15.5	[204 Bytes]
30	ME → USS	Transfer 1024 Bytes of data to the USS through channel 1	
31	ME → UICC	TERMINAL RESPONSE: SEND DATA (immediate) 12.3.3.15.5	[Command performed successfully]

PROACTIVE COMMAND: OPEN CHANNEL 12.3.3.15.1

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 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 12.3.3.15.1

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 Type

Buffer

Buffer size: 1024

PROACTIVE COMMAND: SEND DATA 12.3.3.15.1

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 .. CC (205 Bytes of data)

TERMINAL RESPONSE: SEND DATA 12.3.3.15.1

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 12.3.3.15.2

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

TERMINAL RESPONSE: SEND DATA 12.3.3.15.2

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 12.3.3.15.3

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

TERMINAL RESPONSE: SEND DATA 12.3.3.15.3

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 12.3.3.15.4

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

TERMINAL RESPONSE: SEND DATA 12.3.3.15.4

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: 204 bytes of space available in the Tx buffer

PROACTIVE COMMAND: SEND DATA 12.3.3.15.5

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

TERMINAL RESPONSE: SEND DATA 12.3.3.15.5

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.

12.4 Remote Management use cases

12.4.1 General overview

This section addresses testing of selected use cases for NFC services in environment with possible real data transfer in place.

12.4.2 Conformance requirements

The Requirements tested are referenced in each test case.

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

12.4.3.1.1 Test Sequence No 1: Receiving or send a SMS during BIP data transfer

Initial Conditions

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

Step	Direction	Sequence	Expected Result
1	DUT → UICC	Send Fetch OPEN CHANNEL command	

Step	Direction	Sequence	Expected Result
2	UICC → DUT	OPEN CHANNEL 1.1	
3	DUT → UICC	TERMINAL RESPONSE: OPEN CHANNEL	TR Open Channel successful + SW = 91xx
4		Fetch Send Data (CATTP SYN command for Link establishment)	TR Successful + 90 00
5		Send Event Data Available to the UICC (Reception of CATTP SYN-ACK)	91 XX
6	DUT → UICC	Fetch Receive Data	TR Successful + 91 XX
7		Fetch Send Data (ACK-PDU)	Ask server for downloading data
8	DUT → UICC	Send Event Data Available to the UICC (Reception of data from the server)	91 FF
9	DUT → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
10	DUT → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
11		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.
12	DUT → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
13		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
14		Fetch Send Data store data in Tx buffer (with 0x'zz' data)	TR Successful + 90 00
15		Send Event Data Available to the UICC	
16		Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF
17	DUT → UICC	Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
18		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
19		Fetch Send Data immediate	TR Successful + 90 00
20		Send Event Data Available to the UICC	
21		Fetch Receive Data (with 0xFF data)	TR Successful + 91 FF

Step	Direction	Sequence	Expected Result
22		Fetch Receive Data (with 0xFF data)	TR Successful + 91 yy (last Bytes)
23		Fetch Receive Data (with 0x'yy' data)	TR Successful + 91 zz
24		Fetch Send Data immediate	TR Successful + 91 xx
25		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 description

Bearer type: 03 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: 03 Default Bearer for requested transport layer

Buffer

Buffer size: 1400

12.4.3.2 OTA Data Loading

Test Purpose

Ensure that the Baseband can support the OTA data Loading

Referenced requirement

- TS26_NFC_REQ_078
- TS26_NFC_REQ_079
- TS26_NFC_REQ_081
- TS26_NFC_REQ_120

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, the DUT with a test phone number which can be called and permits to maintain the call for several minutes is necessary.
- Simulated UICC is connected to the DUT
- Prior to this test the DUT 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

Initial Conditions

Set up a network simulator for supported network technology as defined in chapter 2.5.8.

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

immediate Link establishment,

Bearer Type 03 (Default Bearer for requested transport layer),

No Alpha Identifier

Step	Direction	Sequence	Expected Result
1	Server → DUT	Perform Push SMS procedure as defined in section 12.4.3.7.1	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the	

Step	Direction	Sequence	Expected Result
		DUT's port number, which was retrieved within step 1 The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of 0xFF)	
5	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the start of the expected data from the server. 91 XX
6		During CATTP data transfer, Receive and accept an incoming voice call. Operate the Call for the whole test sequence.	Voice call established
7		Repeat steps 8 to 9 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD –Data Available commands may be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
8	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
9	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the remainder of the expected data from the server. 91 XX
10	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
11	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

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) (or other data as specified in the referencing test procedure)

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 (or other value as specified in the referencing test procedure)

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) (or other data as specified in the referencing test procedure)

Channel data length: FF (for the last TERMINAL RESPONSE: RECEIVE DATA the 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 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 VOID

Covered by section 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

Initial Conditions

Set up a network simulator for supported network technology as defined in chapter 2.5.8.

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

Immediate link establishment,

Bearer Type 03 (Default Bearer for requested transport layer)

No alpha identifier

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Perform Push SMS procedure as defined in section 12.4.3.7.1	
2	Server → DUT	Transfer of 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1 The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	

Step	Direction	Sequence	Expected Result
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of 0xFF)	
5	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the start of the expected data from the server. 91 XX
6		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
7		Repeat steps 8 to 9 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD –Data Available commands may be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
8	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
9	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the remainder of the expected data from the server. 91 XX
10	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
11	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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 VOID

Covered by section 12.4.3.2.3

12.4.3.2.5 Test Sequence No 5: BIP CAT-TP data transfer during a Voice Call is established

Initial Conditions

Set up a network simulator for supported network technology as defined in chapter 2.5.8.

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

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 Receiving and accepting an incoming voice call over 2G/3G. Operate the call for the whole test sequence.	Voice call established
2	Server → DUT DUT → Server	Perform Push SMS procedure as defined in section 12.4.3.7.1	
3	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 2. The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
4	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of	91 XX

Step	Direction	Sequence	Expected Result
		data from the server, 60K Bytes of data in the DUT buffer)	
5		Repeat steps 6 to 7 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD –Data Available commands may be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
6	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
7	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the expected data from the server. 91 XX
8	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
9	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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 VOID

Covered by section 12.4.3.2.5

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

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 DUT shall have been powered ON and ISO7816 initialization has been completed.

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 supported network technology as defined in chapter 2.5.8.

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

Test UICC should be configured to No PoR.

Immediate Link establishment,

Bearer Type 03 (Default Bearer for requested transport layer)

No alpha identifier

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT→ Server	Perform Push SMS procedure as defined in section 12.4.3.7.1	

Step	Direction	Sequence	Expected Result
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1 The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4		Repeat steps 5 to 6 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD –Data Available commands shall be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
5	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
6	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the expected data from the server. 91 XX
7	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
8	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically:

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 supported network technology as defined in section 2.5.8.

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

immediate link establishment,

Bearer Type 03 (Default Bearer for requested transport layer)

No alpha identifier

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT→ Server	Perform Push SMS procedure as defined in section 12.4.3.7.2	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1 The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4		Repeat steps 5 to 6 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD – Data Available commands should be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
5	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
6	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the expected data from the server. 91 XX

Step	Direction	Sequence	Expected Result
7	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
8	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically:

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 VOID

12.4.3.3.4 VOID

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 supported network technology as defined in chapter 2.5.8.

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

Immediate link establishment,

Bearer Type 03 (Default Bearer for requested transport layer)

No alpha identifier.

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Perform Push SMS procedure with SPI ’12 22’ as defined in section 12.4.3.7.1 with SPI set to ’12 22’ (PoR only on error)	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1.	

Step	Direction	Sequence	Expected Result
		The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4		Repeat steps 5 to 6 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD – Data Available commands should be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
5	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
6	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the expected data from the server. 91 XX
7	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
8	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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

12.4.3.4.1 Test Sequence No 1

Initial Conditions

Set up a network simulator supported network technology as defined in chapter 2.5.8.

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

- 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 has 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.

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Perform Push SMS procedure as defined in section 12.4.3.7.1 or 12.4.3.7.2	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1. The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of 0xFF)	
5	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the start of the expected data from the server. 91 XX

Step	Direction	Sequence	Expected Result
6		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
7		Repeat steps 8 to 9 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD – Data Available commands may be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
8	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
9	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the remainder of the expected data from the server. 91 XX
10	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
11	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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.

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

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 DUT 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 supported network technology as defined in chapter 2.5.8.

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

Step	Direction	Sequence	Expected Result
1	Server → DUT DUT → Server	Perform Push SMS procedure as defined in section 12.4.3.7.1 or 12.4.3.7.2	
2	Server → DUT	Transfer 60k Bytes of data to the DUT through channel 1 using the DUT's port number, which was retrieved within step 1. The data shall be constructed such that each portion of the data can be unambiguously identified when received by the UICC.	
3	DUT → UICC	ENVELOPE: EVENT DOWNLOAD – Data Available (Reception of data from the server, 60K Bytes of data in the DUT buffer)	91 XX
4	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of 0xFF)	
5	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the start of the expected data from the server.

Step	Direction	Sequence	Expected Result
			91 XX
6		Send SMS from DUT to test phone number. Receive SMS from test phone number on DUT.	SMS Sent and Received
7		Repeat steps 8 to 9 until the complete 60k Bytes of data have been received by the UICC. Additional ENVELOPE: EVENT DOWNLOAD – Data Available commands should be sent by the DUT in between successive PROACTIVE COMMAND: Receive Data commands.	
8	UICC → DUT	PROACTIVE COMMAND: Receive Data 12.1 (with channel data length of YY according to the amount of data available)	
9	DUT → UICC	TERMINAL RESPONSE: RECEIVE DATA 12.1	TR Successful Channel data contains the remainder of the expected data from the server. 91 XX
10	UICC → DUT	PROACTIVE COMMAND: CLOSE CHANNEL 12.1	
11	DUT → UICC	TERMINAL RESPONSE: CLOSE CHANNEL 12.1	[Command performed successfully] TR Successful + 90 00

Logically

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.

12.4.3.6 VOID

12.4.3.7 Link Establishment using Push SMS

Test Purpose

To ensure that the device establishes a connection to a given remote management server on request by an appropriate Push SMS

Referenced requirement

- TS26_NFC_REQ_081

12.4.3.7.1 Test Procedure: OPEN CHANNEL on receiving Push SMS without PoR

Initial Conditions

Set up a network simulator for LTE, terminal is authenticating against LTE.

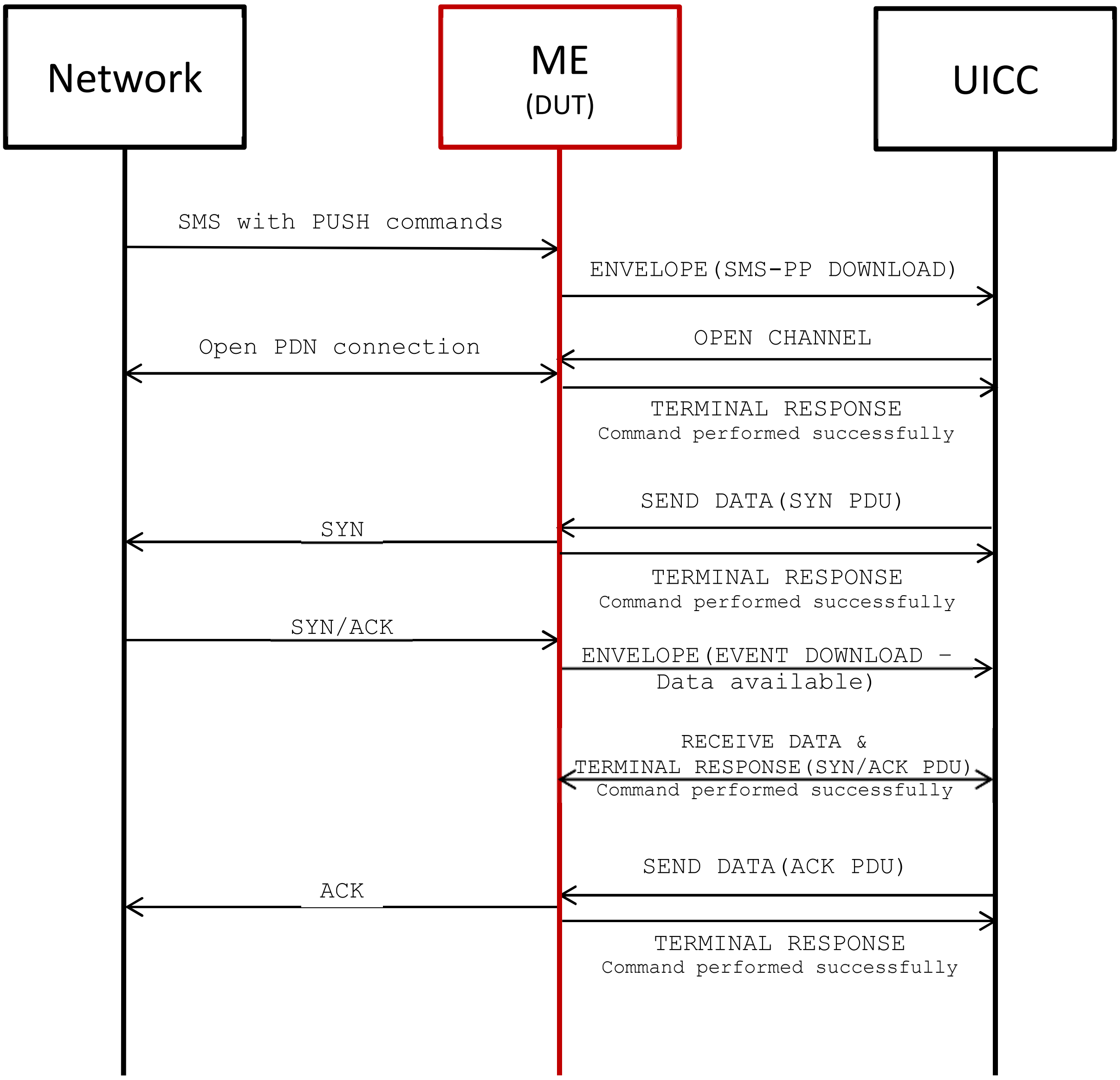
If PUSH SMS is sent through IMS over LTE: Set up an IMS server, DUT is registering with IMS in order to receive and send SMS.

If PUSH SMS is sent through 2G/3G network: Set up a network simulator for 2G/3G, terminal is authenticating against 2G/3G in order to receive and send SMS.

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

Test UICC should send no PoR.

Test Environment



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	Network → DUT	Send "PUSH SMS" (content see below) with two commands: • "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 • "Request for CAT_TP link establishment" as defined in TS 102 226 [23] and TS 102 127 [24]	ENVELOPE(SMS-PP DOWNLOAD), forwarding the PUSH SMS to the UICC
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 → DUT	PROACTIVE COMMAND: OPEN CHANNEL 12.1	• Open PDN connection may take place • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
7	UICC → DUT	PROACTIVE COMMAND: SEND DATA 12.1 (SYN PDU)	• Transfer data to the network through channel 1 to retrieve DUT's port number. • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
8	Network → DUT	Transmit SYN/ACK Packet	ENVELOPE(EVENT DOWNLOAD - Data available 12.1)
9	UICC → DUT	PROACTIVE COMMAND: RECEIVE DATA 12.1 (SYN/ACK PDU)	TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
10	UICC → DUT	PROACTIVE COMMAND: SEND DATA 12.1 (ACK PDU)	• Transfer ACK Packet into network • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1

Logically

<u>PROACTIVE COMMAND: OPEN CHANNEL 12.1</u>

Logically:

Command details

Command number: 1

Command type: OPEN CHANNEL

Command qualifier: immediate link establishment

Device identities

Source device: UICC

Destination device: DUT

Bearer description

Bearer type: 03 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: 03 Default Bearer for requested transport layer

Buffer

Buffer size: 1400

Same as PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 in clause 11.3.3.1.

Same as TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1 in clause 11.3.3.1.

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 DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE DATA 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.

12.4.3.7.2 Test Procedure: OPEN CHANNEL on receiving Push SMS, with PoR

Initial Conditions

Set up a network simulator for LTE, terminal is authenticating against LTE.

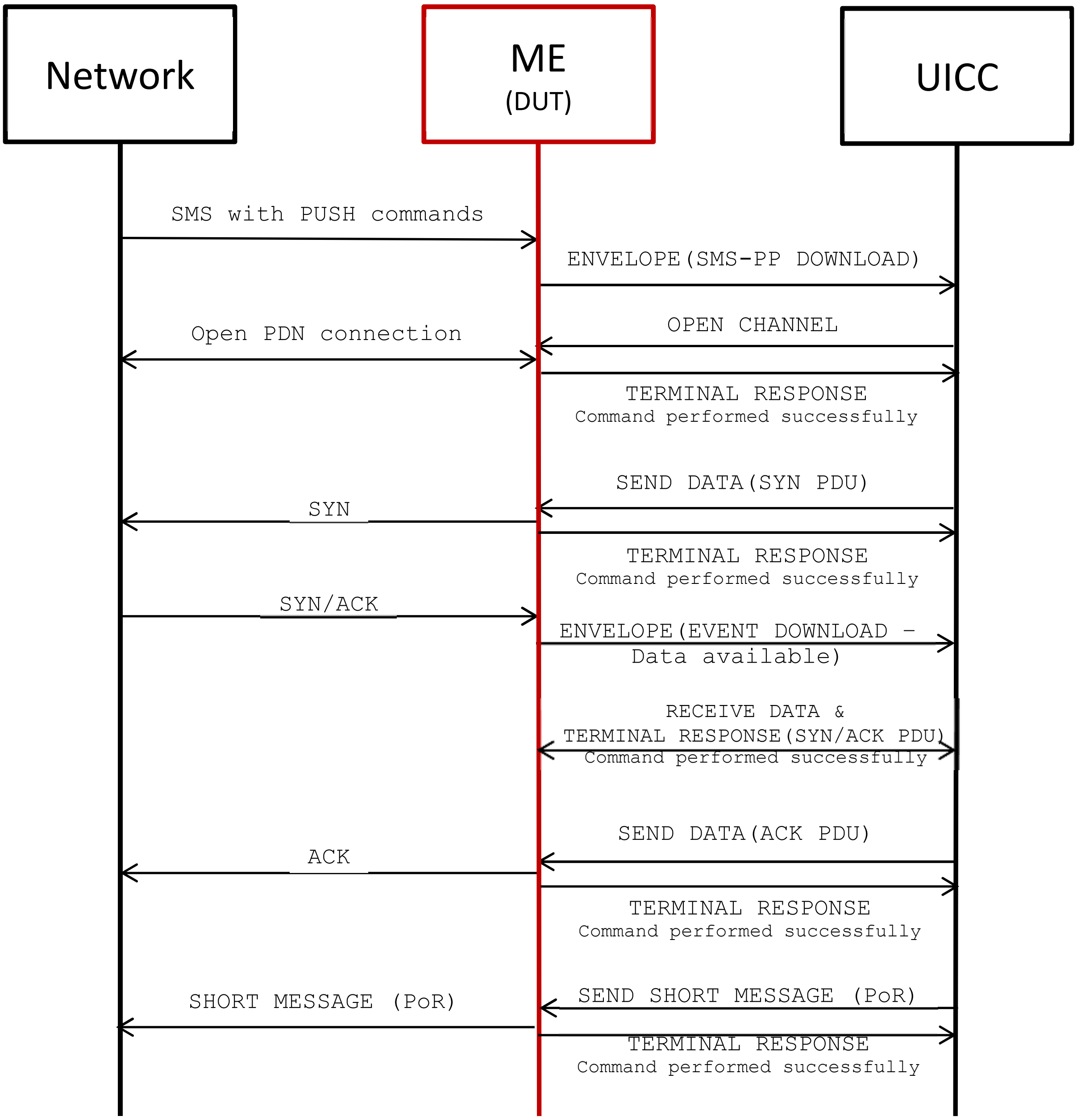
If PUSH SMS is sent through IMS over LTE: Set up an IMS server, DUT is registering with IMS in order to receive and send SMS.

If PUSH SMS is sent through 2G/3G network: Set up a network simulator for 2G/3G, terminal is authenticating against 2G/3G in order to receive and send SMS.

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

Test UICC should send PoR.

Test Environment



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	Network → DUT	Send "PUSH SMS" (content see below) with two commands: • "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 • "Request for CAT_TP link establishment" as defined in TS 102 226 [23] and TS 102 127 [24]	ENVELOPE(SMS-PP DOWNLOAD), forwarding the PUSH SMS to the UICC
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 → DUT	PROACTIVE COMMAND: OPEN CHANNEL 12.1	• Open PDN connection may take place • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
7	UICC → DUT	PROACTIVE COMMAND: SEND DATA 12.1 (SYN PDU)	• Transfer data to the network through channel 1 to retrieve DUT's port number. • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
8	Network → DUT	Transmit SYN/ACK Packet	ENVELOPE(EVENT DOWNLOAD – Data Available 12.1)
9	UICC → DUT	PROACTIVE COMMAND: RECEIVE DATA 12.1 (SYN/ACK PDU)	TERMINAL RESPONSE (Command performed successfully, '91 xx')
10	UICC → DUT	PROACTIVE COMMAND: SEND DATA 12.1 (ACK PDU)	• Transfer ACK Packet into network • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1
11	UICC → DUT	PROACTIVE COMMAND: SEND SHORT MESSAGE (PoR)	• Transfer PoR into network • TERMINAL RESPONSE (Command performed successfully, '91 xx') 12.1

Logically

Same as PROACTIVE COMMAND: SET UP EVENT LIST 11.1.1 in clause 11.3.3.1.

Same as TERMINAL RESPONSE: SET UP EVENT LIST 11.1.1 in clause 11.3.3.1.

Same as PROACTIVE COMMAND: OPEN CHANNEL 12.1 in clause 12.4.3.7.1.

Same as TERMINAL RESPONSE: OPEN CHANNEL 12.1 in clause 12.4.3.7.1.

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 DATA 12.1 in clause 12.4.3.2.1.

Same as TERMINAL RESPONSE: RECEIVE DATA 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.

13.2 Conformance requirements

The Requirements tested are referenced in each test case.

13.3 Test Cases

13.3.1 Secure Element Access API 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

Initial Conditions

An instance of the UICC application **APDU_TestApplication.cap** with AID01 is selectable.

MobileApplication is installed on the DUT and implements a function “Select AID01”. The application is signed and respectively named:

- **GSMA_Mobile_App_SP1_signed** signed with a test certificate #1

13.3.1.1 Test Sequence No 1: Standard

Initial Conditions

The DUT is in Radio OFF state (e.g. Flight Mode, Airplane Mode, etc.)

Step	Direction	Sequence	Expected Result
1	User → DUT	Launch **GSMA_Mobile_App_SP1_signed**	None

Step	Direction	Sequence	Expected Result
2	DUT→ UICC	Call the “Select AID01” 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 = ‘AID01’; Le=00, or empty
3	UICC → DUT	UICC successfully opens a new logical channel to AID01	Expected result of the API called: call to “Select AID01” function returns successfully
4	DUT → UICC	Send **APDU Case 4** P1 = 0x01	The expected C-APDU: C-APDU (’XX 04 01 00 FF’ <Data field of 255 bytes> FF)
5	UICC → DUT	**APDU_TestApplication.cap** returns: P1 = 0x01 R-APDU – <data field of 255 bytes> SW1-SW2	Expected result of the API called: R-APDU - data field of 255 bytes, SW1, SW2

13.3.1.2 VOID

13.3.1.3 VOID

13.3.1.4 Test Sequence No 4: After reboot

Initial Conditions

The DUT is in Radio OFF state (e.g. Flight mode, Airplane Mode, etc.)

Step	Direction	Sequence	Expected Result
1	User → DUT	Power off the DUT	None
2	User → DUT	Power on the DUT	None
3	User → DUT	Launch **GSMA_Mobile_App_SP1_signed**	None

Step	Direction	Sequence	Expected Result
4	DUT→ UICC	Call the “Select AID01” 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 = ‘AID01’; Le=00, or empty
5	UICC → DUT	UICC successfully opens a new logical channel to AID01	Expected result of the API called: call to “Select AID01” function returns successfully
6	DUT → UICC	Send **APDU Case 4** P1 = 0x01	The expected C-APDU: C-APDU (’XX 04 01 00 FF’ <Data field of 255 bytes> FF)
7	UICC → DUT	**APDU_TestApplication.cap** returns: P1 = 0x01 R-APDU – <data field of 255 bytes> SW1-SW2	Expected result of the API called: R-APDU - data field of 255 bytes, SW1, SW2

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_109

Initial Conditions

- ReferenceApplication.cap managing the reference transaction with AID_REF selectable into the reference UICC.
- APDU Application to send APDUs according to the reference transaction.
- Set the DUT to “Radio Off”

13.3.2.1 Test Sequence No 1: Enable, disable

Step	Direction	Sequence	Expected Result
1	User → DUT	Enable NFC on the DUT, if not enabled	None
2	User → DUT	Check in the Wireless Settings option if it sets the current state of NFC to "Enabled"	“NFC enabled” indication is present

Step	Direction	Sequence	Expected Result
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	User → DUT	Disable NFC on the DUT	None
6	User → DUT	Check in the Wireless Settings option if the DUT changes the current state of NFC to "Disabled"	"NFC enabled" indication is absent
7		Perform the reference transaction using a contactless reader	Reference transaction is not performed
8		Try to read a NFC tag	NFC Tag is not read
9	User → DUT	Enable NFC on the DUT	None
10		Perform the reference transaction using a contactless reader	Reference transaction is performed successfully
11		Try to read a NFC tag	NFC Tag is read successfully

13.3.2.2 Test Sequence No 2: Persistence after reboot

Initial Conditions

Step	Direction	Sequence	Expected Result
1	User → DUT	Enable NFC on the DUT, if not enabled	None
2	User → DUT	Check in the Wireless Settings option if it sets the current state of NFC to "Enabled"	"NFC enabled" indication is present
3	User → DUT	Power off the DUT	None
4	User → DUT	Power on the DUT. Check in the Wireless Settings option if it sets the current state of NFC to "Enabled"	"NFC enabled" indication is still present
5	User → DUT	Launch **GSMA_Mobile_App_SP1_signed**	None
6	DUT → UICC	Call "Select AID1" function	Call is successful

13.3.3 Modem and UICC over APDU exchange

Test Purpose

To ensure the Modem support APDU exchange to access UICC for cases 1, 2, 3 & 4 as defined in ISO/IEC 7816-4.

Referenced requirement

- TS26_NFC_REQ_113

Initial Conditions

None

13.3.3.1 Test Sequence No 1

Following Test Cases in Table B.1.2 6.3.1.6.5.6 (transmit(byte[] command)) from Open Mobile API test specification SHALL be passed:

- Test cases ID2 to ID16
- Test cases ID18 to ID21
- Test cases ID23

If the test cases referenced in Table B.1.2 6.3.1.6.5.6 are already referenced in certification programs, then this test sequence should not be referenced in the certification programs.

13.3.4 Modem retrieves the response data to the SELECT command

Test Purpose

To ensure the Modem provides a way for the application processor to retrieve the answer from the UICC after the selection of an AID.

Referenced requirement

- TS26_NFC_REQ_141

Initial Conditions

None

13.3.4.1 Test Sequence No 1: Modem retrieves the response data to the SELECT command

Following Test Cases in Table B.1.2 6.3.1.6.5.4 (getSelectResponse) from Open Mobile API test specification SHALL be passed:

- Test cases ID1, ID2, ID4, ID5, ID6, ID7, ID8

If the test cases referenced in Table B.1.2 6.3.1.6.5.6 are already referenced in certification programs, then this test sequence should not be referenced in the certification programs.

13.3.5 Modem supports 19 logical channels

Test Purpose

To ensure the Modem support 19 logical channels in addition to the basic channel.

Referenced requirement

- TS26_NFC_REQ_142

Initial Conditions

None

13.3.5.1 Test Sequence No 1: Modem supports 19 logical channels

Following Test Cases in Table B.1.2 6.3.1.6.4.8 (openLogicalChannel – Extended logical channels) from Open Mobile API test specification SHALL be passed:

- Test cases ID1, ID2, ID3

If the test cases referenced in Table B.1.2 6.3.1.6.5.6 are already referenced in certification programs, then this test sequence should not be referenced in the certification programs.

13.3.6 Long APDU handling

Test Purpose

To ensure the modem of the DUT handle correctly long APDU

There are 2 ways to handle them, either long APDU are segmented in several smaller segments, or the Modem & SIM Card both support Extended Length APDU and the APDU can be exchanged within one segment.

Referenced requirement

- TS26_NFC_REQ_113
- TS26_NFC_REQ_141
- TS26_NFC_REQ_161

13.3.6.1 Test Sequence No 1: Get Response APDU segmented from UICC (Case2 Command)

Referenced requirement

- TS26_NFC_REQ_113
- TS26_NFC_REQ_141

Initial Conditions

The Applet returns a response of 2500 bytes length to the command sent, where the UICC uses SW = '61XX' multiple times in order to send the response.

App1: An application capable of sending a short APDU Case2 command to the Applet.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App1 on the DUT	Installation is successful
2	App → UICC	Select the Applet	Applet is successfully selected
3	App → UICC	App1 send a case2 short APDU command to the Applet	APDU command is sent
4	UICC → App	Applet answer with a 2500 bytes response	2500 bytes are received by App1 and SW: 90 00 at the end

13.3.6.2 Test Sequence No 2: Get Response APDU segmented from UICC (Case4 Command)

Referenced requirement

- TS26_NFC_REQ_113
- TS26_NFC_REQ_141

Initial Conditions

The Applet return a response of 2500 bytes length to the command sent, where the UICC uses SW = '61XX' multiple times in order to send the response.

App1: An application capable of sending a short APDU Case4 command to the Applet.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App1 on the DUT	Installation is successful
2	App → UICC	Select the Applet	Applet is successfully selected
3	App → UICC	App1 send a case4 short APDU command to the Applet	APDU command is sent
4	App → UICC	Applet answer with a 2500 bytes response	2500 bytes are received by App1 and SW: 90 00 at the end

13.3.6.3 Test Sequence No 3: Long APDU answer from UICC (case 2E command)

Referenced requirement

- TS26_NFC_REQ_113
- TS26_NFC_REQ_141
- TS26_NFC_REQ_161

Initial Conditions

The UICC used for the testing SHALL support extended length APDU

App1: An application capable of sending an APDU case 2E command to the Applet.

The APDU is defined like this: CLA INS P1 P2 Le

Where Le is “00 08 00” (2048 bytes)

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App1 on the DUT	Installation is successful
2	App → UICC	Select the Applet	Applet is successfully selected
3	App → UICC	App1 send case 2E APDU command to the applet with Le “00 08 00”	APDU command is sent
4	UICC → App	Applet answer a 2048 bytes response	2048 bytes are received by app1 followed by SW: 90 00 at the end

13.3.6.4 Test Sequence No 4: Long APDU command + answer from UICC (case 4E command)

Referenced requirement

- TS26_NFC_REQ_113
- TS26_NFC_REQ_141
- TS26_NFC_REQ_161

Initial Conditions

The UICC used for the testing SHALL support extended length APDU

The applet hosted on the UICC returns a response of 2048 bytes length to the command sent

App1: An application capable of sending a long APDU case 4E command to the Applet.

The APDU is defined like this: CLA INS P1 P2 Lc Nc data bytes Le

Where:

- Lc is 00 08 00 (2048 bytes)
- Nc is 2048 bytes length
- Le is “08 00” (2048 bytes)

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App1 on the DUT	Installation is successful
2	App → UICC	Select the Applet	Applet is successfully selected
3	App → UICC	App1 send case 4E APDU command to the applet with Lc “00 08 00” and Le “08 00”	APDU command is sent
4	UICC → App	Applet answer a 2048 bytes response	2048 bytes are received by app1 followed by SW: 90 00 at the end

13.3.7 Terminal Capability TAG 82

Test Purpose

To ensure that during the initialisation of the UICC, the DUT indicates that it supports an SWP link as specified by ETSI TS 102 221 [8]

Referenced requirement

- TS26_NFC_REQ_006
- TS26_NFC_REQ_166

13.3.7.1 Test Sequence No 1: Terminal Capability TAG 82

Initial Conditions

Device is powered off

Step	Direction	Sequence	Expected Result
1	User→DUT	Power on the device If needed enter the PIN code of the UICC	The device is powered on.
2	User →DUT	Inspect the initialization parameters exchanged between the DUT and the UICC. Verify the content of the “TERMINAL CAPABILITY "82" TAG“ sent by the DUT to the UICC.	TERMINALCAPABILITY "82" TAG “is set to “UICC-CLF”: Supported Value of b1 is “1”. See ETSI TS 102 221 Clause 11.1.19.2.3

13.3.8 Reselect previously non-existing applet

Test Purpose

Ensure that after an applet is loaded to the UICC, the selection of the applet is possible without rebooting the device

Referenced requirement

- TS26_NFC_REQ_047

13.3.8.1 Test Sequence No 1: Select non existing applet, deploy applet, select existing applet

Prepare an applet1 identified by AID1 to be installed on the UICC

Initial Conditions

- Applet1 identified by AID06 does not exist on the UICC
- MobileApplication implements the “Select AID06” function.
- MobileApplication is installed on the DUT

Step	Direction	Sequence	Expected Result
1	User→DUT	Launch Mobile Application	none
2	User→DUT	Call the “Select AID06” function	SELECT by AID command with “AID06” is received by the UICC over ISO7816 interface.
3	UICC-→DUT	UICC answer “6A82” to the SELECT by AID command received in Step2	Selecting AID06 fails
4		Deploy applet1 to the UICC	Applet1 is deployed successfully
5	User →DUT	Call the “Select AID06” function	SELECT by AID command with “AID06” is received by the UICC over ISO7816 interface.
6	UICC-→DUT	UICC answer “9000” to the SELECT by AID command received in Step5	Selecting AID06 succeeds

13.3.9 Retrieve CIN and IIN from eSE ISD by mobile application

Test Purpose

To ensure that during the CIN and IIN on the ISD of the eSE are personalized and can be retrieved by a mobile application.

Referenced requirement

- TS26_NFC_REQ_183
- TS26_NFC_REQ_185

Note: these REQs are included in TS26 v12

13.3.9.1 Test Sequence No 1: CIN, IIN retrieval from eSE

Initial Conditions

App2 is installed on the DUT and implements a function “Select by AID_ZERO_LENGTH”.

Note: The “Select by AID_ZERO_LENGTH” function selects the ISD on the eSE.

The Mobile Application is capable of sending GET DATA command to the eSE.

Step	Direction	Sequence	Expected Result
1	App → eSE	Select the ISD on the eSE	ISD is successfully selected
2	App → eSE	App2 send a GET DATA command with P1=00, P2=45 to the ISD to retrieve the CIN	APDU command is sent
3	eSE → App	ISD answers with CIN in the response	CIN is received by App2. The value of the returned CIN equals to the value of Item 3 in Table 2.7.
4	App → eSE	App2 send a GET DATA command with P1=00, P2=42 to the ISD to retrieve the IIN	APDU command is sent
5	eSE → App	ISD answers with IIN in the response	IIN is received by App2. The value of the returned IIN equals to the value of Item 2 in Table 2.7.

14 VOID

15 Android specific test cases

15.1 General overview

This chapter addresses test cases which are related to Android specific requirements.

15.2 Conformance requirements

The Requirements tested are referenced in each test case.

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

15.3.2 Conformance requirements

The Requirements tested are referenced in each test case.

15.3.3 Test Cases

15.3.3.1 Shared library

For further test details, see TS.27 v13.0 [47]

15.3.3.1.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

15.3.3.2 Getting an instance of the “NFC Controller” class

For further test details, see TS.27 v13.0 [47]

15.3.3.2.1 Test Sequence No 1: getDefaultController, valid usage

For further test details, see TS.27 v13.0 [47]

15.3.3.2.2 Test Sequence No 2: getDefaultController, null callback

For further test details, see TS.27 v13.0 [47]

15.3.3.3 Getting the status the NFC Controller

For further test details, see TS.27 v13.0 [47]

15.3.3.3.1 Test Sequence No 1: isEnabled, NFC enabled

For further test details, see TS.27 v13.0 [47]

15.3.3.3.2 Test Sequence No 2: isEnabled, NFC disabled

For further test details, see TS.27 v13.0 [47]

15.3.3.4 Enabling the NFC Controller

For further test details, see TS.27 v13.0 [47]

15.3.3.4.1 Test Sequence No 1: enableNfcController, confirmation dialog cancelled

For further test details, see TS.27 v13.0 [47]

15.3.3.4.2 Test Sequence No 2: enableNfcController, confirmation dialog confirmed

For further test details, see TS.27 v13.0 [47]

15.3.3.4.3 Test Sequence No 3: enableNfcController, NFC already enabled

For further test details, see TS.27 v13.0 [47]

15.4 Accessing the Secure Elements

15.4.1 General overview

This section provides test cases related to the Secure Element access.

15.4.2 Conformance requirements

The Requirements tested are referenced in each test case.

15.4.3 Test Cases

15.4.3.1 Open Mobile Service Layer API

Test Purpose
To ensure that OMAPI service layer API are not available and accessible from the android application.

Referenced requirement

- TS26_NFC_REQ_124
- TS26_NFC_REQ_186

Initial Conditions

- The DUT is powered on

15.4.3.1.1 Test Sequence No 1

Initial Conditions

None

Step	Direction	Sequence	Expected Result
1	App → DUT	Using reflection (“Class.forName”), check if any of the requested OMAPI service layer classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.Provider	Exception “ClassNotFoundException” is thrown
2	App → DUT	Using reflection (“Class.forName”), check if any of the requested OMAPI service layer (Discovery API) classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.SEDiscovery • org.simalliance.openmobileapi.SERecognizer • org.simalliance.openmobileapi.SERecognizerByAID • org.simalliance.openmobileapi.SERecognizerByATR • org.simalliance.openmobileapi.SERecognizerByHistoricalBytes	Exception “ClassNotFoundException” for each class is thrown
3	App → DUT	Using reflection (“Class.forName”), check if any of the requested OMAPI service layer (File Management API) classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.FileViewProvider • org.simalliance.openmobileapi.FileViewProvider.FCP • org.simalliance.openmobileapi.FileViewProvider.Record	Exception “ClassNotFoundException” for each class is thrown
4	App → DUT	Using reflection (“Class.forName”), check if any of the requested OMAPI service layer (Authentication Service API) classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.AuthenticationProvider • org.simalliance.openmobileapi.AuthenticationProvider.PinID	Exception “ClassNotFoundException” for each class is thrown

Step	Direction	Sequence	Expected Result
5	App → DUT	Using reflection ("Class.forName"), check if any of the requested OMAPI service layer (PKCS# 15 Service API) classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.PKCS15Provider • org.simalliance.openmobileapi.PKCS15Provider.Path	Exception "ClassNotFoundException" for each class is thrown
6	App → DUT	Using reflection ("Class.forName"), check if any of the requested OMAPI service layer (Secure Storage API) classes are available in the DUT or not. List of OMAPI Service Layer classes: • org.simalliance.openmobileapi.SecureStorageProvider	Exception "ClassNotFoundException" for each class is thrown

15.4.3.2 Access to GlobalPlatform OMAPI after the device is booted

Test Purpose

To ensure that an application has access to the SE through the OMAPI right after the BOOT_COMPLETED event is received

Referenced requirement

- TS26_NFC_REQ_125

15.4.3.2.1 Test Sequence No 1: OM API access after boot, ARA

Initial Conditions

- An instance of the UICC application **APDU_TestApplication.cap** with AID 01 is selectable and is installed on to the UICC
- **GSMA_Mobile_App_BOOT#1** application signed with a private key corresponding to test certificate #1 and implementing a function "Select AID 01" using the openLogicalChannel() method for the UICC application AID 01
- **GSMA_Mobile_App_BOOT#1** defines a broadcastReceiver as follows
- Registers in its Manifest the following permissions:

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
```

- Define a "BroadcastReceiver" as follows

```
<receiver android:name=".BootUpReceiver">
 <intent-filter>
  <action android:name=
   "android.intent.action.BOOT_COMPLETED"></action>
 </intent-filter>
</receiver>
```


- When the activity **GSMA_Mobile_App_BOOT#1** receive the intent "BOOT_COMPLETED" the activity will send "select AID 01" and "select AID 02" to the UICC immediately
- **GSMA_Mobile_App_BOOT#1** is installed on the DUT
- Access Control is authorizing [**GSMA_Mobile_App_BOOT#1**] to access the applet "**APDU_TestApplication.cap**" on the UICC using AID 01 and preventing access to the applet using AID 02 ..
- The Access Control is using ARA mechanism.

The DUT is powered off.

Step	Direction	Sequence	Expected Result
1	User → DUT	Power on the DUT	Activity listening "BOOT_COMPLETED" is launched.
2	App → UICC	Call "Select AID 01" function	SELECT command is successful and call to "Select AID 01" function returns successfully
3	App → UICC	Call "Select AID 02" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted

15.4.3.2.2 Test Sequence No 2: OM API access after boot, ARF

Initial Conditions

- An instance of the UICC application **APDU_TestApplication.cap** with AID 01 is selectable and is installed on to the UICC

  GSMA_Mobile_App_BOOT#1 application signed with a private key corresponding to test certificate #1 and implementing a function "Select AID 01" using the openLogicalChannel() method for the UICC application AID 01
- **GSMA_Mobile_App_BOOT#1** defines a broadcastReceiver as follows
- Registers in its Manifest the following permissions:

```
<uses-permission android:name="android.permission.RECEIVE_BOOT_COMPLETED" />
```

- Define a "BroadcastReceiver" as follows

```
<receiver android:name=".BootUpReceiver">
 <intent-filter>
  <action android:name=
   "android.intent.action.BOOT_COMPLETED"></action>
 </intent-filter>
</receiver>
```

  - When the activity **GSMA_Mobile_App_BOOT#1** receive the intent "BOOT_COMPLETED" the activity will send "select AID 01" and "select AID 02" to the UICC immediately.

- **GSMA_Mobile_App_BOOT#1** is installed on the DUT

- Access Control is authorizing [**GSMA_Mobile_App_BOOT#1**] to access the applet "**APDU_TestApplication.cap**" on the UICC using AID 01 and preventing access to the applet using AID 02.
- The Access Control is using ARF mechanism.

- The DUT is powered off

Step	Direction	Sequence	Expected Result
1	User → DUT	Power on the DUT	Activity listening "BOOT_COMPLETED" is launched.
2	APP→ UICC	Call "Select AID 01" function	SELECT command is successful and call to "Select AID 01" function returns successfully
3	APP → UICC	Call "Select AID 02" function	Call is unsuccessful, returning an error indicating that the access control rights are not granted.

15.4.3.3 Getting the list of available Secure Elements

For further test details, see TS.27 v13.0 [47]

15.4.3.3.1 Test Sequence No 1: Handset::getavailableSecureElements

For further test details, see TS.27 v13.0 [47]

15.4.3.4 Identical SE Names across device components

To ensure that the framework is using the same Secure Element names across device components.

Referenced requirement

- TS26_NFC_REQ_069
- TS26_NFC_REQ_144

15.4.3.4.1 Test Sequence No 1: Usage of identical SE Names across device components

For further test details, see TS.27 v13.0 [47]

15.4.3.4.2 Test Sequence No 2: Usage of identical SE Names across device components (without using GSMA API)

Initial Conditions

- Application [app01]
- Provides the following features
  - defines an "Off-Host" service [myOffHostService] in its Manifest.
    - with group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
```

```
android:category="payment">
<aid-filter android:name="AID01"/>
</aid-group>
```

- service [myOffHostService] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService>
</offhost-apdu-service>
```

- Retrieves the list of readers using GlobalPlatform OMAPI
- Displays a notification when a transaction event is received
- The notification displays the Secure Element name at the origin of the event

- Applet with [AID01] as AID is installed on the UICC
- Access Control is allowing communication between any applets in the UICC and [app01]

Step	Direction	Sequence	Expected Result
1	APP → DUT	Using GlobalPlatform APIs, get the list of available readers	The name of one of the returned reader is equal to “SIM” or “SIM1” and this string is stored in [SEName]
2	User → DUT	From the “Setting” menu open the “Tap&Pay” entry	List of entries contains a banner where “myOffHostService” is displayed
3	User → DUT	Select entry with “myOffHostService” banner	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with [AID01] as parameter	SW = 90 00 is returned

Step	Direction	Sequence	Expected Result
7	UICC → DUT DUT → APP	Send a transaction event from an applet in the UICC	• A notification linked to the transaction event is displayed by [app01] • The Secure Element is displayed by the notification is equal to the one stored in [SEName]

15.5 NFC Transaction Events

15.5.1 General overview

This section provides test cases for checking reception of NFC Transaction events.

15.5.2 Conformance requirements

The Requirements tested are referenced in each test case.

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 broadcast accordingly “EVT_Transaction” to several applications.

Referenced requirement

- TS26_NFC_REQ_098
-

15.5.3.1.1 Test Sequence No 1: enableMultiEvt_transactionReception, not authorised

Initial Conditions

- PKCS#15 structure on the UICC:
    - A certificate [cert01] is stored in a file [EF01]
    - A EF_CDF file [EF02] as defined by PKCS#15 specification is referencing [EF01]
    - EF_ODF file contains a reference to [EF02] for the “A5” type
- Application [app01] is signed with a 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	App→DUT	From the “Handset” instance, call “enableMultiEvt_transactionReception” method	“SecurityException” is thrown by the system

15.5.3.1.2 Test Sequence No 2: enableMultiEvt_transactionReception, enable, reboot

Initial Conditions

- PKCS#15 structure on the UICC:
  - A certificate [cert01] is stored in a file [EF01]
  - A EF_CDF file [EF02] as defined by PKCS#15 specification is referencing [EF01]
  - EF_ODF file contains a reference to [EF02] for the “A5” type
- Access Control is allowing communication between any applets in the UICC and [app01], [app02] and [app03]
- Application [app01]
  - Defines an “activity” which listens an “intent-filter” as follows

```
<intent-filter>
    <action android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT" />
    <category android:name="android.intent.category.DEFAULT" />
    <data
            android:host="secure"
            android:pathPrefix="/SIM"
            android:port="0"
            android:scheme="nfc"/>
</intent-filter>
```

- Applications [app02] and [app03]
  - Signed with certificate [cert01]
  - Define a “BroadcastReceiver” which an “intent-filter” as follows

```
<intent-filter>
    <action android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT" />
    <category android:name="android.intent.category.DEFAULT" />
    <data
            android:host="secure"
            android:pathPrefix="/SIM"
            android:port="0"
            android:scheme="nfc"/>

</intent-filter>
```

- An instance of the “Handset” class has been retrieved successfully and is not *null*

Step	Direction	Sequence	Expected Result
1	App→DUT	From the “Handset” instance, call “enableMultiEvt_transactionReception” method from [app02]	No exception thrown by the system
2	UICC→ DUT	Send a transaction event from an applet in the UICC	• [app02] and [app03] are receiving the event • [app01] is not receiving the event
3	App→DUT	Restart DUT	

Step	Direction	Sequence	Expected Result
4	UICC →DUT	Send transaction event from an applet in the UICC	• [app01] is receiving the event • [app02] and [app03] are not receiving the event

15.5.3.2 Receiving EVT_Transaction from “BroadcastReceiver”

For further test details, see TS.27 v13.0 [47]

15.5.3.2.1 Test Sequence No 1: enableMultiEvt_transactionReception, path pattern

For further test details, see TS.27 v13.0 [47]

15.5.3.2.2 Test Sequence No 2: enableMultiEvt_transactionReception, path prefix

For further test details, see TS.27 v13.0 [47]

15.5.3.2.3 Test Sequence No 3: enableMultiEvt_transactionReception, invalid intent parameters

For further test details, see TS.27 v13.0 [47]

15.5.3.3 Checking EVT_Transaction data sent through “BroadcastReceiver”

For further test details, see TS.27 v13.0 [47]

15.5.3.3.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

15.5.3.4 Test EVT Transaction event shall be handled only by the appropriate applications

Test Purpose

Test EVT Transaction event shall have a specific format

Referenced requirement

- TS26_NFC_REQ_129
- TS26_NFC_REQ_152.1

Initial Conditions

- GSMA Mobile App UIA #2 and #5 is just a variant of GSMA Mobile App UIA #1 as described in section in 8.3.4

15.5.3.4.1 Test Sequence No 1: Unicast

Initial Conditions

- DUT is running in “unicast” mode (default mode after device power cycle)
- Applets with [AID01] and [AID02] as AID are installed on the UICC
- Access Control is
  - Allowing Communication between applet [AID01] and [app01], [app03]

- Blocking any communication with applet [AID02]

- Following applications are installed in the following order: [app02], [app03], [app01]
- Application [app01]
  - Signed with certificate [cert01]
  - Define an "Activity" which listens an "intent-filter" as follows

```
<intent-filter>
       <action
       android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
       <category android:name="android.intent.category.DEFAULT"/>
       <data
              android:host="secure"
              android:pathPattern="/SIM.*/[AID01]"
              android:scheme="nfc"/>
</intent-filter>
```

This "Activity" displays a dialog box with =”/SIM.*/[AID01]” as message

- Applications [app02] and [app03]
  - Application [app02] is signed with certificate [cert02]
  - Application [app03] is signed with certificate [cert03]
  - Define an "Activity" which listens an "intent-filter" as follows

```
<intent-filter>
       <action
       android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
       <category android:name="android.intent.category.DEFAULT"/>
       <data
              android:host="secure"
              android:pathPattern="/SIM.*/[AID02]"
              android:scheme="nfc"/>
</intent-filter>
```

- This "Activity" display a dialog box with text =”/SIM.*/[AID02] from app02” as message for application [app02]
- This "Activity" display a dialog box with text =”/SIM.*/[AID02] from app03” as message for application [app03]

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID01.	**APDU Application** receives Status Word 90 00

Step	Direction	Sequence	Expected Result
5	PCD	Power off RF field	
6	DUT → UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	Only a dialog box containing "/SIM.*[AID01]" is displayed
8	User → DUT	Close Dialog box	
9	User → DUT	Repeat steps 2-6 with AID02 instead of AID01	
10	UICC → DUT	UICC sends EVT_TRANSACTION with AID02	No dialog box is displayed

15.5.3.4.2 Test Sequence No 2: Unicast

Initial Conditions

- DUT is running in "unicast" mode (default after device power cycle)
- Applets with [AID01] and [AID02] as AID are installed on the UICC
- Access Control is
  - Allowing communication between applet [AID01] and [app01], [app03]
  - Blocking any communication between applet [AID01] and [app02]
  - Allowing communication between applet [AID02] and any applications
- Following applications are installed in the following order: [app02], [app03], [app01]
- Application [app01]
  - Signed with certificate [cert01]
  - Define an "Activity" which listens an "intent-filter" as follows

```
<intent-filter>
      <action
      android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
      <category android:name="android.intent.category.DEFAULT"/>
      <data
            android:host="secure"
            android:pathPattern="/SIM.*/.*"
            android:scheme="nfc"/>
</intent-filter>
```

This "Activity" displays a dialog box with text ="/SIM.*/.*" as message

- Applications [app02] and [app03]
  - Application [app02] is signed with certificate [cert02]
  - Application [app03] is signed with certificate [cert03]
  - Define an "Activity" which listens an "intent-filter" as follows

```
<intent-filter>
   <action android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
   <category android:name="android.intent.category.DEFAULT"/>
```

```
        <data
            android:host="secure"
           android:pathPattern="/SIM.*/[AID01]"
           android:scheme="nfc"/>
    </intent-filter>
```

- This “Activity” displays a dialog box with text =”/SIM.*/[AID01] from app02” as message for application [app02]
- This “Activity” displays a dialog box with text =”/SIM.*/[AID01] from app03” as message for application [app03]

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD → DUT	Perform RF protocol initialisation	
4	PCD → DUT	Using the **APDU application,** send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT → UICC	Send EVT_Field_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	Only a dialog box containing “/SIM.*/[AID01] from app03” is displayed
8	User → DUT	Close Dialog box	
9	User → DUT	Repeat steps 2-6 with AID02 instead of AID01	
10	UICC → DUT	UICC sends EVT_TRANSACTION with AID02	Only a dialog box containing “/SIM.*/.*” is displayed

15.5.3.5 Application Permission for using Open Mobile API

Test Purpose
To ensure that android application has required permissions to use OMAPIs and the Transaction events.

Referenced requirement
- TS26_NFC_REQ_130

Initial Conditions
- APDU_TestApplication.cap is installed on the UICC and is Selectable at AID 01.
- The DUT is powered on

15.5.3.5.1 Test Sequence No 1

Initial Conditions

- **MobileApplication** using Open Mobile APIs is installed on the DUT.
- The following permission is NOT declared in the **MobileApplication** manifest:

```
<permission
android:name="org.simalliance.openmobileapi.SMARTCARD"/>
```

Step	Direction	Sequence	Expected Result
1	App → DUT	Attempt to create a SEService object	“SecurityException” is thrown by the system

15.5.3.6 EVT Transaction is received only by the appropriate application based on priority scheme

Test Purpose

Check that in Unicast mode the EVT Transaction is received only by the appropriate application based on priority scheme.

Referenced requirement

- TS26_NFC_REQ_150
- TS26_NFC_REQ_150.1

15.5.3.6.1 Test Sequence No 1: Check installation order

Initial Conditions

- DUT is running in “unicast” mode (default mode after device power cycle)
- Applet with [AID01] is installed on the UICC
- Access Control is
- Allowing Communication between applet [AID01] and [app01], [app02]
- Following applications are installed in the following order: [app01], [app02]
- Application [app01]
- Signed with certificate [cert01]
- The “android:priority” is set to “-100”
- Define an “Activity” which listens to an “intent-filter” as follows

```
<intent-filter>
       <action
       android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
       <category android:name="android.intent.category.DEFAULT"/>
       <data
              android:host="secure"
              android:pathPattern="/SIM.*/[AID01]"
              android:scheme="nfc"/>
</intent-filter>
```

- Applications [app02]
  - Application [app02] is signed with certificate [cert02]

- The “android:priority” is set to “-100”
- Define an “Activity” which listens an “intent-filter” as follows

```
<intent-filter>
       <action
       android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
       <category android:name="android.intent.category.DEFAULT"/>
       <data
              android:host="secure"
              android:pathPattern="/SIM.*/[AID01]"
              android:scheme="nfc"/>
</intent-filter>
```

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD→ DUT	Perform RF protocol initialisation	
4	PCD→ DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT→ UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	Transaction Event Activity [app01] is started.

15.5.3.6.2 Test Sequence No 2: Check that “android : priority” has priority over the installation order

Initial Conditions

- DUT is running in “unicast” mode (default mode after device power cycle)
- Applet with [AID01] is installed on the UICC
- Access Control is
  - Allowing Communication between applet [AID01] and [app01], [app02]
- Following applications are installed in the following order: [app02], [app01]
- Application [app01]
  - Signed with certificate [cert01]
  - The “android:priority” is set to “-10”
  - Define an “Activity” which listens to an “intent-filter” as follows

```
<intent-filter>
       <action
       android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
       <category android:name="android.intent.category.DEFAULT"/>
       <data
              android:host="secure"
              android:pathPattern="/SIM.*/[AID01]"
```

```
                android:scheme="nfc"/>
</intent-filter>
```

- Applications [app02]
- Application [app02] is signed with certificate [cert02]
- The “android:priority” is set to “-100”
- Define an “Activity” which listens an “intent-filter” as follows

```
<intent-filter>
        <action
        android:name="com.gsma.services.nfc.action.TRANSACTION_EVENT"/>
        <category android:name="android.intent.category.DEFAULT"/>
        <data
                android:host="secure"
                android:pathPattern="/SIM.*/[AID01]"
                android:scheme="nfc"/>
</intent-filter>
```

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on.	
2	PCD	Power on RF field	
3	PCD→ DUT	Perform RF protocol initialisation	
4	PCD→ DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
5	PCD	Power off RF field	
6	DUT→ UICC	Send EVT_FIELD_OFF	
7	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	Transaction Event Activity [app01] is started.

15.6 VOID

15.7 Multiple Card Emulation Environment

15.7.1 General overview

This section provides test cases for checking features linked to Multiple Card Emulation Environment.

15.7.2 Conformance requirements

The Requirements tested are referenced in each test case.

15.7.3 Test Cases

15.7.3.1 Dynamic AIDs Registration APIs – “com.gsma.services.nfc.NfcController” class

For further test details, see TS.27 v13.0 [47]

15.7.3.1.1 Test Sequence No 1: defineOffHostService, valid call

For further test details, see TS.27 v13.0 [47]

15.7.3.1.2 Test Sequence No 2: defineOffHostService, null description

For further test details, see TS.27 v13.0 [47]

15.7.3.1.3 Test Sequence No 3: defineOffHostService, invalid SEName

For further test details, see TS.27 v13.0 [47]

15.7.3.1.4 Test Sequence No 4: defineOffHostService, null description and invalid SEName

For further test details, see TS.27 v13.0 [47]

15.7.3.1.5 Test Sequence No 5: Off-host service with Tap&Pay

For further test details, see TS.27 v13.0 [47]

15.7.3.1.6 Test Sequence No 6: deleteOffHostService, null service

For further test details, see TS.27 v13.0 [47]

15.7.3.1.7 Test Sequence No 7: getOffHostServices, multiple applications, GSMA APIs and manifest

For further test details, see TS.27 v13.0 [47]

15.7.3.1.8 Test Sequence No 8: getOffHostServices, no off-host services defined

For further test details, see TS.27 v13.0 [47]

15.7.3.1.9 Test Sequence No 9: getDefaultOffHostService, single application

For further test details, see TS.27 v13.0 [47]

15.7.3.1.10 Test Sequence No 10: getDefaultOffHostService, multiple applications

For further test details, see TS.27 v13.0 [47]

15.7.3.1.11 Test Sequence No 11: Tap&Pay after reboot

For further test details, see TS.27 v13.0 [47]

15.7.3.2 Dynamic AIDs Registration APIs – "com.gsma.services.nfc.OffHostService" class

For further test details, see TS.27 v13.0 [47]

15.7.3.2.1 Test Sequence No 1: getLocation

For further test details, see TS.27 v13.0 [47]

15.7.3.2.2 Test Sequence No 2: defineAidGroup, CATEGORY_PAYMENT

For further test details, see TS.27 v13.0 [47]

15.7.3.2.3 Test Sequence No 3: defineAidGroup, CATEGORY_OTHER

For further test details, see TS.27 v13.0 [47]

15.7.3.2.4 Test Sequence No 4: defineAidGroup, null description

For further test details, see TS.27 v13.0 [47]

15.7.3.2.5 Test Sequence No 5: defineAidGroup, invalid category

For further test details, see TS.27 v13.0 [47]

15.7.3.2.6 Test Sequence No 6: getAidGroups, deleteAidGroup

For further test details, see TS.27 v13.0 [47]

15.7.3.2.7 Test Sequence No 7: setBanner

For further test details, see TS.27 v13.0 [47]

15.7.3.2.8 Test Sequence No 8: Persistence of "OffHostService"

For further test details, see TS.27 v13.0 [47]

15.7.3.3 Dynamic AIDs Registration APIs – "com.gsma.services.nfc.AidGroup" class

For further test details, see TS.27 v13.0 [47]

15.7.3.3.1 Test Sequence No 1: AIDGroup, general usage

For further test details, see TS.27 v13.0 [47]

15.7.3.3.2 Test Sequence No 2: Persistence of "AidGroup"

For further test details, see TS.27 v13.0 [47]

15.7.3.4 "Long Press" on "Tap&Pay" menu entries

For further test details, see TS.27 v13.0 [47]

15.7.3.4.1 Test Sequence No 1: Off-host service defined in manifest

For further test details, see TS.27 v13.0 [47]

15.7.3.4.2 Test Sequence No 2: Off-host service defined using GSMA APIs

For further test details, see TS.27 v13.0 [47]

15.7.3.5 Routing in Multiple CEE model

For further test details, see TS.27 v13.0 [47]

15.7.3.5.1 VOID

15.7.3.5.2 Test Sequence No 2: Off-host payment service via GSMA APIs

For further test details, see TS.27 v13.0 [47]

15.7.3.5.3 Test Sequence No 3: Default route HCE, host payment service (selected in Tap&Pay), off-host payment service via GSMA APIs

For further test details, see TS.27 v13.0 [47]

15.7.3.5.4 Test Sequence No 4: Default route HCE, host payment service, off-host payment service via GSMA APIs (selected in Tap&Pay), “other” service via GSMA APIs

For further test details, see TS.27 v13.0 [47]

15.7.3.5.5 Test Sequence No 5: device off, “other” routing

For further test details, see TS.27 v13.0 [47]

15.7.3.5.6 Test Sequence No 6: HCE entry selected in Tap&Pay, device off, payment routing

For further test details, see TS.27 v13.0 [47]

15.7.3.5.7 Test Sequence No 7: off-host entry selected in Tap&Pay, device off, payment routing

For further test details, see TS.27 v13.0 [47]

15.7.3.5.8 Test Sequence No 8: screen off, “other” routing.

For further test details, see TS.27 v13.0 [47]

15.7.3.5.9 Test Sequence No 9: HCE entry selected in Tap&Pay, screen off, payment routing

For further test details, see TS.27 v13.0 [47]

15.7.3.5.10 Test Sequence No 10: off-host entry selected in Tap&Pay, screen off, payment routing

For further test details, see TS.27 v13.0 [47]

15.7.3.6 AID Conflict Resolution Mechanism

Test Purpose

Ensure DUT provide AID Conflict Resolution mechanism.

Referenced requirement

- TS26_NFC_REQ_068
- TS26_NFC_REQ_068.01
- TS26_NFC_REQ_068.02

Initial Conditions
None

15.7.3.6.1 Test Sequence No 1

For further test details, see TS.27 v13.0 [47]

15.7.3.6.2 Test Sequence No 2 (without using GSMA API)

Initial Conditions

No default service for category “Other” is present in the DUT

- Applet with [AID01] as AID is installed on the UICC.

  When the Applet is selected it shall send a response APDU 9000 + ‘4f 46 46 48 4f 53 54

- NFC is enabled on the DUT

Application [app01]

- define an “Off-Host” service in its Manifest, with description “Offhost”.
- With group "other" as category and containing [AID01] as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - [app01] service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

Application [app02]

- defined as “Host” service in its Manifest, with description “Host”.
- With group "other" as category and containing [AID01] as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID01"/>
</aid-group>
```

- [app02] service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
</intent-filter>
```

- When the application is selected it shall send a response APDU 9000 + ‘48 43 45’’

- Install the [app01] and [app02] on to the DUT for registering their respective NFC services.

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	“SELECT APDU” command is sent with AID01 as parameter	DUT should present a message asking the user which service is to be invoked. See Note
4	User → DUT	Select “Offhost”	
5	User → PCD	Power Off the field	
6	User → PCD	Power ON the field	
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 + extra data ‘4f 46 46 48 4f 53 54’ is returned by the off-host application
Note: TS26_NFC_REQ_068.02 is implemented in Android by asking the user to select the preferred service.			

15.7.3.7 Routing table update after NFC Application is uninstalled or disabled in Multiple CEE model

Test Purpose

Ensure DUT removes all of the application entries related to a disabled or uninstalled application.

Referenced requirement

- TS26_NFC_REQ_063
- TS26_NFC_REQ_063.1
- TS26_NFC_REQ_064

15.7.3.7.1 Test Sequence No 1: Application uninstalled

For further test details, see TS.27 v13.0 [47]

15.7.3.7.2 Test Sequence No 2: Application disabled and re-enabled

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- In the NFC Controller the default AID route is set to HCE (see section 2.6.1)
- Application [app01] defined an "OffHost" other service [serv01] in its Manifest.
  - With group "other" as category and containing AID01 as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name= [AID 01]/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- Applet with [AID01] as AID is installed on the UICC. [AID01] is of size 16 bytes.
- When the applet is selected, it shall send the response APDU 9000 + '4f 46 46 48 4f 53 54'
- NFC is enabled on the DUT

Step	Direction	Sequence	Expected Result
1	User → DUT	Install the Application [app01] on to the DUT	No exception is expected
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power ON the field	

Step	Direction	Sequence	Expected Result
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: 90 00 + extra data ‘4f 46 46 48 4f 53 54’ is returned by the host application
5	User → DUT	Disable the Application [app01] using If the DUT supports O_BEFORE_ANDROID_NOUGAT: “adb shell pm hide <package_name>” Else: “adb shell pm disable-user <package_name>”	Check that the adb command reports success
6	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
7	User → PCD	Power ON the field	
8	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: not equal 90 00
9	User → DUT	Enable the Application [app01] using If the DUT supports O_BEFORE_ANDROID_NOUGAT: “adb shell pm unhide <package_name>” Else: “adb shell pm enable <package_name>”	Check that the adb command reports success.
10	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
11	User → PCD	Power ON the field	
12	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: 90 00 + extra data ‘4f 46 46 48 4f 53 54’ is returned by the host application

15.7.3.7.3 Test Sequence No 3: Application uninstalled (without using GSMA API)

Initial Conditions

- Determine N the maximum capacity of the routing table using the procedure 2.6.2
- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- In the NFC Controller the default AID route is set to HCE (see section 2.6.1)
- Application [app01] defined an “OffHost” other service [serv01] in its Manifest.
  - With group “other” as category and containing AID01 as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name= [AID 01]/>
</aid-group>
```

- service [serv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- Applet with [AID01] as AID is installed on the UICC. [AID01] is of size 16 bytes.
- When the applet is selected, it shall send the response APDU 9000 + ‘4f 46 46 48 4f 53 54’
- NFC is enabled on the DUT
- Application [Fillrouteapp01] implements the registerAidsForService method
- Application [Fillrouteapp01] defined an “OffHost” other service [fillrouteserv01] in its Manifest.
  - With group “other” as category and containing TestAID01 as defined below

```
<aid-group android:description="@string/aidfillroute"
android:category="other">
<aid-filter android:name= [TestAID01]/>
</aid-group>
```

- service [fillrouteserv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- Every [TestAID xx] are of size 16 bytes and for the same target. [TestAID xx] SHALL be any random AID of 16 bytes and is not equal to [AID 01].

Steps 2 to 3 are used to fill the routing table (N-1) so that only AID01 of the [app01] can be installed.

Step	Direction	Sequence	Expected Result
1	App → DUT	Call the “registerAidsForService” method of Fillrouteapp01 with N-1 different AIDs [TestAID xx] with “other” category to register them for [fillrouteserv01] service	registerAidsForService method returns a boolean for success

Step	Direction	Sequence	Expected Result
2	User → DUT	Install the Application [app01] on to the DUT	No exception is expected
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
4	User → PCD	Power ON the field	
5	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: 90 00 + extra data ‘4f 46 46 48 4f 53 54’ is returned by the host application
6	User → DUT	From the “Setting -> apps” menu, remove the Application [app01]	
7	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
8	User → PCD	Power ON the field	
9	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: not equal 90 00

15.7.3.8 Routing update when Application is updated / upgraded in Multiple CEE model.

Test Purpose

To ensure that when an NFC application is updated, the device SHALL update the routing table according to the new registration information

Referenced requirement

- TS26_NFC_REQ_064

Test execution:

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

15.7.3.8.1 Test Sequence No 1: Off-host service

For further test details, see TS.27 v13.0 [47]

15.7.3.8.2 Test Sequence No 2: Host service

Initial Conditions

- The default AID route is set to HCE. (See section 2.6.1)
- Application [app02]
  Defined a “Host” service [serv02] in its Manifest.

  With group " other " as category and containing AID01 as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name= [AID 01]/>
</aid-group>
```

- your service [serv02] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
</intent-filter>
```

- When the [app02] is selected it shall send the response APDU 9000 + ‘48 43 45’’

Step	Direction	Sequence	Expected Result
1	User → DUT	Install Application [app02]	
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power ON the field	
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with [AID 01] as parameter	SW: 90 00 + extra data ‘48 43 45’ is returned by the host application
5		Now update the [app02] with [AID 02] via manifest and install it onto the DUT <aid-group android:description="@string/aiddescription" android:category="other"> <aid-filter android:name=[AID 02]/> </aid-group>	

Step	Direction	Sequence	Expected Result
6	User → PCD	Remove the DUT from the RF field and Place it again in the RF field	
7	PCD → DUT DUT → UICC	Send "SELECT APDU" command with [AID 01] as parameter	SW other than 90 00 will be returned

15.7.3.8.3 Test Sequence No 3: Off-host service (without using GSMA API)

Initial Conditions

- The default AID route is set to HCE. (See section 2.6.1)
- Applet with [AID 01] as AID is installed on the UICC. [AID 01] is of size 16 bytes.

When the Applet is selected it shall send the response APDU 9000 + '4f 46 46 48 4f 53 54'

- Application [app01]

  Defined an "OffHost" service [serv01] in its Manifest.

  With group "other" as category and containing AID01 as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name= [AID 01]/>
</aid-group>
```

- [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/
>
</intent-filter>
```

Step	Direction	Sequence	Expected Result
1	User → DUT	Install Application [app01]	
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power ON the field	

Step	Direction	Sequence	Expected Result
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID1 as parameter	SW: 90 00 + extra data ‘4f 46 46 48 4f 53 54’ is returned by the off-host application
5		Now update the [app01] with [AID 02] via manifest and install it onto the DUT <aid-group android:description="@string/aiddescription" android:category="other"> <aid-filter android:name=[AID 02]/> </aid-group>	
6	User → PCD	Remove the DUT from the RF field and Place it again in the RF field	
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with [AID 01] as parameter	SW other than 90 00 will be returned

15.7.3.9 NFC Controller routing table

Test Purpose

Ensure DUT handles correctly situations when NFC Controller routing is full.

Referenced requirement

- TS26_NFC_REQ_134
- TS26_NFC_REQ_134.2
- TS26_NFC_REQ_134.3
- TS26_NFC_REQ_135
- TS26_NFC_REQ_136

15.7.3.9.1 Test Sequence No 1: REQ_134 menu not available by default

Initial Conditions

- A Factory Reset has been performed
- Application [app01]

  An “Off-Host” service has been defined in the Manifest with

  - “myOffHostService-App01” as description
  - A group with "other" as category and containing one AID named [AID01]

Step	Direction	Sequence	Expected Result
1	User → DUT	Install application [app01]	

Step	Direction	Sequence	Expected Result
2	User → DUT	Open the “Setting” menu	An additional menu allowing the end-user to enable/disable group of AIDs belonging to the category “other” is not available or visible to the end user

15.7.3.9.2 VOID

15.7.3.9.3 Test Sequence No 3: Default route HCE, off-host service added via Tap&Pay menu, check REQ_134 menu

For further test details, see TS.27 v13.0 [47]

15.7.3.9.4 Test Sequence No 4: Default route HCE, off-host service added via manifest, service enabled, contactless session

For further test details, see TS.27 v13.0 [47]

15.7.3.9.5 Test Sequence No 5: Default route HCE, off-host service added via Tap&Pay menu, check REQ_134 menu (without using GSMA API)

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE (see section 2.6.1)
- Application [app01]
  An “Off-Host” service has been defined in the Manifest with
  - “myOffHostService-App01” as description
  - A group with "payment" as category and containing one AID named [AID01]. Group description is “myOtherGroup-App01”
- Application [app02]
  An “Off-Host” service has been defined in the Manifest with
  - “myOffHostService-App02” as description
  - A banner where it is displayed “myOffHostService-App02”
  - A group with "payment" as category containing [AID01] and [AID02] Group description is “myOtherGroup-App02”
- Application [app01] is installed before application [app02]
- In the “Tap&Pay” menu, “myOffHostService-App01” is selected
- After installing application [app01] and application [app02], the NFC Controller routing table is not full

- Application [Fillrouteapp02] defined an “OffHost” other service [fillrouteserv02] in its Manifest.
  - With group “other” as category and containing TestAIDFill01 as defined below

```
<aid-group android:description="@string/aidfillroute"
android:category="other">
<aid-filter android:name= [TestAIDFill01]/>
```

```
</aid-group>
```

- service [fillrouteserv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- It dynamically fills the routing table with different TestAIDFillxx as defined below with “other” category for service [fillrouteserv02] until the registerAidsForService method returns false. To fill the routing table the registerAidsForService method is called repeatedly first with one TestAIDFillxx than with two TestAIDFillxx-s than with three TestAIDFillxx-sand so on -always increasing the number of AID-s to be registered by one- until the method returns false.

AID byte	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
value	0x01 for TestAIDFill01 0x02 for TestAIDFill02 … 0x64 for TestAIDFill100 ….. 0xFF for TestAIDFill255	0x 02	0x 03	0x 04	0x 05	0x 06	0x 07	0x 08	0x 09	0x 10	0x 11	0x 12	0x 13	0x 14	0x 15	0x 03

- After application [app01] and application [app02] are launched, application [Fillrouteapp02] is launched to fill the NFC Controller routing table.

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu open the “Tap&Pay” entry	• At least, 2 entries with “myOffHostService-App01” and “myOffHostService-App02” as banner are displayed. • “myOffHostService-App01” banner is selected
2	User → DUT	Select Entry with “myOffHostService-App02” banner	• User is informed that the NFC Service proposed by the application cannot be used. • Displayed message shall propose to the end-user a way to disable NFC services previously installed

Step	Direction	Sequence	Expected Result
3	User → DUT	Open the “Setting” menu	An additional menu allowing the end-user to enable/disable group of AIDs belonging to the category “other” is visible
4	User → DUT	Open the additional “Setting” menu	Groups “myOffHostService-App01” “myOffHostService-App02” are not displayed as entry

15.7.3.9.6 Test Sequence No 6: Default route HCE, off-host service added via manifest, service enabled, contactless session (without using GSMA API)

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE (see section 2.6.1)
- Application [app01]
  An “Off-Host” service has been defined in the Manifest with
  - “myOffHostService-App01” as description
  - A group with "other" as category and containing one AID named [AID01]. Group description is “myOtherGroup-App01”
- Application [app01] is not yet installed on the DUT
- Application [Fillrouteapp02] is launched to fill the NFC Controller routing table.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install application [app01]	• User is informed that the NFC Service proposed by the application cannot be used. • Displayed message shall propose to the end-user a way to disable NFC services previously installed
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power on the field	
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is not returned
5	User → PCD	Power off the field	

Step	Direction	Sequence	Expected Result
6	User → DUT	Remove the DUT from the area where the field is powered on	
7	User → DUT	Open the “Setting” menu	An additional menu allowing the end-user to enable/disable group of AIDs belonging to the category “other” is visible
8	User → DUT	Open the additional “Setting” menu	• At least, the following groups are displayed: - “myOtherGroup-App01” - the group created by application [Fillrouteapp02] • “myOtherGroup-App01” group is seen as disabled and cannot be enabled
9	User → DUT	Disable the group created by application [Fillrouteapp02]	The group created by application [Fillrouteapp02] is disabled
10	User → DUT	Enable “myOtherGroup-App01”	“myOtherGroup-App01” group is seen as enabled
11	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
12	User → PCD	Power on the field	
13	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned
14	PCD → DUT DUT → UICC	Send “SELECT APDU” command with TestAIDFill01	SW: 90 00 is not returned

15.7.3.10 Tap&Pay menu – routing of APDUs for payment services

Test Purpose

Test the DUT for correct configuration of routing table in response to changes made in the Tap and Pay menu settings:

- Default Processor
- Default Payment Application

And test the DUT for persistence of Tap&Pay menu setting after reboot.

Referenced requirement

- TS26_NFC_REQ_147
- TS26_NFC_REQ_148

- TS26_NFC_REQ_148.1

Initial Conditions

- **ReferenceApplication.cap** is installed with AID_REF on the UICC
- **APDU Application** to send APDUs according to the reference transaction
- NFC enabled on the DUT

15.7.3.10.1 Test Sequence No 1: Tap&Pay routing to UICC

Initial Conditions

- App01: an android application which registers an off_host_apdu_service for AID_REF and specifies the category as "payment".
- App02: an android application which registers host_apdu_service (HCE) for AID_REF and specifies the category as "payment". This application will respond to the APDU application similar to the ReferenceApplication.cap
- All NFC applications on the DUT are uninstalled except applications that are preinstalled

NOTE: It is not possible to configure scenarios 1 and 2 in TS26_NFC_REQ_147. This is because at least 1 payment service must be selected as default in the Tap&Pay settings menu. As a result, these test scenarios have been omitted from the table below.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	Install App02	The application is installed successfully
3	User → DUT	In the NFC Controller set the default AID route to UICC (see section 2.6.1.2).	
4	User → DUT	In NFC Tap&Pay settings set App01 as the default payment service	The active payment service has been set to App01, (Off Host)
5	PCD → DUT DUT → UICC	With the screen off perform the reference transaction using the **APDU application**	App01 (Off Host) responds
6	PCD → DUT DUT → UICC	With the screen on and the device locked perform the reference transaction using the **APDU application**	App01 (Off Host) responds

Step	Direction	Sequence	Expected Result
7	PCD → DUT DUT → UICC	With the screen on and the device unlocked perform the reference transaction using the **APDU application.**	App01 (Off Host) responds
8	PCD → DUT DUT → UICC	With the device in flight mode perform the reference transaction using the APDU application (with NFC switched ON)	App01 (Off Host) responds
9	PCD → DUT DUT → UICC	With the device powered off perform the reference transaction using the **APDU application**	App01 (Off Host) responds
10	User → DUT	Power on the DUT and deactivate flight mode	
11	User → DUT	In NFC Tap&Pay settings set App02 as the default payment service	The active payment service has been set to App02, (Host)
12	PCD → DUT DUT → UICC	With the screen off perform the reference transaction using the **APDU application**	App02 (Host) selection fails with error code '6A82' OR App02 (Host) responds
13	PCD → DUT DUT → UICC	With the screen on and the device locked perform the reference transaction using the **APDU application**	APDUs are routed to the host
14	PCD → DUT DUT → UICC	With the screen on and the device unlocked perform the reference transaction using the **APDU application.**	APDUs are routed to the host
15	PCD → DUT DUT → UICC	With the screen on and the device in flight mode perform the reference transaction using the **APDU application**. (with NFC switched ON)	APDUs are routed to the host
16	PCD → DUT DUT → UICC	With the device powered off perform the reference transaction using the **APDU application**	App02 (Host) selection fails with error code '6A82'

15.7.3.10.2 Test Sequence No 2: Tap&Pay routing to HCE

Initial Conditions

- App01: an android application which registers an off_host_apdu_service for AID_REF and specifies the category as "payment".

- App02: an android application which registers host_apdu_service (HCE) for AID_REF and specifies the category as “payment”. This application will respond to the APDU application similar to the ReferenceApplication.cap
- All NFC applications on the DUT are uninstalled except applications that are preinstalled

NOTE: It is not possible to configure scenarios 1 and 2 in TS26_NFC_REQ_147. This is because at least 1 payment service must be selected as default in the Tap&Pay settings menu. As a result, these test scenarios have been omitted from the table below.

Step	Direction	Sequence	Expected Result
1	User → DUT	Install App01	The application is installed successfully
2	User → DUT	Install App02	The application is installed successfully
3	User → DUT	In the NFC Controller set the default AID route to HCE (see section 2.6.1.1).	
4	User → DUT	In NFC Tap&Pay settings set App01 as the default payment service	The active payment service has been set to App01, (Off Host)
5	PCD → DUT DUT → UICC	With the screen off perform the reference transaction using the **APDU application**	App01 (Off Host) responds
6	PCD → DUT DUT → UICC	With the screen on and the device locked perform the reference transaction using the **APDU application**	App01 (Off Host) responds
7	PCD → DUT DUT → UICC	With the screen on and the device unlocked perform the reference transaction using the **APDU application**.	App01 (Off Host) responds
8	PCD → DUT DUT → UICC	With the device in flight mode perform the reference transaction using the **APDU application** (with NFC switched ON)	App01 (Off Host) responds
9	PCD → DUT DUT → UICC	With the device powered off perform the reference transaction using the **APDU application**	App01 (Off Host) responds
10	User → DUT	Power on the DUT and deactivate the flight mode	

Step	Direction	Sequence	Expected Result
11	User → DUT	In NFC Tap&Pay settings set App02 as the default payment service	The active payment service has been set to App02, (Host)
12	PCD → DUT DUT → UICC	With the screen off perform the reference transaction using the **APDU application**	App02 (Host) selection fails with error code ‘6A82’ OR App02 (Host) responds
13	PCD → DUT DUT → UICC	With the screen on and the device locked perform the reference transaction using the **APDU application**	APDUs are routed to the host
14	PCD → DUT DUT → UICC	With the screen on and the device unlocked perform the reference transaction using the **APDU application**.	APDUs are routed to the host
15	PCD → DUT DUT → UICC	With the screen on and the device in flight mode perform the reference transaction using the **APDU application**. (with NFC switched ON)	APDUs are routed to the host
16	PCD → DUT DUT → UICC	With the device powered off perform the reference transaction using the **APDU application.**	App02 (Host) selection fails with error code ‘6A82’

15.7.3.10.3 Test Sequence No 3: Tap&Pay after reboot (without using GSMA API)

Initial Conditions

- Application [app01] defined an “Off-Host” other service

  [myOffHostService01-App01] in its Manifest.

  - With group "other" as category and containing [AID01]

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID01"/>
</aid-group>
```

  - [app01] service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```


- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService01-App01>
</offhost-apdu-service>
```

- Application [app01] is installed for registering its NFC services
- Applet with [AID01] as AID is installed on the UICC
- NFC is enabled on the DUT

Step	Direction	Sequence	Expected Result
1	User→ DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 1 entry with “myOffHostService01-App01” as banner is displayed
2	User→ DUT	Select entry with “myOffHostService01-App01” banner	
3	User→ DUT	Power off the Device and then power on the Device	
4	User→ DUT	From the “Setting” menu, open the “Tap&Pay” entry	“myOffHostService01-App01” banner is selected
5	User→ DUT	While the field is off, place the DUT in the area where the field will be powered on	
6	User→ PCD	Power on the field	
7	PCD→ DUT DUT→ UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned

15.7.3.11 Dynamic & Automatic switch of AID default Route

Test Purpose

The aims of these tests are to ensure the coexistence between HCE and UICC-based NFC services in the case were many AIDs are used & registered.

Referenced requirements:

- TS26_NFC_REQ_134
- TS26_NFC_REQ_134.1
- TS26_NFC_REQ_135
- TS26_NFC_REQ_143

15.7.3.11.1 Test Sequence No 1: One card emulation environment overflow – Automatic Management

This test ensure that the automatic route switching (without user interaction) feature works in a one ecosystem overflow scenario

Initial Conditions:

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to UICC. (See section 2.6.1.2)
- The UICC contains 3 cardlets with known AIDs [AID01, AID02, AID03].
- AID01, AID02, AID03 are available on the UICC
- AID01 is not registered by any application
- DUT is powered ON and DUT is unlocked and the screen is ON

- Dynamic_Other_HCE: An application able to register a configurable list of non-payment AID with a length of 16 bytes on HCE using the dynamic registration API of Android. [registerAidsForService()]
- The AIDs list used by the application SHALL be different then AIDs used by the 3 others applications

- Static_Other_255AIDs_OffHost: An application able to register a list of 255 non-payment AID with a length of 16 bytes on the OffHost (UICC) using the Manifest of the application
- The AIDs list used by the application SHALL be different than AIDs used by the 3 others applications

- Static_Other_2AIDs_HCE: An application able to register 2 AIDs with a length of 16 bytes on the Host (HCE) referred below as AID04 an AID05 from the Manifest
- Those 2 AIDs are not present in the list of AID used by any other application

- Static_Other_2AIDs_OffHost: An application able to register 2 AIDs with a length of 16 bytes on the OffHOST (UICC) from the Manifest of the application.
- The 2 AIDs chosen SHALL exist on the UICC, referred below as AID02 and AID03
- Those 2 AIDs are not present in the list of AID used by any other application

Step	Direction	Sequence	Expected Result
1	User → DUT	Unregister 254 TestAIDUICC that were registered using 2.6.1.2 procedure (keep one TestAIDUICC)	
2	User → DUT	Install Application Static_Other_2AIDs_OffHost	Installation successful Registration of the 2 OffHost AIDs is successful
3	User → DUT	Install Application Dynamic_Other_HCE	Installation successful

Step	Direction	Sequence	Expected Result
4	User → DUT	Use Dynamic_Other_HCE and register 255 HCE AIDs	No error while registering the AIDs
5	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
6	User → PCD	Power on the field	
7	PCD → DUT	Send "SELECT APDU" command with AID01 as parameter	SW: 6A 82 is returned [default route was previously switched to HCE and AID01 not reachable because not in routing table]
8	PCD → DUT	Send "SELECT APDU" command with AID02 as parameter	SW: 90 00 is returned [AID02 is present in the routing table routed to UICC]
9	PCD → DUT	Send "SELECT APDU" command with AID03 as parameter	SW: 90 00 is returned [AID03 is present in the routing table routed to UICC]
10	User → DUT	Uninstall Dynamic_Other_HCE application	Uninstall is successful
11	User → DUT	Uninstall Static_Other_2AIDs_OffHost application	Uninstall is successful
12	User → DUT	Install Static_Other_2AIDs_HCE application	Install is successful Registration of the 2 Host AIDs is successful
13	User → DUT	Install Static_Other_255AIDs_OffHost application	Install is successful No error while registering the 255 AIDs
14	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
15	User → PCD	Power on the field	
16	PCD → DUT	Send "SELECT APDU" command with AID01 as parameter	SW: 90 00 is returned [default route was previously switched to UICC]
17	PCD → DUT	Send "SELECT APDU" command with AID04 as parameter	SW: 90 00 is returned [AID04 is present in the routing table routed to HCE]
18	PCD → DUT	Send "SELECT APDU" command with AID05 as parameter	SW: 90 00 is returned [AID05 is present in the routing table routed to HCE]

Step	Direction	Sequence	Expected Result
19	User → DUT	Uninstall Static_Other_255AIDs_OffHost application	Uninstall is successful
20	User → DUT	Uninstall Static_Other_2AIDs_HCE application	Uninstall is successful

15.7.3.11.2 Test Sequence No 2: Both card emulation environment overflow - Without payment apps

For further test details, see TS.27 v13.0 [47]

15.7.3.11.3 Test Sequence No 3: Both card emulation environment overflow - Without payment apps (without using GSMA API)

The purpose of this test case is to ensure compliance with TS26_NFC_REQ_135.

Initial Conditions:

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to UICC. (See section 2.6.1.2)
- Know how many 16 bytes AIDs the Routing Table of the DUT may contain (RTS)
- See 2.6 section: “Procedure to identify the AID routing table max size”)
- The UICC contains 3 cardlets with known AIDs [AID01, AID02, AID03].
- AID01, AID02, AID03 are only available on the UICC
- AID01 is not registered by any application
- DUT is powered on and DUT is unlocked and the screen is ON
- Dynamic_Other_HCE: An application able to register a configurable list of non-payment AID with a length of 16 bytes on the HCE using the dynamic registration API of Android [registerAidsForService()]
- The AIDs list used by the application SHALL be different then AIDs used by any other applications. The AIDs of the application have a length of 16 bytes

- Fill_Other_OffHost: defined an “OffHost” other service [fillrouteserv01] in its Manifest.
    - With group “other” as category and containing TestAID01 as defined below

```
<aid-group android:description="@string/aidfillroute"
android:category="other">
<aid-filter android:name= [TestAID01]/>
</aid-group>
```

    - service [fillrouteserv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
```

```
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- Every [TestAID xx] are of size 16 bytes and for the same target. [TestAID xx] SHALL be any random AIDs of 16 bytes and is not equal to any AIDs used by any other applications.

- Application [Fill_Other_OffHost] implements the registerAidsForService method

- Static_Other_2AIDs_HCE: An application able to register 2 AIDs with a length of 16 bytes on HCE referred below as AID04 an AID05 from the Manifest of the application
- Those 2 AIDs must not be present in the list of AID used by any other application

- Static_Other_2AIDs_OffHost: An application able to register 2 AIDs with a length of 16 bytes on the OffHOST (UICC) using the Manifest of the Application.
- The 2 AIDs chosen are expected to exist on the UICC, referred below as AID02 an AID03 from the Manifest of the application
- Those 2 AIDs must not be present in the list of AID used by any other application

Step	Direction	Sequence	Expected Result
1	User → DUT	Unregister 254 TestAIDUICC that were registered using 2.6.1.2 procedure (keep one TestAIDUICC)	
2	User → DUT	Install “Fill_Other_OffHost” application	Installation successful
3	User → DUT	Call the “registerAidsForService” method of “Fill_Other_OffHost” application with RTS-2 different AIDs [TestAID xx] with “other” category to register them for [fillrouteserv01] service	registerAidsForService method returns a boolean for success
4	User → DUT	Install “Static_Other_2AIDs_OffHost” application	Installation successful No error while registering the 2 AIDs
5	User → DUT	Install “Dynamic_Other_HCE” application	
6	User → DUT	User “Dynamic_Other_HCE” to write (RTS-1) AIDs	No error occurs
7	User → DUT	Install "Static_2AIDs_HCE" application	Installation is successful A message is displayed to the user (per REQ_135) Menu entry is available in “Settings” (per REQ_134) The group of AID registered by the "Static_2AIDs_HCE" application is disabled

Step	Direction	Sequence	Expected Result
8	User → DUT	Exit the menu (Home button)	It’s possible to exit menu
9	User → PCD	While the field is off, place the DUT in the area where the field will be powered on	
10	User → PCD	Power on the field	
11	PCD → DUT	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned
12	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	SW: 90 00 is returned
13	PCD → DUT	Send “SELECT APDU” command with AID03 as parameter	SW: 90 00 is returned
14	PCD → DUT	Send “SELECT APDU” command with AID04 as parameter	SW: 6A 82 is returned [not present in routing table]
15	PCD → DUT	Send “SELECT APDU” command with AID05 as parameter	SW: 6A 82 is returned [not present in routing table]
16	User → PCD	Power off the field	
17	User → DUT	Uninstall “Fill_Other_OffHost” application Uninstall “Static_Other_2AIDs_OffHost” application	Uninstall successful
18	User → DUT	Open the menu as defined by TS26_REQ_134 Enable the group defined by “Static_2AIDs_HCE”	
19	User → PCD	While the field is off, place the DUT in the area where the field will be powered on	
20	User → PCD	Power on the field	
21	PCD → DUT	Verify that default route is switched to HCE: Send “SELECT APDU” command with AID01 as parameter	SW: 6A 82 is returned
22	PCD → DUT	Verify that HostAID are now reachable Send “SELECT APDU” command with AID04 as parameter	SW: 90 00 is returned

Step	Direction	Sequence	Expected Result
23	PCD → DUT	Send “SELECT APDU” command with AID05 as parameter	SW: 90 00 is returned
24	User → PCD	Power off the field	
25	User → DUT	Uninstall all the apps	Uninstall is successful

15.7.3.12 Routing in Multiple CEE model without using GSMA API

Test Purpose

To ensure routing between different CEE environments is performed correctly in a multiple CEE model.

Referenced requirement

- TS26_NFC_REQ_065.1
- TS26_NFC_REQ_095
- TS26_NFC_REQ_147

Initial Conditions

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

15.7.3.12.1 Test Sequence No 1: Off-host payment service via manifest, host “other” service

Initial Conditions

- No AID is registered in the CLF routing table.
- Applet with [AID01] as AID is installed on the UICC.
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01] defined an “Off-Host” payment service [serv01] in its Manifest.
  - With group "payment" as category and containing one AID as defined below

    ```
    <aid-group android:description="@string/aiddescription"
    android:category="payment">
    <aid-filter android:name=" AID01"/>
    </aid-group>
    ```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app02] defined a “Host” non-payment service [serv02] in its Manifest.

with group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID02"/>
</aid-group>
```

  - your service [serv02] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
</intent-filter>
```

- Application [app02] should respond to SELECT Command for AID 2 with response APDU ’9000’ with extra data ’48 43 45’

Step	Direction	Sequence	Expected Result
1	App → DUT	Install the application [app01] for registering the “Off-Host” for payment services Install the [app02] for registering the “Host” for other (non-payment) services	
2	App → DUT	Open the “Tap&Pay” menu	Service [serv01] is one entry as banner “myOffHostService01” in the “Tap&Pay” menu
3	User → DUT	Select “myOffHostService01” banner	

Step	Direction	Sequence	Expected Result
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54” As UICC applet will answer to the AID 1 Select.
7	PCD → DUT	Send “SELECT APDU” command with AID 2 as parameter	HCE application will answer to the AID 2 Select. SW: 90 00 is returned with extra data ’48 43 45’
8	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54” As UICC applet will answer to the AID 1 Select.

15.7.3.12.2 Test Sequence No 2: Off-host payment service via manifest

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01] defined an “Off-Host” payment service [myOffHostService-App01] in its Manifest.
    - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID01"/>
</aid-group>
```

    - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-App01>
</offhost-apdu-service>
```

- Application [app01] is installed for registering its NFC services
- Applets with [AID01] & [AID02] as AID are installed on the UICC
  - When they are selected from a POS, SW:90 00 is returned + extra data "4f 46 46 48 4f 53 54"

Step	Direction	Sequence	Expected Result
1	User → DUT	From the "Setting" menu open the "Tap&Pay" entry	At least, 1 entry with "myOffHostService-App01" as banner is displayed
2	User → DUT	Select entry with "myOffHostService01-App01" banner	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
4	User → PCD	Power on the field	
5	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter	SW: 90 00 is returned with extra data "4f 46 46 48 4f 53 54"
6	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID02 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.12.3 Test Sequence No 3: Default route HCE, host payment service (selected in Tap&Pay), off-host payment service via manifest

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines "HCE" service as follows
  - "myHCEService-App01" as description
  - A banner where it is displayed "myHCEService-App1"
  - A group with "payment" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data "48 43 45"
- Application [app02] defined an "Off-Host" payment service [myOffHostService-App02] in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID02"/>
</aid-group>
```

- your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
App02>
</offhost-apdu-service>
```

- Application [app02] is installed for registering its NFC services
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 2 entries with “myHCEService-App01” and “myOffHostService-App02” as banner are displayed
2	User → DUT	Select entry with “myHCEService-App01” banner	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
4	User → PCD	Power on the field	
5	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	SW: 90 00 is returned with extra data “48 43 45”

Step	Direction	Sequence	Expected Result
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.12.4 Test Sequence No 4: Default route HCE, host payment service, off-host payment service via manifest (selected in Tap&Pay), “other” service via manifest

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService-App01” as description
  - A banner where it is displayed “myHCEService-App1”
  - A group with "payment" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”
- Application [app02] defined an “Off-Host” service [myOffHostService-App02] in its Manifest.

  - With group "payment" as category and containing one AID as defined below

    ```
    <aid-group android:description="@string/aiddescription"
    android:category="payment">
    <aid-filter android:name="AID02"/>
    </aid-group>
    ```

  - and with group "other" as category and containing one AID as defined below

    ```
    <aid-group android:description="@string/aiddescription"
    android:category="other">
    <aid-filter android:name="AID03"/>
    </aid-group>
    ```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

    ```
    <intent-filter>
    <action android:name =
    "android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
    </intent-filter>
    ```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-App02>
</offhost-apdu-service>
```

- Application [app02] is installed for registering its NFC services
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID03] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 2 entries with “myHCEService-App01” and “myOffHostService-App02” as banner are displayed
2	User → DUT	Select entry with “myHCEService-App01” banner	
3	User → DUT	Power off Device	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82
8	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID03 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”

15.7.3.12.5 Test Sequence No 5: device off, “other” routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService-App01” as description
  - A banner where it is displayed “myHCEService-App1”
  - A group with "other" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”
- Application [app02] defined an “Off-Host” other service [myOffHostService-App02] in its Manifest.

  - With group "other" as category and containing one AID as defined below

    ```
    <aid-group android:description="@string/aiddescription"
    android:category="other">
    <aid-filter android:name="AID02"/>
    </aid-group>
    ```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

    ```
    <intent-filter>
    <action android:name =
    "android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
    </intent-filter>
    ```

- A banner where it is displayed “myOffHostService01”

  ```
  < offhost-apdu-service
  android:apduServiceBanner="@drawable/myOffHostService-
  App02>
  </offhost-apdu-service>
  ```

- Application [app02] is installed for registering its NFC services
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	Power off Device	
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power on the field	
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
5	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”

15.7.3.12.6 Test Sequence No 6: HCE entry selected in Tap&Pay, device off, payment routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService-App01” as description
  - A banner where it is displayed “myHCEService-App1”
  - A group with "payment" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”
- Application [app02] defined an “Off-Host” payment service [myOffHostService-App02] in its Manifest.

  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID02"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
```

```
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
App02>
</offhost-apdu-service>
```

- [app01] is installed before [app02]
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 2 entries with “myHCEService-App01” and “myOffHostService-App02” as banner are displayed
2	User → DUT	Select entry with “myHCEService-App01” banner	
3	User → DUT	Power off Device	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.12.7 Test Sequence No 7: off-host entry selected in Tap&Pay, device off, payment routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled

- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService-App01” as description
  - A banner where it is displayed “myHCEService-App1”
  - A group with "payment" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”
- Application [app02] defined an “Off-Host” payment service [myOffHostService-App02] in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID02"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-App02>
</offhost-apdu-service>
```

- Application [app02] is installed for registering its NFC services
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 2 entries with “myHCEService-App01” and “myOffHostService-App02” as banner are displayed

Step	Direction	Sequence	Expected Result
2	User → DUT	Select entry with “myOffHostService-App02” banner	
3	User → DUT	Power off Device	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”

15.7.3.12.8 Test Sequence No 8: screen off, “other” routing.

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService-App01” as description
  - A banner where it is displayed “myHCEService-App1”
  - A group with "other" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”
- Application [app02] defined an “Off-Host” other service [myOffHostService-App02] in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name="AID02"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
```

```
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
App02>
</offhost-apdu-service>
```

- Application [app02] is installed for registering its NFC services
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- No default Tap&Pay service is selected

Step	Direction	Sequence	Expected Result
1	User → DUT	Ensure that the Device screen is off	
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power on the field	
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82 OR App01 (Host) responds: SW:90 00 is returned + extra data “48 43 45”
5	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”

15.7.3.12.9 Test Sequence No 9: HCE entry selected in Tap&Pay, screen off, payment routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows

- "myHCEService-App01" as description
- A banner where it is displayed "myHCEService-App1"
- A group with "payment" as category and containing one AID named [AID01]
- When it is selected from a POS, SW:90 00 is returned + extra data "48 43 45"

- Application [app02] defined an "Off-Host" payment service [myOffHostService-App02] in its Manifest.

  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name="AID02"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed "myOffHostService01"

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
App02>
</offhost-apdu-service>
```

- [app01] is installed before [app02]
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data "4f 46 46 48 4f 53 54"
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data "4f 46 46 48 4f 53 54"

Step	Direction	Sequence	Expected Result
1	User → DUT	From the "Setting" menu, open the "Tap&Pay" entry	At least, 2 entries with "myHCEService-App01" and "myOffHostService-App02" as banner are displayed
2	User → DUT	Select entry with "myHCEService-App01" banner	

Step	Direction	Sequence	Expected Result
3	User → DUT	Ensure that the Device screen is off	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter	Contactless selection fails with SW: 6A 82 OR App01 (Host) responds: SW:90 00 is returned + extra data "48 43 45"
7	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID02 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.12.10 Test Sequence No 10: off-host entry selected in Tap&Pay, screen off, payment routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines "HCE" service as follows
  - "myHCEService-App01" as description
  - A banner where it is displayed "myHCEService-App1"
  - A group with "payment" as category and containing one AID named [AID01]
  - When it is selected from a POS, SW:90 00 is returned + extra data "48 43 45"
- Application [app02] defined an "Off-Host" payment service [myOffHostService-App02] in its Manifest.

  - With group "payment" as category and containing one AID as defined below

    ```
    <aid-group android:description="@string/aiddescription"
    android:category="payment">
    <aid-filter android:name="AID02"/>
    </aid-group>
    ```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

    ```
    <intent-filter>
    ```



```
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
App02>
</offhost-apdu-service>
```

- Application [app02] is installed for registering its NFC services
- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 2 entries with “myHCEService-App01” and “myOffHostService-App02” as banner are displayed
2	User → DUT	Select entry with “myOffHostService-App02” banner	
3	User → DUT	Ensure that the Device screen is off	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”

15.7.3.13 Routing in Multiple CEE model with eSE

Test Purpose

To ensure routing between different CEE environments is performed correctly in a multiple CEE model with eSE.

Referenced requirement

- TS26_NFC_REQ_094
- TS26_NFC_REQ_094.1
- TS26_NFC_REQ_095
- TS26_NFC_REQ_147
- TS26_NFC_REQ_173
- TS26_NFC_REQ_173.1

Initial Conditions

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

15.7.3.13.1 Test Sequence No 1: Off-host (eSE) “other” service, host payment service

Initial Conditions

- No AID is registered in the CLF routing table.
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01] defines an “Off-Host” other service [serv01] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name="AID08"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv01] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app02] defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "other" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

Step	Direction	Sequence	Expected Result
1	App → DUT	Install the [app01] for registering the “Off-Host” for other (non-payment) services Install the application [app02] for registering the “Host” for payment services	
2	App → DUT	Open the “Tap&Pay” menu	At least “myHCEService01” as banner is displayed.
3	User → DUT	Select “myHCEService01” banner	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	

Step	Direction	Sequence	Expected Result
6	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID08 as parameter	SW: 90 00 is returned with extra data “65 53 45” As eSE applet will answer to the AID08 Select.
7	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	HCE application will answer to the AID02 Select. SW: 90 00 is returned with extra data ’48 43 45’
8	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID08 as parameter	SW: 90 00 is returned with extra data “65 53 45” As eSE applet will answer to the AID08 Select.

15.7.3.13.2 Test Sequence No 2: Off-host (eSE) “other” service

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01] defines an “Off-Host” other service [serv01] for eSE in its Manifest.

  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name="AID08"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as defined below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv01] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
      <se-ext-group>
            <se-id name="eSE1"/>
      </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app01] is installed for registering its NFC services

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID08 as parameter	SW: 90 00 is returned with extra data “65 53 45”
4	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID07 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.13.3 Test Sequence No 3: Off-host (UICC) payment service, off-host (eSE) payment service (selected in Tap&Pay), host “other” service

Initial Conditions

- No AID is registered in the CLF routing table.
- Applet with [AID01] as AID is installed on the UICC.
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01] defines an “Off-Host” payment service [serv01] for UICC in its Manifest.

    - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

    - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app02] defines a “Host” non-payment service [serv02] in its Manifest.

with group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID02"/>
</aid-group>
```

    - your service [serv02] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.HOST_APDU_SERVICE"/>
</intent-filter>
```

- Application [app02] should respond to SELECT Command for AID 2 with response APDU ’9000’ with extra data ’48 43 45’

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.

- With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

- your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app01], application [app02] and application [app03] are installed to register their NFC services.

Step	Direction	Sequence	Expected Result
1	App → DUT	Open the “Tap&Pay” menu	At least, 2 entries with “myOffHostServiceeSE01” and “myOffHostService01” as banner are displayed.

Step	Direction	Sequence	Expected Result
2	User → DUT	Select "myOffHostServiceeSE01" banner	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
4	User → PCD	Power on the field	
5	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID07 as parameter	SW: 90 00 is returned with extra data "65 53 45" As eSE applet will answer to the AID07 Select.
6	PCD → DUT	Send "SELECT APDU" command with AID 2 as parameter	HCE application will answer to the AID 2 Select. SW: 90 00 is returned with extra data '48 43 45'
7	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.13.4 Test Sequence No 4: Various services

Initial Conditions

- No AID is registered in the CLF routing table.
- Two applets with [AID01] and [AID03] as AID are installed on the UICC.
  - When it is selected from a POS, SW:90 00 is returned + extra data "4f 46 46 48 4f 53 54"
- The default AID route is set to HCE. (See section 2.6.1)

- Application [app01] defines an "Off-Host" payment service [serv01] for UICC in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
```



```
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app02] defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "payment" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
```

```
        <se-ext-group>
            <se-id name="eSE"/>
        </se-ext-group>
    <AID-based>true</AID-based>
    </extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app04] defines an “Off-Host” other service [serv04] for UICC in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID03"/>
</aid-group>
```

  - your service [serv04] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService02>
</offhost-apdu-service>
```

- Application [app05] defines an “Off-Host” other service [serv05] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
```

```
<aid-filter android:name=" AID08"/>
</aid-group>
```

- your service [serv05] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv05] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app06] defines an “Off-Host” other service [serv06] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID09"/>
</aid-group>
```

  - your service [serv06] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv06] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE03>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03], application [app04] , application [app05] and application [app06] are installed to register their NFC services.

Step	Direction	Sequence	Expected Result
1	App → DUT	Open the “Tap&Pay” menu	At least, 3 entries with “myHCEService01” and “myOffHostServiceeSE01” and “myOffHostService01” as banner are displayed.
2	User → DUT	Select “myHCEService01” banner	
3	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	

Step	Direction	Sequence	Expected Result
4	User → PCD	Power on the field	
5	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	HCE application will answer to the AID02 Select. SW: 90 00 is returned with extra data ’48 43 45’
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID07 as parameter	Contactless selection fails with SW: 6A 82
8	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID03 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54” As UICC applet will answer to the AID03 Select.
9	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID08 as parameter	SW: 90 00 is returned with extra data “65 53 45” As eSE applet will answer to the AID08 Select.
10	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID09 as parameter	SW: 90 00 is returned with extra data “65 53 45” As eSE applet will answer to the AID09 Select.

15.7.3.13.5 Test Sequence No 5: screen off, payment routing, off-host (eSE) entry selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
    - “myHCEService01” as description
    - A banner where it is displayed “myHCEService01”
    - A group with "payment" as category and containing one AID named [AID02]
    - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app02] defines an “Off-Host” payment service [serv02] for UICC in its Manifest.

- With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

- your service [serv02] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
      <se-ext-group>
            <se-id name="eSE"/>
      </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03] are installed to register their NFC services.

- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

Step	Direction	Sequence	Expected Result
1	User → DUT	From the “Setting” menu, open the “Tap&Pay” entry	At least, 3 entries: “myHCEService01” and “myOffHostService01” and “myOffHostServiceeSE01” as banner are displayed
2	User → DUT	Select entry with “myOffHostService-eSE01” banner	
3	User → DUT	Ensure that the Device screen is off	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	

Step	Direction	Sequence	Expected Result
6	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82
7	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
8	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID07 as parameter	SW: 90 00 is returned with extra data “65 53 45”

15.7.3.13.6 Test Sequence No 6: Screen off, various routing

Initial Conditions

- No AID is registered in the CLF routing table.
- Two applets with [AID01] and [AID03] as AID are installed on the UICC.
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- The default AID route is set to HCE. (See section 2.6.1)

- Application [app01] defines an “Off-Host” payment service [serv01] for UICC in its Manifest.

  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
```

```
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app02] defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "payment" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app04] defines an “Off-Host” other service [serv04] for UICC in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID03"/>
</aid-group>
```

  - your service [serv04] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService02>
</offhost-apdu-service>
```

- Application [app05] defines an “Off-Host” other service [serv05] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID08"/>
</aid-group>
```

  - your service [serv05] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
```



```
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv05] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app06] defines an “Off-Host” other service [serv06] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID09"/>
</aid-group>
```

  - your service [serv06] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv06] declaration must contain com.gsma.services.nfc.extensions


```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE03>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03], application [app04] , application [app05] and application [app06] are installed to register their NFC services.

Step	Direction	Sequence	Expected Result
1	App → DUT	Open the “Tap&Pay” menu	At least, 3 entries with “myHCEService01” and “myOffHostServiceeSE01” and “myOffHostService01” as banner are displayed.
2	User → DUT	Select “myOffHostServiceeSE01” banner	
3	User → DUT	Ensure that the Device screen is off	
4	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
5	User → PCD	Power on the field	
6	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82

Step	Direction	Sequence	Expected Result
7	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter	Contactless selection fails with SW: 6A 82
8	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID07 as parameter	SW: 90 00 is returned with extra data "65 53 45"
9	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID03 as parameter	SW: 90 00 is returned with extra data "4f 46 46 48 4f 53 54" As UICC applet will answer to the AID03 Select.
10	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID08 as parameter	SW: 90 00 is returned with extra data "65 53 45" As eSE applet will answer to the AID08 Select.
11	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID09 as parameter	SW: 90 00 is returned with extra data "65 53 45" As eSE applet will answer to the AID09 Select.

15.7.3.14 Routing in Multiple CEE model with eSE in Battery Low Mode

Test Purpose

To ensure routing between different CEE environments is performed correctly in a multiple CEE model with eSE in Battery Low Mode.

Referenced requirement

- TS26_NFC_REQ_021
- TS26_NFC_REQ_094
- TS26_NFC_REQ_094.1
- TS26_NFC_REQ_095
- TS26_NFC_REQ_147
- TS26_NFC_REQ_173
- TS26_NFC_REQ_173.1

Initial Conditions

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

15.7.3.14.1 Test Sequence No 1: Battery Low, “other” routing

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "other" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app02] defines an “Off-Host” other service [serv02] for UICC in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID03"/>
</aid-group>
```

  - your service [serv02] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService02>
</offhost-apdu-service>
```

- Application [app03] defines an “Off-Host” other service [serv03] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID08"/>
```

```
</aid-group>
```

- your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv01] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03] are installed to register their NFC services.

- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID03] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

The following initial conditions need to be executed after the previous initial conditions are executed and in the following order:

- No default Tap&Pay service is selected
- Ensure that the Device is in battery power low mode

Step	Direction	Sequence	Expected Result
1	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82
2	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID03 as parameter	SW: 90 00 is returned with extra data “4f 46 46 48 4f 53 54”
3	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID08 as parameter	SW: 90 00 is returned with extra data “65 53 45”

15.7.3.14.2 Test Sequence No 2: Battery Low, payment routing, off-host (eSE) entry selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]
  This application defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "payment" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app02] defines an “Off-Host” payment service [serv02] for UICC in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - your service [serv02] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03] are installed to register their NFC services.

- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

The following initial conditions need to be executed after the previous initial conditions are executed and in the following order:

- Select “myOffHostServiceeSE01” banner from the “Tap&Pay” menu
- Ensure that the Device is in battery power low mode

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	
3	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82
5	PCD → DUT DUT → eSE	Send “SELECT APDU” command with AID07 as parameter	SW: 90 00 is returned with extra data “65 53 45”

15.7.3.14.3 Test Sequence No 3: Battery Low, payment routing, host entry selected in Tap&Pay

Initial Conditions

- All NFC applications on the DUT are uninstalled except applications that are preinstalled
- The default AID route is set to HCE. (See section 2.6.1)
- Application [app01]

  This application defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”

- A group with "payment" as category and containing one AID named [AID02]
- When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app02] defines an “Off-Host” payment service [serv02] for UICC in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - your service [serv02] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03] are installed to register their NFC services.

- An applet with [AID01] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”
- An applet with [AID02] as AID is installed on the UICC
  - When it is selected from a POS, SW:90 00 is returned + extra data “4f 46 46 48 4f 53 54”

The following initial conditions need to be executed after the previous initial conditions are executed and in the following order:

- Select “myHCEService01” banner from the “Tap&Pay” menu
- Ensure that the Device is in battery power low mode

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	

Step	Direction	Sequence	Expected Result
3	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID02 as parameter	Contactless selection fails with SW: 6A 82
4	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter	Contactless selection fails with SW: 6A 82
5	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID07 as parameter	Contactless selection fails with SW: 6A 82

15.7.3.14.4 Test Sequence No 4: Battery Low, various routing

Initial Conditions

- No AID is registered in the CLF routing table.
- Two applets with [AID01] and [AID03] as AID are installed on the UICC.
  - When it is selected from a POS, SW:90 00 is returned + extra data "4f 46 46 48 4f 53 54"
- The default AID route is set to HCE. (See section 2.6.1)

- Application [app01] defines an "Off-Host" payment service [serv01] for UICC in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID01"/>
</aid-group>
```

  - your service [serv01] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed "myOffHostService01"

```
< offhost-apdu-service
```

```
android:apduServiceBanner="@drawable/ myOffHostService01>
</offhost-apdu-service>
```

- Application [app02] defines “HCE” service as follows
  - “myHCEService01” as description
  - A banner where it is displayed “myHCEService01”
  - A group with "payment" as category and containing one AID named [AID02]
  - When it is selected from a POS, SW:90 00 is returned + extra data “48 43 45”

- Application [app03] defines an “Off-Host” payment service [serv03] for eSE in its Manifest.
  - With group "payment" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="payment">
<aid-filter android:name=" AID07"/>
</aid-group>
```

  - your service [serv03] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv03] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE01”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE01>
</offhost-apdu-service>
```

- Application [app04] defines an “Off-Host” other service [serv04] for UICC in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID03"/>
</aid-group>
```

  - your service [serv04] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- A banner where it is displayed “myOffHostService02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/ myOffHostService02>
</offhost-apdu-service>
```

- Application [app05] defines an “Off-Host” other service [serv05] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID08"/>
</aid-group>
```

  - your service [serv05] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
```



```
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service [serv05] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE02”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE02>
</offhost-apdu-service>
```

- Application [app06] defines an “Off-Host” other service [serv06] for eSE in its Manifest.
  - With group "other" as category and containing one AID as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name=" AID09"/>
</aid-group>
```

  - your service [serv06] declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service [serv06] declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>true</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostServiceeSE03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostServiceeSE03>
</offhost-apdu-service>
```

- Application [app01], application [app02], application [app03], application [app04] , application [app05] and application [app06] are installed to register their NFC services.

The following initial conditions need to be executed after the provious initial conditions are executed and in the following order:

- Select “myOffHostServiceeSE01” banner from the “Tap&Pay” menu
- Ensure that the Device is in battery power low mode

Step	Direction	Sequence	Expected Result
1	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
2	User → PCD	Power on the field	
3	PCD → DUT	Send “SELECT APDU” command with AID02 as parameter	Contactless selection fails with SW: 6A 82
4	PCD → DUT DUT → UICC	Send “SELECT APDU” command with AID01 as parameter	Contactless selection fails with SW: 6A 82

Step	Direction	Sequence	Expected Result
5	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID07 as parameter	SW: 90 00 is returned with extra data "65 53 45"
6	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID03 as parameter	SW: 90 00 is returned with extra data "4f 46 46 48 4f 53 54" As UICC applet will answer to the AID03 Select.
7	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID08 as parameter	SW: 90 00 is returned with extra data "65 53 45" As eSE applet will answer to the AID08 Select.
8	PCD → DUT DUT → eSE	Send "SELECT APDU" command with AID09 as parameter	SW: 90 00 is returned with extra data "65 53 45" As eSE applet will answer to the AID09 Select.

15.7.3.15 nonAID based services registration and conflict management

Test Purpose

Ensure DUT handles the registration of nonAID based services.

Referenced requirement

- TS26_NFC_REQ_094
- TS26_NFC_REQ_094.01
- TS26_NFC_REQ_094.02
- TS26_NFC_REQ_170
- TS26_NFC_REQ_170.1
- TS26_NFC_REQ_172
- TS26_NFC_REQ_175
- TS26_NFC_REQ_176

Initial Conditions

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

15.7.3.15.1 Test Sequence No 1: nonAID based service registration and selection on RF technology level (UICC service selection succeeds)

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- The NFC reader is establishing an ISO 14443-3 communication over type A.
- Install an Applet on the UICC, to handle CLT=A mode or use an intrinsic UICC mechanism (e.g. MIFARE Classic). When activated the Applet requests the Contactless paramsters according to “Mifare classic parameters” in Table 2 of GSMA SGP12 [42]
- Application [app01] defines a nonAID based “Off-Host” service for UICC in its Manifest.

  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="SIM"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService03>
</offhost-apdu-service>
```

- Application [app02] defines a nonAID based “Off-Host” service for eSE in its Manifest.

- your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService-eSE04”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
eSE04>
</offhost-apdu-service>
```

Step	Direction	Sequence	Expected Result
1	App → DUT	Install [app01] to register it’s NFC services	Installation is successful
2	App → DUT	Install [app02] to register it’s NFC services	The user is directed to a menu entry in “Settings” that lists the following conflicting services: myOffHostService03 myOffHostService-eSE04 The user is presented an option to select one and only one of these services.
3	User → DUT	Select myOffHostService03	myOffHostService03 is selected

Step	Direction	Sequence	Expected Result
4	PCD → DUT DUT → UICC	Use a contactless reader to exchange command with the UICC applet while remaining at ISO 14443-3 communication level (e.g. send a MIFARE authenticate command).	the command is received by the UICC and UICC response is received by the contactless reader
5	PCD → DUT	The test tool verifies the following contactless protocol parameters: GP Tag '80' – UID (LV) GP Tag '81' - SAK GP Tag '82' - ATQA GP Tag '83' – ATS (LV) GP Tag '84 - FWI/SFGI GP Tag '85' – CID support GP Tag '86' - Data_Rate Max	The values of these parameters are matching the values of profile 2 as defined in Table 3 of GSMA SGP12 [42]

15.7.3.15.2 Test Sequence No 2: nonAID based service registration and selection on RF technology level (UICC service selection fails)

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- The NFC reader is establishing an ISO 14443-3 communication over type A.
- Install an Applet on the UICC, to handle CLT=A mode or use an intrinsic UICC mechanism (e.g. MIFARE Classic)
- Application [app01] defines a nonAID based "Off-Host" service for UICC in its Manifest.

  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="SIM1"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService03>
</offhost-apdu-service>
```

- Application [app02] defines a nonAID based “Off-Host” service for eSE in its Manifest.
    - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService-eSE04”


```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
eSE04>
</offhost-apdu-service>
```

Step	Direction	Sequence	Expected Result
1	App → DUT	Install [app01] to register it's NFC services	Installation is successful
2	App → DUT	Install [app02] to register it's NFC services	The user is directed to a menu entry in "Settings" that lists the following conflicting services: myOffHostService03 myOffHostService-eSE04 The user is presented an option to select one and only one of these services.
3	User → DUT	Select myOffHostService-eSE04	myOffHostService-eSE04 is selected
4	PCD → DUT DUT → UICC	Use a contactless reader to exchange command with the UICC applet while remaining at ISO 14443-3 communication level (e.g. send a MIFARE authenticate command).	the command is not received by the UICC and the expected UICC response is not received by the contactless reader

15.7.3.15.3 Test Sequence No 3: nonAID based service registration and selection on RF protocol level (UICC service selection succeeds)

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- Install an applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCA. Note: The reader shall not explicitly select the Applet by AID. When activated the Applet requests the Contactless paramsters according to "DESFire EV1" in Table 2 of GSMA SGP12 [42]
- Application [app01] defines a nonAID based "Off-Host" service for UICC in its Manifest.
  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
```

```
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

- your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data  android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

- with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="SIM1"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService03>
</offhost-apdu-service>
```

- Application [app02] defines a nonAID based “Off-Host” service for eSE in its Manifest.
  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data  android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
```

```
        <se-ext-group>
            <se-id name="eSE"/>
        </se-ext-group>
    <AID-based>false</AID-based>
    </extensions>
```

- A banner where it is displayed “myOffHostService-eSE04”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
eSE04>
</offhost-apdu-service>
```

Step	Direction	Sequence	Expected Result
1	App → DUT	Install [app01] to register it’s NFC services	Installation is successful
2	App → DUT	Install [app02] to register it’s NFC services	The user is directed to a menu entry in “Settings” that lists the following conflicting services: myOffHostService03 myOffHostService-eSE04 The user is presented an option to select one and only one of these services.
3	User → DUT	Select myOffHostService03	myOffHostService03 is selected
4	PCD → DUT DUT → UICC UICC → DUT DUT → PCD	Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) to the UICC applet using a contactless reader Note: The reader shall access the UICC applet without explicitly selecting it by AID.	Status Word 90 00 is returned
5	PCD → DUT	The test tool verifies the following contactless protocol parameters: GP Tag ‘80’ – UID (LV) GP Tag ‘81’ - SAK GP Tag ‘82’ - ATQA GP Tag ‘83’ – ATS (LV) GP Tag ’84 - FWI/SFGI GP Tag ‘85’ – CID support GP Tag ‘86’ - Data_Rate Max	The values of these parameters are matching the values of profile 3 as defined in Table 3 of GSMA SGP12 [42]

15.7.3.15.4 Test Sequence No 4: nonAID based service registration and selection on RF protocol level (UICC service selection fails)

Initial Conditions

- If the phone supports a mechanism to change the default technology, protocol or Default AID route, then do a factory reset before the test
- The NFC reader is polling in type A only or provide a mechanism to make sure the NFC transaction will be performed using RF type A.
- Install an applet on the UICC implementing External Authenticate according to Annex A.4.4, implicitly selectable via NFCA. Note: The reader shall not explicitly select the Applet by AID

- Application [app01] defines a nonAID based “Off-Host” service for UICC in its Manifest.
  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="SIM"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService03”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService03>
</offhost-apdu-service>
```

- Application [app02] defines a nonAID based “Off-Host” service for eSE in its Manifest.
  - your service declaration must contain an intent filter in the meta-data element as define below

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

  - your service declaration must contain com.gsma.services.nfc.extensions

```
<meta-data   android:name="com.gsma.services.nfc.extensions"
android:resource="@xml/nfc_se"/>
```

  - with nfc_se xml file as defined bellow

```
<extensions xmlns:android="http://www.gsma.com"
android:description="@string/servicedesc">
     <se-ext-group>
          <se-id name="eSE"/>
     </se-ext-group>
<AID-based>false</AID-based>
</extensions>
```

- A banner where it is displayed “myOffHostService-eSE04”

```
< offhost-apdu-service
android:apduServiceBanner="@drawable/myOffHostService-
eSE04>
</offhost-apdu-service>
```

Step	Direction	Sequence	Expected Result
1	App → DUT	Install [app01] to register it’s NFC services	Installation is successful
2	App → DUT	Install [app02] to register it’s NFC services	The user is directed to a menu entry in “Settings” that lists the following conflicting services: myOffHostService03 myOffHostService-eSE04 The user is presented an option to select one and only one of these services.

Step	Direction	Sequence	Expected Result
3	User → DUT	Select myOffHostService-eSE04	myOffHostService-eSE04 is selected
4	PCD → DUT DUT → UICC UICC → DUT DUT → PCD	Send EXTERNAL AUTHENTICATE (acc to Annex A.4.4) to the UICC applet using a contactless reader Note: The reader shall try to access the UICC applet without explicitly selecting it by AID.	Status Word not equal 90 00 is returned

15.8 Platform Dependant Properties

15.8.1 General overview

This section provides test cases for checking platform dependant properties.

15.8.2 Conformance requirements

The Requirements tested are referenced in each test case.

15.8.3 Test Cases

15.8.3.1 "getVersion" API

For further test details, see TS.27 v13.0 [47]

15.8.3.2 "getProperty" API

For further test details, see TS.27 v13.0 [47]

15.8.3.2.1 Test Sequence No 1: OMAPI

For further test details, see TS.27 v13.0 [47]

15.8.3.2.2 Test Sequence No 2: MULTIPLE_ACTIVE_CEE

For further test details, see TS.27 v13.0 [47]

15.8.3.2.3 Test Sequence No 3: HCI_SWP, BATTERY_LOW_MODE

For further test details, see TS.27 v13.0 [47]

15.8.3.2.4 Test Sequence No 4: BATTERY_POWER_OFF_MODE, FELICA, MIFARE_CLASSIC, MIFARE_DESFIRE

For further test details, see TS.27 v13.0 [47]

15.8.3.2.5 Test Sequence No 5: Invalid argument

For further test details, see TS.27 v13.0 [47]

15.8.3.2.6 Test Sequence No 6: BATTERY_POWER_OFF_MODE, FELICA, MIFARE_CLASSIC, MIFARE_DESFIRE

For further test details, see TS.27 v13.0 [47]

15.9 Security

15.9.1 General overview

This section provides test cases for checking security requirements.

15.9.2 Conformance requirements

The Requirements tested are referenced in each test case.

15.9.3 Test Cases

15.9.3.1 Permissions

Ensure DUT implements correctly the requested permissions for using NFC services.

Referenced requirement

- TS26_NFC_REQ_130.1
- TS26_NFC_REQ_131
- TS26_NFC_REQ_190
- TS26_NFC_REQ_191

15.9.3.1.1 Test Sequence No 1: Protection level for NFC Permission

Initial Conditions

- Application [app01]
  Registers in its Manifest the following permissions:
  - android.permission.NFC
  - For devices before Android 9: org.simalliance.openmobileapi.SMARTCARD
    Note: For devices based on Android 9 or following Android releases: org.simalliance.openmobileapi.SMARTCARD will not be used by the device .
  - For devices before Android 9: com.gsma.services.nfc.permission.TRANSACTION_EVENT
  - For devices based on Android 9 or following Android releases: android.permission.NFC_TRANSACTION_EVENT.
  It is allowed to use the same application for both devices before Android 9 and devices based on Android 9 or following Android releases
  Provides the following features

- Retrieves the list of readers via OMAPI
- Displays a notification when a transaction event is received

- Application [app01] is built with the following parameters:
  - "compileSdkVersion" >= 23
  - "targetSdkVersion" >=23
  - "minSdkVersion"<23
- Application [app01] is not yet installed on the DUT
- Access Control is allowing communication between any applet in the UICC and [app01]

Step	Direction	Sequence	Expected Result
1	User → DUT	Install [app01] without using "adb install" command	• For devices based on an Android release before "Marshmallow", the framework is requesting to accept, at least, the following permissions: - org.simalliance.openmobileapi. SMARTCARD - com.gsma.service.nfc.permission. TRANSACTION_EVENT • For devices based on "Marshmallow" or following Android releases, the framework is not requesting to accept the following permissions - org.simalliance.openmobileapi. SMARTCARD - com.gsma.service.nfc.permission. TRANSACTION_EVENT - android.permission.NFC_TRANSACTION_EVENT
2	App → DUT	Retrieve and display a list of available readers	Android is not requesting to accept any additional permissions
3	PCD	Power on RF field	
4	PCD→ DUT	Perform RF protocol initialisation	
5	PCD→ DUT	Using the **APDU application**, send a SELECT command with AID01	**APDU Application** receives Status Word 90 00
6	PCD	Power off RF field	
7	DUT→ UICC	Send EVT_FIELD_OFF	
8	UICC → DUT	UICC sends EVT_TRANSACTION with AID01	• The DUT does not request to accept any additional permissions The application displays a notification linked to the transaction event

15.9.3.1.2 Test Sequence No 2: Permissions for using NFC services

Initial Conditions

- Application [app01]
  Registers in its Manifest the following permissions:
  - android.permission.NFC
  - For devices before Android 9:
  com.gsma.services.nfc.permission.TRANSACTION_EVENT
  - For devices based on Android 9 or following Android releases:
  android.permission.NFC_TRANSACTION_EVENT.
  It is allowed to use the same application for both devices before Android 9 and devices based on Android 9 or following Android releases.
  Registers an activity for receiving a transaction event based on [AID01]

- Application [app02]
  Registers in its Manifest the following permissions:
  - android.permission.NFC
  Registers an activity for receiving a transaction event based on [AID02]

- Application [app03] VOID

- Application [app04]
  Does not registers in its Manifest the following permissions:
  - android.permission.NFC
  - com.gsma.services.nfc.permission.TRANSACTION_EVENT
  - android.permission.NFC_TRANSACTION_EVENT.
  It is allowed to use the same application for both devices before Android 9 and devices based on Android 9 or following Android releases.
  Registers an activity for receiving a transaction event based on [AID04]

- Access Control is allowing communication between any applets in the UICC and any applications

Step	Direction	Sequence	Expected Result
1	App → DUT	Generate a transaction event (see procedure 2.6.3) based on [AID01]	• Transaction Event Activity from [app01] is launched
2	App → DUT	Generate a transaction event (see procedure 2.6.3) based on [AID02]	• Transaction event Activity from [app02] is not launched
3	App → DUT	Generate a transaction event (see procedure 2.6.3) based on [AID04]	• Transaction event Activity from [app04] Is not launched

15.9.3.2 APDU Logs

Ensure DUT avoid to log any sensitive information such as APDU exchange

Referenced requirement

- TS26_NFC_REQ_163

15.9.3.2.1 Test Sequence No 1: APDU Logs for contactless transaction

Initial Conditions

- Application [app01] define an “OffHost” other service [serv01] in its Manifest.
    - With group “other” as category and containing AID01 as defined below

```
<aid-group android:description="@string/aiddescription"
android:category="other">
<aid-filter android:name= [AID 01]/>
</aid-group>
```

    - your service [serv01] declaration must contain an intent filter

```
<intent-filter>
<action android:name =
"android.nfc.cardemulation.action.OFF_HOST_APDU_SERVICE"/>
</intent-filter>
```

Registers in its Manifest the following permissions:
- android.permission.NFC
- For devices before Android 9: org.simalliance.openmobileapi.SMARTCARD
  Note: For devices based on Android 9 or following Android releases: org.simalliance.openmobileapi.SMARTCARD will not be used by the device.
- Before Android 9: com.gsma.services.nfc.permission.TRANSACTION_EVENT
- For devices based on Android 9 or following Android releases: android.permission.NFC_TRANSACTION_EVENT.

It is allowed to use the same application for both devices before Android 9 and devices based on Android 9 or following Android releases

app01 is built to receive transaction event from AID01 cardlet.

- Applet with [AID01] as AID is installed on the UICC. [AID01] is of size 16 bytes.
    - When the cardlet is selected from the contactless interface, a transaction event is sent to the DUT containing additional data generated by the cardlet.
      The additional data shall be constructed such that its occurrence in the logcat file guarantees that it originated from the transaction event. Examples: the AID of the cardlet, or random bytes of sufficient length.
- Access Control is allowing communication between any applet in the UICC and [app01]

Step	Direction	Sequence	Expected Result
1	User → DUT	Clear pre-existing logs on the device using the following command: adb logcat -b radio -b main –c	
2	User → DUT	While the field is off, place the DUT in the area where the field will be powered on	
3	User → PCD	Power on the field	
4	PCD → DUT DUT → UICC	Send "SELECT APDU" command with AID01 as parameter.	SW: 90 00 is returned App01 has received the push transaction event from the cardlet containing the additional data provided by the cardlet
5	User → PCD	Power off the field	
6	User →DUT	Extract the logcat "main" and "radio" logs of the devices using the following command: adb logcat -v time -d > main.txt adb logcat -b radio -v time -d > radio.txt See Note	No occurrence of AID01 is found in logs No occurrence of the additional data generated by the cardlet is found in logs

Note: In order to ensure that the logcat content is complete the test tool needs to ensure that the main.txt and radio.txt contains the complete log data from Step2 to Step5.

15.9.3.2.2 Test Sequence No 2: APDU Logs for OMAPI access

Initial Conditions

- Application [app01] registers in its Manifest the following permissions:
  - android.permission.NFC
  - For devices before Android 9: org.simalliance.openmobileapi.SMARTCARD
    Note: For devices based on Android 9 or following Android releases: org.simalliance.openmobileapi.SMARTCARD will not be used by the device.

  It is allowed to use the same application for both devices before Android 9 and devices based on Android 9 or following Android releases

- Applet with [AID01] as AID is installed on the UICC. [AID01] is of size 16 bytes.

Step	Direction	Sequence	Expected Result
1	User → DUT	Clear pre-existing logs on the device using the following command: adb logcat -b radio -b main –c	
2	DUT → UICC	Send “SELECT APDU” command with AID01 as parameter on the contact interface using OpenMobileAPI	SW: 90 00 is returned
3	User → DUT	Extract the logcat “main” and “radio” logs of the devices using the following command: adb logcat -v time -d > main.txt adb logcat -b radio -v time -d > radio.txt and verify if AID01 is found in the logs See Note	No occurrence of AID01 is found

Note: In order to ensure that the logcat content is complete the test tool needs to ensure that the main.txt and radio.txt contains the complete log data of Step2.

16 VOID

17 VOID

18 VOID

19 Other OS specific test cases

Other OS specific test cases can be added based on contributions.

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.4.1, 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 05 59 50 00 00 00 00 00 00 52 41 44 50
- Applet A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 41

A.3 Structure File

The structure file of the applet test is as follows:

5F 00 (DR) Folder

1F 00 (EF) First 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 A.1: 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 A.2: 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 A.3: 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	

Code	Value	Meaning
INS	82	
P1	00	
P2	00	
Lc	08	
Data	9E EA C0 F9 4D 60 53 34	

Table A.4: External Authenticate command details

A.5 Source Code (Java)

The Java Source Code can be obtained from the GSMA TSG NFC Public GitHub here:

https://github.com/GSMATerminals/NFC-Test-Book-Public

Annex B Reference to other test plan

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

B.1 GlobalPlatform OMAPI

Note: The SIMalliance group published the "OMAPI Transport API Test Specification" until version 2.2 and Second Errata. The specification has thereafter moved to GlobalPlatform.

Reference test Specification: The test book refers to "SIMallianceGlobalPlatform Open Mobile API test specification for Transport API [5]

"GlobalPlatform 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 based on SE type:

Options	Name	GSMA Status for UICC	GSMA Status for eSE
access to the basic channel is blocked by the DUT	OP-002	Mandatory	Optional
access to the basic channel is allowed by the DUT	OP-003	SHALL not be supported	Optional
access to the default applet is blocked by the DUT	OP-011	Mandatory	Optional
access to the default applet is allowed by the DUT	OP-010	SHALL not be supported	Optional

Table B.1.1: Optional Features that are mandatory

Note: for some specific behaviour of the test tool when testing the "GlobalPlatform OMAPI Transport API Test Specification" [5], see section 2.5.1.1.

The test cases listed in Table B.1.2 are applicable according to the applicability table of the referred GlobalPlatform test specification:

Each test case listed below contains one, or more ID-s listed explicitly in "GlobalPlatform OMAPI Transport API Test Specification" [5]. The ID-s shall be handled as separate test cases.

The "TS.26 versions" column gives the item in the "Test Case number and description" column the applicable requirements version:

- Version x.y "onwards": if the requirement is applicable from this TS.26 version and for the later versions
- "Up to" Version x.y: if the requirement is applicable until this TS.26 version and it has been changed or replaced for the later TS.26 versions

TS.27 Numbering	GP OMAPI Section [5]	Test case number and description	Test case IDs	TS.26 versions
6.3.1.6.1.1	6.1.1	GlobalPlatform OMAPI - Constructor: SEService(Context context, SEService.CallBack listener)	ID1 – ID6	6.0 onwards
6.3.1.6.1.2	6.1.2	GlobalPlatform OMAPI - Method: Reader[] getReaders()	ID1	6.0 onwards
6.3.1.6.1.3	6.1.3	GlobalPlatform OMAPI - Method: boolean isConnected()	ID1, ID2	6.0 onwards
6.3.1.6.1.4	6.1.4	GlobalPlatform OMAPI - Method: void shutdown()	ID1 - ID3	6.0 onwards
6.3.1.6.1.5	6.1.5	GlobalPlatform OMAPI - Method: String getVersion()	ID1	6.0 onwards
6.3.1.6.2.1	6.2.1	GlobalPlatform OMAPI - Method: void serviceConnected(SEService service)	ID1	6.0 onwards
6.3.1.6.3.1	6.3.1	GlobalPlatform OMAPI - Method: String getName()	ID1	6.0 onwards
6.3.1.6.3.1eSE	6.3.1	GlobalPlatform OMAPI - Method: String getName()	ID1	11.0 onwards
6.3.1.6.3.2	6.3.2	GlobalPlatform OMAPI - Method SEService getSEService()	ID1	6.0 onwards
6.3.1.6.3.3	6.3.3	GlobalPlatform OMAPI - Method: boolean isSecureElementPresent()	ID1, ID2	6.0 onwards
6.3.1.6.3.3eSE	6.3.3	GlobalPlatform OMAPI - Method: boolean isSecureElementPresent()	ID1	11.0 onwards
6.3.1.6.3.4	6.3.4	GlobalPlatform OMAPI - Method: Session openSession()	ID1 – ID3	6.0 onwards
6.3.1.6.3.5	6.3.5	GlobalPlatform OMAPI - Method: void closeSessions()	ID1, ID2	6.0 onwards
6.3.1.6.3.6	6.3.6	Reader:Event types value	ID1, ID2	10.0 onwards
6.3.1.6.3.7	6.3.7	Method:void registerReaderEventCallback (Reader.EventCallBack cb)	ID1 - ID8	10.0 onwards
6.3.1.6.3.8	6.3.8	Method:void unregisterReaderEventCallback (Reader.EventCallBack cb)	ID1 – ID5	10.0 onwards
6.3.1.6.4.1	6.4.1	GlobalPlatform OMAPI - Method: Reader getReader()	ID1, ID2	6.0 onwards
6.3.1.6.4.2	6.4.2	GlobalPlatform OMAPI - Method: byte[] getATR()	ID1 – ID3	6.0 onwards

TS.27 Numbering	GP OMAPI Section [5]	Test case number and description	Test case IDs	TS.26 versions
6.3.1.6.4.3	6.4.3	GlobalPlatform OMAPI - Method: void close()	ID1, ID2	6.0 onwards
6.3.1.6.4.4	6.4.4	GlobalPlatform OMAPI - Method: boolean isClosed()	ID1, ID2	6.0 onwards
6.3.1.6.4.5	6.4.5	GlobalPlatform OMAPI - Method: void closeChannels()	ID1, ID2	6.0 onwards
6.3.1.6.4.6	6.4.6	GlobalPlatform OMAPI - Method: Channel openBasicChannel()	ID7	6.0 onwards
6.3.1.6.4.7	6.4.7	GlobalPlatform OMAPI - Method: Channel openLogicalChannel()	ID1, ID2, ID3b, ID5a, ID5c, ID6 – ID17	6.0 onwards
			ID5b, ID5d	6.0 to 12.0
6.3.1.6.4.7b	6.4.7	GlobalPlatform OMAPI - Method: Channel openLogicalChannel()	ID18 – ID23	10.0 onwards
6.3.1.6.4.7eSE	6.4.7	GlobalPlatform OMAPI - Method: Channel openLogicalChannel()	ID1, ID2, ID3a, ID5a, , ID5c, , ID6, ID7, ID9 – ID23	11.0 onwards
			ID5b, ID5d	11.0 to 12.0
6.3.1.6.4.8	6.4.8	GlobalPlatform OMAPI - Method: Channel openLogicalChannel – Extended logical channels	ID1, ID3	8.0 onwards
			ID2	8.0 to 12.0
6.3.1.6.4.9	6.4.9	GlobalPlatform OMAPI - Method: Channel openBasicChannel (with P2)	ID7	8.0 onwards
6.3.1.6.4.10	6.4.10	GlobalPlatform OMAPI - Method: Channel openLogicalChannel (with P2)	ID1, ID2, ID3b, ID5a, ID5c, ID6 – ID20	8.0 onwards
			ID5b, ID5d	8.0 to 12.0
6.3.1.6.4.10b	6.4.10	GlobalPlatform OMAPI - Method: Channel openLogicalChannel (with P2)	ID21 – ID26	10.0 onwards
6.3.1.6.4.11	6.4.11	GlobalPlatform OMAPI - Method: Channel openLogicalChannel (with P2) – Extended logical channels	ID1, ID3	8.0 onwards
			ID2	8.0 to 12.0
6.3.1.6.5.1	6.5.1	SIMalliance APIs GlobalPlatform OMAPI - Method: void close()	ID1, ID3 – ID6	6.0 onwards

TS.27 Numbering	GP OMAPI Section [5]	Test case number and description	Test case IDs	TS.26 versions
6.3.1.6.5.2	6.5.2	GlobalPlatform OMAPI - Method: boolean isBasicChannel()	ID2	6.0 onwards
6.3.1.6.5.3	6.5.3	GlobalPlatform OMAPI - Method: boolean isClosed()	ID1, ID2	6.0 onwards
6.3.1.6.5.4	6.5.4	GlobalPlatform OMAPI - Method: byte[] getSelectResponse()	ID1, ID2, ID4 – ID12	6.0 onwards
6.3.1.6.5.4b	6.5.4	GlobalPlatform OMAPI - Method: byte[] getSelectResponse()	ID13 – ID32	10.0 onwards
6.3.1.6.5.5	6.5.5	GlobalPlatform OMAPI - Method: Session getSession()	ID1	6.0 onwards
6.3.1.6.5.6	6.5.6	GlobalPlatform OMAPI - Method: byte[] transmit(byte[] command)	ID2 – ID21, ID23 – ID29	6.0 onwards
6.3.1.6.5.6b	6.5.6	GlobalPlatform OMAPI - Method: byte[] transmit(byte[] command)	ID30 – ID39	10.0 onwards
6.3.1.6.5.6eSE	6.5.6	GlobalPlatform OMAPI - Method: byte[] transmit(byte[] command)	ID2 – ID7, ID9 – ID11, ID15 – ID17, ID19 – ID21, ID23 – ID39	11.0 onwards
6.3.1.6.5.7	6.5.7	GlobalPlatform OMAPI - Method: Boolean[] selectNext()	ID1 – ID5, ID7 – ID9	6.0 onwards
6.3.1.6.5.7eSE	6.5.7	GlobalPlatform OMAPI - Method: Boolean[] selectNext()	ID1 – ID4, ID7 – ID9	11.0 onwards
6.3.1.6.5.8	6.5.8	GlobalPlatform OMAPI - Method: Boolean[] isOpen()	ID1, ID2	13.0 onwards

Table B.1.2: GlobalPlatform OMAPI test cases

B.2 EMVCo

The GSMA requires device manufacturer to pass the EMVCo Level 1 testing according to EMVCo test plan in the scope of a device evaluation. This applies for Analog, Digital [38], Performance and Interoperability testing [39].

Completion of EMVCo testing is not considered a pre-requisite for a device vendor to start testing for all test cases in defined in the GSMA TS.27 NFC Handset Test Book. A device vendor may have all test cases defined in the GSMA TS.27 NFC Handset Test Book conducted before testing with EMVCo or in parallel with testing with EMVCo.

B.3 VOID

B.4 ETSI TS 102 613 SWP

Reference test Specification: ETSI TS 102 694-1 [11]

ETSI TS 102 694-1 [11] specifies a number of optional features for the device. The following table lists which optional features from ETSI TS 102 694-1 [11] 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
7	Window size of 3	O_WS_3	R
8	Window size of 4	O_WS_4	R
9	HCI as per ETSI TS 102 622 [10]	O_102_622	M
11	CLT, ISO/IEC 18092 [28]	O_CLT_F	M
18	Card Emulation, ISO/IEC 14443-4 type A	O_CE_A	M
19	Card Emulation, ISO/IEC 14443-4 type B	O_CE_B	M
21	Terminal supports CLT, ISO/IEC 14443-3 [5] Type A independently of whether the UICC indicates support of extended bit durations	O_CLT_A_FULL	C002
22	Terminal supports CLT, ISO/IEC 14443-3 [5] Type A only when the UICC indicates support of extended bit durations down to 0,590 µs	O_CLT_A_EXTENDED_ONLY	C002
C001: IF O_BAT_OFF THEN M ELSE O C002: Either O_CLT_A_FULL or O_CLT_A_EXTENDED_ONLY shall be supported but not both.Note 1: The option “Class C low power mode” is related to the option O_BAT_OFF as defined within TS.27 Section 2.1.4. The test cases which are conditional for “Class C low power mode” will be performed according to condition C001 as defined above.			

Table B.4.1: Optional Features from ETSI TS 102 694-1

The following test cases are applicable:

1) Test cases verified by GCF WI 133 are listed in the Table below. 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)

Index	TC Title
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 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

Index	TC Title
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
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 B.4.2: List of applicable test cases from GCF WI 133

2) Test cases verified by GCF WI 190 [26] are listed in Table B.4.3. 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)

Index	TC Title
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 B.4.3: List of additional test cases

B.5 ETSI TS 102 622 [10] 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 supported	O_CE_TypeA	M
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	M
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	M

Item	Option	Mnemonic	GSMA Status
10	CLT for Type A as specified in ETSI TS 102 613 [9] 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			

Table B.5.1: Optional Features from ETSI TS 102 695-1

The following test cases shall be verified:

1) Test cases verified by GCF WI 133 are listed in Table B.5.2. 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
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

Index	TC Title
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 B.5.2: List of applicable test cases from GCF WI 133

2) Test cases verified by GCF WI 190 [26] are listed in Table B.5.3. These test cases are validated by GCF.

Index	TC Title
5.6.1.4	Test case x: RF gate of type F
5.6.4.1.4	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
5.6.4.2.3	Test case 2: Routing EVT_FIELD_ON and EVT_FIELD_OFF to RF Gate with lowest GID
5.6.4.4.2	Test case 1: ISO/IEC 18092 Type F
5.6.4.4.3	Test case y: RF off during ISO/IEC 18092 [28] Type F commands handling
5.6.4.4.4	Test case 3: Routing EVT_FIELD_ON and EVT_FIELD_OFF to RF Gate with lowest GID
5.7.2.3.1.2	Test case 1: ISO/IEC 14443-4 compliant type A

Table B.5.3: List of additional test cases

B.6 ETSI TS 102.384 [13], 3GPP 31.124

Reference test Specification: ETSI TS 102 384 [13] and 3GPP TS 31.124 v10.0.0

The test cases in Table B.6.1 are applicable to verify TS26_NFC_REQ_078 as following:

1) Applicable test cases verified by GCF WI 035 are listed in Table B.6.1. 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)
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

Ref Spec	Index	TC Title	Sequence Name
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 B.6.1: List of applicable test cases from GCF WI – 035 [15]

The applicable test cases to verify TS26_NFC_REQ_079

1) The applicable test case from 3GPP TS 31.124 is listed in Table B.6.2.

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 B.6.2: applicable test cases from GCF WI 035

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 B.6.3

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 B.6.3: 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 B.6.4

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 B.6.4: Applicable test cases

The test cases in Table B.6.5 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 B.6.5: List of additional test cases

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
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)	No
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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)	Yes
5.4.2.2		2	ACCESS_DENIED_ERROR_APDU_AR_DO_Bad_value (c0-62-b5)	Yes
5.4.2.3		3	ACCESS_DENIED_ERROR_NFC_AR_DO_Bad_Length (c0-90-88)	Yes
5.4.2.4		4	ACCESS_DENIED_ERROR_NFC_AR_DO_Bad_Value (c0-d7-66)	Yes
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)	Yes
5.4.4.2		2	ALGORITHM_A_SUCCESS__R1_SEApp1_DevApp1__R2_All_All__request_DevApp2_SEApp1 (ff-ee-09)	Yes
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)	Yes
5.4.5.2		2	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1__R2_All_All__request_DevApp2_SEApp1 (ff-d4-fb)	Yes
5.4.5.3		3	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_ALWAYS__R2_All_SEApp_DevApp2_NEVER (c0-40-30)	Yes
5.4.5.4		4	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_ALWAYS__R2_All_SEApp_DevApp2_NEVER (ff-40-30)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.5.5		5	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_NEVER__R2_All_SEApp_DevApp2_ALWAYS (c0-80-26)	Yes
5.4.5.6		6	ALGORITHM_C_SUCCESS__R1_All_SEApp_DevApp1_NEVER__R2_All_SEApp_DevApp2_ALWAYS (ff-80-26)	Yes
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)	Yes
5.4.6.2		2	ALGORITHM_D_SUCCESS__All_SEApp_All_DevApp_APDU_access_ALWAYS (ff-2e-b0)	Yes
5.4.6.3		3	ALGORITHM_D_SUCCESS__No_Rule (c0-cf-16)	Yes
5.4.6.4		4	ALGORITHM_D_SUCCESS__No_Rule (ff-cf-16)	Yes
5.4.7.1	ANNEX_D_EXAMPLE_01	1	ANNEX_D_SUCCESS__example_01 (c0-38-cb)	Yes
5.4.7.2		2	ANNEX_D_SUCCESS__example_01 (ff-38-cb)	Yes
5.4.8.1	ANNEX_D_EXAMPLE_02	1	ANNEX_D_SUCCESS__example_02 (c0-a1-46)	Yes
5.4.8.2		2	ANNEX_D_SUCCESS__example_02 (ff-a1-46)	Yes
5.4.9.1	ANNEX_D_EXAMPLE_03	1	ANNEX_D_SUCCESS__example_03 (c0-39-49)	Yes
5.4.9.2		2	ANNEX_D_SUCCESS__example_03 (ff-39-49)	Yes
5.4.10.1	ANNEX_D_EXAMPLE_04	1	ANNEX_D_SUCCESS__example_04 (c0-2f-62)	Yes
5.4.10.2		2	ANNEX_D_SUCCESS__example_04 (ff-2f-62)	Yes
5.4.11.1	ANNEX_D_EXAMPLE_05	1	ANNEX_D_SUCCESS__example_05 (c0-d2-97)	Yes
5.4.11.2		2	ANNEX_D_SUCCESS__example_05 (ff-d2-97)	Yes
5.4.12.1	ANNEX_D_EXAMPLE_06	1	ANNEX_D_SUCCESS__example_06 (c0-51-08)	Yes
5.4.12.2		2	ANNEX_D_SUCCESS__example_06 (ff-51-08)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.13.1	ANNEX_D_EXAMPLE_07	1	ANNEX_D_SUCCESS__example_07 (c0-e0-83)	Yes
5.4.13.2		2	ANNEX_D_SUCCESS__example_07 (ff-e0-83)	Yes
5.4.14.1	ANNEX_D_EXAMPLE_09	1	ANNEX_D_SUCCESS__example_09 (c0-4c-cb)	Yes
5.4.14.2		2	ANNEX_D_SUCCESS__example_09 (ff-4c-cb)	Yes
5.4.15.1	ANNEX_D_EXAMPLE_10	1	ANNEX_D_SUCCESS__example_10 (c0-ad-7d)	Yes
5.4.15.2		2	ANNEX_D_SUCCESS__example_10 (ff-ad-7d)	Yes
5.4.16.1	ANNEX_D_EXAMPLE_11	1	ANNEX_D_SUCCESS__example_11 (c0-f4-52)	Yes
5.4.16.2		2	ANNEX_D_SUCCESS__example_11 (ff-f4-52)	Yes
5.4.17.1	ANNEX_D_EXAMPLE_12	1	ANNEX_D_SUCCESS__example_12 (c0-24-84)	Yes
5.4.17.2		2	ANNEX_D_SUCCESS__example_12 (ff-24-84)	Yes
5.4.18.1	ANNEX_D_EXAMPLE_13	1	ANNEX_D_SUCCESS__example_13 (c0-29-86)	Yes
5.4.18.2		2	ANNEX_D_SUCCESS__example_13 (ff-29-86)	Yes
5.4.19.1	ANNEX_D_EXAMPLE_14	1	ANNEX_D_SUCCESS__example_14 (c0-72-b7)	Yes
5.4.19.2		2	ANNEX_D_SUCCESS__example_14 (ff-72-b7)	Yes
5.4.20.1	ANNEX_D_EXAMPLE_15	1	ANNEX_D_SUCCESS__example_15 (c0-db-cb)	Yes
5.4.20.2		2	ANNEX_D_SUCCESS__example_15 (ff-db-cb)	Yes
5.4.21.1	ANNEX_D_EXAMPLE_16	1	ANNEX_D_SUCCESS__example_16 (c0-cf-17)	Yes
5.4.21.2		2	ANNEX_D_SUCCESS__example_16 (ff-cf-17)	Yes
5.4.22.1	ANNEX_D_EXAMPLE_17	1	ANNEX_D_SUCCESS__example_17 (c0-0b-29)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.22.2		2	ANNEX_D_SUCCESS__example_17 (ff-0b-29)	Yes
5.4.23.1	ANNEX_D_EXAMPLE_18	1	ANNEX_D_SUCCESS__example_18 (c0-71-16)	Yes
5.4.23.2		2	ANNEX_D_SUCCESS__example_18 (ff-71-16)	Yes
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)	Yes
5.4.24.6		6	APDU_FILTER_DEFINITION_SUCCESS__FILTER__2_filters__true (ff-6c-24)	Yes
5.4.24.7		7	APDU_FILTER_DEFINITION_SUCCESS__FILTER__3_filters__true (c0-44-7d)	Yes
5.4.24.8		8	APDU_FILTER_DEFINITION_SUCCESS__FILTER__3_filters__true (ff-44-7d)	Yes
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)	Yes
5.4.25.2		2	BIG_RULES_MANAGEMENT_One_Big_Rule (ff-64-2b)	Yes
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
5.4.26.17		17	CONFLICT_SUCCESS__1_SEApp_1_DevApp__R1_apdu_always__R2_apdu_always (c0-73-51)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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)	Yes
5.4.26.92		92	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_0_match__R2_apdu_always (ff-b0-96)	Yes
5.4.26.93		93	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (c0-52-54)	Yes
5.4.26.94		94	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_1_match__R2_apdu_always (ff-52-54)	Yes
5.4.26.95		95	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (c0-70-6c)	Yes
5.4.26.96		96	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_1_filter_2_match__R2_apdu_always (ff-70-6c)	Yes
5.4.26.97		97	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (c0-58-13)	Yes
5.4.26.98		98	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_2_filters_1_match_each__R2_apdu_always (ff-58-13)	Yes
5.4.26.99		99	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (c0-4b-98)	Yes
5.4.26.100		100	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_0_match (ff-4b-98)	Yes
5.4.26.101		101	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (c0-38-d9)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.102		102	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_1_match (ff-38-d9)	Yes
5.4.26.103		103	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (c0-85-6b)	Yes
5.4.26.104		104	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_1_filter_2_match (ff-85-6b)	Yes
5.4.26.105		105	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (c0-3c-9f)	Yes
5.4.26.106		106	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_2_filters_1_match_each (ff-3c-9f)	Yes
5.4.26.107		107	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (c0-06-0b)	Yes
5.4.26.108		108	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_always (ff-06-0b)	Yes
5.4.26.109		109	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (c0-c8-59)	Yes
5.4.26.110		110	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_always__R2_apdu_never (ff-c8-59)	Yes
5.4.26.111		111	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (c0-76-bc)	Yes
5.4.26.112		112	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_filter (ff-76-bc)	Yes
5.4.26.113		113	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (c0-b0-d6)	Yes
5.4.26.114		114	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_filter__R2_apdu_never (ff-b0-d6)	Yes
5.4.26.115		115	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (c0-09-ca)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.26.116		116	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_always (ff-09-ca)	Yes
5.4.26.117		117	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (c0-b1-4f)	Yes
5.4.26.118		118	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_filter (ff-b1-4f)	Yes
5.4.26.119		119	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (c0-74-d7)	Yes
5.4.26.120		120	CONFLICT_SUCCESS__All_SEApp_All_DevApp__R1_apdu_never__R2_apdu_never (ff-74-d7)	Yes
5.4.26.121		121	CONFLICT_SUCCESS__R1_1_SEApp_1_DevApp_ALWAYS__R2_All_SEApp_1_DevApp_NEVER (c0-61-bf)	Yes
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
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)	Yes
5.4.30.2		2	RULES_UPDATED_SUCCESS__All_rules_deleted (ff-53-95)	Yes
5.4.30.3		3	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_ALWAYS__New_All_SEApp_All_DevApp_ALWAYS (c0-60-75)	Yes
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)	Yes
5.4.30.6		6	RULES_UPDATED_SUCCESS__Old_All_SEApp_DevApp1_NEVER__New_All_SEApp_DevApp2_ALWAYS (ff-d9-54)	Yes
5.4.30.7		7	RULES_UPDATED_SUCCESS__Rule_All_SEApp_DevApp1_NEVER__Delete_All_SEApp_All_DevApp (c0-df-ce)	Yes

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
5.4.30.8		8	RULES_UPDATED_SUCCESS__Rule_All_SEApp_DevApp1_NEVER__Delete_SEApp1_DevApp1_ALWAYS (c0-07-84)	Yes
5.4.30.9		9	RULES_UPDATED_SUCCESS__rule_equally_restrictive_added__filters_merged (c0-e5-98)	Yes
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)	Yes
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)	Yes
5.4.30.14		14	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_ALWAYS_to_NEVER (ff-06-c3)	Yes
5.4.30.15		15	RULES_UPDATED_SUCCESS__rule_modified_from_APDU_access_NEVER_to_ALWAYS (c0-90-64)	Yes
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)	Yes
5.4.30.18		18	RULES_UPDATED_SUCCESS__rule_more_restrictive_added (ff-90-c9)	Yes
5.4.30.19		19	RULES_UPDATED_SUCCESS__rule_with_higher_priority_added (c0-a2-fd)	Yes
5.4.30.20		20	RULES_UPDATED_SUCCESS__rule_with_higher_priority_added (ff-a2-fd)	Yes
5.4.30.21		21	RULES_UPDATED_SUCCESS__rule_with_lower_priority_added (c0-93-41)	Yes
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

TS.27 Numbering	Requirement	ID Test Case	GlobalPlatform Test case	Included
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
5.4.32.10		10	SPECIFIC_RULES_PRIORITY_SUCCESS__R1_All_SEApp_1_DevApp__R2_All_SEApp_All_DevApp (ff-9b-05)	Yes

Table B.8.1: GlobalPlatform Secure Element Access Control Test Cases

B.9 NFC Forum Tag Operation, Analog and Digital Testing

B.9.1 Tag Operation

Reference test Specification: The test book refers to “NFC Forum Test Cases for Type 1 Tag Operation” specification.

For further test details, see TS.27 v13.0 [47]

Reference test Specification: The test book refers to “NFC Forum Test Cases for Type 2 Tag Operation” specification.

The following table indicates which test cases are included in the current version of the Test Book:

TS.27 Numbering	NFC Forum	Test case description
3.3.3.24.2.3.5.2.1	3.5.2.1	NDEF Read from Version 1.2 Type 2 tag [TC_T2T_NDA_BV_1]
3.3.3.24.2.3.5.2.2	3.5.2.2	NDEF Read from Version 2.0 Type 2 tag [TC_T2T_NDA_BV_2]
3.3.3.24.2.3.5.4.1	3.5.4.1	NDEF Detection and Read from Type 2 tag (x = 0 to 2) [TC_T2T_NDA_BV_3_x]
3.3.3.24.2.3.5.4.2	3.5.4.2	NDEF Write on INITIALIZED Type 2 tag (x = 0 to 2) [TC_T2T_NDA_BV_4_x]
3.3.3.24.2.3.5.4.3	3.5.4.3	NDEF Write on READ-ONLY Type 2 tag [TC_T2T_NDA_BV_5]
3.3.3.24.2.3.5.4.4	3.5.4.4	Transitions from READ/WRITE to READ-ONLY (x = 0 to 2) [TC_T2T_NDA_BV_6_x]

Table B.9.2: NFC Forum Test Cases for Type 2 Tag Operation

Reference test Specification: The test book refers to “NFC Forum Test Cases for Type 3 Tag Operation”.

The following table indicates which test cases are included in the current version of the Test Book:

TS.27 Numbering	NFC Forum	Test case description
3.3.3.24.3.3.1.1.1	3.1.1.1	Manufacture Parameter (PMm) and Maximum Timing [TC_T3T_MEM_BV_1]
3.3.3.24.3.3.3.1.1	3.3.1.1	Frame Structure and Communication Protocol [TC_T3T_FTH_BV_1]
3.3.3.24.3.3.4.1.1	3.4.1.1	Update Command and Check Command [TC_T3T_CSE_BV_1]
3.3.3.24.3.3.4.2.1	3.4.2.1	Block List Format [TC_T3T_CSE_BV_2]
3.3.3.24.3.3.5.1.1	3.5.1.1	NDEF Management data RWFlag [TC_T3T_NDA_BV_1]
3.3.3.24.3.3.5.2.1	3.5.2.1	Versioning (x=0 to 2) [TC_T3T_NDA_BV_2_x]
3.3.3.24.3.3.5.3.1	3.5.3.1	NDEF Detection and Read Sequence (x=0 or 1) [TC_T3T_NDA_BV_3_x]
3.3.3.24.3.3.5.3.2	3.5.3.2	NDEF Write Sequence [TC_T3T_NDA_BV_4]

Table B.9.3: NFC Forum Test Cases for Type 3 Tag Operation

Reference test Specification: The test book refers to “NFC Forum Test Cases for Type 4 Tag”.The following table indicates which test cases are included in the current version of the Test Book:

TS.27 Numbering	NFC Forum	Test case description
3.3.3.24.4.3.5.2.1	3.5.2.1	NDEF Read from Version 2.2 Type 4 Tag [TC_T4T_NDA_BV_1]
3.3.3.24.4.3.5.2.2	3.5.2.2	NDEF Read from Version 3.0 Type 4 Tag [TC_T4T_NDA_BV_2]
3.3.3.24.4.3.5.4.1	3.5.4.1	NDEF Detection and Read from Type 4 Tag [TC_T4T_NDA_BV_3]
3.3.3.24.4.3.5.4.2	3.5.4.2	NDEF Write on INITIALISED Type 4 Tag [TC_T4T_NDA_BV_4]
3.3.3.24.4.3.5.4.3	3.5.4.3	NDEF Write on READ-ONLY Type 4 Tag [TC_T4T_NDA_BV_5]

Table B.9.4: NFC Forum Test Cases for Type 4 Tag Operation

B.9.2 Analog Tests

This Annex refers to test cases from any version including “NFC Forum Test Cases for Analog” [46].

B.9.2.1 NFC Forum Test Cases for Analog (all valid versions)

The following table lists the test cases relevant for all referenced versions of NFC Forum Analog test specifications.

TS.27 Numbering	NFC Forum	Test case description
3.3.3.25.9.1.1.1	9.1.1.1	Power Reception Test for NFC-A at Minimum Conditions
3.3.3.25.9.1.1.2	9.1.1.2	Power Reception Test for NFC-A at Nominal Conditions
3.3.3.25.9.1.1.3	9.1.1.3	Power Reception Test for NFC-A at Maximum Conditions
3.3.3.25.9.1.1.4	9.1.1.4	Power Reception Test for NFC-B at Minimum Conditions
3.3.3.25.9.1.1.5	9.1.1.5	Power Reception Test for NFC-B at Nominal Conditions
3.3.3.25.9.1.1.6	9.1.1.6	Power Reception Test for NFC-B at Maximum Conditions
3.3.3.25.9.1.1.7	9.1.1.7	Power Reception Test for NFC-F at Minimum Conditions
3.3.3.25.9.1.1.8	9.1.1.8	Power Reception Test for NFC-F at Nominal Conditions
3.3.3.25.9.1.1.9	9.1.1.9	Power Reception Test for NFC-F at Maximum Conditions
3.3.3.25.9.1.1.11	9.1.1.11	Carrier Frequency Test
3.3.3.25.9.1.2.1	9.1.2.1	Modulation Polling Device to Listening Device at Limit Conditions – NFC-A
3.3.3.25.9.1.2.2	9.1.2.2	Modulation Polling Device to Listening Device at Limit Conditions – NFC-B
3.3.3.25.9.1.2.3	9.1.2.3	Modulation Polling Device to Listening Device at Limit Conditions – NFC-F
3.3.3.25.9.1.3.4	9.1.3.4	Subcarrier Modulation – NFC-A
3.3.3.25.9.1.3.5	9.1.3.5	Subcarrier Modulation – NFC-B
3.3.3.25.9.2.1.2	9.2.1.2	Maximum Power Emission Measurement
3.3.3.25.9.2.1.3	9.2.1.3	Carrier Frequency Measurement

TS.27 Numbering	NFC Forum	Test case description
3.3.3.25.9.2.1.5	9.2.1.5	Threshold Level Test
3.3.3.25.9.2.2.1	9.2.2.1	Modulation Polling Device to Listening Device – NFC-A
3.3.3.25.9.2.2.2	9.2.2.2	Modulation Polling Device to Listening Device – NFC-B
3.3.3.25.9.2.2.3	9.2.2.3	Modulation Polling Device to Listening Device – NFC-F

Table B.9.5: NFC Forum Test Cases for Analog (all valid versions)

B.9.2.2 VOID

B.9.2.3 NFC Forum Test Cases for Analog V2.0 only

The following table lists the test cases specific for NFC Forum Analog V2.0.

TS.27 Numbering	NFC Forum	Test case description
3.3.3.27.9.1.1.10	9.1.1.10	Loading Effect Measurement
3.3.3.27.9.1.1.12	9.1.1.12	Power On and Off Test for NFC-A
3.3.3.27.9.1.1.13	9.1.1.13	Power On and Off Test for NFC-B
3.3.3.27.9.1.1.14	9.1.1.14	Power On and Off Test for NFC-F
3.3.3.27.9.1.3.1	9.1.3.1	Load Modulation Amplitude for NFC-A
3.3.3.27.9.1.3.2	9.1.3.2	Load Modulation Amplitude for NFC-B
3.3.3.27.9.1.3.3	9.1.3.3	Load Modulation Amplitude for NFC-F
3.3.3.27.9.2.1.1	9.2.1.1	Minimum Power Emission Measurement
3.3.3.27.9.2.1.4	9.2.1.4	Reset Characteristics Measurement
3.3.3.27.9.2.3.1	9.2.3.1	Load Modulation Reception Test for NFC-A
3.3.3.27.9.2.3.2	9.2.3.2	Load Modulation Reception Test for NFC-B
3.3.3.27.9.2.3.3	9.2.3.3	Load Modulation Reception Test for NFC-F

Table B.9.6: NFC Forum Test Cases for Analog V2.0 only

B.9.3 Digital Tests

The device manufacturers shall prove the correct implementation of NFC Forum Digital and Activity specification. This proof can be provided by confirming

a) that the DUT uses a CLF with NFC Forum Certification Release [19] and complies to the related rules for integrating a certified platform into the DUT.

Or

b) that the tests according to NFC Forum-TS-Digital Protocol- [19] and NFC Forum-TS-Activity [19] have been successfully performed by the device manufacturer.

Starting 1st January 2019 option a) SHALL be completed.

B.10 ETSI TS 102 221 UICC-Terminal interface

Reference test Specification: ETSI TS 102 230-1 [41]

The following test cases are applicable:

1) Test cases verified by GCF WI 263 are listed in the table below. These test cases are validated by GCF.

Index	TC Title
9.1.1	TERMINAL CAPABILITY – Additional interfaces support

Table B.10.1: List of applicable test cases from GCF WI 263

Annex C Reference Tags - Real NFC Tags

The following is a list of currently available reference NFC tags that can be used in the scope of this document. Other suitable tags can also be used; for example, tags listed in previous version of TS.27 are considered suitable.

Tag type	Formatting	Model	Tag supplier	Memory size (Bytes)	Antenna Size (mm)
2	Dynamic, NDEF formatted	NTAG210	NXP	48	22 * 22
		NTAG213	NXP	144/504/888	23 (round)
		NTAG216	NXP	144/504/888	23 (round)
	Static, NDEF formatted	NTAG210	NXP	48	22 * 22
		NTAG213	NXP	144/504/888	23 (round)
		NTAG216	NXP	144/504/888	23 (round)
3	NDEF formatted	RC-S966	SONY	208	43 * 43
		RC-SA01	SONY	3968	80 * 50
4A		NTAG 413 DNA	NXP	160	45 * 76
		ST25TA02K	STM	256	25 * 25
4B		RF430CL330H associated with RF430CL330HTB board antenna coil	Texas Instruments	3k	40 * 35

Below are the older references which are still allowed to be used until further notice.

- NTAG213 (Type 2 Tag, NXP) NDEF Formatted for 144 bytes of Dynamic User Memory, Antenna Size = 42mm diameter
- NTAG213 (Type 2 Tag, NXP) NDEF Formatted for up to 48 bytes of Static User Memory, Antenna Size = 42mm diameter
- Mifare Ultralight EV1 (Type 2 Tag, NXP), to be NDEF formatted, for 48 Bytes of Static User Memory, Antenna Size = 80x50mm
- FeliCa Lite-S (Type 3 Tag, Sony Corp.) NDEF Formatted for 208 bytes of User Memory, Antenna Size = 43x43mm
- Mifare DESFire (Type 4A Tag, NXP) NDEF Formatted for 2k bytes of User Memory, Antenna Size = 80x50mm

Annex D NFC Device Implementation statement (Informative)

The xls below indicates the device features and all test cases from the present version of the Test Book:

TSGNFC_280_TS27_
Device_feature_state

Annex E Test Case configuration files

E.1 Reference PKCS#15 files

E.1.1 Directory file (EF_DIR)

Files Type	Elementary File	
Data Structure	Record	
File ID	3F00 2F00	
	Access Conditions	
	READ	Always

Data Object						Additional record for EF_DIR
T	***T***	***T***	***T***	***T***	***L***	***V***
61					14	
	4F				0C	A000000063504B43532D3135
	51				04	3F00 7F50

E.1.2 Object Directory File (EF_ODF)

Files Type	Elementary File	
Data Structure	Transparent	
File ID	5031	
	Access Conditions	
	READ	Always

Data Object						
T	***T***	***T***	***T***	***T***	***L***	***V***
A5					06	
	30				04	
		04			02	5205
A7					06	
	30				04	
		04			02	5207

E.1.3 Data Object Directory File (EF_DODF)

Files Type	Elementary File	
Data Structure	Transparent	
File ID	5207	
	Access Conditions	
	READ	Always

Data Object						
T	***T***	***T***	***T***	***T***	***L***	***V***
A1					29	
	30				00	
	30				0F	
		0C			0D	4750205345204163632043746C
	A1				14	
		30			12	
			06		0A	2A864886FC6B81480101
			30		04	
				04	02	4200

E.1.4 Certificate Directory File (EF_CDF)

Files Type	Elementary File	
Data Structure	Transparent	
File ID	5205	
	Access Conditions	
	READ	Always

Data Object						Certificate #1
T	***T***	***T***	***T***	***T***	***L***	***V***
30					1D	
	30				0C	
		0C			0A	47534D41203031204341
	30				03	
		04			01	01
	A1				08	
		30			06	

			30		04	
				04	02	4361
Data Object						Certificate #2
T	***T***	***T***	***T***	***T***	***L***	***V***
30					1D	
	30				0C	
		0C			0A	47534D41203032204341
	30				03	
		04			01	02
	A1				08	
		30			06	
			30		04	
				04	02	4362

E.2 Reference GSMA files for PKCS#15 structure

E.2.1 Certificate Files

The content of the file is not described in this document but it is understood that it should be filled by X.509 certificates [29].

Certificate #01		
Files Type	Elementary File	
Data Structure	Transparent	
File ID	4361	
	Access Conditions	
	READ	Always

Certificate #02		
Files Type	Elementary File	
Data Structure	Transparent	
File ID	4362	
	Access Conditions	
	READ	Always

E.2.2 Access Control Files

EF ACMain		
Files Type	Elementary File	
Data Structure	Transparent	
File ID	4200	
	Access Conditions	
	READ	Always

Data Object						
T	**T**	**T**	**T**	**T**	**L**	**V**
30					10	
	04				08	0102030405060708
30					04	
	04				02	4300

EF ACRules		
Files Type	Elementary File	
Data Structure	Transparent	
File ID	4300	
	Access Conditions	
	READ	Always

E.3 AIDs referenced by PKCS#15 files

The reference PKCS#15 structures are using the following AID-s:
AID01 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31’

AID02 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32’

AID03 = ‘A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 33’

E.4 Specific configuration files for test case 5.3.1.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

E.5 Specific configuration files for test case 5.3.1.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 00

E.6 Specific configuration files for test case 5.3.1.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.7 Specific configuration files for test case 5.3.1.4

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

E.8 Specific configuration files for test case 5.3.1.5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 11

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

ACConditions2:

30 16 04 14 [Hash of Certificate #02 (20 bytes)]

E.9 Specific configuration files for test case 5.3.1.6

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]
30 16 04 14 [Hash of Certificate #02 (20 bytes)]

E.10 Specific configuration files for test case 5.3.1.7

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)

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 [Hash of Certificate #01 (20 bytes)]

ACConditions2:
30 16 04 14 [Hash of Certificate #02 (20 bytes)]

E.11 Specific configuration files for test case 5.3.1.8

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32 30 04 04 02 43 10

ACConditions1:
30 16 04 14 [Hash of Certificate #01 (20 bytes)]

E.12 Specific configuration files for test case 5.3.1.9

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)

ACRules:
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 11

ACConditions1:
30 16 04 14 [Hash of Certificate #01 (20 bytes)]

ACConditions2:
30 00

E.13 Specific configuration files for test case 5.3.2.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:
30 00

E.14 Specific configuration files for test case 5.3.2.1 Step5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:
30 08 04 06 [6 first bytes of the hash of Certificate #01] <!-- corrupted Hash -->

E.15 Specific configuration files for test case 5.3.2.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 00

E.16 Specific configuration files for test case 5.3.3.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

E.17 Specific configuration files for test case 5.3.3.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

PKCS#15 file system

|- EF ACRules (4300) --> shall reference EF ACConditions files

|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #02 (20 bytes)]

E.18 Specific configuration files for test case 5.3.4.1

PKCS#15 file system

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 16 04 14 [Hash of Certificate #01 (20 bytes)]

E.19 Specific configuration files for test case 5.3.5.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32 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 [Hash of Certificate #01 (20 bytes)]

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 [Hash of Certificate #02 (20 bytes)]

E.20 Specific configuration files for test case 5.3.5.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)
|- EF ACConditions2 (4311)
|- EF ACConditions3 (4312)

ACRules:
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10
30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32 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 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 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 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 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 [Hash of Certificate #01 (20 bytes)]

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 [Hash of Certificate #02 (20 bytes)]

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.21 Specific configuration files for test case 5.3.6.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

ACRules:

<!-- ACRF is absent -->

ACConditions:

<!-- ACCF is absent -->

E.22 Specific configuration files for test case 5.3.6.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files

ACRules:

<!-- ACRF is present but empty -->

ACConditions:

<!-- ACCF is absent -->

E.23 Specific configuration files for test case 5.3.6.4

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 30 04 04 02 43 10

ACConditions1:

30 12 04 10 [16 first bytes of the hash of Certificate #01]

E.24 Specific configuration files for test case 5.3.6.5

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 1A A0 12 04 10 A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31 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.25 Specific configuration files for test case 8.3.4.1

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:

30 08 82 00 30 04 04 02 43 10

ACConditions1:

30 00

E.26 Specific configuration files for test case 8.3.4.2

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:
30 08 82 00 30 04 04 02 43 10

ACConditions1:
30 00

E.27 Specific configuration files for test case 8.3.4.3

PKCS#15 application (AID: A0 00 00 00 63 50 4B 43 53 2D 31 35)

|- EF ACRules (4300) --> shall reference EF ACConditions files
|- EF ACConditions1 (4310)

ACRules:
30 08 82 00 30 04 04 02 43 10

ACConditions1:
30 16 04 14 [Hash of Certificate #02 (20 bytes)]

Annex F Configuration for Device with eSE

In order to run the TS.27 test cases a device with eSE shall be configured as described below. This is the responsibility of the device vendor to set this configuration for the devices under test.

- No nonAID based applications are installed on the eSE
- The eSE shall be configured with an ISD personalized with CIN and IIN
- The following applets shall be installed on the eSE under the ISD:
  - Two instances of Applet1 - defined in 2.5.3.5 eSE Applications – with the following instance AIDs:
    - AID01: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 31
    - AID03: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 33

    For the installation parameters see – Annex F.1

  - One instance of Applet2 – defined in 2.5.3.5 eSE Applications – with the following instance AID:
    - AID02: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 32
  - Three instances of Applet3 – defined in 2.5.3.5 eSE Applications – with the following instance AIDs:
    - AID07: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 37
    - AID08: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 38
    - AID09: A0 00 00 05 59 50 00 00 00 00 00 00 52 41 44 39

    For the installation parameters see – Annex F.1

  - GlobalPlatform Test Applets – available at website [43]
    For the installation parameters see – Annex F.2

- The eSE shall be configured with an ARA-M applet complying GP SEAC specification [7].
  - The AID of this application is defined in Section 2.1 of GP SEAC specification [7]
  - The ARA-M applet shall contain the following access rules:
    - Access rules as defined in GlobalPlatform OMAPI Test Specification [5] – AnnexB - Access Control Applet (ARA)
    - Access rule to allow APDU access to AID01, AID02, AID03, AID07, AID08, AID09 from any mobile application (implicitely part of “allow all” rule)

  A sample ARA applet containing the access rules listed above is available at [44].
  For the installation parameters see – Annex F.3

F.1 Installation parameters for the GSMA applets

Applet	Cap file (as available on GSMA GitHUB)	Applet AID (Instance AID)	EM AID (Class AID)	ELF AID (Load AID)	Load params	Installation Params	Perso data	Privileges
Applet1 - AID01	com.gsma.test.nfc.Applet1_A0xxx31.cap	A000000559500000000000052414431	same as Applet AID	A000000559500000000000052414401	NA	C9 00 See note	NA	000000'
Applet1 - AID03	com.gsma.test.nfc.Applet1_A0xxx33.cap	A000000559500000000000052414433	same as Applet AID	A000000559500000000000052414403	NA	C9 00 See note	NA	000000'
Applet2 - AID02	com.gsma.test.nfc.Applet2_A00xxx32.cap	A000000559500000000000052414432	same as Applet AID	A000000559500000000000052414402	NA	C9 00 See note	NA	000000'
Applet3 - AID07	com.gsma.test.nfc.Applet3_A00xxx37.cap	A000000559500000000000052414437	same as Applet AID	A000000559500000000000052414407	NA	C9 00 See note	NA	000000'
Applet3 - AID08	com.gsma.test.nfc.Applet3_A0xxx38.cap	A000000559500000000000052414438	same as Applet AID	A000000559500000000000052414408	NA	C9 00 See note	NA	000000'
Applet3 - AID09	com.gsma.test.nfc.Applet3_A0xxx39.cap	A000000559500000000000052414439	same as Applet AID	A000000559500000000000052414409	NA	C9 00 See note	NA	000000'

Note: The Installation Parameters shall be chosen so that the instance shall be explicitly selectable on the contactless interface based on AID. It may require to use Contactless Protocol Parameters also in the Installation Parameters e.g.: EF0EA00C80028182810101A5038201C0.

F.2 Installation parameters for the GlobalPlatform applets

Applet	Cap file	Applet AID (Instance AID)	EM AID (Class AID)	ELF AID (Load AID)	Load params	Installation Params	Perso data	Privileges
TestApp	omapitest1.cap	A000000600010001EE0501	A000000600010001EE0501	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6999	omapitest1.cap	A000000600010001EE0502	A000000600010001EE0502	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6280	omapitest1.cap	A000000600010001EE0503	A000000600010001EE0503	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6283	omapitest1.cap	A000000600010001EE0504	A000000600010001EE0504	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6310	omapitest1.cap	A000000600010001EE0505	A000000600010001EE0505	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW63C1	omapitest1.cap	A000000600010001EE0506	A000000600010001EE0506	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_selectresponse	omapitest1.cap	A000000600010001EE0507	A000000600010001EE0507	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6280_selectresponse	omapitest1.cap	A000000600010001EE0508	A000000600010001EE0508	A000000600010001FF05	NA	C9 00	NA	000000'

TestApp_SW6283_selectresponse	omapitest1.cap	A000000600010001EE0509	A000000600010001EE0509	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6310_selectresponse	omapitest1.cap	A000000600010001EE050A	A000000600010001EE050A	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW63C1_selectresponse	omapitest1.cap	A000000600010001EE050B	A000000600010001EE050B	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_p1p2	omapitest1.cap	A000000600010001EE050C	A000000600010001EE050C	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_clains	omapitest1.cap	A000000600010001EE050D	A000000600010001EE050D	A000000600010001FF05	NA	C9 00	NA	000000'
Partial_1_instance_1	omapitest1.cap	A000000600010001EE050E01	A000000600010001EE050E	A000000600010001FF05	NA	C9 00	NA	000000'
Partial_1_instance_2	omapitest1.cap	A000000600010001EE050E02	A000000600010001EE050E	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6280_partial_instance1	omapitest1.cap	A000000600010001EE050F01	A000000600010001EE050F	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6280_partial_instance2	omapitest1.cap	A000000600010001EE050F02	A000000600010001EE050F	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6283_partial_instance1	omapitest1.cap	A000000600010001EE051001	A000000600010001EE0510	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW6283_partial_instance2	omapitest1.cap	A000000600010001EE051002	A000000600010001EE0510	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_SW61XX	omapitest1.cap	A000000600010001EE0511	A000000600010001EE0511	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_Multi_SW61xx	omapitest1.cap	A000000600010001EE0512	A000000600010001EE0512	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_Get_Response	omapitest1.cap	A000000600010001EE0513	A000000600010001EE0513	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_Case4_Swwarning	omapitest1.cap	A000000600010001EE0514	A000000600010001EE0514	A000000600010001FF05	NA	C9 00	NA	000000'
TestApp_multiselectable	omapitest2.cap	A000000600010001EE5501	A000000600010001EE5501	A000000600010001FF55	NA	C9 00	NA	000000'
TestApp_Case4_SWwarning_nodata	omapitest3.cap	A000000600010001EE5601	A000000600010001EE5601	A000000600010001FF56	NA	C9 00	NA	000000'
AID_TestApp_p1p2_etsi	omapitest3.cap	A000000600010001EE5602	A000000600010001EE5602	A000000600010001FF56	NA	C9 00	NA	000000'
TestApp_SW6280_selectresponse_etsi	omapitest3.cap	A000000600010001EE5603	A000000600010001EE5603	A000000600010001FF56	NA	C9 00	NA	000000'
TestApp_SW6283_selectresponse_etsi	omapitest3.cap	A000000600010001EE5604	A000000600010001EE5604	A000000600010001FF56	NA	C9 00	NA	000000'
TestApp_SW6310_selectresponse_etsi	omapitest3.cap	A000000600010001EE5605	A000000600010001EE5605	A000000600010001FF56	NA	C9 00	NA	000000'
TestApp_SW63C1_selectresponse_etsi	omapitest3.cap	A000000600010001EE5606	A000000600010001EE5606	A000000600010001FF56	NA	C9 00	NA	000000'
Length_6	omapitest4.cap	A00000060002	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_7	omapitest4.cap	A0000006000200	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_8	omapitest4.cap	A000000600020001	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_9	omapitest4.cap	A000000600020001EE	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_10	omapitest4.cap	A000000600020001EE05	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_11	omapitest4.cap	A000000600020001EE0515	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_12	omapitest4.cap	A000000600020001EE051501	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_13	omapitest4.cap	A000000600020001EE05150101	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_14	omapitest4.cap	A000000600020001EE0515010101	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_15	omapitest4.cap	A000000600020001EE051501010101	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'
Length_16	omapitest4.cap	A000000600020001EE05150101010101	A000000600020001EE05150101010101	A000000600020001FF05	NA	C9 00	NA	000000'

F.3 Installation parameters for the GP ARA applet

Applet	Cap file	Applet AID (Instance AID)	EM AID (Class AID)	ELF AID (Load AID)	Load params	Installation Params	Perso data	Privileges
ARA-M	ara.cap	A00000015141434C00	A00000015141434C00	A00000015141434C	NA	C9 00	NA	00'

Annex G Document History

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
1.2	10/09/13	First published version	GSM Associatio n	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 GlobalPlatform 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.
5.0	12/01/15	The changes to the TS.27 NFC Handset Test Book V5.0 include the following: <u>Change to existing test cases:</u> • Test case improvements throughout sections 3, 7, 8, 12, 13, 15. • Renaming of AID RID from ”undefined” to GSMA ID. • Inclusion of details for handling of application certificates. • Applicability table updated. Adding iOS and “Other OS” - contents for FFS. • Removal of redundant tables in Annexes. • Improved description of tables in section 2. • Various editorial improvements throughout the document. <u>New and removed test cases:</u>	TSG	Kay Fritz

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
		• 36 new tests added to sections 8, 12, 13 and 15. • 103 new tests from GlobalPlatform SEAC Test Plan referenced in section 5. • 2 tests removed from section 13 (due to redundancy).		
6.0	08/06/15	<u>Change to existing test cases:</u> • Test case improvements in sections 3, 7, 8, 12, 13, 15. • Baseline requirement reference to TS.26 V7.0. • Java Source code in Annex A.5 moved to GitHub. • 11 tests removed (VOID'ed) from Sect 3 (FFS), 7 (FFS) and 12. None of these tests were allocated in GCF WI. (4 FFS tests left). • Applicability table expanded with reference to TS.26 version. (Proposed by GCF LS) • Annex D with complete list of test cases moved to separate excel (Proposed by GCF LS). This excel also provide a copy (Non-Normative) of table with Optional Features (sect 2.1.4) and list of external test cases (Annex B1, B.8, B.9). • New Optional Features added, sect 2.1.4. • Clarification of mandatory features in TS.27 vs. optional features in external standards (sect 2.2.1). • Various editorial improvements throughout the document. <u>New test cases:</u> • 20 new tests added to section 12, 13 and 15. • 3 tests updated from FFS to complete status in section 12. • 26 new tests from NFC Forum Tag Operation tests referenced in section 3.	TSG	Kay Fritz
6.1	20/07/15	Due to a duplicate test numbering in Annex B.9, one extra digit is added after the main number 3.3.3.24. This change applies only for the test cases listed in Annex B.9.	TSG	Kay Fritz
7.0	21/12/15	<u>Specification alignment:</u> TS.26 V8.0 alignment incl. TC updates to align GSMA API Existing TCs updated with conditional branching for GSMA API version New section 2.6 with procedures for default HCE/UICC routing and routing table handling New DUT options for section 12 and 15 <u>New tests:</u> 4 Secure Element Access API tests (OPMAPI) 18 Android specific section 15 33 Analogue NFC Forum test cases <u>Removed tests:</u>	TSG	Kay Fritz

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
		1 GP SEAC test 5 tests for NFC Forum Tag Type 1 with static memory General improvements: New table for information to be provided by vendor, sect 2.1.6 Updated applicability and new conditions for tests in section 7, 12, and 15 Various TC improvements to section 3, 7, 8, 12,13, 15 Updated reference Tags.		
8.0	22/02/16	Note 13 changes: • Section 2.1.4: Optional feature “Support of NFC Forum Analog Test” and related Note 13 are removed. • Section 2.1.5: Test case applicability for 3.3.3.25 changed to Mandatory and related condition removed. Other technical corrections: • Section 2.5.3.2: Correction/clarifications in definition of Device Applications. • Section 2.6.2: Corrections to Applications needed and test steps. • Section 2.6.3: Comment added to “Procedure to send a transaction event”. Editorial corrections: • Removal and replacement of references to ISO 14443. • Other editorial corrections	TSG	Kay Fritz

9.0	24/06/16	The changes to the TS.27 V9.0 include the following:
		• Compliant to and referencing latest TS.26 V9.0.
		• 10 new test cases to reduce testing gaps in the following sections:
		  • 7. Multiple Card Emulation Environment (2 new tests)
		  • 12. Remote Management of NFC Services (2 new tests)
		  • 13. General Device Support (3 new tests)
		  • 15. Android specific test cases (3 new tests)
		• 2 Voided tests from section
		  • 7. Multiple Card Emulation Environments. See list of test cases below.
		• The versions of referenced ETSI and 3GPP specifications are updated to reference a newer or the latest versions.
		• Reference to EMVCo updated Mobile Level 1 Analog, Digital, Interoperability and performance testing requirements.
		• New reference Type 2 Tag with static memory added. TS.27 Annex C.
			TSG	Kay Fritz

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
		• Naming suffix added for test sequences which previously did not have specific sequence names. See details below. • Various corrections and improvements to existing test cases • Various editorial improvements. A version of TS.27 with track changes can be requested at GSMA.		

10.0	23/12/16	The changes include the following:
		• Compliant with TS.26 V10.0.
		• New test cases introduced in the following sections:
		  • Section 3. RF Protocol compliance - referencing NFC Forum Analog V2.0 tests.
		  • Section 6. Secure Element Access API - referencing new Open Mobile API tests.
		  • Section 7. Multiple Card Emulation Environments. New test to cover the new requirements TS26_NFC_REQ_167 for size of routing table.
		  • Section 13. General Device Support. New test to address issue from field with re-selecting applet.
		  • Section 15: Android specific test cases. New test to cover that FELICA is mandatory. New test to return the version of OMAPI version implemented.
		• Tests removed (Voided) in the section
		  • Section 12.4: Redundant test cases removed.
		• New device options (Item 28+29) in table 2.1.4.
		• Applicability table 2.1.5 updated.
		• Updated versions of ETSI, 3GPP, OMAPI, NFC Forum specifications.
		• Procedure to identify the size of the AID routing table updated in section 2.6.2.
		• New general procedure to check if UICC is accessible, section 2.6.4.
		• Various corrections and improvements to existing test cases in the following sections:
		  • Section 7, 8, 12, 13, 15.
		• Table B4.1 and Table B5.1 are updated to reflect ISO/IEC 18092 Type F is mandatory and to include changes of the tables from ETSI TS 102 694-1 and TS 102 695-1 respectively. Please note still some of the options are Mandatory in TS.27 while Optional in the ETSI specifications.
			TSG	Kay Fritz

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
		• Various editorial improvements incl. re-numbering of tables/figures. A version of TS.27 with track changes can be requested at GSMA.		
11.0	12/06/17	This release included the following changes: • Section 2.1.4: A new DUT option (31) for Android Nougat introduced. • Section 2.6.2: The procedures for Initial Default Routing to UICC and HCE updated. • Various test cases corrections in sections 8.3, 13.3, 15.5.3, 15.7.3, and 15.9.3. • Test 13.3.3.1, 13.3.4.1, 13.3.5.1: Clarification that these tests are also referenced in 6.3.1.6.5.6 and can be removed from work items. • New Annex B.10 referencing ETSI TS 102 221: 9.1 TERMINAL CAPABILITY – from GCF WI-263. A version of TS.27 with track changes between V10 and V11 can be requested from GSMA. See section associated Liaison Statement for detailed list of TS.27 changes.	TSG#28	Kay Fritz
12.0	04/12/17	This release includes the following: • 7 new tests for card emulation with Display Off/Locked and Device Switched Off. • 6 new tests for extended APDU length. • 5 new tests for Non-AID based services. • 16 new tests for eSE based NFC services. • NFC Forum Digital testing compliance to be proved by referencing a NFC Forum compliant CLF chipset or corresponding testing. • Test cases using and testing GSMA API have been made optional. 18 new tests using Android native API are introduced to replace tests which were using GSMA API. • Various improvements to existing tests. A version of TS.27 with track changes between V11 and V12 can be requested from GSMA. See also associated Liaison Statement for detailed list of TS.27 new test cases.	TSG#30	Kay Fritz

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
13.0	04/06/18	This release includes the following: • **Android 9** impacted test cases are updated and new Device Option introduced to manage test applicability for devices implementing before Android 9 and from Android 9 and onwards. This is implemented in the Device Options section 2.1.4 and the Applicability Table section 2.1.5. A number of existing test cases are Not Applicable for devices implementing Android 9 and onwards. Updated Device Application compatible with Android 9 is described in section 2.5.3.1.1. For more details on impact, see also separate LS: TSGNFC_211_LS_AndroidP_impact_V7. • **Single CEE** tests voided (2 tests) due to removal of Single CEE requirements. • **eSE applet description:** Improved description and references to the applets used for eSE test cases, including new Annex F with eSE applet installation parameters. • **NFC Forum Analog V1.0** tests have been removed. Section 3.3.3.26 and Annex B.9.2.2 Voided. • **Existing TS.26 requirements** referenced as tested within existing test cases: The following 7 requirements referenced as tested within existing tests cases: TS26_NFC_REQ_084, TS26_NFC_REQ_122, TS26_NFC_REQ_122.2, TS26_NFC_REQ_152, TS26_NFC_REQ_152.2, TS26_NFC_REQ_162, TS26_NFC_REQ_173, TS26_NFC_REQ_173.1. • **Document Cross Reference** updated with reference to newer specifications. • **Various improvements** to existing tests.	TSG	Kay Fritz/ Vodafone
13.0	13/08/18	A number of "*Error! Reference source not found*" identified in the PDF version and corrected in the Word version. A new PDF version has been created.	TSG	Claus Madsen/ COMPRION

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
14.0	04/12/18	This version is implementing following changes: ▪ **NFC Forum Type 1 Tag** tests are not applicable for TS.26 V14 and onwards. ▪ **Battery Power-Off Mode** tests are not applicable for TS.27 V14 and onwards. ▪ **AID Routing Table Overflow** tests are made optional if the Device supports more than 40 AIDs in the Routing Table. ▪ **GSMA API tests:** Following the deprecation of the GSMA API requirements in TS.26, the related test cases are not applicable from TS.26 V14 and onwards. ▪ **Optional Features:** Some Device Optional Features have been removed. ▪ **OMAPI applicability:** The specific Android Pie applicability for OMAPI tests is removed as well some other applicability changes are implemented. See Annex B.1. ▪ **New OMAPI test:** 1 new OMAPI test added. ▪ **Redundant TS.26 requirement information** is removed from TS.27. ▪ **Document Cross Reference** updated with reference to newer specifications. – Including reference to NFC Forum Certification Release which identifies the version of specific NFC Forum test specifications. – References to SIMalliance specifications in relation to OMAPI are replaced with GlobalPlatform. ▪ **Device OS specific applicability** for Windows and BlackBerry is removed. ▪ Various improvements to existing tests. A version of TS.27 with track changes between V13 and V14 can be requested from GSMA.	TSG	Anders Olsson/Sony

Version	Date	Brief Description of Change	Approval Authority	Editor / Company
14.1	18/01/19	The following corrections have been introduced: 1. Section 2.1.5: Applicability for the test 8.3.4.3 changed to be applicable only before Android 9. Changed from M to C029. 2. Annex B.1: In TS.27 V14 some 5 new sub tests were introduced to manage the applicability. These sub tests are removed and are covered by existing test cases. The following sub-tests were deleted: • 6.3.1.6.4.7c, • 6.3.1.6.4.7eSEb, • 6.3.1.6.4.8b, • 6.3.1.6.4.10c, • 6.3.1.6.4.11b 3. Annex B.1: The test ID 4a in test 6.3.1.6.4.7eSE is removed. This ID is FFS in GlobalPlatform OMAPI Test Specification and therefore incorrectly included. 4. Annex F.2: The "EM AID (Class AID)" have been corrected for the following Applets: • TestApp_SW6283_partial_instance1 • TestApp_SW6283_partial_instance2 • TestApp_SW61XX	TSG	Claus Madsen / COMPRION

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.