ETSITS 101 955 V7.4.2 (2002-09)

Technical Specification

Digital cellular telecommunications system (Phase 2+); Test specification for SIM API for Java cardTM (3GPP TS 11.13 version 7.4.2 Release 1998)



Reference RTS/TSGT-031113v742 Keywords GSM

ETSI

650 Route des Lucioles F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - NAF 742 C Association à but non lucratif enregistrée à la Sous-Préfecture de Grasse (06) N° 7803/88

Important notice

Individual copies of the present document can be downloaded from: <u>http://www.etsi.org</u>

The present document may be made available in more than one electronic version or in print. In any case of existing or perceived difference in contents between such versions, the reference version is the Portable Document Format (PDF). In case of dispute, the reference shall be the printing on ETSI printers of the PDF version kept on a specific network drive within ETSI Secretariat.

Users of the present document should be aware that the document may be subject to revision or change of status.

Information on the current status of this and other ETSI documents is available at

http://portal.etsi.org/tb/status/status.asp

If you find errors in the present document, send your comment to: editor@etsi.fr

Copyright Notification

No part may be reproduced except as authorized by written permission. The copyright and the foregoing restriction extend to reproduction in all media.

© European Telecommunications Standards Institute 2002. All rights reserved.

DECTTM, **PLUGTESTS**TM and **UMTS**TM are Trade Marks of ETSI registered for the benefit of its Members. **TIPHON**TM and the **TIPHON logo** are Trade Marks currently being registered by ETSI for the benefit of its Members. **3GPP**TM is a Trade Mark of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners.

Intellectual Property Rights

IPRs essential or potentially essential to the present document may have been declared to ETSI. The information pertaining to these essential IPRs, if any, is publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: "Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards", which is available from the ETSI Secretariat. Latest updates are available on the ETSI Web server (http://www.etsi.org/legal/home.htm).

Pursuant to the ETSI IPR Policy, no investigation, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Foreword

This Technical Specification (TS) has been produced by the ETSI 3rd Generation Partnership Project (3GPP).

The present document may refer to technical specifications or reports using their 3GPP identities, UMTS identities or GSM identities. These should be interpreted as being references to the corresponding ETSI deliverables.

The cross reference between GSM, UMTS, 3GPP and ETSI identities can be found under www.etsi.org/key.

Contents

Intelle	ectual Property Rights	2
Forew	vord	2
Forew	vord	g
1	Scope	10
2	References	10
3	Definitions and abbreviations.	11
3.1	Definitions	
3.2	Abbreviations	
4	Test Environment	12
4.1	Applicability	
4.2	Test environment description	
4.3	Tests format	
4.3.1	Test Area Reference	
4.3.1.1	1 · · · · · · · · · · · · · · · · · · ·	
4.3.1.2		
4.3.1.3		
4.3.1.4		
4.4	Initial Conditions	
4.5	Package name	
4.6	AID Coding	
4.6.1 4.6.2	Specific Test Applet Name for Francycell	
4.0.2	Specific Test Applet Name for Framework	
4.7.1	APDU tool	
4.7.2	Util package	
4.7.3	Applet installation parameters	
4.7.3.1		
4.7.3.2	• •	
4.8	Testing methodology	
4.8.1	Test interfaces and facilities	
5	Test plan	18
6	API Test Plan	18
6.1	Package sim.access:	
6.1.1	Interface SIMView	
6.1.1.1	Constants	18
6.1.1.2	Method select(short fid, byte[] fci, short fciOffset, short fciLength)	19
6.1.1.3		23
6.1.1.4		
6.1.1.5	, and the state of	
6.1.1.6		
6.1.1.7		
6.1.1.8	*	
6.1.1.9 6.1.1.1		
6.1.1.1		
6.1.1.1		
6.1.2	Class SIMSystem	
6.1.2.1	·	
6.1.3	Class SIMViewException	
6.1.3.1		
6.1.3.2		
6.2	Package sim.toolkit	60

6.2.1	Interface ToolkitConstants	60
6.2.1.1	Constants	60
6.2.2	Interface ToolkitInterface	60
6.2.2.1	Method processToolkit	60
6.2.3	Class EditHandler	61
6.2.4	Class EnvelopeHandler	
6.2.4.1	Method getEnvelopeTag	
6.2.4.2	Method getItemIdentifier	
6.2.4.3	Method getSecuredDataLength	
6.2.4.4	Method getSecuredDataOffset	
6.2.4.5	Method getTheHandler	
6.2.4.6	Method getTPUDLOffset	
6.2.4.7	Method getLength	
6.2.4.8	Method copy	
6.2.4.9	Method findTLV	
6.2.4.10	Method getValueLength	
6.2.4.11	Method getValueByte	
6.2.4.12	Method copyValue	
6.2.4.13	Method compareValue	
6.2.4.14	Method findAndCopyValue(byte tag, byte[] dstBuffer, short dstOffset)	
6.2.4.15	Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short	61
0.2.4.13	dstOffset, short dstLength)dstOffset, short dstLength	
6.2.4.16	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	
6.2.4.17	Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[]	63
0.2.4.17		00
() 5	compareBuffer, short compareOffset, short compareLength)	
6.2.5	Class EnvelopeResponseHandler	
6.2.5.1	Method getTheHandler	
6.2.5.2	Method post	
6.2.5.3	Method postAsBERTLV	
6.2.5.4	Method getLength	
6.2.5.5	Method copy	
6.2.5.6	Method findTLV	
6.2.5.7	Method getValueLength	
6.2.5.8	Method getValueByte	
6.2.5.9	Method copyValue	
6.2.5.10	Method compare Value	
6.2.5.11	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	
6.2.5.12	Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short	
	dstOffset, short dstLength)	
6.2.5.13	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	112
6.2.5.14	Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[]	
	compareBuffer, short compareOffset, short compareLength)	
6.2.5.15	Method appendArray	
6.2.5.16	Method appendTLV(byte tag, byte value)	
6.2.5.17	Method appendTLV(byte tag, byte value1, byte value2)	
6.2.5.18	Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)	
6.2.5.19	Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length	
6.2.5.20	Method clear	
6.2.6	Class MEProfile	
6.2.6.1	Method check (byte index)	
6.2.6.2	Method check (byte [] mask, short offset, short length)	129
6.2.7	Class ProactiveHandler	131
6.2.7.1	Method getTheHandler	131
6.2.7.2	Method init	132
6.2.7.3	Method initDisplayText	134
6.2.7.4	Method initGetInkey	137
6.2.7.5	Method initGetInput	140
6.2.7.6	Method send	
6.2.7.7	Method getLength	145
6.2.7.8	Method copy	
6.2.7.9	Method findTLV	
6.2.7.10	Method getValueLength	

6.2.7.11	Method getValueByte	.151
6.2.7.12	Method copyValue	
6.2.7.13	Method compare Value	
6.2.7.14	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	
6.2.7.15	Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short	
	dstOffset, short dstLength)	160
6.2.7.16	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	
6.2.7.17	Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[]	. 1 0 0
0.2.7.17	compareBuffer, short compareOffset, short compareLength)	165
6.2.7.18	Method appendArray	
6.2.7.19	Method appendTLV(byte tag, byte value)	
6.2.7.19	Method appendTLV(byte tag, byte value)	
6.2.7.21	Method appendTLV(byte tag, byte Value, short valueoffset, short valuelength)	
6.2.7.21	Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length).	
6.2.7.23	Method clear	
6.2.8	Class ProactiveResponseHandler	
6.2.8.1	Method copyAdditionalInformation	
6.2.8.2	Method copyTextString	
6.2.8.3	Method getAdditionalInformationLength	
6.2.8.4	Method getGeneralResult	
6.2.8.5	Method getItemIdentifier	
6.2.8.6	Method getTextStringCodingScheme	
6.2.8.7	Method GetTextStringLength	
6.2.8.8	Method getTheHandler	
6.2.8.9	Method getLength	
6.2.8.10	Method copy	
6.2.8.11	Method findTLV	
6.2.8.12	Method getValueLength	
6.2.8.13	Method getValueByte	
6.2.8.14	Method copyValue	.206
6.2.8.15	Method compareValue	
6.2.8.16	Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)	.210
6.2.8.17	Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short	
	dstOffset, short dstLength)	.213
6.2.8.18	Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)	.216
6.2.8.19	Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[]	
	compareBuffer, short compareOffset, short compareLength)	.218
6.2.9	Class ToolkitRegistry	
6.2.9.1	Method allocateTimer	
6.2.9.2	Method changeMenuEntry	
6.2.9.3	Method clearEvent	
6.2.9.4	Method disableMenuEntry	
6.2.9.5	Method enableMenuEntry	
6.2.9.6	Method getEntry	
6.2.9.7	Method getPollInterval	
6.2.9.8	Method initMenuEntry	
6.2.9.9	Method isEventSet	
6.2.9.10	Method releaseTimer	
6.2.9.11	Method requestPollInterval	
6.2.9.11	Method setEvent	
6.2.9.13	Method setEventList	
6.2.10	Class ViewHandler	
6.2.11	Class ToolkitException	
6.2.11.1	Exception Constants	
6.2.11.2	Constructor ToolkitException	
6.2.11.3	Method throwIt	
6.3	SIM Toolkit Framework	
6.3.1	Minimum Handler Availability	
6.3.1.1	ProactiveHandler	
6.3.1.2	ProactiveResponseHandler	
6.3.1.3	EnvelopeHandler	
6311	EnveloneResponseHandler	280

Note: Di	data in multitriggeringdata	
6.3.2	Handler Integrity	
6.3.2.1	ProactiveHandler Proact	
6.3.2.2	ProactiveResponseHandler	
6.3.2.3	EnvelopeHandler	
6.3.3	Applet Triggering	
6.3.3.1	EVENT_PROFILE_DOWNLOAD	
6.3.3.2	EVENT_MENU_SELECTION	
6.3.3.3	EVENT_MENU_SELECTION_HELP_REQUEST	
6.3.3.4	EVENT_FORMATTED_SMS_PP_ENV	
6.3.3.5	EVENT_UNFORMATTED_SMS_PP_ENV	
6.3.3.6	EVENT_CALL_CONTROL_BY_SIM	
6.3.3.7	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	310
6.3.3.8	EVENT_TIMER_EXPIRATION	312
6.3.3.9	EVENT_UNFORMATTED_SMS_CB	313
6.3.3.10		
6.3.3.11	EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	
6.3.3.12		
6.3.3.13		
6.3.3.14		
6.3.3.15		
6.3.3.16		
6.3.3.17	EVENT_UNRECOGNIZED_ENVELOPE	
6.3.3.18		
6.3.4	Proactive Command Sending by the STF	
6.3.4.1	System Proactive Commands	
6.3.4.2	Interaction with GSM commands	
6.3.5	Exception Handling	
6.3.5.1	Hide Exceptions from the ME	
6.3.5.2 6.3.6	Interaction with Multiple Triggering	
6.3.6.1	Input Data	
6.3.6.2	Output Data	
	ea Reference: FWK FWS OUDA	
6.3.7	Envelope Response Posting	
6.3.7.1	EVENT_CALL_CONTROL_BY_SIM	
6.3.7.2	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	
6.3.7.3	EVENT_UNRECOGNIZED_ENVELOPE	
6.3.8	Toolkit Installation	
6.3.8.1	Timers Allocation	338
6.3.8.2	Item Identifier	
6.3.8.3	Item Position	343
6.3.8.4	Maximum Text Length for a menu entry	344
6.3.8.5	Maximum number of menu entries	346
6.3.8.6	Access Domain	347
6.3.8.7	Priority Level	351
6.3.9	File System Context	
6.3.9.1	Initial Context	
6.3.9.2	Context Preservation (current file)	
6.3.9.3	Context Preservation (current record pointer)	
6.3.10	Other parts transferred to framework from API	
6.3.10.1	A handler is a temporary JCRE Entry Point object	
6.3.10.2		
6.3.10.3	11	
	A (normative): Class and Methods AID numbering and acronyms	
A.1 S	Sim.access	363
A.1.1	SIMView methods	
A.1.2	SIMSystem methods	
Λ 1 2	SIMVious Execution methods	363

A.2	Sim.toolkit	
A.2.1	ToolkitConstants	
A.2.2 A.2.3	ToolkitInterface methods	
A.2.3 A.2.4	EnvelopeHandler methods	
A.2.5	EnvelopeResponseHandler methods	
A.2.6	MEProfile methods	
A.2.7	ProactiveHandler methods	
A.2.8	ProactiveResponseHandler methods	
A.2.9 A.2.10	ToolkitRegistry methods	
A.2.10		
	•	
	ex B (normative): Script file syntax and format description	
B.1	Syntax description	
B.2	Semantics	
B.3	Example	370
B.4	Style and formatting	371
Anne	ex C (normative): Default Prepersonalisation	372
C.1	General Default Prepersonalisation	372
C.2	Sim.Access.SimView test default prepersonalisation	
C.2.1	DF _{SIMTEST} (SIM Test)	
C.2.2	EF _{TNR} (Transparent Never Read)	
C.2.3	EF _{TNU} (Transparent Never Update)	
C.2.4	EF _{TARU} (Transparent Always Read and Update)	
C.2.5 C.2.6	EF _{CNR} (Cyclic Never Read) EF _{CNU} (Cyclic Never Update)	
C.2.7	EF _{CNIC} (Cyclic Never Increase)	
C.2.8	EF _{CNIV} (Cyclic Never Invalidate)	
C.2.9	EF _{CNRH} (Cyclic Never Rehabilitate)	
C.2.10	Critic C C	
C.2.11		
C.2.12	LIVE \ - I	
C.2.13 C.2.14	zime (377
C.2.15		
	11010	
C.2.16		
C.2.17	Chie	
C.2.18	CIMI ·	
C.2.19	EF _{CNRI} (Cyclic Never Rehabilitate Invalidated)	378
Anne	ex D (normative): sim.test.util package and loading , testing and cleaning script examples	379
Anne	ex E (normative): Test Area files	380
Anne	ex F (Normative): AID numbering and acronyms for Framework tests	381
F.1	Toolkit Installation Parameters (TIN)	
F.2	Minimum Handler Availability (MHA)	
F.3	Handler Integrity (HIN)	
F.4	Applet Triggering (APT)	
F.5	Proactive Command Sending (PCS)	
F.6	Envelope Response Posting (ERP)	382

F.7	Framework Security (FWS)	382
F.8	File System Context (FSC)	383
F.9	Exception Handling (EXH)	383
F.10	Other parts transferred to framework from API (API)	383
Anne	ex G (Normative): Configuration Parameters File	384
G.1	Syntax	384
G.2	File Contents and Organisation	384
G.2.1		
G.2.2		
G.2.3		
G.2.4		
G.2.5		
G.3	Full example	386
Anne	ex H (informative): Change history	388
Histo	orv	389

Foreword

This Technical Specification (TS) has been produced by the 3rd Generation Partnership Project (3GPP).

The contents of the present document are subject to continuing work within the TSG and may change following formal TSG approval. Should the TSG modify the contents of the present document, it will be re-released by the TSG with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

- x the first digit:
 - 1 presented to TSG for information;
 - 2 presented to TSG for approval;
 - 3 or greater indicates TSG approved document under change control.
- y the second digit is incremented for all changes of substance, i.e. technical enhancements, corrections, updates, etc.
- z the third digit is incremented when editorial only changes have been incorporated in the document.

1 Scope

The present document covers the minimum characteristics considered necessary in order to provide compliance to GSM 03.19 "SIM API for Java Card TM" [7].

The present document describes the technical characteristics and methods of test for testing the SIM API for Java Card (TM) [7] implemented in the subscriber identity modules (SIMs) for GSM. It specifies the following parts:

- test applicability
- test environment description
- tests format
- test area reference
- conformance requirements
- test auite files
- test procedure
- test coverage and,
- a description of the associated testing tools that shall be used.

2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non-specific.
- For a specific reference, subsequent revisions do not apply.
- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1]	(void)
[2]	(void)
[3]	GSM 11.11: "Digital cellular telecommunication system (Phase 2+); Specification of the Subscriber Identity Module - Mobile Equipment (SIM - ME) interface".
[4]	GSM 11.14: "Digital cellular telecommunications system (Phase 2+); Specification of the SIM application toolkit for the Subscriber Identity Module – Mobile Equipment (SIM – ME) interface".
[5]	GSM 11.17: "Subscriber Identity Module" (SIM) conformance test specification".
[6]	(void)
[7]	GSM 03.19 Rel-98: "Digital cellular telecommunications system (Phase 2+); Subscriber Identity Module Application Programming Interface (SIM API); SIM API for Java Card TM ; Stage 2".
[8]	GSM 03.48 Rel-99: "Digital cellular telecommunications system (Phase 2+); Security Mechanisms for the SIM application toolkit; Stage 2"
[9]	ISO/IEC 7816-3 (1997): "Identification cards - Integrated circuit(s) cards with contacts, Part 3:

Electronic signals and transmission protocols".

[10]	GSM 02.19: "Digital cellular telecommunications system (Phase 2+, Release 98); Subscriber Identity Module Application Programming Interface (SIM API); Service description; Stage 1".
[11]	SUN Java Card Specification: "Java Card 2.1 API Specification".
[12]	SUN Java Card Specification: "Java Card 2.1 Runtime Environment Specification".
[13]	SUN Java Card Specification: "Java Card 2.1 VM Architecture Specification".
[14]	ETSI TS 101 220: "Numbering System for Telecommunication IC card applications".
[15]	GSM 11.10-1: "Digital cellular telecommunication system (Phase 2+); Mobile Station (MS) conformance specification; Part 1: Conformance specification".

3 Definitions and abbreviations

3.1 Definitions

The definitions specified in GSM 11.10-1 [15] clause 3.3 shall apply, unless otherwise specified in the present clause.

Applet: An Applet is an application built up using a number of classes which will run under the control of the Java Card virtual machine.

Applet installation parameters: Default values for applet installation parameters.

Applet loading script: File containing the APDU commands that will load and install the test applet in the card.

CleanUp Script file: File containing the APDU commands that will restore the Default Initial Conditions on the SIM

Conformance Requirement Reference: Description of the expected card behaviour according to 03.19 specification.

Expected state: the state in which the SIM is supposed to be after the execution of the test procedure applied on the relevant initial conditions

Security parameters: Minimum security requirements defined for the applet installation process.

Test Area: Set of Test Cases applicable to a specific part (class method, framework behaviour,...) of the 03.19 specification.

Test Case: Elementary test that checks for compliance with one or more Conformance Requirement References.

Test Output file: TBD.

Test procedure: the sequence of actions/commands to perform all the test cases defined in a test area.

Test Script file: File containing the APDU commands that will execute and verify the test results.

Test Toolkit Applet: Applet designed to test a specific functionality of the SIM API 03.19 specification.

3.2 Abbreviations

For the purpose of the present document, the following abbreviations apply, in addition to those listed in GSM 01.04 [2]:

AC Application Code AID Application Identifier

APDU Application Protocol Data Unit API Application Programming Interface

CAD Card Acceptance Device FFS For Further Study IFD Interface Device

JCRE Java CardTM Run Time Environment

JVM Java Virtual Machine SIM Subscriber Identity Module

SE Sending Entity

4 Test Environment

This clause specifies requirements that shall be met and the testing rules that shall be followed during the test procedure.

4.1 Applicability

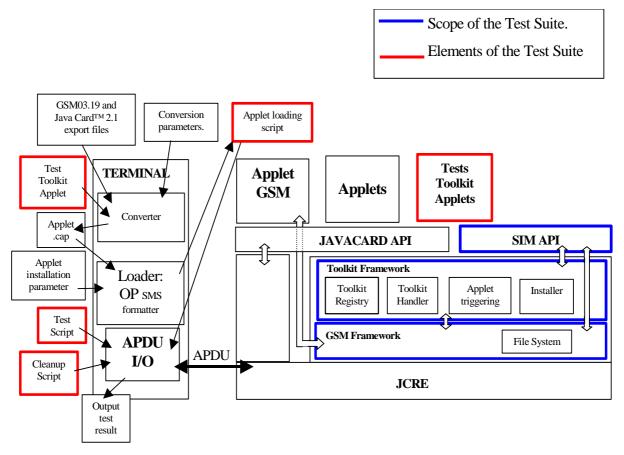
The tests defined in this specification shall be performed taking into account the services supported by the card as specified in the EF_{SST} file.

This specification contains tests that test interoperability at the API level. This specification does not currently contain tests for interoperability at the SIM API framework and at the byte code level. These are for further study.

The test defined in this specification are applicable to cards implementing TS 03.19 [7] version 7.4.0 unless otherwise stated.

4.2 Test environment description

The general architecture for the test environment is:



Note: This diagram shows the test architecture required to test interoperability at both API and bytcode level. The latter is currently not included in the current specification. The diagram is for information.

4.3 Tests format

4.3.1 Test Area Reference

```
Each test area is referenced as follows:
```

```
API Testing:: 'API_[package name]_[classname]_[methodname]' where
      package name:
         sim.access package: '1'
         sim.toolkit package: '2'
      class name:
         yyy: 3 letters for each class.
         See Annex A for full classes acronyms list.
      method name:
         zzzz[input parameters]:
         See Annex A for full methods name acronyms list.
  FWK: framework testing
      Chapter name:
         xxx: 3 letters for each chapter
         See annex F for full chapter acronyms list
      Subchapter name
         yyyy: : 4 letters for each subchapter
         See annex F for full subchapter acronyms list
  LDR: loader testing
      [TBD]
```

4.3.1.1 Conformance requirements

The conformance requirements are expressed in the following way:

- Method prototype as listed in GSM 03.19 [7]specification.
- Normal execution:
 - Contains normal execution and correct parameters limit values, each referenced as a Conformance Requirement Reference Normal (CRRN)
- Parameters error:
 - Contains parameter errors and incorrect parameter limit values, each referenced as a Conformance Requirement Reference Parameter Error (CRRP)
- Context error:
 - Contains errors due to the context the method is used in, each referenced as a Conformance Requirement Reference Context Error (CRRC)

4.3.1.2 Test Area files

The files included in the Test Area use the following naming convention:

- Test Script: [Test Area Reference]_[Test script number].scr

- Test Applet: [Test Area Reference]_[Test applet number].java

- Load Script: [Test Area Reference]_[Load Script number].ldr

- Cleanup Script: [Test Area Reference]_[Cleanup Script number].clr

- Parameter File: [Test Area Reference]_[Parameter File number].par

The test script, applet, installation parameters, load script, cleanup script and conversion parameters numbers start from '1'

The test script, load script and cleanup script shall share a common syntax and format (see Annex B).

The parameter file has an own syntax (see Annex G) and contains parameters to be used for CAP-file conversion and loading/cleanup script generation.

Scripts file shall be run in the following order:

[Test Area Reference]_1.ldr

[Test Area Reference]_1.scr

[Test Area Reference]_1.clr

[Test Area Reference]_2.ldr

[Test Area Reference]_2.scr

[Test Area Reference]_2.clr

. . . .

[Test Area Reference]_n.ldr

[Test Area Reference]_n.scr

[Test Area Reference]_n.clr

In case that one of the files is not needed, it shall be skipped during the tests execution.

4.3.1.3 Test Procedure

Each test procedure contains a table to indicate the expected responses form the API and/or the APDU level as follows:

Test Case			
Id Description API Expectation APDU Expectation		APDU Expectation	
	Test Case detailed description	!	Expected response at APDU level.

4.3.1.4 Test Coverage

The table at the end of each test procedure indicates the correspondence between the Conformance Requirements Reference (CRR) and the different test cases.

4.4 Initial Conditions

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

- unless otherwise stated, the file system and the files' content shall fulfil the requirements described in the "Default Prepersonalisation" paragraph;
- unless otherwise stated, before installing the applet(s) relevant to the current test procedure, all packages specific to other test procedures shall not be present.

When both statements apply, a test procedure is said to be in the "Default Initial Conditions" state.

4.5 Package name

Java packages integrating this Test Suite shall follow this naming convention:

sim.test.access.[Test Area Reference]: Java Card packages containing Test Area References for the GSM 03.19 sim.access package.

sim.test.framework.[Test Area Reference]: Java Card packages containing Test Area References for the GSM 03.19 framework.

sim.test.util: for the Test util package defined in this Test Suite.

sim.test.toolkit.[Test Area Reference]: Java Card packages containing Test Area References for the GSM 03.19[7] sim.toolkit package.

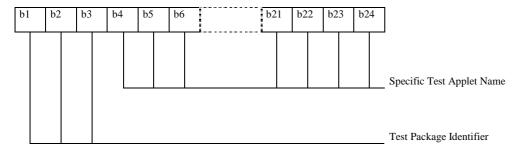
Example:

The package ../sim.test.access.[Test Area Reference] creates the following directory structure ../sim/test/access/[Test Area Reference]/API_1_..._[1..n].*, where 'API_1_..._[1..n].*' are the different test applets Java source files used in [Test Area Reference].

4.6 AID Coding

The AID coding for the Test Packages, Applet classes and Applet shall be as specified in TS 101 220 [14]. In addition, the following TAR values are defined for use within the present document:

TAR Coding (3 bytes/24 bits):



Test package Identifier(bits b1-b3):

000 reserved (as TAR= '00.00.00' is reserved for Card Manager)

001 API

010 Framework

011 Loader

111 sim.test.util

other values are RFU

Application Provider specific data (1 byte):

'00' for Package

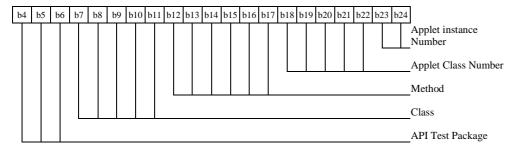
'01' for Applet class

'02' for Applet Instance

For example, the AID of Package sim.test.util is 'A0 00 00 00 09 00 02 FF FF FF FF 89 E0 00 00 00'

4.6.1 Specific Test Applet Name for API

Specific applet test name (bits b4-b24):



for API Test Package(3 bits)

001 sim.access

010 sim.toolkit

other are RFU

Class (5 bits): need to be assigned specification order see Annex A for the full list

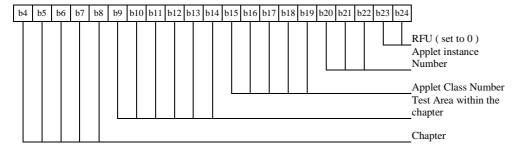
Method (6 bits): need to be assigned specification order see Annex A for the full list

Applet Class Number (5 bits): linked to Test Area, it shall start with 1 for classes and shall be 0 for package.

Applet Instance Number (2 bits) defined in the test procedure it shall start with 01 for applet instance and shall be 00 for package and class.

4.6.2 Specific Test Applet Name for Framework

Specific applet test name (bits b4-b24):



for Chapter (5 bits)

00001 Toolkit Installation Parameters

00010 Minimum Handler Availability

00011 Handler Integrity

00100 Applet Triggering

00101 Proactive Command Sending

00110 Framework Security

00111 Envelope Response Posting

01000 File System Context

01001 Exception Handling

01010 Other parts transferred to framework from API

other are RFU

Test Area within the chapter (6 bits): values are defined in Annex F

Applet Class number (5 bits): linked to Test Area, it shall start with 1 for classes and shall be 0 for package.

Applet Instance number (3 bits) defined in the test procedure it shall start with 01 for applet instance and shall be 00 for package and class.

4.7 Test Equipment

These subclauses recommend a minimum specification for each of the items of test equipment referenced in the tests.

4.7.1 APDU tool

This test tool shall meet the following requirements:

- be able to send command to the card TPDU;
- be able to check none, only a part, or all of the data returned;
- be able to check none, only part, or all of the status returned;
- be able to accept all valid status codes returned;
- be able to support Reader commands;
- be able to generate a log file for each test execution.
- if more data is returned than defined in the test specification, the tool shall continue;
- if less data is returned than defined in the test specification, the tool shall aborts and return an error;
- if there is an error in data or status returned, the tool shall abort and return an error.

The log file produced by the test tool shall include the following information:

- all commands issued;
- all data returned;
- all status returned;
- all errors codes;
- expected data and status in case of error;
- comments from the scripts;
- a log message to report success or failure of the test.

4.7.2 Util package

Annex D includes java source code for the sim.test.util package as well as loading, testing and cleaning script examples.

4.7.3 Applet installation parameters

4.7.3.1 Security parameters

Loading scripts shall use the following security parameters as stated in GSM 03.48 [8] for applet installation:

Parameter	Value in hexadecimal
SPI	0A 00
KIC	00
KID	11
TAR	00 00 00
CNTR	00 00 00 00 01
PCNTR	00
Kev	01 23 45 67 89 AB CD EF

4.7.3.2 Loading components

Cap files in loading scripts shall not include the descriptor component as described in Java Card 2.1 VM Architecture Specification [13].

4.8 Testing methodology

4.8.1 Test interfaces and facilities

The SIM-ME interface provides the main transport interface for the purpose of performing conformance tests.

The SIM API interface provides the main test interface for the purpose of performing conformance tests.

5 Test plan

The test plan is divided according to the SIM API specification, that way the tests will follow the class hierarchy for the sim.toolkit and sim.access package; for the SIM Toolkit framework this test plan describes the different points that will be tested with the present test specification.

6 API Test Plan

6.1 Package sim.access:

6.1.1 Interface SIMView

Note: The Test applet shall be run on a class that implements this interface.

6.1.1.1 Constants

Test Area Reference: API_1_SVW_CONST

6.1.1.1.1 Conformance Requirements

This section does not describe the conformance requirements for a method, but rather for the constants of the interface.

Normal execution

CRRN1: The constants shall have the same name and value that is defined in GSM 03.19 [7].

6.1.1.1.2 Test Suite Files

None.

6.1.1.1.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.1.1.2 Method select(short fid, byte[] fci, short fciOffset, short fciLength)

Test Area Reference: API_1_SVW_ SLCTS_BSS

6.1.1.2.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: If the desired file is selected, the length of the FCI (File Control Information) which has been written to the array fci is returned.

CRRN2: If the length fciLength is greater than or equal to the length of the FCI structure, the whole FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.

CRRN3: If the length fciLength is less than the length of the FCI structure, the first part of the FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.

CRRN4: After selecting a DF/MF no EF is selected.

CRRN5: After selecting a linear fixed EF no record is selected.

CRRN6: After selecting a cyclic EF the first record which is the last updated record is selected.

CRRN7: The current files (file context) of any other applets shall not be changed. See GSM 03.19 [7] - §5.2. This will be tested during the testing of the framework.

CRRN8: The information returned by fci shall be formatted as described in GSM 11.11 [3], §9.2.1.

CRRN9: The file with a File-ID that matches fid shall be found according to the following selection rules:

- 1) An immediate child EF or DF of the current MF/DF can be selected,
- 2) A sibling DF of the current DF can be selected,
- 3) The current MF/DF it self can be selected,
- 4) The parent MF/DF of the current DF can be selected,

5) The MF can always be selected.

Parameter errors

CRRP1: If the array fci is null, an instance of NullPointerException shall be thrown.

CRRP2: If fciOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: If fciLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP4: If fciOffset plus fciLength is greater than the length of the array fci.length, or fciOffset equals fci.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: If the file with a File-ID which matches fid could not be found according to the selection rules listed in CRRN9, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_NOT_FOUND.

CRRC2: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC3: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.2.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_SLCTS_BSS_1.scr

Test Applet: API_1_SVW_SLCTS_BSS_1.java

Load Script: API_1_SVW_SLCTS_BSS_1.ldr

Cleanup Script: API_1_SVW_SLCTS_BSS_1.clr

6.1.1.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	Select EFICCID in MF (Transparent EF)	No exception shall be thrown.	
	<pre>fid = SIMView.FID_EF_ICCID byte[] fci = new byte[34]</pre>	Shall return a value not greater than 20.	
	fciOffset = 0	tnan 20.	
	fciLength = 20	<description fci:<="" of="" th=""><th></th></description>	
	select()	XX XX	
		XX XX	
		2F E2	
		04	
	Colort FF :: MF /Transport FF\	No constitute of the state of t	
2	Select EF _{ICCID} in MF (Transparent EF) fid = SIMView.FID_EF_ICCID	No exception shall be thrown. Shall return 13.	
	fciOffset = 0	fci shall contain the first 13 bytes of	
	fciLength = 13	the FCI structure.	
3	Select DF _{GSM} in MF	No exception shall be thrown.	
3	fid = SIMView.FID_DF_GSM	Shall return 7.	
	fciOffset = 0	fci shall contain the first 7 bytes	
	fciLength = 7	of the FCI.	
	select()	<pre><description fci:<="" of="" pre=""></description></pre>	
		XX XX	
		XX XX	
		7F 20	
		02	
3	Select DF _{GSM} in MF	No exception shall be thrown.	
~	fid = SIMView.FID_DF_GSM	Shall return 7.	
	fciOffset = 0	fci shall contain the entire FCI	
	<pre>fciLength = 7 select()</pre>	structure.	
	Belect()	<description fci:<="" of="" th=""><th></th></description>	
		XX XX	
		XX XX 7F 20	
		02	
		>	
4	Select EF _{ACM} in DF _{GSM} (CyclicEF)	No exception shall be thrown.	
	<pre>fid = SIMView.FID_EF_ACM fciOffset = 0</pre>	Shall return a value between 15	
	fciLength = 20	and 20. (Cyclic EF)	
	select()	fci shall contain the first 15 or more bytes of the FCI structure.	
		fci[14] shall have the value 3	
		(length of record).	
5	Select MF	No exception shall be thrown.	
	<pre>fid = SIMView.FID_MF fciOffset = 0</pre>	Shall return a value between 22	
	fciLength = 34	and 34. fci shall contain the entire FCI	
	select()	structure.	
6	Select DF _{TELECOM} in MF	No exception shall be thrown.	
	fid = SIMView.FID_DF_TELECOM	Shall return 20.	
	<pre>fci[0] = fci[1] = '05' fciOffset = 2</pre>	fci shall contain the first 20 bytes of	
	fciLength = 20	the FCI structure starting at index	
	select()	2. The first two bytes shall (still) have the value '05'.	
7	Select EF _{FDN} in DF _{TELECOM} (Linear FixedEF)	No exception shall be thrown.	
′	fid = SIMView.FID_EF_FDN	Shall return 15.	
	fciOffset = 0	fci shall contain the first 15 bytes of	
	<pre>fciLength = 15 select()</pre>	the FCI structure.	
	BETECC()	fci[14] shall have the value 28	
_	fatta	(length of record).	
8	fci is null fid = SIMView.FID_EF_FDN	Shall throw	
	byte[] nullBuffer = null	java.lang.NullPointerException.	
	fciOffset = 0		
	fciLength = 15		
<u></u>	select()		

ld	Description	API Expectation	APDU Expectation
9	fciOffset < 0	Shall throw	
	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = -1	xception.	
	fciLength = 15	'	
40	select()	Ob all the same	
10	fciLength < 0 fid = SIMView.FID_EF_FDN	Shall throw	
	fciOffset = 0	java.lang.ArrayIndexOutOfBoundsE	
	fciLength = -1	xception.	
	select()		
11	fciOffset + fciLength > fci.length	Shall throw	
	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = 20	xception.	
	<pre>fciLength = 15 select()</pre>		
12	fciOffset >= fci.length	Shall throw	
'-	fid = SIMView.FID_EF_FDN	java.lang.ArrayIndexOutOfBoundsE	
	fciOffset = 34	xception	
	fciLength = 1		
12	select()	4. No expension shall be three:	
13	Selection possibilities 1 - fid = SIMView.FID_MF	1 – No exception shall be thrown.	
	fciOffset = 0	2 – No exception shall be thrown.3 – No exception shall be thrown.	
	fciLength = 15	4 – No exception shall be thrown.	
	select()	5 – No exception shall be thrown.	
	2 - fid = SIMView.FID_DF_TELECOM	6 – No exception shall be thrown.	
	<pre>select() 3 - fid = SIMView.FID_DF_GRAPHICS</pre>	7 – No exception shall be thrown.	
		8 – No exception shall be thrown.	
	4 - fid = SIMView.FID_DF_TELECOM	9 – No exception shall be thrown.	
	select()	o the enterprise enter to the enterprise	
	5 - fid = SIMView.FID_DF_GRAPHICS		
	select()		
	<pre>6 - fid = SIMView.FID_MF select()</pre>		
	7 - fid = SIMView.FID_DF_GSM		
	select()		
	8 - fid = SIMView.FID_DF_TELECOM		
	<pre>select() 9 - fid = SIMView.FID_DF_TELECOM</pre>		
	select()		
	-		
14	EF not selected after MF/DF selection	1 - No exception shall be thrown.	
	1 - fid = SIMView.FID_MF	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	<pre>fid = SIMView.FID_EF_ICCID select()</pre>	reason code NO_EF_SELECTED.	
	2 - fid = SIMView.FID_MF		
	select()		
<u></u>	readBinary()		
15	No selection of non-reachable file	1 – No exception shall be thrown.	
	<pre>1 - fid = SIMView.FID_MF select()</pre>	2 – Shall throw	
	2 - fid = SIMView.FID_EF_ACM	sim.access.SIMViewException with	
	select()	reason code FILE_NOT_FOUND.	
16	No record is selected after selecting linear	1 – No exception shall be thrown.	
	fixed EF	2 – No exception shall be thrown.	
	1 - fid = SIMView.FID_MF	3 – No exception shall be thrown.	
	<pre>select() 2 - fid = FID_DF_SIMTEST</pre>	4 – Shall throw	
		sim.access.SIMViewException with	
	3 - fid = FID_EF_LARU	reason code	
	select()	RECORD_NUMBER_NOT_AVAIL	
	4 - recNumber = 0	ABLE.	
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord()</pre>		
	reauxecoru(/	i	

ld	Description	API Expectation	APDU Expectation
17	Select() 3 - fid = FID EF CARU	 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - The contents of data1 and data2 shall be identical. 	

6.1.1.2.4 Test Coverage

CRR Number	Test Case Number
N1	1-7
N2	3, 5
N3	1, 2, 4, 6, 7
N4	14
N5	16
N6	17
N8	1, 3
N9	1-7, 13
P1	8
P2	9
P3	10
P4	11, 12
C1	15
C2, C3	Not Tested

6.1.1.3 Method select (short fid)

Test Area Reference: API_1_SVW_SLCTS

6.1.1.3.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: If the desired file is selected, no exception is thrown.

CRRN2: After selecting a DF/MF no EF is selected.

CRRN3: After selecting a linear fixed EF no record is selected.

CRRN4: After selecting a cyclic EF the first record which is the last updated record is selected.

CRRN5: The current files (file context) of any other applets shall not be changed [03.19 - §5.2]. This will be tested during the testing of the framework.

CRRN6: The file with a File-ID that matches fid shall be found according to the following selection rules:

- 1) An immediate child EF or DF of the current MF/DF can be selected,
- 2) A sibling DF of the current DF can be selected,
- 3) The current MF/DF it self can be selected,
- 4) The parent MF/DF of the current DF can be selected,

5) The MF can always be selected.

Parameter errors

No requirements.

Context errors

CRRC1: If the file with a File-ID which matches fid could not be found according to the selection rules listed in CCRN6, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_NOT_FOUND.

CRRC2: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC3: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.3.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_SLCTS_1.scr

Test Applet: API_1_SVW_SLCTS_1.java

Load Script: API_1_SVW_SLCTS_1.ldr

Cleanup Script: API_1_SVW_SLCTS_1.clr

6.1.1.3.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	Select EF _{ICCID} in MF (Transparent EF)	No exception shall be thrown.	
	fid = SIMView.FID_EF_ICCID		
2	EF not selected after MF/DF selection	1 - No exception shall be thrown.	
	1 - fid = SIMView.FID_MF	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	fid = SIMView.FID_EF_ICCID	reason code NO_EF_SELECTED.	
	<pre>select() 2 - fid = SIMView.FID_MF</pre>		
	select()		
	readBinary()		
3	No record is selected after selecting linear	1 – No exception shall be thrown.	
	fixed EF 1 - fid = SIMView.FID_MF	2 – No exception shall be thrown.	
	- = SIMVIEW.FID_MF select()	3 – No exception shall be thrown.4 – Shall throw	
	2 - fid = FID_DF_SIMTEST	sim.access.SIMViewException with	
	select()	reason code	
	<pre>3 - fid =FID_EF_LARU select()</pre>	RECORD_NUMBER_NOT_AVAIL	
	4 - recNumber = 0	ABLE.	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
4	readRecord()	4. No expension about he shrows	
4	Record pointer in selected cyclic EF 1 - fid = SIMView.FID_MF	1 - No exception shall be thrown.2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid =FID_DF_SIMTEST	4 - No exception shall be thrown.	
	<pre>select() 3 - fid = FID_EF_CARU</pre>	5 - The contents of data1 and data2	
	select()	shall be identical.	
	4 - byte[] data1 = { 1,2,3 }		
	updateRecord(data1)		
	5 - fid = FID_EF_CARU select()		
	readRecord(data2)		
<u> </u>	compare data1 to data2	4 1	
5	Selection possibilities 1 - fid = SIMView.FID_MF	1 – No exception shall be thrown.	
	select()	2 – No exception shall be thrown.3 – No exception shall be thrown.	
	2 - fid = SIMView.FID_DF_TELECOM	4 – No exception shall be thrown.	
	select()	5 – No exception shall be thrown.	
	<pre>3 - fid = SIMView.FID_DF_GRAPHICS select()</pre>	6 - No exception shall be thrown.	
	4 - fid = SIMView.FID_DF_TELECOM	7 – No exception shall be thrown.	
	select()	8 – No exception shall be thrown.	
	<pre>5 - fid = SIMView.FID_DF_GRAPHICS select()</pre>	9 – No exception shall be thrown.	
	6 - fid = SIMView.FID_MF		
	select()		
	7 - fid = SIMView.FID_DF_GSM select()		
	8 - fid = SIMView.FID_DF_TELECOM		
	select()		
	<pre>9 - fid = SIMView.FID_DF_TELECOM select()</pre>		
6	No selection of unreachable file	1 – No exception shall be thrown.	
"	1 - fid = SIMView.FID_MF	2 – Shall throw	
	select()	sim.access.SIMViewException with	
	2 - fid = SIMView.FID_EF_ACM	reason code FILE_NOT_FOUND.	
	select()		

6.1.1.3.4 Test Coverage

CRR Number	Test Case Number
N1	1
N2	2
N3	3
N4	4
N6	5
C1	6
C2, C3	Not Tested

6.1.1.4 Method status

Test Area Reference: API_1_SVW_STAT_BSS

6.1.1.4.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The FCI (File Control Information) of the current DF (or MF) is returned in the same format as for a SELECT command in case of selecting an MF/DF (described in GSM 03.19 [7], §9.2.1).

CRRN2: If the length fciLength is greater than or equal to the length of the FCI structure, the whole FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.

CRRN3: If the length fciLength is less than the length of the FCI structure, the first part of the FCI structure is copied into the array fci and the length of the FCI which has been written to the array fci is returned.

Parameter errors

CRRP1: If the array fci is null, an instance of NullPointerException shall be thrown.

CRRP2: If fciOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: If fciLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP4: If fciOffset plus fciLength is greater than the length of the array fci.length, or fciOffset equals fci.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC2: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.4.2 Test Suite Files

Additional requirements for the GSM personalisation:

Test Script: API_1_SVW_STAT_BSS_1.scr
Test Applet: API_1_SVW_STAT_BSS_1.java
Load Script: API_1_SVW_STAT_BSS_1.ldr
Cleanup Script: API_1_SVW_STAT_BSS_1.clr

6.1.1.4.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	Status of MF byte[] fci = new byte[34] fciOffset = 0 fciLength = 7 status()	No exception shall be thrown. Shall return 7. fci shall contain the entire FCI structure. <description 00="" 01="" 3f="" fci:="" of="" xx=""></description>	
2	Status after select EF _{ICCID} in MF 1 - fid = SIMView.FID_DF_GSM fciOffset = 0 fciLength = 34 len = select() 2 - byte[] fci2 = new byte[34] len2 = status() 3 - Compare len and len2 4 - Compare the len bytes of fci and fci2	 No exception shall be thrown. Shall return a value between 22 and 34. No exception shall be thrown. Shall return 22 or more. len and len2 shall be identical fci and fci2 shall be identical 	
3	Status of DFTelecom 1 - fid = SIMView.FID_DF_TELECOM select() 2 - fciOffset = 0 fciLength = 100 status()	1 - No exception shall be thrown. Shall return a value between 22 and 34. 2 - No exception shall be thrown. Shall return a value between 22 and34. fci shall contain the entire FCI structure (check that returned value is equal to 13 plus the "length of following data" - fci[12]).FID of the returned fci (fci[4:5]) is FID_DF_TELECOM.	
4	Status DFTELECOM fciOffset = 0 fciLength = 7 status()	No exception shall be thrown. Shall return 7. fci shall contain the first 7 bytes of the FCI structure starting at index 0. FID of the returned fci (fci[4:5]) is FID_DF_TELECOM.	
5	<pre>fci is null byte[] nullBuffer = null fciOffset = 0 fciLength = 34 status()</pre>	Shall throw java.lang.NullPointerException.	
6	<pre>fciOffset < 0 fciOffset = -1 fciLength = 34 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
7	<pre>fciLength < 0 fciOffset = 0 fciLength = -1 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
8	<pre>fciOffset + fciLength > fci.length fciOffset = 20 fciLength = 15 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
9	<pre>fciOffset >= fci.length fciOffset = 34 fciLength = 1 status()</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	

6.1.1.4.4 Test Coverage

CRR Number	Test Case Number
N1	1-4
N2	2, 3
N3	1, 4
P1	5
P2	6
P3	7
P4	8, 9
C1, C2	Not Tested

6.1.1.5 Method readBinary

Test Area Reference: API_1_SVW_REDBS_BSS

6.1.1.5.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: If data can be accessed at the specified offset, the value respOffset plus respLength are returned and the data bytes of the currently selected transparent file are returned in resp.

Parameter errors

CRRP1: If fileOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.

CRRP2: If fileOffset plus respLength exceeds the length of the file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.

CRRP3: If the array resp is null, an instance of NullPointerException shall be thrown.

CRRP4: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP5: If respLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP6: If respOffset plus respLength is greater than the length of the array resp.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.

CRRC2: If the currently selected EF is not transparent, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE INCONSISTENT.

CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.

CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for the reading of an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.5.2 Test Suite Files

Additional requirements for the GSM personalisation: none.

Test Script: API_1_SVW_REDBS_BSS_1.scr

Test Applet: API_1_SVW_REDBS_BSS_1.java

Load Script: API_1_SVW_REDBS_BSS_1.ldr

Cleanup Script: API_1_SVW_REDBS_BSS_1.clr

6.1.1.5.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored	
1	Read from EFICCID in MF (Transparent EF) 1 - fid = SIMView.FID_EF_ICCID select() 2 - fileOffset = 0 byte[] resp = new byte[20] resp[0:19] = '55' respOffset = 10 respLength = 10 readBinary()	1 - No exception shall be thrown. 2 - No exception shall be thrown. Shall return 20. resp shall contain the entire contents of EFICCID starting at index 10. <description 0f="" 55="" ff="" of="" resp:=""></description>	
2	Read from EFICCID in MF resp[0:19] = '55' fileOffset = 5 respOffset = 10 respLength = 5 readBinary()	No exception shall be thrown. Shall return 15. resp shall contain the last 5 bytes of EFICCID starting at index 10. <description 55="" ff="" of="" resp:=""></description>	
3	Offset into File out of bounds fileOffset = -1 respOffset = 0 respLength = 10 readBinary()	Shall throw sim.access.SIMViewException with reason code OUT_OF_FILE_BOUNDARIES.	
4	<pre>fileOffset + respLength > EF length fileOffset = 9 respOffset = 0 respLength = 2 readBinary()</pre>	Shall throw sim.access.SIMViewException with reason code OUT_OF_FILE_BOUNDARIES.	
5	<pre>resp is null byte[] nullBuffer = null fileOffset = 0 respOffset = 0 respLength = 10 readBinary()</pre>	Shall throw java.lang.NullPointerException.	
6	<pre>respOffset < 0 fileOffset = 0 respOffset = -1 respLength = 10 readBinary()</pre>	Shall throw java.lang. ArrayIndexOutOfBoundsException.	
7	<pre>respLength < 0 fileOffset = 0 respOffset = 0 respLength = -1 readBinary()</pre>	Shall throw java.lang. ArrayIndexOutOfBoundsException.	

ld	Description	API Expectation	APDU Expectation
8	respOffset + respLength > resp.length	Shall throw	
	fileOffset = 0	java.lang.	
	respOffset = 10	ArrayIndexOutOfBoundsException.	
	respLength = 11 readBinary()		
-		1 No expension shall be through	
9	EF is not Transparent	1 - No exception shall be thrown.	
	1 - fid = FID_DF_SIMTEST select()	2 - No exception shall be thrown.	
	2 - fid = FID_EF_LARU	3 - Shall throw	
	select()	sim.access.SIMViewException with	
	3 - fileOffset = 0	reason code	
	respOffset = 0	FILE_INCONSISTENT.	
	respLength = 1		
	readBinary()		
10	Access condition not fulfilled	Shall throw	
	1 - fid = DFSIMTTEST	sim.access.SIMViewException with	
	select()	reason code	
	2 - fid = EFTNR	AC_NOT_FULFILLED.	
	select()		
	3 - fileOffset = 0		
	respOffset = 0		
	respLength = 1 readBinary()		
11	EF is invalidated	1 No expension shall be through	
	1 - fid = EFTNU	1 - No exception shall be thrown.	
	invalidate()	2 - Shall throw	
	2 - readBinary()	sim.access.SIMViewException with	
	3 - rehabilitate()	reason code	
	5 ISMADILISAGO()	INVALIDATION_STATUS_CONTR	
		ADICTION.	
		3 - No exception shall be thrown.	
12	No EF selected	1 - No exception shall be thrown.	
	1- fid = SIMView.FID_MF	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 readBinary()	reason code NO_EF_SELECTED.	

6.1.1.5.4 Test Coverage

CRR Number	Test Case Number
N1	1-2
P1	3
P2	4
P3	5
P4	6
P5	7
P6	8,
C1	12
C2	9
C3	10
C4	11
C5, C6	Not Tested

6.1.1.6 Method updateBinary

Test Area Reference: API_1_SVW_UPDBS_BSS

6.1.1.6.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The currently selected transparent file is updated starting at fileOffset, with the string of dataLength bytes in the array data starting at dataOffset.

Parameter errors

CRRP1: If fileOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_FILE_BOUNDARIES.

CRRP2: If fileOffset plus dataLength exceeds the length of the file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT OF FILE BOUNDARIES.

CRRP3: If the array data is null, an instance of NullPointerException shall be thrown.

CRRP4: If dataOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP5: If dataLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP6: If dataOffset plus dataLength greater than the length of the array data.length an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.

CRRC2: If the currently selected EF is not transparent, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.

CRRC3: If the calling applet does not fulfil the access condition, UPDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.

CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for updating of an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.6.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_UPDBS_BSS_1.scr

Test Applet: API_1_SVW_UPDBS_BSS_1.java

Load Script: API_1_SVW_UPDBS_BSS_1.ldr

Cleanup Script: API_1_SVW_UPDBS_BSS_1.clr

6.1.1.6.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	No EF selected	Shall throw	
	fileOffset = 0	sim.access.SIMViewException with	
	<pre>byte[] data = new byte[20]</pre>	reason code NO_EF_SELECTED.	
	data[0] = '55'	1000011 0000 110_21 _02220120.	
	dataOffset = 0		
	dataLength = 10		
	updateBinary()		
2	Update Transparent EF	1 – No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 – No exception shall be thrown.	
	select()	3 – No exception shall be thrown.	
	2 - fid = EFTARU	4 – No exception shall be thrown.	
	select()	Data in resp[0] shall be '55'.	
	3 - fileOffset = 3	Data in resp[o] snail be 33.	
	data[0] = '55'		
	dataOffset = 0		
	dataLength = 1		
	updateBinary()		
1	4 - fileOffset = 3 respOffset = 0		
	respLength = 1		
1	readBinary()		
3	1 - fileOffset = 254	1 - No exception shall be thrown.	
٥	data[0] = '55'	2 - No exception shall be thrown.	
1	data[1] = 'AA'		
	data[2] = '66'	Data in resp shall be	
	dataOffset = 0	resp[0] = '55'	
	dataLength = 3	resp[1] = 'AA'	
	updateBinary()	resp[2] = '66'	
	2 - fileOffset = 254		
	respOffset = 0		
	respLength = 3		
	readBinary()		
4	Offset into File out of bounds	Shall throw	
	fileOffset = -1	sim.access.SIMViewException with	
	dataOffset = 0	reason code	
	dataLength = 10	OUT_OF_FILE_BOUNDARIES.	
-	updateBinary() fileOffset + dataLength > EF length	Shall throw	
5	fileOffset = 259		
	dataOffset = 0	sim.access.SIMViewException with	
	dataLength = 2	reason code	
	updateBinary()	OUT_OF_FILE_BOUNDARIES.	
6	data is null	Shall throw	
"	byte[] nullBuffer = null	java.lang.NullPointerException.	
	fileOffset = 0	Javanang.i tam omtorexoeption.	
1	dataOffset = 0		
	dataLength = 10		
	updateBinary()		
7	dataOffset < 0	Shall throw	
	fileOffset = 0	java.lang.	
		ArrayIndexOutOfBoundsException.	
	dataOffset = -1	,	
	dataLength = 10		
<u></u>	updateBinary()	01 11 11	
8	dataLength < 0	Shall throw	
1	fileOffset = 0	java.lang.	
	dataOffset = 0	ArrayIndexOutOfBoundsException.	
1	dataLength = -1		
9	updateBinary() dataOffset + dataLength > data.length	Shall throw	
9	fileOffset = 0		
1	dataOffset = 10	java.lang.	
1	dataLength = 11	ArrayIndexOutOfBoundsException.	
	updateBinary()		

ld	Description	API Expectation	APDU Expectation
10	<pre>EF is not Transparent 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select() 3 - fileOffset = 0 data[0] = '55' dataOffset = 0 dataLength = 1 updateBinary()</pre>	No exception shall be thrown. No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code FILE_INCONSISTENT.	
11	Access condition not fulfilled 1 - fid = DFSIMTEST select() fid = EFTNU select() 2 - fileOffset = 0 data[0] = '55' dataOffset = 0 dataLength = 1 updateBinary()	No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED.	
12	EF is invalidated 1 - fid = EFTNR invalidate() 2 - fileOffset = 0 data[0] = '55' dataOffset = 0 dataLength = 1 updateBinary() 3 - rehabilitate()	1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION. 3 - No exception shall be thrown.	

6.1.1.6.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3
P1	4
P2	5
P3	6
P4	7
P5	8
P6	9
C1	1
C2	10
C3	11
C4	12
C5. C6	Not Tested

6.1.1.7 Method readRecord

Test Area Reference: API_1_SVW_REDRSBS_BSS

6.1.1.7.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The data bytes from the record, specified by mode and recNumber of the currently selected linear fixed or cyclic EF, is read at recOffset. A total of respLength bytes of this data is copied to the array resp at respOffset.

CRRN2: If the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT:

- if recNumber is not 0, the record addressed by recNumber will be read;
- if recNumber is 0 the current selected record will be read; and
- the current record pointer shall not change.

CRRN3: If the access mode is REC_ACC_MODE_NEXT:

- the next record relative to the current selected record will be selected and read;
- if no current record is selected, the first record will be selected and read;
- if the current record pointer is set to the last record for a cyclic EF the record pointer is set to the first record and the record is read;
- the current record pointer of any other applet shall not be changed.

CRRN4: If the access mode is REC_ACC_MODE_PREVIOUS:

- the previous record relative to the current selected record will be selected and read;
- if no current record is selected, the last record will be selected and read;
- if the current record pointer is set to the first record, for a linear fixed EF the method responses with an error exception and for a cyclic EF the record pointer is set to the last record and the record is read;
- the current record pointer of any other applet shall not be changed.

Parameter errors

CRRP1: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and recNumber is less than 0 or greater than records available, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.

CRRP2: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT, recNumber is 0 and there is no current record selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.

- CRRP3: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_NEXT and the current record pointer is set to the last record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP4: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_PREVIOUS and the current record pointer is set to the first record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP5: If the specified offset into the selected record recOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP6: If recOffset plus respLength is greater than the record length, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP7: If the access mode is not between 2 and 4 inclusive (2 = REC_ACC_MODE_NEXT, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP8: If the array resp is null, an instance of NullPointerException shall be thrown.
- CRRP9: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP10: If respLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP11: If respOffset plus respLength is greater than the length of the array resp.length, or respOffset equals resp.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is neither linear fixed nor cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for reading an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.
- CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY PROBLEM.
- CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.7.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API 1 SVW REDRSBS BSS 1.scr

Test Applet: API_1_SVW_REDRSBS_BSS_1.java

Load Script: API_1_SVW_REDRSBS_BSS_1.ldr

Cleanup Script: API_1_SVW_REDRSBS_BSS_1.clr

6.1.1.7.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	No EF selected	Shall throw	
	recNumber = 1	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code NO_EF_SELECTED.	
	recOffset = 0		
	byte[] resp = new byte[20]		
	respOffset = 0		
	respLength = 10		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
2	Read Absolute and Current from Linear Fixed	1 - No exception shall be thrown.	
	EF	2 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	3 - No exception shall be thrown.	
	select()	resp shall be:	
	2 - fid = EFLARU	resp[0] = '55'	
	select()	resp[1] = '55'	
	// Record pointer not set.		
	3 - recNumber = 0	resp[2] = '55'	
	mode = REC_ACC_MODE_NEXT	resp[3] = '55'	
	recOffset = 0	4 - No exception shall be thrown.	
	respOffset = 0	resp shall be:	
	respLength = 4	resp[0] = 'AA'	
	<pre>readRecord() 4 - recNumber = 2</pre>	resp[1] = 'AA'	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp[2] = 'AA'	
	readRecord()	resp[3] = 'AA'	
	5 - recNumber = 1	5 - No exception shall be thrown.	
	readRecord()	resp shall be:	
	6 - recNumber = 0	resp[0] = '55'	
	resp[0] = resp[1] = resp[2] = resp[3] =	resp[1] = '55'	
	'00'	resp[2] = '55'	
	readRecord()	resp[3] = '55'	
		6 - No exception shall be thrown.	
		resp shall be:	
		resp[0] = '55'	
		resp[1] = '55'	
		resp[2] = '55'	
		resp[3] = '55'	
3	Read Next from Linear Fixed EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = 'AA'	
	recOffset = 0	resp[1] = 'AA'	
	respOffset = 0	resp[2] = 'AA'	
	respLength = 4	resp[3] = 'AA'	
	readRecord()		
4	Read Next from Linear Fixed EF	Shall throw	
	recNumber = 0	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_NEXT	reason code	
	recOffset = 0	RECORD_NUMBER_NOT_AVAIL	
	respOffset = 0	ABLE.	
	respLength = 4	ABLL.	
	readRecord()		
5	Read Previous from Linear Fixed EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_PREVIOUS	resp[0] = '55'	
	recOffset = 0 respOffset = 0	resp[1] = '55'	
	respLength = 4	resp[2] = '55'	
	readRecord()	resp[3] = '55'	
	Caanceora()		
6	Read Previous from Linear Fixed EF	Shall throw	
1	recNumber = 0	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_PREVIOUS	reason code	
	recOffset = 0	RECORD_NUMBER_NOT_AVAIL	
	respOffset = 0	ABLE.	
	respLength = 4	ADLE.	
	readRecord()		

ld	Description	API Expectation	APDU Expectation
7	Read Absolute and Current from Cyclic EF	1 - No exception shall be thrown.	•
1	1 - fid = EFCARU	2 - No exception shall be thrown.	
	select()	resp shall be:	
	2 - recNumber = 2	resp[0] = ' AA'	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp[1] = ' AA'	
	recOffset = 0	resp[2] = 'AA'	
	respOffset = 0		
	respLength = 3	3 - No exception shall be thrown.	
	readRecord()	resp shall be:	
	3 - recNumber = 1	resp[0] = '55'	
	readRecord()	resp[1] = '55'	
	4 - recNumber = 0	resp[2] = '55'	
	resp[0] = resp[1] = resp[2] = '00 readRecord()	4 - No exception shall be thrown.	
	leaurecold()	resp shall be:	
		resp[0] = '55'	
		resp[1] = '55'	
		resp[2] = '55'	
8	Read Next from Cyclic EF		
0	recNumber = 0	No exception shall be thrown. resp shall be:	
	mode = REC_ACC_MODE_NEXT		
	recOffset = 0	resp[0] = 'AA'	
	respOffset = 0	resp[1] = 'AA'	
	respLength = 3	resp[2] = 'AA'	
	readRecord()		
9	Read Next from Cyclic EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_NEXT	resp[0] = '55'	
	recOffset = 0	resp[0] = 33 resp[1] = '55'	
	respOffset = 0		
	respLength = 3	resp[2] = '55'	
	readRecord()		
10	Read Previous from Cyclic EF	No exception shall be thrown.	
	recNumber = 0	resp shall be:	
	mode = REC_ACC_MODE_PREVIOUS	resp[0] = 'AA'	
	recOffset = 0	resp[1] = 'AA'	
	respOffset = 0	resp[2] = 'AA'	
	respLength = 3	1.504[=]	
11	readRecord()	No expension shall be through	
11	Read Previous from Cyclic EF recNumber = 0	No exception shall be thrown.	
	mode = REC_ACC_MODE_PREVIOUS	resp shall be:	
	recOffset = 0	resp[0] = '55'	
	respOffset = 0	resp[1] = '55'	
	respLength = 3	resp[2] = '55'	
	readRecord()		
12	Read Absolute from Linear Fixed EF beyond	1 – No exceptionshall be thrown.	
-	Records	2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select()	reason code	
	2 - recNumber = -1	RECORD_NUMBER_NOT_AVAIL	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	ABLE.	
	recOffset = 0	3 - Shall throw	
	respOffset = 0 respLength = 4	sim.access.SIMViewException with	
	readRecord()	reason code	
	3 - recNumber = 3	RECORD_NUMBER_NOT_AVAIL	
	readRecord()		
40		ABLE.	
13	No current record in linear fixed EF, read	1 - No exception shall be thrown.	
	current 1 - fid = EFLARU	2 - Shall throw	
	select() // No curr rec	sim.access.SIMViewException with	
	2 - recNumber = 0 // curr rec	reason code	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	RECORD_NUMBER_NOT_AVAIL	
	recOffset = 0	ABLE.	
	respOffset = 0		
	respLength = 4		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
14	recOffset < 0	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - Shall throw	
	<pre>select() 2 - recNumber = 1 // rec 1</pre>	sim.access.SIMViewException with	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	reason code	
	recOffset = -1	OUT_OF_RECORD_BOUNDARIE	
	respOffset = 0	S.	
	respLength = 4 readRecord()		
15	recOffset + respLength > Record Length	1 - No exception shall be thrown.	
'	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 1	reason code	
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 2</pre>	OUT_OF_RECORD_BOUNDARIE	
	respOffset = 0	S	
	respLength = 4		
	readRecord()		
16	Reading with invalid mode	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - Shall throw	
	select() 2 - recNumber = 0	sim.access.SIMViewException with	
	mode = 1	reason code INVALID_MODE.	
	recOffset = 0	3 - Shall throw	
	respOffset = 0	sim.access.SIMViewException with	
	respLength = 4 readRecord()	reason code INVALID_MODE.	
	3 - mode = 5		
	readRecord()		
17	resp is null	Shall throw	
	byte[] nullBuffer = null	java.lang.NullPointerException.	
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT resp0ffset = 0</pre>		
	respLength = 10		
	readRecord()		
18	respOffset < 0	Shall throw	
	respOffset = -1	java.lang.	
	respLength = 10	ArrayIndexOutOfBoundsException.	
	readRecord ()		
19	respLength < 0	Shall throw	
	respOffset = 0	java.lang.	
	respLength = -1 readRecord ()	ArrayIndexOutOfBoundsException.	
20	respOffset + respLength > resp.length	Shall throw	
	respOffset = 10	java.lang.	
	respLength = 11	ArrayIndexOutOfBoundsException.	
24	readRecord ()		
21	EF is neither Cyclic nor Linear Fixed 1 - fid = DFSIMTEST	1 - No exception shall be thrown.2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - fid = EFTNU	sim.access.SIMViewException with	
	select()	reason code	
	<pre>3 - resp0ffset = 0 respLength = 4</pre>	FILE_INCONSISTENT.	
	readRecord()	_	
22	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFCNR	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	<pre>2 - respLength = 3 readRecord()</pre>	reason code	
		AC_NOT_FULFILLED.	
23	EF is invalidated	1 - No exception shall be thrown.	
	<pre>1 - fid = EFCNU invalidate()</pre>	2 - Shall throw	
	2 - readRecord()	sim.access.SIMViewException with	
	3 - rehabilitate()	reason code	
		INVALIDATION_STATUS_CONTR ADICTION.	
		3 - No exception shall be thrown.	
		יווטשוו.	

6.1.1.7.4 Test Coverage

CRR Number	Test Case Number
N1	2-5, 7-11
N2	2, 7
N3	3, 8, 9
N4	5, 10, 11
P1	12
P2	13
P3	4
P4	6
P5	14
P6	15
P7	16
P8	17
P9	18
P10	19
P11	20
C1	1
C2	21
C3	22
C4	23
C5, C6	Not Tested

6.1.1.8 Method updateRecord

Test Area Reference: API_1_SVW_UPDRSBS_BSS

6.1.1.8.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: dataLength bytes of the record specified by mode and recNumber of the current selected linear fixed or cyclic EF are updated at recOffset, by using the string of bytes in the array data starting at dataOffset.

CRRN2: If the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and the file is a linear fixed EF:

- the record addressed by recNumber will be updated;
- if recNumber is 0 the current selected record will be updated; and
- the current record pointer shall not change.

CRRN3: If the access mode is REC_ACC_MODE_NEXT and the file is a linear fixed EF:

- the next record relative to the current selected record will be selected and updated;
- if no current record is selected, the first record will be selected and updated;
- the current record pointer of any other applet shall not be changed.

CRRN4: If the access mode is REC_ACC_MODE_PREVIOUS:

- the previous record relative to the current selected record will be selected and updated;

- if no current record is selected, the last record will be selected and updated;
- if a cyclic EF is updated, the oldest record will be updated independent of the current record pointer and this record becomes record number 1 and the current record;
- the current record pointer of any other applet shall not be changed in case of a linear fixed EF.

Parameter errors

- CRRP1: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT and recNumber is less than 0 or greater than records available, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP2: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_ABSOLUTE_CURRENT, recNumber is 0 and there is no current record selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP3: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_NEXT and the current record pointer is set to the last record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP4: If the currently selected EF is linear fixed and the access mode is REC_ACC_MODE_PREVIOUS and the current record pointer is set to the first record; an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.RECORD_NUMBER_NOT_AVAILABLE.
- CRRP5: If the specified offset into the selected record recOffset is less than 0, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP6: If recOffset plus dataLength is greater than the record lengh, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP7: If the access mode is not between 2 and 4 inclusive (2 = REC_ACC_MODE_NEXT, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP8: If the currently selected EF is cyclic and the mode of record access mode is not REC_ACC_MODE_PREVIOUS, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP9: If the array data is null, an instance of NullPointerException shall be thrown.
- CRRP10: If dataOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP11: If dataLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP12: If dataOffset plus dataLength, is greater than the length of the array data.length, or dataOffset equals data.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is neither linear fixed nor cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, UPDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for updating an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.8.2 Test Suite Files

Additional requirements for the GSM personalisation: This test is based on the assumption that the contents of the EFs in $DF_{SIMTEST}$ are identical to those defined in the default pre-personalisation and the current record pointers have not been altered.

Test Script: API_1_SVW_UPDRSBS_BSS_1.scr

Test Applet: API_1_SVW_UPDRSBS_BSS_1.java

Load Script: API_1_SVW_UPDRSBS_BSS_1.ldr

Cleanup Script: API_1_SVW_UPDRSBS_BSS_1.clr

6.1.1.8.3 Test Procedure

SIM Initialisation Responses ignored.	ctation	APDU Expectat	API Expectation	Description	ld
No EF selected reconnected REC_ACC_MODE_ABSOLUTE_CURRENT recoff fact = 0 byte[] data = new byte[20] dataOffset = 0 da			•		
mode = RRC_ROC_MODE_ABSOLUTE_CURRENT recoffast = 0 byte1 data = new byte[20] datablessed d					
mode = RRC_ROC_MODE_ABSOLUTE_CURRENT recoffast = 0 byte1 data = new byte[20] datablessed d			sim.access.SIMViewException with	recNumber = 1	
byte() dataOffset = 0					
dataOffset = 0 dataDegree = 10 updateRecord() 2					
dataLength = 10 updateRecord()					
1 - No exception shall be thrown. 2 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - No exception shall be thrown. 6 - 11 11 12 11 12 13 13 14 14 14 15 14 15 14 15 15					
Fixed EF 1 - fid = DPSIMTEST select() 2 - fid = EPLARU select() // Record pointer not set. 3 - recNumber = 2 mode = REC_ACC_MODE_ABSOLUTE_CURRENT data[0:3] = '11' recoffset = 0 datalength = 4 updateRecord() respOffset = 0 respLength = 4 recommber = 0 respLength = 4 recommber = 1 recommber = 1 readrecord() redateRecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 recommber = 1 readrecord() respOffset = 0 respLength = 4 respOffset = respOffset = 0 respLength = 4 respLeng					
1 - fid = DPSIMTEST select() 2 - fid = EPLARU select() // Record pointer not set. Resp fell etc.		= 4		-	2
select() 2 - fid = FFLARU Select() // Record pointer not set. Resp[0] = 11' Resp[1] = '11' Resp[2] = '11' Resp[2] = '11' Resp[3] = '11' Res			•		
2 - fid = FFLARU select() // Record pointer not set. 3 - recNumber = 2 mode = REC_ACC_MODE_ABSOLUTE_CURRENT data(0:3) = '11' recoffset = 0 dataLength = 4 updateRecord() respOrtset = 0 repLength = 4 recNumber = 0 repLength = 4 recNumber = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() // read result with mode "absolute" repOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 mode = REC_ACC_MODE_NEXT recOffset = 0 repLength = 4 recNumber = 1 recNumber = 1 recNumber = 1 recNumber = 1 recNumber = 0 repLength = 4 recNumber = 1 rec					
select() // Record pointer not set. 3 - recNumber = 2 mode = REC_ACC_MODE_ARSOLUTE_CURRENT data[0:3] = '11' recOffset = 0 dataOffset = 0 dataLength = 4 updateRecord() respOffset = 0 readRecord() 2 - fid = FFLARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respOff				` '	
// Record pointer not set. Resp[2] = 111' Resp[2] = 111' Resp[2] = 111' Resp[3] = '11' Resp[3] =					
Resp[3] = '11' Resp[6]					
mode = REC_ACC_MODE_ABSOLUTE_CURRENT data[0:3] = "11" recoffset = 0 dataOffset = 0 respLength = 4 updateRecord() respOffset = 0 readRecord() respLength = 0 readRecord() 2 - fid = FILARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = "22" resp[2] = "22" resp[3] = "22' resp[3] = "22' resp[3] = "22' resp[6] = "22 resp[6] = "22" resp[7] = "22" resp[7] = "22" resp[7] = "22" resp[8] = "22" resp[8] = "22" resp[9] = "23" resp[9] = "3" recoffset = 0 data[0:3] = "20" 1 - No exception shall be thrown. 3 - No exception shall be					
data[0:3] = '11' reoOffset = 0 dataLength = 4 updateRecord() respOffset = 0 respLength = 0 readRecord() 2 - fid = DFSIMTEST select() 2 - fid = EFLARU select() 3 - reoNumber = 0 mode = REC_ACC_MODE_NEXT reoOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - reeNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" reapOffset = 0 respLength = 4 reoNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" reapOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" reapOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" readRecord() 2 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select select 3 - recNumber = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataIngth = respLength = 4 updateRecord() // read result respOffset = 0 dataIngth = respLength = 4 updateRecord() // read result respOffset = 0 dataIngth = respLength = 4 updateRecord() // read result respOffset = 0 dataIngth = respLength = 4 updateRecord()			Resp[3] = 11		
recoffset = 0 dataOffset = 0 dataLength = 4 updateRecord() 3 Update Current from Linear Fixed EF 1 - fid = DFSINTEST select() 2 - fid = EFLARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 dataIor3] = '00' dataOffset = 0 dataIor3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // write data with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readresult with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readrecord() Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SINTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 respLength = 4 reconsider = 0 respLength = 4 reconsider = 0 respLength = 0					
dataOffset = 0 dataLength = 4 updateRecord() respOffset = 0 readRecord() 1 - fid = DFSIMTEST select() 2 - fid = BELARU select() 3 - recoNumber = 0 recoNation = 0 dataIO:3] = '20' dataIO:3] = '20' dataIO:3] = '30' dataIO:3] = 71 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_CR_ACC_MODE_ABSOLUTE_CURRENT recoNumber = 0 dataIO:3] = '30' dataOffset = 0 dataIO:3] = '30' dataOffset = 0 dataIO:3] = '33' dataOffset = respOffset = 0 dataIO:3] = respOffset = 0 dataIO:3					
<pre>updateRecord() resp0ffset = 0 respLength = 0 readRecord() 1 - fid = DFSIMTEST select() 2 - fid = EFLARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 data[0:3] = '00' dataoffset = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // write data with mode "absolute" resp0ffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" resp0ffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 2 - fid = FID_DF_SIMTEST select() 2 - fid = FID_DF_SIMTEST select() 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataoffset = resp0ffset = 0 data[0:3] = '33' Resp[2] = '33' Resp[3] = '33' Resp[3</pre>					
respOffset = 0 readRecord() 3				_	
respLength = 0 readRecord() 3					
TeadRecord() 3				-	
3 Update Current from Linear Fixed EF 1 - fid = DFSIMTEST select() 2 - fid = EFLARU select() 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' datalength = 4 updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() 1 - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 1 - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - No exception shall be thrown. 6 - REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' Resp[1] = '33' Resp[2] = '33' Resp[2] = '33' Resp[3] = '33'					
1 - fid = DFSIMTEST select() 2 - fid = EFLARU select() 3 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() 2 - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 3 - No exception shall be thrown. 8 - No exception shall be thrown. 9 - No exception shall be t					
select() 2 - fid = EFLARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respDength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respDength = 4 updateRecord() A - No exception shall be thrown. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. 5 - No exception shall be thrown. 6 - REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0 data[0:3] = '33' Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[2] = '33' Resp[3] = '33' Resp[3] = '33'					3
2 - fid = EFLARU select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "absolute" respLength = 4 recNumber = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 datalength = respLength = 4 updateRecord() 4 UpdateNext from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 datalength = respLength = 4 updateRecord()					
select() // Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()				` '	
// Set record pointer with mode "next". 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() // Resp(1] = '32' resp[2] = '22'					
mode = RBC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '00' dataOffset = 0 dataAlength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = RBC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = RBC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4					
recoffset = 0 data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4					
<pre>data[0:3] = '00' dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>					
<pre>dataOffset = 0 dataLength = 4 updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recoffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre> 1 - No exception shall be thrown. 2 - No exception shall be thrown. Resp shall be: Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[3] = '33' Re					
<pre>updateRecord() // write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respoffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>			lesp[0] = 22		
<pre>// write data with mode "current" 4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" resp0ffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>				dataLength = 4	
<pre>4 - recNumber = 0 data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>					
<pre>data[0:3] = '22' mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record</pre>					
<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT updateRecord() // read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>					
<pre>// read result with mode "absolute" respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4</pre>					
respOffset = 0 respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() 1 - No exception shall be thrown. 2- No exception shall be thrown. Resp shall be: Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[3] = '33'				updateRecord()	
respLength = 4 recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4					
recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4					
mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord() 4 Update Next from Linear Fixed EF, no record				1 3	
4 Update Next from Linear Fixed EF, no record pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() 1 - No exception shall be thrown. 2- No exception shall be thrown. 3 - No exception shall be thrown. Resp shall be: Resp[0] = '33' Resp[2] = '33' Resp[3] = '33'					
<pre>pointer set 1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre> 2- No exception shall be thrown. 3 - No exception shall be thrown. Resp shall be: Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[3] = '33'					
1 - fid = FID_DF_SIMTEST select() 2 - fid = FID_EF_LARU Resp shall be: 3 - recNumber = 0 Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[3]					4
<pre>select() 2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre> Resp shall be: Resp[0] = '33' Resp[2] = '33' Resp[2] = '33' Resp[3] = '33'					
2 - fid = FID_EF_LARU select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord() Resp[0] = '33' Resp[1] = '33' Resp[3] = '33'			•		
<pre>select 3 - recNumber = 0 mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre> Resp[0] = '33' Resp[1] = '33' Resp[2] = '33' Resp[3] = '33'				` '	
mode = REC_ACC_MODE_NEXT recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()					
recOffset = 0 data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()					
<pre>data[0:3] = '33' dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre>					
<pre>dataOffset = respOffset = 0 dataLength = respLength = 4 updateRecord()</pre>					
updateRecord()					
				· · · · · · · · · · · · · · · · · ·	
<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT readRecord()</pre>					
5 Update Next from Linear Fixed EF, record 1 - No exception shall be thrown.			1 - No exception shall be thrown.		5
pointer set 2 - No exception shall be thrown.				=	
1 - recNumber = 0 resp shall be:			·	1 - recNumber = 0	
mode = REC_ACC_MODE_NEXT resp[0] = '44'					
recOffset = 0 data[0:3] = '44'					
resp[2] = '44'			resp[2] = '44'	aaoa[0.0] - 11	

ld	Description	API Expectation	APDU Expectation
	dataOffset = 0	resp[3] = '44'	
	dataLength = 4	1	
	updateRecord()		
	2 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()		

ld	Description	API Expectation	APDU Expectation
6	Update Next from Linear Fixed EF, no more	Shall throw	
	records	sim.access.SIMViewException with	
	recNumber = 0	reason code	
	mode = REC_ACC_MODE_NEXT	RECORD_NUMBER_NOT_AVAIL	
	recOffset = 0	ABLE.	
	<pre>data[0:3] = '55' dataOffset = 0</pre>		
	dataLength = 4		
	updateRecord()		
7	Update Previous from Linear Fixed EF, no	1 - No exception shall be thrown.	
	record pointer set	2 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	3 - No exception shall be thrown.	
	select()	4 - No exception shall be thrown.	
	2 - fid = EFLARU select()	resp shall be:	
	3 - recNumber = 0	resp[0] = '66'	
	mode = REC_ACC_MODE_PREVIOUS	resp[1] = '66'	
	recOffset = 0	resp[2] = '66'	
	data[0:3] = '66'	resp[3] = '66'	
	<pre>dataOffset = respOffset = 0 dataLength = respLength = 4</pre>		
	updateRecord()		
	4 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	readRecord()		
8	I = -	1 - No exception shall be thrown	
	pointer set	2 - No exception shall be thrown.	
	1 - recNumber = 0 mode = REC_ACC_MODE_PREVIOUS	Resp shall be:	
	recOffset = 0	Resp[0] = '7744'	
	data[0:3] = '77'	Resp[1] = '7744'	
	dataOffset = respOffset = 0	Resp[2] = '7744'	
	dataLength = respLength = 4	Resp[3] = '7744'	
	<pre>updateRecord() readRecord()</pre>		
	2 - mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
9	Update Previous from Linear Fixed EF , no	Shall throw	
	more records	sim.access.SIMViewException with	
	recNumber = 0	reason code	
	mode = REC_ACC_MODE_PREVIOUS	RECORD_NUMBER_NOT_AVAIL	
	recOffset = 0 data[0:3] = '88'	ABLE.	
	dataOffset = respOffset = 0		
	dataLength = respLength = 4		
	updateRecord()		
10	Update Previous from Cyclic EF	1 - No exception shall be thrown.	
	<pre>1 - fid = FID_DF_SIMTEST select()</pre>	2 - No exception shall be thrown.	
	2 - fid = FID_EF_CARU	3 - No exception shall be thrown.	
	select()	4 - No exception shall be thrown.5 - No exception shall be thrown.	
	3 - recNumber = 2	resp shall be:	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	resp[0] = data[0]	
	recOffset = 0 respOffset = 0	resp[0] = data[0] resp[1] = data[1]	
	respLength = 3	resp[2] = data[1]	
	readRecord()		
	4 - recNumber = 2		
	<pre>mode = REC_ACC_MODE_PREVIOUS data[0:2] = resp[0:2] ^ 'FF'</pre>		
	dataOffset = 0		
	dataLength = 3		
	updateRecord()		
	5 - recNumber = 0 mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	respOffset = 0		
	respLength = 3		
	readRecord()		
11	1 •		
	Records	2 - Shall throw	
	1 - fid = EFLARU select()	sim.access.SIMViewException with	
	2 -recNumber = -1	reason code	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	RECORD_NUMBER_NOT_AVAIL ABLE.	
	recOffset = 0	3 - Shall throw	
	dataOffset = 0	sim.access.SIMViewException with	
	dataLength = 4	Sim.access. Oliviview Lxception with	

ld	Description	API Expectation	APDU Expectation
	2 - recNumber = 3	reason code RECORD_NUMBER_NOT_AVAIL ABLE.	

ld	Description	API Expectation	APDU Expectation
12	No current record in linear fixed EF, update	1 - No exception shall be thrown.	•
	current	2 - Shall throw	
	1 - fid = EFLARU	sim.access.SIMViewException with	
	select() // No curr rec	reason code	
	2 - recNumber = 0 // curr rec	RECORD_NUMBER_NOT_AVAIL	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	ABLE.	
	recOffset = 0 dataOffset = 0		
	dataOffset = 0 dataLength = 4		
	updateRecord()		
13	recOffset < 0	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 1 // rec 1	reason code	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	OUT_OF_RECORD_BOUNDARIE	
	recOffset = -1 dataOffset = 0	S.	
	dataDength = 4		
	updateRecord()		
14	recOffset + dataLength > Record Length	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 1	reason code	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT	OUT_OF_RECORD_BOUNDARIE	
	recOffset = 2 dataOffset = 0	S.	
	dataOffset = 0 dataLength = 4		
	updateRecord()		
15	Updating with invalid mode	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - recNumber = 0 mode = 1	reason code INVALID_MODE.	
	recOffset = 0	3 - Shall throw	
	dataOffset = 0	sim.access.SIMViewException with	
	dataLength = 4	reason code INVALID_MODE.	
	updateRecord()		
	3 - mode = 5		
10	updateRecord()	4 1 1 1 1	
16	Updating Cyclic EF with invalid mode 1 - fid = DFSIMTEST	1 - No exception shall be thrown.	
	select()	2 - No exception shall be thrown.	
	2 - fid = EFCARU	3 - Shall throw	
	select()	sim.access.SIMViewException with	
	3 - recNumber = 0	reason code INVALID_MODE. 4 - Shall throw	
	mode = REC_ACC_MODE_NEXT	sim.access.SIMViewException with	
	recOffset = 0 data[0:2] = '00'	reason code INVALID_MODE.	
	dataOffset = 0	5 - Shall throw	
	dataLength = 3	sim.access.SIMViewException with	
	updateRecord()	reason code INVALID_MODE.	
	4 - recNumber = 0	TOUGHT TOUGHT TO THE TOUGHT.	
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	updateRecord() 5 - recNumber = 2		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
1	updateRecord()		
17	data is null	Shall throw	
	byte[] nullBuffer = null	java.lang.NullPointerException.	
	dataOffset = 0		
	<pre>dataLength = 10 updateRecord()</pre>		
18	dataOffset < 0	Shall throw	
10	dataOffset = -1	java.lang.	
	dataLength = 10	ArrayIndexOutOfBoundsException.	
	updateRecord()	, may mack out or bounds exception.	
19	dataLength < 0	Shall throw	
	dataOffset = 0	java.lang.	
1	dataLength = -1	ArrayIndexOutOfBoundsException.	
20	updateRecord()		
20	dataOffset + dataLength > data.length	Shall throw	
	<pre>dataOffset = 10 dataLength = 11</pre>	java.lang.	
	updateRecord()	ArrayIndexOutOfBoundsException.	
21	EF is neither Cyclic nor Linear Fixed	1 - No exception shall be thrown.	
<u> </u>	- ,		

ld	Description	API Expectation	APDU Expectation
	<pre>1 - fid = DFSIMTEST select() 2 - fid = EFTNR select() 3 - dataOffset = 0 dataLength = 4 updateRecord()</pre>	2 - No exception shall be thrown. 3 - Shall throw sim.access.SIMViewException with reason code FILE_INCONSISTENT.	
22	Access condition not fulfilled 1 - fid = EFCNU select() 2 - recOffset = 0 dataOffset = 0 dataLength = 1 mode = REC_ACC_MODE_PREVIOUS updateRecord() 3 - fid = EFLNU select() 4 - recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0 dataOffset = 0 dataLength = 1 updateRecord()	No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED. No exception shall be thrown. Shall throw sim.access.SIMViewException with reason code AC_NOT_FULFILLED.	
23	EF is invalidated 1 - fid = EFCNR invalidate() 2 - updateRecord() 3 - rehabilitate()	 1 - No exception shall be thrown. 2 - Shall throw sim.access.SIMViewException with reason code INVALIDATION_STATUS_CONTR ADICTION. 3 - No exception shall be thrown. 	

6.1.1.8.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3,4, 5, 7, 8, 10
N2	2, 3
N3	5, 6
N4	7, 8, 9, 10
P1	11
P2	12
P3	6
P4	9
P5	13
P6	14
P7	15
P8	16
P9	17
P10	18
P11	19
P12	20
C1	1
C2	21
C3	22
C4	23
C5, C6	Not Tested

6.1.1.9 Method seek

Test Area Reference: API_1_SVW_SEEKB_BSS

6.1.1.9.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

short pattOffset,
 short pattLength)
throws java.lang.NullPointerException,
 java.lang.ArrayIndexOutOfBoundsException,
 SIMViewException

Normal execution

- CRRN1: If the pattern in patt with the length pattLength at offset pattOffset is found in the record being specified by mode, the current record pointer is set to that record and the record number is returned. The record pointer of any other applet is not changed. This will be tested during the testing of the framework.
- CRRN2: If mode is SEEK_FROM_BEGINNING_FORWARD, the search starts with the first record forward towards the end of the file.
- CRRN3: If mode is SEEK_FROM_END_BACKWARD, the search starts with the last record backward towards the beginning of the file.
- CRRN4: If mode is SEEK_FROM_NEXT_FORWARD, the search starts from the next record after the current record pointer forward towards the end of file. If no current record pointer is selected, the search starts with the first record.
- CRRN5: If mode is SEEK_FROM_PREVIOUS_BACKWARD, the search starts from the previous record before the current record pointer backward towards the beginning of the file. If no current record pointer is selected the search starts with the last record.
- CRRN6: If pattern in patt is not found, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.PATTERN_NOT_FOUND.
- CRRN7: If mode is SEEK_FROM_NEXT_FORWARD and the record pointer is at the last record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.PATTERN_NOT_FOUND.
- CRRN8: If mode is SEEK_FROM_PREVIOUS_BACKWARD and the record pointer is at the first record, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.PATTERN_NOT_FOUND.

Parameter errors

- CRRP1: If mode is not between 0 and 3 inclusive (0 = SEEK_FROM_BEGINNING_FORWARD, etc.), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALID_MODE.
- CRRP2: If the pattern array patt is null, an instance of NullPointerException shall be thrown.
- CRRP3: If pattOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP4: If pattLength is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.
- CRRP5: If pattLength is greater than the size of the record of the currently selected EF, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.OUT_OF_RECORD_BOUNDARIES.
- CRRP6: If pattOffset plus pattLength is greater than the length of the pattern array patt.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is not linear fixed, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If the calling applet does not fulfil the access condition, READ, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.

CRRC4: If the currently selected EF is invalidated and the file status of the EF does not allow for reading an invalidated file, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC5: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC6: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.9.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_SEEKB_BSS_1.scr

Test Applet: API_1_SVW_SEEKB_BSS_1.java

Load Script: API_1_SVW_SEEKB_BSS_1.ldr

Cleanup Script: API_1_SVW_SEEKB_BSS_1.ldr

6.1.1.9.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	•
1	No EF selected	Shall throw	
	Byte[] patt = new byte[20]	sim.access.SIMViewException with	
	pattOffset = 0	reason code NO_EF_SELECTED.	
	pattLength = 10		
	mode = SEEK_FROM_BEGINNING_FORWARD		
	seek()		
2	Pattern not Found	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	<pre>select() 2 - fid = EFLARU</pre>	3 - Shall throw	
		sim.access.SIMViewException with	
	3 - patt[0] = 'DA'	reason code	
	pattOffset = 0	PATTERN_NOT_FOUND.	
	pattLength = 1		
	mode = SEEK_FROM_BEGINNING_FORWARD		
	seek()		
3	Seek from Beginning Forward	No exception shall be thrown. Shall	
	patt[0:2] = '55'	return 1	
	pattOffset = 0		
	pattLength = 3		
	mode = SEEK_FROM_BEGINNING_FORWARD		
4	Seek () Seek from End Backward	No evention shall be thrown. Chall	
4	patt[0:2] = '55'	No exception shall be thrown. Shall	
	pattOffset = 0	return 1	
	pattLength = 3		
	mode = SEEK_FROM_END_BACKWARD		
	seek()		
5	Seek from Next Forward	No exception shall be thrown. Shall	
	patt[0:2] = 'AA'	return 2	
	pattOffset = 0		
	pattLength = 3		
	mode = SEEK_FROM_NEXT_FORWARD		
	seek()	Chall through	
6	Last Record, Seek from Next Forward	Shall throw	
	<pre>mode = SEEK_FROM_NEXT_FORWARD seek()</pre>	sim.access.SIMViewException with	
	Beer ()	reason code	
<u> </u>		PATTERN_NOT_FOUND.	
7	Seek from Previous Backward	No exception shall be thrown. Shall	
	patt[0:2] = '55'	return 1	
	pattOffset = 0		
	<pre>pattLength = 3 mode = SEEK_FROM_PREVIOUS_BACKWARD</pre>		
	seek()		
8	First Record, Seek from Previous Backward	Shall throw	
	i ii st itoosiu, oeek ii oiii i Tevious Dackwalu	Onan unow	

ld	Description	API Expectation	APDU Expectation
	SEEK_FROM_PREVIOUS_BACKWARD	sim.access.SIMViewException with	7 II 2 C Expectation
	seek()	reason code	
		PATTERN_NOT_FOUND.	
9	Pattern not Found (out of reach)	Shall throw	
	patt[0:2] = '55'	sim.access.SIMViewException with	
	pattOffset = 0	reason code	
	pattLength = 3	PATTERN_NOT_FOUND.	
	mode = SEEK_FROM_NEXT_FORWARD	/ \	
40	seek()	4 01 11 11	
10	Invalid mode 1 - mode = 4	1 - Shall throw	
	seek()	sim.access.SIMViewException with	
	2 - mode = -1	reason code INVALID_MODE 2 - Shall throw	
	seek()		
		sim.access.SIMViewException with	
11	patt is null	reason code INVALID_MODE Shall throw	
11	byte[] nullBuffer = null		
	mode = SEEK_FROM_BEGINNING_FORWARD	java.lang.NullPointerException.	
	seek ()		
12	pattOffset < 0	Shall throw	
	patt[0:2] = '55'	java.lang.	
	pattOffset = -1	ArrayIndexOutOfBoundsException	
	<pre>pattLength = 3 mode = SEEK_FROM_BEGINNING_FORWARD</pre>		
	seek()		
13	pattLength < 0	Shall throw	
	patt[0:2] = '55'	java.lang.	
	pattOffset = 0	ArrayIndexOutOfBoundsException	
	pattLength = -1	'	
	<pre>mode = SEEK_FROM_BEGINNING_FORWARD seek()</pre>		
14	pattLength > size of record	Shall throw	
14	patt[0:4] = '55'	sim.access.SIMViewException with	
	pattOffset = 0	reason code	
	pattLength = 4	OUT_OF_RECORD_BOUNDARIE	
	mode = SEEK_FROM_BEGINNING_FORWARD	S	
15	pattOffset + pattLength > patt.length	Shall throw	
15	patt[0:2] = '55'		
	pattOffset = 1	java.lang. ArrayIndexOutOfBoundsException	
	pattLength = 3	ArrayindexOdiOiBodildSException	
	mode = SEEK_FROM_BEGINNING_FORWARD		
	seek()		
16	EF is not Linear Fixed	1 - No exception shall be thrown.	
	1 - fid = EFTNU select()	2 - Shall throw	
	2 - pattOffset = 0	sim.access.SIMViewException with	
	pattLength = 3	reason code	
	mode = SEEK_FROM_BEGINNING_FORWARD	FILE_INCONSISTENT	
	seek()	3 - Shall throw	
	3 - fid = EFCNU	sim.access.SIMViewException with reason code	
	<pre>select() seek()</pre>	FILE_INCONSISTENT	
17	Access condition not fulfilled	1 - No exception shall be thrown.	
''	1 - fid = EFLNR	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - patt[0] = '55'	reason code	
	pattOffset = 0	AC_NOT_FULFILLED.	
	pattLength = 1		
	<pre>mode = SEEK_FROM_BEGINNING_FORWARD seek()</pre>		
18	EF is invalidated	1 - No exception shall be thrown.	
	1 - fid = EFLARU	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - invalidate()	sim.access.SIMViewException with	
	3 - patt[0] = '55	reason code	
	<pre>pattOffset = 0 pattLength = 1</pre>	INVALIDATION_STATUS_CONTR	
	mode = SEEK_FROM_BEGINNING_FORWARD	ADICTION.	
	seek()	4 - No exception shall be thrown.	
	4 - rehabilitate()		

6.1.1.9.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3 - 6, 7
N2	3
N3	4
N4	5
N5	7
N6	2, 6, 8, 9
N7	6
N8	8
P1	10
P2	11
P3	12
P4	13
P5	14
P6	15
C1	1
C2	16
C3	17
C4	18
C5, C6	Not Tested

6.1.1.10 Method increase

Test Area Reference: API_1_SVW_INCR_BS_BS

6.1.1.10.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The value in the array incr is added to the value of the last increased / updated record in the currently selected cyclic EF. The result is stored in the oldest record and returned in the array resp. The updated record becomes record number 1 and is selected as current record. The number of bytes of valid data in resp is returned.

Parameter errors

CRRP1: If the array incr is null, an instance of NullPointerException shall be thrown.

CRRP2: If incrOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: If incrOffset plus the value 3, is greater than the length of the array incr.length, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP4: If the result of the addition is greater than the maximum value of the record (represented by all bytes set to 'FF'), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MAX_VALUE_REACHED.

CRRP5: If the array resp is null, an instance of NullPointerException shall be thrown.

CRRP6: If respOffset is less than 0, an instance of ArrayIndexOutOfBoundsException shall be thrown.

CRRP7: If the remaining length of the array resp at the offset respOffset is less than the length of the record, an instance of ArrayIndexOutOfBoundsException shall be thrown.

Context errors

- CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.
- CRRC2: If the currently selected EF is not cyclic, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC3: If increase is not allowed as indicated by the FCI byte 8 (GSM 11.11: FCI structure of an EF returned by the SELECT command), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.FILE_INCONSISTENT.
- CRRC4: If the calling applet does not fulfil the access condition, INCREASE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.
- CRRC5: If the currently selected EF is invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION STATUS CONTRADICTION.
- CRRC6: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY PROBLEM.
- CRRC7: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.10.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_INCR_BS_BS_1.scr
Test Applet: API_1_SVW_INCR_BS_BS_1.java
Load Script: API_1_SVW_INCR_BS_BS_1.ldr

Cleanup Script: API_1_SVW_INCR_BS_BS_1.clr

6.1.1.10.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
2	No EF selected byte[] incr = new byte[4] byte[] resp = new byte[4] incrOffset = 0 respOffset = 0 increase() Increase, verify response 1 - fid = DFSIMTEST select() 2 - fid = EFCARU select() 3 - //Set both records to 00 00 00 mode = REC_ACC_MODE_PREVIOUS data[0:3] = 0 dataOffset = 0 dataLength = 3 updateRecord() updateRecord() 4 - incrOffset = 0 incr[2] = 1 respOffset = 0	Shall throw sim.access.SIMViewException with reason code NO_EF_SELECTED. 1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. 4 - No exception shall be thrown. resp[] shall contain {0,0,1,0}.	
3	<pre>increase()</pre>	1 - No exception shall be thrown.	
	<pre>incr[2] = 0, incr[3] = 2 respOffset = 1 increase() 2 - resp[3] = 0 recNumber = 0 mode = REC_ACC_MODE_ABSOLUTE_CURRENT</pre>	resp[] shall contain {0,0,0,3}. 2 - No exception shall be thrown. resp[] shall contain {0,0,3,0}.	

ld	Description	API Expectation	APDU Expectation
	recOffset = 0	7 ti 1 = Apoctution	THE DE EMPORTATION
	respOffset = 0		
	respLength = 0		
	readRecord()		
4	incr is null	Shall throw	
	<pre>byte[] nullBuffer = null incrOffset = 0</pre>	java.lang.NullPointerException.	
	respOffset = 0		
	increase()		
5	incrOffset < 0	Shall throw	
	incrOffset = -1	java.lang.	
	respOffset = 0	ArrayIndexOutOfBoundsException.	
	increase()	-	
6	incrOffset + 3 > incr.length	Shall throw	
	<pre>incrOffset = 2 respOffset = 0</pre>	java.lang.	
	increase()	ArrayIndexOutOfBoundsException.	
7	Reach Maximum Value	Shall throw	
	incr[0] = incr[1] = incr[2] = 'FF'	sim.access.SIMViewException with	
	incrOffset = 0	reason code	
	respOffset = 0	MAX_VALUE_REACHED.	
0	increase()	Shall throw	
8	resp is null incr[0] = incr[1] = 0x00'	java.lang.NullPointerException.	
	incr[2] = '02'	java.iang.ivum omterexception.	
	incrOffset = 0		
	byte[] respNull = null		
	respOffset = 0		
_	increase()	Chall throw	
9	respOffset < 0 incrOffset = 0	Shall throw java.lang.	
	respOffset = -1	ArrayIndexOutOfBoundsException.	
	increase()	ArrayindexOdiOiBodildsException.	
10	respOffset + recordLength > resp.length	Shall throw	
	incrOffset = 0	java.lang.	
	respOffset = 2	ArrayIndexOutOfBoundsException.	
11	EF is not Cyclic	1 - No exception shall be thrown.	
' '	1 - fid = EFTARU	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - incrOffset = 0	reason code	
	respOffset = 0	FILE_INCONSISTENT.	
	<pre>increase() 3 - fid = EFLARU</pre>	3 - No exception shall be thrown.	
	select()	4 - Shall throw	
	4 - incrOffset = 0	sim.access.SIMViewException with	
	respOffset = 0	reason code	
	increase()	FILE_INCONSISTENT.	
12	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFCNIC	2 - Shall throw	
	<pre>select() 2 - incrOffset = 0</pre>	sim.access.SIMViewException with	
	respOffset = 0	reason code	
	increase()	AC_NOT_FULFILLED.	
13	EF is invalidated	1 - No exception shall be thrown.	
	1 - fid = EFCARU	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	<pre>2 - invalidate() 3 - incrOffset = 0</pre>	sim.access.SIMViewException with	
	respOffset = 0	reason code	
	increase()	INVALIDATION_STATUS_CONTR	
	4 - rehabilitate()	ADICTION.	
1.4	0111	4 - No exception shall be thrown.	
14	Check increase not allowed from FCI	1 - No exception shall be thrown.	
	1 - fciOffset = 0	Bit 7 of resp[7] shall not be set (0),	
	fciLength = 8	indicating that increase is not	
	select (FID_EF_CINA, fci)	allowed.	
	Verify FCI byte 8 (fci[7])	2 – Shall throw sim.access.SIMViewException with	
	2 - incrOffset = 0	reason code	
	respOffset = 0	FILE_INCONSISTENT	
	<pre>increase()</pre>	1.122_114001401012141	

6.1.1.10.4 Test Coverage

CRR Number	Test Case Number
N1	2, 3
P1	4
P2	5
P3	6
P4	7
P5	8
P6	9
P7	10
C1	1
C2	11
C3	14
C4	12
C5	13
C6, C7	Not Tested

6.1.1.11 Method invalidate

Test Area Reference: API_1_SVW_INVL

6.1.1.11.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The currently selected EF of the calling applet shall be invalidated, i.e. the flag in the EF file status shall be changed accordingly.

Parameter errors

No requirements.

Context errors

CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.

CRRC2: If the calling applet does not fulfil the access condition, INVALIDATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.

CRRC3: If the currently selected EF is already invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC4: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC5: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL ERROR.

6.1.1.11.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_INVL_1.scr

Test Applet: API_1_SVW_INVL_1.java

Load Script: API_1_SVW_INVL_1.ldr

Cleanup Script: API_1_SVW_INVL_1.clr

6.1.1.11.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	No EF is selected	1 - Shall throw	
	1 - invalidate()	sim.access.SIMViewException with	
		reason code NO_EF_SELECTED.	
2	Invalidate EF	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = EFTNR	4 - No exception shall be thrown.	
	<pre>select() 3 - invalidate()</pre>	·	
	4 - rehabilitate()		
3	Access condition not fulfilled	1 - No exception shall be thrown.	
_	1 - fid = EFCNIV	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - invalidate()	reason code	
		AC_NOT_FULFILLED.	
4	EF is already invalidated	1 - No exception shall be thrown.	
	1 - fid = EFTNR	2 - No exception shall be thrown.	
	select()	3 - Shall throw	
	2 - invalidate()	sim.access.SIMViewException with	
	3 - invalidate()	reason code	
		INVALIDATION STATUS CONTR	
		ADICTION.	

6.1.1.11.4 Test Coverage

CRR number	Test Case Number	
N1	2	
C1	1	
C2	3	
C3	4	
C4, C5	Not Tested	

6.1.1.12 Method rehabilitate

Test Area Reference: API_1_SVW_REHA

6.1.1.12.1 Conformance Requirements

The method with the following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The currently selected EF of the calling applet shall be rehabilitated, i.e. the flag in the EF file status shall be changed accordingly.

Parameter errors

No requirements.

Context errors

CRRC1: If the calling applet has currently no EF selected, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.NO_EF_SELECTED.

CRRC2: If the calling applet does not fulfil the access condition, REHABILITATE, to perform this function, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.AC_NOT_FULFILLED.

CRRC3: If the currently selected EF is not invalidated, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INVALIDATION_STATUS_CONTRADICTION.

CRRC4: If the method call causes a memory problem (e.g. memory access error), an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.MEMORY_PROBLEM.

CRRC5: If the method call causes an error to occur that is not expected and thus not handled, an instance of SIMViewException shall be thrown. The reason code shall be SIMViewException.INTERNAL_ERROR.

6.1.1.12.2 Test Suite Files

Additional requirements for the GSM personalisation: None

Test Script: API_1_SVW_REHA_1.scr

Test Applet: API_1_SVW_REHA_1.java

Load Script: API_1_SVW_REHA_1.ldr

Cleanup Script: API_1_SVW_REHA_1.clr

6.1.1.12.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	No EF is selected	1 - Shall throw	
	1 - rehabilitate()	sim.access.SIMViewException with	
		reason code NO_EF_SELECTED.	
2	Rehabilitate invalidated File	1 - No exception shall be thrown.	
	1 - fid = DFSIMTEST	2 - No exception shall be thrown.	
	select()	3 - No exception shall be thrown.	
	2 - fid = EFCNR	4 - No exception shall be thrown.	
	<pre>select() 3 - invalidate()</pre>	5 - No exception shall be thrown.	
	4 - rehabilitate()	resp[] shall contain {0,0,1}.	
	5 - byte[] incr = new byte[3] = {0,0,1}		
	incrOffset = 0		
	<pre>byte[] resp = new byte[1] = 1</pre>		
	respOffset = 0		
	increase()		
3	Access condition not fulfilled	1 - No exception shall be thrown.	
	1 - fid = EFCNRH	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - rehabilitate()	reason code	
		AC_NOT_FULFILLED	
4	Rehabilitate validated File	1 - No exception shall be thrown.	
	1 - fid = EFCNR	2 - Shall throw	
	select()	sim.access.SIMViewException with	
	2 - rehabilitate()	reason code	
		INVALIDATION_STATUS_CONTR	
		ADICTION.	

6.1.1.12.4 Test Coverage

CRR number	Test Case Number	
N1	2	
C1	1	
C2	3	
C3	4	
C4, C5	Not Tested	

6.1.2 Class SIMSystem

6.1.2.1 Method getTheSIMView

Test Area Reference: API_1_SSY_GETS

6.1.2.1.1 Conformance Requirement:

The method with following header shall compliant to its definition in the API.

public static SIMView getTheSIMView()

Normal execution

CRRN1: returns a reference to class which implements the SIMView interface

Parameters error

No requirements

Context errors

No requirements

6.1.2.1.2 Test suite files

No additional requirements for the GSM personalisation:

Test Script: API_1_SSY_GETS_1.scr

Test Applet: API_1_SSY_GETS_1.java

Load Script: API_1_SSY_GETS_1.ldr

Cleanup Script: API_1_SSY_GETS_1.clr

6.1.2.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	reference not equal null after execute	The returned reference shall be not null after execute	
2	reference to the GSM interface	Returned a reference to the GSM interface	

6.1.2.1.4 Test Coverage

CRR number	Test case number
N1	1.2

6.1.3 Class SIMViewException

6.1.3.1 Method throwlt

Test Area Reference: API_1_SVE_THITS

6.1.3.1.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Throws the JCRE instance of SIMViewException with the specified reason

CRRN2: Extends javacard.framework.CardRuntimeException

Parameter errors

No requirements

Context errors

No requirements

6.1.3.1.2 Test Suite Files

No additional requirements for the GSM personalisation

Test Script: API_1_SVE_THITS_1.scr

Test Applet: API_1_SVE_THITS_1.java

Load Script: API_1_SVE_THITS_1.ldr

Cleanup Script: API_1_SVE_THITS_1.clr

6.1.3.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Throws the JCRE instance of	Reason = 0	
	SIMViewException with the specified reason		
2	Throws the JCRE instance of	Reason = 1	
	SIMViewException with the specified reason		
3	Throws the JCRE instance of	Reason = 15	
	SIMViewException with the specified reason		
4	SIMViewException extends	Reason = 0	
	javacard.framework.CardRuntimeException		
5	SIMViewException extends	Reason = 1	
	javacard.framework.CardRuntimeException		
6	SIMViewException extends	Reason = 15	
	javacard.framework.CardRuntimeException		

6.1.3.1.4 Test Coverage

CRR number	Test case number
N1	1,2,3
N2	4,5,6

6.1.3.2 Constructor

Test Area Reference: API_1_SVE_COORS

6.1.3.2.1 Conformance Requirement:

The method with following header shall compliant to its definition in the API.

Normal execution

CRRN1: Construct a SIMViewException with the specified reason

Parameters error

No requirements

Context errors

No requirements

6.1.3.2.2 Test suite files

No additional requirements for the GSM personalisation

Test Script: API_1_SVE_COORS_1.scr

Test Applet: API_1_SVE_COORS_1.java

Load Script: API_1_SVE_COORS.ldr

Cleanup Script: API_1_SVE_COORS.clr

6.1.3.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	SIMViewException with the specified reason	Reason (specified)	
	(The reason shall set with setReason and		
	compare the Exception with getReason)		

6.1.3.2.4 Test Coverage

CRR number	Test case number
N1	1

6.1.3.3 Reason Codes

Test Area Reference: API_1_SVE_CONS

6.1.3.3.1 Conformance Requirement:

There is no API, only constants. This constants shall compliant to its definition in the API.

Normal execution

CRRN1: The Constants of the class SIMViewException shall all have the same name and value defined in the GSM03.19

CRRN2: Constructs SIMViewException a Exception with the specified reason

Parameters error

No requirements

Context errors

No requirements

6.1.3.3.2 Test suite files

None

6.1.3.3.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed

6.2 Package sim.toolkit

6.2.1 Interface ToolkitConstants

6.2.1.1 Constants

Test Area Reference: API_2_TKC_CONS

6.2.1.1.1 Conformance Requirement:

There is no API, only constants. This constants shall be compare to its definition in the API.

Normal execution

CRRN1: The Toolkit Constants shall all have the same name and value defined in the GSM03.19 normalization.

Parameters error

No requirements

Context errors

No requirements

6.2.1.1.2 Test suite files

None

6.2.1.1.3 Test Procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.2.2 Interface ToolkitInterface

6.2.2.1 Method processToolkit

Test Area Reference: API_1_TKI_PRTKB

6.2.2.1.1 Conformance Requirement:

The method with following prototype shall compliant to its definition in the API.

Normal execution

CRRN1: This interface must be implemented by a Toolkit applet (which extends the javacard.framework.Applet class) so that it can be triggered by the Toolkit Handler according to the registration information.

CRRN2: The Toolkit applet will have to implement the processToolkit shared method so that the following events can be notified:

Event	Description	
EVENT_PROFILE_DOWNLOAD	Terminal Profile command reception	
EVENT_FORMATTED_SMS_PP_ENV	03.48 formatted envelope SMS-PP Data	
	Download reception	
EVENT_FORMATTED_SMS_PP_UPD	03.48 formatted Update Record EF SMS	
EVENT_UNFORMATTED_SMS_PP_ENV	Unformatted Envelope SMS-PP Data Download	
	reception	
EVENT_UNFORMATTED_SMS_PP_UPD	Unformatted Update Record EF SMS	
EVENT_UNFORMATTED_SMS_CB	Unformatted Cell Broadcast Data Download	
	command reception	
EVENT_MENU_SELECTION	Envelope Menu Selection command reception	
EVENT_MENU_SELECTION_HELP_REQUEST	Envelope Menu Selection Help Request	
	command reception	
EVENT_CALL_CONTROL_BY_SIM	Envelope Call Control by SIM command	
	reception	
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	Envelope MO Short Message Control by SIM	
	command reception	
EVENT_TIMER_EXPIRATION	Envelope Timer Expiration	
EVENT_EVENT_DOWNLOAD_MT_CALL	Envelope Event Download - MT call	
EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	Envelope Event Download - Call connected	
EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED	Event Download - Call disconnected	
EVENT_EVENT_DOWNLOAD_LOCATION_STATUS	Envelope Event Download - Location status	
EVENT_EVENT_DOWNLOAD_USER_ACTIVITY	Envelope Event Download - User activity	
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE	Envelope Event Download - Idle screen available	
EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	Envelope Event Download - Card Reader Status	
EVENT_STATUS_COMMAND	Status APDU command event	
EVENT_UNRECOGNIZED_ENVELOPE	Unrecognized Envelope command reception	

Parameters error

No requirements

Context errors

No requirements

6.2.2.1.2 Test suite files

The method is tested in the Framework

6.2.2.1.3 Test Coverage

CRR number	Test case number
N1	Tested in Framework
N2	Tested in Framework

6.2.3 Class EditHandler

It is not possible to test the methods provided by this class as it is declared 'abstract'; it will be done in the class inheriting it: EnvelopeResponseHandler, ProactiveHandler.

6.2.4 Class EnvelopeHandler

6.2.4.1 Method getEnvelopeTag

Test Area Reference: API_2_ENH_GENT

6.2.4.1.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

```
public byte getEnvelopeTag()
```

Normal execution

CRRN1: The method shall return the Envelope BER-TLV tag.

CRRN2: The Envelope BER TAG is available for all triggered toolkit applets from the invocation to the termination of their processToolkit method if the EnvelopeHandler is available.

Parameters error

Context errors

6.2.4.1.2 Test suite files

Test Script: API_2_ENH_GENT_1.scr

Test Applet: API_2_ENH_GENT_1.java

Load Script: API_2_ENH_GENT_1.ldr

Cleanup Script: API_2_ENH_GENT_1.clr

6.2.4.1.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	getEnvelopeTag called just after triggering of the	Returns 0xD1	
	application.		
2	getEnvelopeTag called after a proactive	Returns 0xD1	
	command.		
3	getEnvelopeTag called after a second proactive	Returns 0xD1	
	command.		

6.2.4.1.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	1, 2, 3

6.2.4.2 Method getItemIdentifier

Test Area Reference: API_2_ENH_GIID

6.2.4.2.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1:The method shall return the item identifier byte value.

CRRN2:The item identifier byte value returned shall be from the first Item Identifier TLV element.

CRRN3: If the element is available it becomes the TLV selected.

CRRN4: The item identifier is available for all triggered toolkit applets from the invocation to the termination of their processToolkit method if the EnvelopeHandler is available.

Parameters error

No requirements.

Context errors

CRRC1: The method shall throw ToolkitException (UNAVAILABLE_ELEMENT) if the item identifier TLV is not present.

CRRC2: The method shall throw ToolkitException (OUT_OF_TLV_BOUNDARIES) if the item identifier byte is missing in the Item Identifier Simple TLV.

6.2.4.2.2 Test suite files

Test Script: API_2_ENH_GIID_1.scr

Test Applet: API_2_ENH_GIID_1.java

Load Script: API_2_ENH_GIID_1.ldr

Cleanup Script: API_2_ENH_GIID_1.clr

6.2.4.2.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Send envelope SMS-PP Formatted with item identifier TLV and identifier value of 03	Returns 03	
2	Send envelope SMS-PP Formatted with two item identifier TLV with first value FF and second 44	Returns FF	
3	Send envelope SMS-PP Formatted with two item identifier TLV with first value 81 and second 44, call twice the method getItemIdentifier	Returns 81 Returns 81	
4	Send envelope SMS-PP Formatted with item identifier TLV and value of 66. FindTLV with TAG 02. getItemIdentifier and then getValueByte with offset 0	getItemIdentifier=getValueByte	
5	Send envelope SMS-PP Formatted without item identifier TLV and getItemIdentifier	ToolkitException (UNAVAILABLE_ELEMENT)	
6	Send Envelope SMS-PP Formatted with item identifier TLV (66), send proactive command. Then getItemIdentifier	Returns 66	
7	Send Envelope SMS-PP Formatted with item identifier TLV but without item number	ToolkitException (OUT_OF_TLV_BOUNDARIES)	

6.2.4.2.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	2, 3
N3	4
N4	6
C1	5
C2	7

6.2.4.3 Method getSecuredDataLength

Test Area Reference: API_2_ENH_GSDL

6.2.4.3.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the length of the secured data contained in a SMS TPDU TLV.

CRRN2: The length is from the first SMS TPDU TLV.

CRRN3: The length should not include padding bytes.

CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV and if the SMS TP-UD is formatted according to GSM03.48.

CRRN5: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD and if the SMS TP-UD is formatted according to GSM03.48.

CRRN6: If the method is successful, the selected TLV should be the SMS TPDU TLV.

Parameters error

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element.

CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of missing Secured Data.

6.2.4.3.2 Test suite files

Specific triggering:

- SMS CB
- FORMATTED SMS PP UPD
- UNFORMATED SMS PP ENV
- For Formatted triggering if CC/RC/DS is used, the security parameters are the one used for downloading applications.

Test Script: API_2_ENH_GSDL_1.scr
Test Applet: API_2_ENH_GSDL_1.java
Load Script: API_2_ENH_GSDL_1.ldr
Cleanup Script: API_2_ENH_GSDL_1.clr

6.2.4.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Test with TP-OA length of 2	Returns 0x2A	
2	Test with TP-OA length of 6	Returns 0x2A	
3	Test with TP-OA length of 12	Returns 0x2A	
4	Test with RC/CC/DS length of 0	Returns 0x10	
5	Test with RC/CC/DS length of 8	Returns 0x10	
6	Test with PCNTR = 0	Returns 0x10	

7	Test with PCNTR = 7	Returns 0x05
8	Test with SecuredDataLength = 00	Returns 0x00
	Test with UserDataLength = 0x33	Returns 0x33
	Test with UserDataLength = 0x 6C	Returns 0x 6C
11	Test with UserDataLength = 0x 6D	Returns 0x 6D
12	Test with UserDataLength = maximum length:	Returns 0x 79
	0x79	
13	Verify it is the first TPDU TLV:	Returns 0x05
	Send a SMS PP with 2 TPDU TLV and inside two	
	different secured data lengths: 5 and 10	
14	Same test as 1 but with	Returns 0x2A
	FORMATTED_SMS_PP_UPD	
15	Same test as 2 but with	Returns 0x2A
	FORMATTED_SMS_PP_UPD	
16	Same test as 3 but with	Returns 0x2A
	FORMATTED_SMS_PP_UPD	
17	Same test as 4 but with	Returns 0x10
	FORMATTED_SMS_PP_UPD	
18	Same test as 5 but with	Returns 0x10
	FORMATTED_SMS_PP_UPD	
19	Same test as 3 but with	Returns 0x10
	FORMATTED_SMS_PP_UPD	
20	Same test as 4 but with	Returns 0x05
	FORMATTED_SMS_PP_UPD	
21	Same test as 5 but with	Returns 0x00
	FORMATTED SMS_PP_UPD	
22	Same test as 6 but with	Returns 0x33
	FORMATTED_SMS_PP_UPD	
23	Same test as 7 but with	Returns 0x 6C
	FORMATTED_SMS_PP_UPD	Trotains ox oo
24	Same test as 8 but with	Returns 0x 6D
- '	FORMATTED_SMS_PP_UPD	
25	Same test as 9 but with	Returns 0x 79
_	FORMATTED_SMS_PP_UPD	
26	Same test as 10 but with	Returns 0x05
	FORMATTED_SMS_PP_UPD	1.0101110 0700
27	Verify after call of the method the current TLV is	getValueByte returns 0x40
	the TPDU TLV:	95.13.3527.5.534110.57.15
	findTLV device identities, getSecuredDataLength	
	and then getValueByte to verify that the current	
	TLV is the TPDU TLV	
28	Send an envelope SMS CB,	ToolkitException
-0	getSecuredDataLength	UNAVAILABLE_ELEMENT
29	Send an envelope SMS PP unformatted	ToolkitException
23	Cond an envelope owe i i unionnatieu	UNAVAILABLE_ELEMENT
<u> </u>		ONA A VIEWOFF TEFFINITIALI

6.2.4.3.4 Test Coverage

This method has only been tested with call control and the tests shall be improved during 03.48 tests.

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
	11, 12, 13
N2	13
N3	6, 7
N4	1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
	11, 12, 13, 25
N5	14, 15, 16, 17, 18, 19, 20,
	21, 22, 23, 24, 25, 26
N6	27
C1	28
C2	29

6.2.4.4 Method getSecuredDataOffset

Test Area Reference: API_2_ENH_GSDO

6.2.4.4.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the offset of the secured data first byte contained in a SMS TPDU TLV.

CRRN2: The offset is from the first SMS TPDU TLV.

CRRN3: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV and if the SMS TP-UD is formatted according to GSM03.48.

CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD and if the SMS TP-UD is formatted according to GSM03.48.

CRRN5: If the method is successful, the selected TLV should be the SMS TPDU TLV.

Parameters error

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element.

CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of missing Secured Data.

6.2.4.4.2 Test suite files

Specific triggering:

SMS CB

FORMATTED SMS PP UPD UNFORMATED SMS PP ENV

For Formatted triggering if CC/RC/DS is used, the security parameters are the one used for downloading applications.

Test Script: API_2_ENH_GSDO_1.scr

Test Applet: API_2_ENH_GSDO_1.java

Load Script: API_2_ENH_GSDO_1.ldr

Cleanup Script: API_2_ENH_GSDO_1.clr

6.2.4.4.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Test with TP-OA length of 2	Returns 0x21	
2	Test with TP-OA length of 6	Returns 0x23	
3	Test with TP-OA length of 12	Returns 0x26	
4	Test with RC/CC/DS length of 0	Returns 0x21	
5	Test with RC/CC/DS length of 8	Returns 0x29	
6	Send a SMS PP with 2 TPDU TLV and inside two	Returns 0x24 (the first offset)	
	different secured data offsets		

7	Same test as 1 but with FORMATTED_SMS_PP_UPD	Returns 0x21	
8	Same test as 2 but with FORMATTED_SMS_PP_UPD	Returns 0x23	
9	Same test as 3 but with FORMATTED_SMS_PP_UPD	Returns 0x26	
10	Same test as 4 but with FORMATTED_SMS_PP_UPD	Returns 0x21	
11	Same test as 5 but with FORMATTED_SMS_PP_UPD	Returns 0x29	
12	Same test as 6 but with FORMATTED_SMS_PP_UPD	Returns 0x24 (the first offset)	
13	Verify after call of the method the current TLV is the TPDU TLV: findTLV device identities, getSecuredDataOffset and then getValueByte to verify that the current TLV is the TPDU TLV	Returns 0x40	
14	Send an envelope SMS CB, getSecuredDataOffset	ToolkitException UNAVAILABLE_ELEMENT	
15	Send an envelope SMS PP unformatted	ToolkitException UNAVAILABLE_ELEMENT	
16	Send an envelope SMS-PP formatted with no secured data, getSecuredDataOffset	Returns 0x21	

6.2.4.4.4 Test Coverage

This method has only been tested with call control and the tests shall be improved during 03.48 tests.

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 16
N2	6, 12
N3	1, 2, 3, 4, 5, 6, 16
N4	7, 8, 9, 10, 11
N5	13
C1	14
C2	15

6.2.4.5 Method getTheHandler

Test Area Reference: API_2_ENH_GTHD

6.2.4.5.1 Conformance Requirements

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the single system instance of the EnvelopeHandler class.

CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12])

Parameters error

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

6.2.4.5.2 Test suite files

Test Script: API_2_ENH_GTHD_1.scr

Test Applet: API_2_ENH_GTHD_1.java

Load Script: API_2_ENH_GTHD_1.ldr

Cleanup Script: API_2_ENH_GTHD_1.clr

6.2.4.5.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler twice	The returned objects shall be the	
		same	
2	Verify that getTheHandler returns an	The reference returned shall be an	
	EnvelopeHandler	EnvelopeHandler (checkcast)	
	GetTheHandler		
3	Verify the returned value is not null	The reference returned shall not be	
	GetTheHandler	null.	

6.2.4.5.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	To be checked in
	Framework tests and
	insert here cross
	reference
C1	To be checked in
	Framework tests and
	insert here cross
	reference

6.2.4.6 Method getTPUDLOffset

Test Area Reference: API_2_ENH_GTPO

6.2.4.6.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the TPUDL offset in a SMS TPDU TLV.

CRRN2: The offset is from the first SMS TPDU TLV.

CRRN3: The method can be used if the event is EVENT_FORMATTED_SMS_PP_ENV.

CRRN4: The method can be used if the event is EVENT_FORMATTED_SMS_PP_UPD.

CRRN5: The method can be used if the event is EVENT_UNFORMATTED_SMS_PP_ENV.

CRRN6: The method can be used if the event is EVENT_UNFORMATTED_SMS_PP_UPD.

CRRN7: If the method is successful, the selected TLV should be the SMS TPDU TLV.

Parameters error

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) in case of unavailable SMS TPDU TLV element.

CRRC2: The method shall thrown ToolkitException (UNAVAILABLE_ELEMENT) if the TPUDL field does not exist.

6.2.4.6.2 Test suite files

Specific triggering:

FORMATTED SMS PP UPD UNFORMATTED SMS PP UPD UNFORMATTED SMS PP ENV SMS CB

Test Script: API_2_ENH_GTPO_1.scr

Test Applet: API_2_ENH_GTPO_1.java

Load Script: API_2_ENH_GTPO_1.ldr

Cleanup Script: API_2_ENH_GTPO_1.clr

6.2.4.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Test with TP-OA length of 2	Returns 0x0D	
	Test with TP-OA length of 6	Returns 0x0F	
3	Test with TP-OA length of 12	Returns 0x12	
4	Send a SMS PP with 2 TPDU TLV and inside two different UDL offsets	Returns 0x10 (the first offset)	
5	Same test as 1 but with FORMATTED_SMS_PP_UPD	Returns 0x0D	
6	Same test as 2 but with FORMATTED_SMS_PP_UPD	Returns 0x0F	
7	Same test as 3 but with FORMATTED_SMS_PP_UPD	Returns 0x12	
8	Same test as 4 but with FORMATTED_SMS_PP_UPD	Returns 0x10 (the first offset)	
9	Same test as 1 but with UNFORMATTED_SMS_PP_UPD	Returns 0x0D	
10	Same test as 2 but with UNFORMATTED_SMS_PP_UPD	Returns 0x0F	
11	Same test as 3 but with UNFORMATTED_SMS_PP_UPD	Returns 0x12	
12	Same test as 4 but with UNFORMATTED_SMS_PP_UPD	Returns 0x12 (the first offset)	
13	Same test as 1 but with UNFORMATTED_SMS_PP_ENV	Returns 0x0D	
14	Same test as 2 but with UNFORMATTED_SMS_PP_ENV	Returns 0x0F	
15	Same test as 3 but with UNFORMATTED_SMS_PP_ENV	Returns 0x12	
16	Same test as 4 but with UNFORMATTED_SMS_PP_ENV	Returns 0x10 (the first offset)	
17	Verify after call of the method the current TLV is the TPDU TLV: findTLV device identities, getTPUDLOffset and then	Returns 0x40	

	getValueByte to verify that the current TLV is the TPDU TLV		
18	Send an envelope SMS CB, getTPUDLOffset	ToolkitException UNAVAILABLE_ELEMENT	

6.2.4.6.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4, 5, 6, 7, 8, 9, 10,
	11, 12, 13, 14, 15, 16, 17
N2	4
N3	1, 2, 3, 4, 17
N4	5, 6, 7, 8
N5	13, 14, 15, 16
N6	9, 10, 11, 12
N7	17
C1	18
C2	Don't no how to test

6.2.4.7 Method getLength

Test Area Reference: API_2_ENH_GLEN

6.2.4.7.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

throws ToolkitException

Normal execution

CRRN1: returns the length in bytes of the TLV list.

Parameter Error

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.4.7.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GLEN_1.scr

Test Applet: API_2_ENH_GLEN_1.java

Load Script: API_2_ENH_GLEN_1.ldr

Cleanup Script: API_2_ENH_GLEN_1.clr

6.2.4.7.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Send an envelope SMS PP with BER length of 0x31	Result of getLength() is 0x0031	
2	Send an envelope SMS PP with BER length of 0x7F	Result of getLength() is 0x007Fh	
3	Send an envelope SMS PP with BER length of 81 80	Result of getLength() is 0x0080h	
4	Send an envelope SMS PP with BER length of 81 FC	Result of getLength() is 0x00FCh	

6.2.4.7.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3, 4
C1	Does not apply for
	EnvelopeHandler

6.2.4.8 Method copy

Test Area Reference: API_2_ENH_COPY_BSS

6.2.4.8.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: copies the simple TLV list contained in the handler to the destination byte array.

 $CRRN2:\ returns\ dstOffset+dstLength.$

Parameter errors

CRRP1: if dstBuffer is null a NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.

CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.8.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_COPY_BSS_1.scr
Test Applet: API_2_ENH_COPY_BSS_1.java

Load Script: API_2_ENH_COPY_BSS_1.ldr

Cleanup Script: API_2_ENH_ COPY _ BSS _1.clr

6.2.4.8.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to dstBuffer	NullPointerException is thrown	
2	dotOffoot > dotDuffor longth	ArrayIndexOutOfBoundsException is	
_	dstOffset ≥ dstBuffer.length dstBuffer.length = 5	thrown	
	dstOffset = 5	trirown	
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsException is	
3	dstBuffer.length = 5	thrown	
	dstOffset = -1	unown	
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsException is	
	dstBuffer.length = 5	thrown	
	dstOffset = 0		
	dstLength = 6		
5	DstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsException is	
	DstBuffer.length = 5	thrown	
	DstOffset = 3		
	DstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsException is	
	dstBuffer.length = 5	thrown	
	dstOffset = 0		
	dstLength = -1		
7	DstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_BO	
	DstBuffer.length = 48	UNDARIES is thrown	
	DstOffset = 0		
_	DstLength = 48 Successful call, dstBuffer is the whole buffer	D	
8	DstBuffer.length = 47	Result of copy() is 0X0047	
	DstBuller.length = 47		
	DstLength = 47		
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0032	
10	DstBuffer.length = 50	Result of copy() is 0x0032	
	dstOffset = 3		
	dstLength = 47		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0009	
'-	dstBuffer.length = 15	. 100011 01 0000	
	dstOffset = 3		
	dstLength = 6		
13	Compare the whole buffer	Result of arrayCompare() is 0	
14	Successful call, dstBuffer is part of a buffer	Result of copy() is 0X0104	
	dstBuffer.length = 260	170	
	dstOffset = 257		
	dstLength = 3		
15	Compare the whole buffer	Result of arrayCompare() is 0	
16	Successful call, copy with length =0	Result of copy() is 0x104	
	dstBuffer.length = 260		
	dstOffset = 260		
	dstLength = 0		

6.2.4.8.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13, 15
N2	8, 10, 12, 14, 16
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1	Does not apply for
	EnvelopeHandler

6.2.4.9 Method findTLV

Test Area Reference: API_2_ENH_FINDBB

6.2.4.9.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.

CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.

CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.

CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.

CRRN5: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.9.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_ENH_FINDBB_1.scr

Test Applet: API_2_ENH_ENH_FINDBB_1.java

Load Script: API_2_ENH_ENH_FINDBB_1.ldr

Cleanup Script: API_2_ENH_FINDBB_1.clr

6.2.4.9.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	Trig the applet with SMS PP including one more		
	tag 02 and one TAG 04		
1	Invalid input parameter	ToolkitException.BAD_INPUT_P	
	Occurrence = 0	ARAMETER is thrown	
2	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 02h		
	Occurrence = 1		
3	Call the getValueLength() method	Result is 0x02	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	Tag = 06h		
	Occurrence = 1		

5	Call the getValueLength() method	Result is 0x05h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	Tag = $03h$		
	Occurrence = 1		
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown.	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	Tag = 02h		
	Occurrence = 3		
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown.	
10	Search the TLV	Result is	
	Tag = 02h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 2		
11	Search the TLV	Result is	
	Tag = 04h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1		
12	Search tag 81h	Result is TLV_FOUND_CR_SET	
	Tag = 86h		
	Occurrence = 1		
13	Search tag 84h	Result is	
	Tag = 84h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1		

6.2.4.9.4 Test Coverage

CRR number	Test case number
N1	3, 5
N2	2, 4
N3	10, 11
N4	6, 7,8, 9
N5	12, 13
P1	1
C1	Does not apply for EnvelopeHandler

6.2.4.10 Method getValueLength

Test Area Reference: API_2_ENH_GVLE

6.2.4.10.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.10.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GVLE_1.scr

Test Applet: API_2_ENH_GVLE_1.java

Load Script: API_2_ENH_GVLE_1.ldr

Cleanup Script: API_2_ENH_GVLE_1.clr

6.2.4.10.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 33, Length C8		
1	getValueLength()	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown	
2	Search TLV 02h		
	getValueLength()	Result is 0X0002	
3	Search TLV 0Bh		
	getValueLength()	Result is 0X0024	
4	Search TLV 33h		
	getValueLength()	Result is 0X00C8	

6.2.4.10.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4
C1	Does not apply for
	EnvelopeHandler
C2	1

6.2.4.11 Method getValueByte

Test Area Reference: API_2_ENH_GVBYS

6.2.4.11.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_public} \mbox{public byte getValueByte(short valueOffset)} \\ \mbox{throws ToolkitException}$

Normal execution

CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

Parameter errors

CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.11.2 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_GVBY.scr

Test Applet: API_2_ENH_GVBY_1.java

Load Script: API_2_ENH_GVBY.ldr

Cleanup Script: API_2_ENH_GVBY_1.clr

6.2.4.11.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 33, Length C8 Value 01 02		
1	getValueByte(0)	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
2	Search TLV 02h		
	getValueByte(2)	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
3	Search TLV 02h		
	getValueByte(1)	Result is 0x81	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 83h (Source)	
5	Search TLV 33h		
	getValueByte(7E)	Result is 0x7F	
6	Search TLV 33h		
	getValueByte(80)	Result is 0x81	
7	getValueByte(7F)	Result is 0x80	
8	Search TLV B3h		
	getValueByte(C7)	Result is 0xC8	

6.2.4.11.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5, 6, 7, 8
P1	2
C1	Does not apply for
	EnvelopeHandler
C2	1

6.2.4.12 Method copyValue

Test Area Reference: API_2_ENH_CPYVS_BSS

6.2.4.12.1 Conformance Requirement

The method with following header shall be compliant with its definition in the API.

Normal execution

CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.12.3 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_CPYVS_BSS_1.scr

Test Applet: API_2_ENH_CPYVS_BSS_1.java

Load Script: API_2_ENH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_ENH_CPYVS_BSS_1.clr

6.2.4.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Search TLV 02h	·	
	copyValue() with a null dstBuffer	NullPointerException is thrown	
2	Search TLV 0Bh		
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 5		
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
_	dstLength = 1	A do do 0.000 p do F ati-	
4	dstLength >dstBuffer.length dstBuffer.length = 5	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 0	n is thrown	
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3	II IS UIIOWII	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = -1		
7	Search TLV 06h		
	valueOffset ≥ TLV Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 6	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
8	dstLength = 1 valueOffset < 0	ToolkitEveention OUT OF TIV	
0	valueOffset = -1	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	DOUNDAKIES IS INFOWN	
	dstOffset = 0		
	1		

	dstLength = 1		
9	dstLength > TLV length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	dstBuffer.length = 15	BOOMB/ INTEG IS II II OWIT	
	dstOffset = 0		
	dstLength = 7		
10	valueOffset + dstLength > TLV length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 2	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 5		
11	Search TLV 01h		
	copyValue()	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown on the	
		copyValue() method	
12	Search TLV 06h		
	Successful call	Result of copyValue() is 0x0006	
	valueOffset = 0		
	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		
13	Compare buffer	Result is 00h	
	buffer = 81 11 22 33 44 F5		
14	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of copyValue() is 0x0007	
	valueOffset = 1		
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 4		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 11 22		
	33 44 55 55 55		
	55 55 55 55 55 55 55 55 55 55		
10		Deput of convivalue() is 20	
16	Successful call, copy with length =0 dstBuffer.length = 20	Result of copyValue() is 20	
	dstOffset = 20		
	dstLength = 0		

6.2.4.12.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14, 16
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	EnvelopeHandler
C2	11

6.2.4.13 Method compareValue

Test Area Reference: API_2_ENH_CPRVS_BSS

6.2.4.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Compares the last found TLV element with a buffer:

CRRN1: returns 0 if identical.

CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.

CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.4.13.3 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_CPRVS_BSS_1.scr

Test Applet: API_2_ENH_CPRVS_BSS_1.java

Load Script: API_2_ENH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_ENH_CPRVS_BSS_1.clr

6.2.4.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Search TLV 02h	74 T EXPOSITION	7.1 DO Expodiano.
•	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	Search TLV 0Bh		
	<pre>compareOffset ≥ compareBuffer.length compareBuffer.length = 5 compareOffset = 5 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>compareOffset < 0 compareBuffer.length = 5 compareOffset = -1 compareLength = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>compareLength >compareBuffer.length compareBuffer.length = 5 compareOffset = 0 compareLength = 6</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>compareOffset + compareLength</pre>	ArrayIndexOutOfBoundsException is thrown	
6	<pre>compareLength < 0 compareBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	

	compareOffset = 0	1	
	compareUffset = 0 compareLength = -1		
7	Search TLV 06h		
		ToolkitEveention OUT OF TLV	
	valueOffset ≥ TLV Length valueOffset = 6	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	compareBuffer.length = 15	BOONDAKIES IS UIIOWII	
	compareOffset = 0		
	compareLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
_	compareLength = 1 compareLength > TLV length	Tablist Counties OUT OF TIV	
9	valueOffset = 0	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	compareBuffer.length = 15	BOUNDARIES IS thrown	
	compareOffset = 0		
	compareLength = 7		
10	valueOffset + compareLength > TLV length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 2	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
4.4	compareLength = 5 Search TLV 01h	Popult in TLV/ NOT FOUND	
11		Result is TLV_NOT_FOUND	
	compareValue()	ToolkitException.UNAVAILABLE ELEMENT is thrown	
10	0	_ELEMENT IS thrown	
12	Search TLV 06h		
	Initialise compareBuffer		
	compareBuffer = 81 11 22 33 44 F5		
	Compare buffers	Result is 00h	
	valueOffset = 0	Result is oon	
	compareOffset = 0		
	compareLength = 6		
13	Initialise compareBuffer		
	compareBuffer =		
	7F 11 22 33 44 F5		
	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer		
	compareBuffer =		
	83 11 22 33 44 F5 Compare buffers with same parameters	Result is -1	
15	Initialise compareBuffer	Result is -1	
15	compareBuffer =		
	55 55 55 81 11 22 33 44 F5		
	55 55 55 55		
	Compare buffers	Result is 00h	
	valueOffset = 1		
	compareOffset = 4		
42	compareLength = 5		
16	Initialise compareBuffer		
	compareBuffer = 55 55 55 81 10 22 33 44 F5		
	55 55 55 55 55		
	Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer	1 COUNT IO 1 I	
''	compareBuffer =		
	55 55 55 81 12 22 33 44 F5		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
18	Successful call, compareValue with length =0	Result of compareValue() is 0	
	CompareBuffer.length = 15		
	CompareOffset = 15		
<u></u>	CompareLength = 0		

6.2.4.13.4 Test Coverage

CRR number	Test case number
N1	12, 15
N2	13, 16, 18
N3	14, 17
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	EnvelopeHandler
C2	11

6.2.4.14 Method findAndCopyValue(byte tag, byte[] dstBuffer, short dstOffset)

Test Area Reference: API_2_ENH_FACYB_BS

6.2.4.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.14.3 Test Suite files

Specific triggering: None

Test Script: API_2_ENH_FACYB_BS_1.scr

Test Applet: API_2_ENH_FACYB_BS_1.java

Load Script: API_2_ENH_FACYB_BS_1.ldr

Cleanup Script: API_2_ENH_FACYB_BS_1.clr

6.2.4.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44	·	•
	Tag 33, Length C4 Value 01 02		
1	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 06h	n is thrown	
	dstBuffer.length = 06		
_	dstOffset = 06	A de de (
3	<pre>dstOffset < 0 dstBuffer.length = 06</pre>	ArrayIndexOutOfBoundsExceptio	
	dstOffset = -1	n is thrown	
4	length > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 05	n is thrown	
	dstOffset = 0		
5	DstOffset + length >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	DstBuffer.length = 06	n is thrown	
	DstOffset = 1		
	Coloot o TI V /tom 02h)		
6	Select a TLV (tag 02h) findAndCopyValue()	ToolkitEveention LINAVALLARI F	
	tag = 03h	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
7	Successful call	_ELEMENT is thrown.	
/	Tag = 06h	Result of findAndCopyValue () is 0006	
	DstBuffer.length = 06	0006	
	DstOffset = 0		
8	Compare buffer	Result is 00h	
	buffer = 81 11 22 33 44 F5		
9	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue () is	
	<pre>dstBuffer.length = 12 dstOffset = 2</pre>	0008	
10	Compare buffer	Result is 00h	
10	buffer =	Tresuit is out	
	55 55 81 11 22 33 44 F5 55 55 55 55		
11	Successful call	Result of findAndCopyValue () is	
	tag = 02h	0002	
	<pre>dstBuffer.length = 2 dstOffset = 0</pre>		
12	Compare buffer	Result is 00h	
	buffer = 83 81	Tresult is out	
13	Successful call (with tag 82h)	Result of findAndCopyValue () is	
	tag = 82h	0002	
	dstBuffer.length = 02		
14	dstOffset = 0 Compare buffer	Result is 00h	
14	buffer = 83 81	Legalit 12 0011	
15	Successful call (with tag B3h)	Result of findAndCopyValue () is	
	tag = B3h	00C4	
	dstBuffer.length = C4		
	dstOffset = 0		
16	Compare buffer	Result is 00h	
	buffer = 01 02 C4		

6.2.4.14.4 Test Coverage

CRR number	Test case number	
N1	8, 10, 12	
N2	6	
N3	7, 9, 11	
N4	13, 14, 15, 16	
P1	1	
P2	2, 3, 4, 5	
C1 Does not apply for		
	EnvelopeHandler	

6.2.4.15 Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference: API_2_ENH_FACYBS_BSS

6.2.4.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.15.3 Test Suite files

Test Script: API_2_ENH_FACYBS_BSS_1.scr

Test Applet: API_2_ENH_FACYBS_BSS_1.java

Load Script: API_2_ENH_FACYBS_BSS_1.ldr

Cleanup Script: API_2_ENH_FACYBS_BSS_1.clr

6.2.4.15.3 Test procedure

Fill the SMS PP with TLV: Tag 02 Value 22 44 Tag 33, Length C4 Value 01 02 1 findAndCopyValue() with a null dstBuffer 2 dstOffset ≥ dstBuffer.length	ation
1 findAndCopyValue() with a null dstBuffer 2 dstOffset ≥ dstBuffer.length tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 5 dstCoffset = 5 dstCoffset = 0 dstBuffer.length = 5 dstCoffset = -1 dstLength = 1 dstCoffset = -1 dstLength = 0 dstBuffer.length = 5 dstCoffset = 0 dstBuffer.length = 5 dstCoffset = 0 dstCoffset = -1 dstLength = 0 dstCoffset = 3 dstCoffset = 0	
1 findAndCopyValue() with a null dstBuffer NullPointerException is thrown	
2 dstOffset ≥ dstBuffer.length tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 5 dstOffset = 5 dstLength = 1 dstDeffset = 5 dstLength = 1 dstDeffset = 0 dstBuffer.length = 5 dstLength = 1 dstLength = 1 dstLength = 1 dstLength = 5 dstOffset = 0 dstLength = 1 7 valueOffset ≥ Value Length tag = 06h, occurrence = 1 valueOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 1 8 valueOffset = 0 dstLength = 1 8 valueOffset = 0 dstLength = 1 9 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 1 10 valueOffset = 0 dstLength = 1 11 Select a TLV (tag 02h) 12 tag = 06h, occurrence = 1 valueOffset = 0 dstDeffset = 0 dstDeffs	
tag = 06h, occurrence = 1 valueOffset = 0 datBuffer.length = 5 datDtfset = 5 datDtfset = 5 datDtfset = 1 3	
tag = 06h, occurrence = 1 valueOffset = 0 datBuffer.length = 5 datDtfset = 5 datDtfset = 5 datDtfset = 1 3	
distDiffer 1 5 distOffset 5 distLength 1 3 distLength 2 1 distLength 1 distLength 2 distDiffset 3 distLength 2 distLength 2 distLength 3 distLength 3 distLength 3 distLength 4 distLength 3 distLength 4 distLength 4 distLength 4 distLength 4 distLength 5 distOffset 5 distOffset 5 distOffset 4 stLength 3 distLength 6 distLength 7 distLength	
dstDffset = 5" dstLength = 1 dstOffset < 0 ArrayIndexOutOfBoundsException n is thrown	
dstLength = 1 ArrayIndexOutOfBoundsException n is thrown	
dstBuffer length 5 dstOffset 0 dstBuffer length 1 dstLength	
dstBuffer.length = 5 dstLength > dstLength > dstBuffer.length dstBuffer length = 5 dstOffset = 0 dstLength = 5 dstOffset = 3 dstLength = 5 dstOffset = 3 dstLength = 5 dstOffset = 0 dstBuffer.length = 5 dstOffset = 0 dstBuffer.length = 5 dstOffset = 0 dstLength = 1 7	
dstLength = 1 4	
dstLength > dstBuffer.length dstBuffer.length dstBuffer.length = 5 dstOffset = 0 o dstLength = 6	
dstBuffer.length = 5 dstOffset = 0 dstLength = 6 5 dstOffset + dstLength > dstBuffer.length dstBuffer.length = 5 dstOffset = 3 dstLength = 3 6 dstBuffer.length = 5 dstOffset = 0 dstLength = -1 7 valueOffset ≥ Value Length tag = 06h, occurrence = 1 valueOffset = 6 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 8 valueOffset = 0 dstLength = 1 9 dstLength = 1 9 dstLength > Value length valueOffset = 0 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 9 dstLength > Value length valueOffset = 0 dstBuffer.length = 15 dstOffset = 0 dstLength = 7 10 valueOffset + dstLength > ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown ToolkitException.UNAVAILABLE ELEMENT is thrown ToolkitException.UNAVAILABLE ELEMENT is thrown. Result of findAndCopyValue() is 6 dstSuffer.length = 06 dstSuffer.length = 06 dstSufferst = 0 dstBuffer.length = 06 dstSufferst = 0 dstBuffer.length = 06 dstSufferst = 0 dstBuffer.length = 06	
dstOffset = 0 dstLength = 6 5	
dstLength = 6 dstOffset + dstLength > dstBuffer.length dstBuffer length = 5 dstOffset = 3 dstLength = 3	
Stoffset + dstLength > dstBuffer.length ArrayIndexOutOfBoundsException n is thrown	
dstDeffer length = 5 dstOffset = 3 dstLength = 3 dstLength = 3 dstLength = 3 dstLength = 5 dstOffset = 10 dstBuffer length = 5 dstOffset = 0 dstLength = -1 7	
dstOffset = 3 dstLength = 3 6 dstBuffer.length = 5 dstOffset = 0 dstLength = -1 7 valueOffset ≥ Value Length tag = 06h, occurrence = 1 valueOffset = 0 dstLength = 1 8 valueOffset = 0 dstLength = 1 8 valueOffset = -1 dstDuffset - length = 15 dstOffset = 0 dstLength = 1 9 dstLength > ValueOffset < 0 dstLength = 1 9 dstLength > ValueOffset = 0 dstLength = 1 10 dstDuffset = 0 dstLength = 7 10 valueOffset = 2 dstLength = 15 dstOffset = 0 dstLength = 7 10 valueOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 7 10 cstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 1 11 Select a TLV (tag 02h) findAndCopyValue() tag = 06h occurrence = 2 Call the getValueLength() method 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstLength = 06 dstDuffset = 0	
dstLength = 3 6	
dstBuffer.length = 5 dstOffset = 0 dstLength = -1 7	
dstBuffer.length = 5 dstOffset = 0 dstLength = -1 7	
dstLength = -1 7 valueOffset ≥ Value Length tag = 06h, occurrence = 1 valueOffset = 6 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 dstDeffset = 0 dstLength = 1 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 dstBuffer.length = 15 dstOffset = 0 dstLength > Value length valueOffset = 0 dstLength = 7 dstBuffer.length = 15 dstOffset = 0 dstLength = 7 dstBuffer.length = 15 dstOffset = 0 dstLength = 5 dstOffset = 0 dstDeffset =	
ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown valueOffset ≥ Value Length valueOffset = 6 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 8	
tag = 06h, occurrence = 1 valueOffset = 6 dstBuffer.length = 15 dstoffset = 0 dstLength = 1 8 valueOffset < 0 valueOffset = -1 dstBuffer.length = 15 dstoffset = 0 dstLength = 1 9 dstLength > Value length valueOffset = 0 dstLength = 15 dstoffset = 0 dstLength = 1 10 valueOffset + dstLength > ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown ToolkitException.UNAVAILABLE ELEMENT is thrown ToolkitException.UNAVAILABLE ELEMENT is thrown. ToolkitException.UNAVAILABLE ELEMENT is thrown. Result of findAndCopyValue() is 6 dstBuffer.length = 06	
valueOffset = 6 dstBuffer.length = 15 dstOffset = 0 dstLength = 1	
dstBuffer.length = 15 dstOffset = 0 dstLength = 1 8	
dstOffset = 0 dstLength = 1 8	
Solution	
valueOffset = -1 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 9	
dstBuffer.length = 15 dstOffset = 0 dstLength > Value length valueOffset = 0 dstBuffer.length = 15 dstoffset = 0 dstBuffer.length = 15 dstoffset = 0 dstLength = 7 10 valueOffset + dstLength > Text String length valueOffset = 2 dstBuffer.length = 15 dstoffset = 0 dstLength = 5 11 Select a TLV (tag 02h) findAndCopyValue() tag = 06h occurrence = 2 Call the getValueLength() method 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0	
dstOffset = 0 dstLength = 1 9	
dstLength = 1 g	
9	
dstBuffer.length = 15 dstOffset = 0 dstLength = 7 10	
dstoffset = 0 dstLength = 7 10 valueOffset + dstLength > Text String length valueOffset = 2 dstBuffer.length = 15 dstoffset = 0 dstLength = 5 11 Select a TLV (tag 02h) tag = 06h occurrence = 2 Call the getValueLength() method 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstBuffer.length = 06 dstOffset = 0 dstBuffer.length = 06 dstOffset = 0 dstBuffer.length = 06 dstOffset = 0	
dstLength = 7	
ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown	
valueOffset = 2 dstBuffer.length = 15 dstOffset = 0 dstLength = 5 11 Select a TLV (tag 02h) findAndCopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown tag = 06h occurrence = 2 ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 Result of findAndCopyValue() is 6 dstBuffer.length = 06 dstOffset = 0 6	
dstBuffer.length = 15 dstOffset = 0 dstLength = 5 11	
dstLength = 5 11	
11 Select a TLV (tag 02h) findAndCopyValue() tag = 06h occurrence = 2 Call the getValueLength() method ToolkitException.UNAVAILABLE _ELEMENT is thrown ToolkitException.UNAVAILABLE _ELEMENT is thrown. 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0	
findAndCopyValue() tag = 06h occurrence = 2 Call the getValueLength() method ToolkitException.UNAVAILABLE _ELEMENT is thrown ToolkitException.UNAVAILABLE _ELEMENT is thrown. 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0	
tag = 06h occurrence = 2 Call the getValueLength() method ToolkitException.UNAVAILABLE _ELEMENT is thrown. 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0	
Call the getValueLength() method ToolkitException.UNAVAILABLE ELEMENT is thrown.	
Call the getValueLength() method ToolkitException.UNAVAILABLE _ELEMENT is thrown. 12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0 ToolkitException.UNAVAILABLE _ELEMENT is thrown. Result of findAndCopyValue() is 6	
ELEMENT is thrown. 12	
12 Successful call tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0 Result of findAndCopyValue() is 6	
<pre>tag = 06h, occurrence = 1 valueOffset = 0 dstBuffer.length = 06 dstOffset = 0</pre>	
<pre>valueOffset = 0 dstBuffer.length = 06 dstOffset = 0</pre>	
dstOffset = 0	
dstLength = 06	
13 Compare buffer Result is 00h	
buffer = 81 11 22 33 44 F5	
14 initialise dstBuffer	
dstBuffer = 55 55 55	
Successful call Result of findAndCopyValue () is	
tag = 06h, occurrence = 1 0007	
valueOffset = 2	

	dstBuffer.length = 12		
	dstOffset = 3		
	dstLength = 04		
15	Compare buffer	Result is 00h	
15		Result is our	
	buffer =		
4.0	55 55 55 22 33 44 F5 55 55 55 55 55	D 1: (6: 14 10 1/1 0):	
16	Successful call	Result of findAndCopyValue() is	
	tag = 02h, occurrence = 1	0002	
	valueOffset = 0		
	dstBuffer.length = 12		
	dstOffset = 0		
	dstLength = 2		
17	Compare buffer	Result is 00h	
	buffer = 83 81 55 55		
18	Successful call	Result of findAndCopyValue() is	
	tag = 02h, occurrence = 2	0002	
	valueOffset = 0		
	dstBuffer.length = 12		
	dstOffset = 0		
	dstLength = 2		
19	Compare buffer	Result is 00h	
	buffer = 22 44 55 55		
20	Successful call (with tag 82h)	Result of findAndCopyValue () is	
	tag = 82h	0002	
	occurrence = 1		
	3 055 1 0		
	<pre>valueOffset = 0</pre>		
	dstBuffer.length = 12		
	dstBuffer.length = 12		
21	<pre>dstBuffer.length = 12 dstOffset = 0</pre>	Result is 00h	
21	<pre>dstBuffer.length = 12 dstOffset = 0 dstLength = 02</pre>	Result is 00h	
21	<pre>dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer</pre>		
	<pre>dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55</pre>	Result is 00h Result of findAndCopyValue () is 0002	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h)	Result of findAndCopyValue () is	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h	Result of findAndCopyValue () is	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2	Result of findAndCopyValue () is	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0	Result of findAndCopyValue () is	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02	Result of findAndCopyValue () is 0002	
	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer	Result of findAndCopyValue () is	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02	Result of findAndCopyValue () is 0002	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer	Result of findAndCopyValue () is 0002 Result is 00h	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer Buffer = 22 44 55 55 Successful call, findAndCopyValue with	Result of findAndCopyValue () is 0002 Result is 00h Result of findAndCopyValue () is	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer Buffer = 22 44 55 55 Successful call, findAndCopyValue with length = 0	Result of findAndCopyValue () is 0002 Result is 00h	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer Buffer = 22 44 55 55 Successful call, findAndCopyValue with length = 0 DstBuffer.length = 12	Result of findAndCopyValue () is 0002 Result is 00h Result of findAndCopyValue () is	
22	dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer buffer = 83 81 55 55 Successful call (with tag 82h) tag = 82h occurrence = 2 valueOffset = 0 dstBuffer.length = 12 dstOffset = 0 dstLength = 02 Compare buffer Buffer = 22 44 55 55 Successful call, findAndCopyValue with length = 0	Result of findAndCopyValue () is 0002 Result is 00h Result of findAndCopyValue () is	

6.2.4.15.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18, 24
N4	20, 21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for EnvelopeHandler

6.2.4.16 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference: API_2_ENH_FACRB_BS

6.2.4.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical returns 0.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.16.3 Test Suite files

Test Script: API_2_ENH_FACRB_BS_1.scr

Test Applet: API_2_ENH_FACRB_BS_1.java

Load Script: API_2_ENH_FACRB_BS_1.ldr

Cleanup Script: API_2_ENH_FACRB_BS_1.clr

6.2.4.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Fill the SMS PP with TLV: Tag 02 Value 22 44		
	Tag 33, Length C4 Value 01 02		
1	findAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
2	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 06h	n is thrown	
	compareBuffer.length = 12		
	compareOffset = 12		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 12	n is thrown	
	compareOffset = -1		
4	length > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 05	n is thrown	
	compareOffset = 0		
5	compareOffset + length >	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length	n is thrown	
	compareBuffer.length = 12		
	compareOffset = 7		
6	Select a TLV (tag 02h)		

	<pre>findAndCompareValue() tag = 03h</pre>	ToolkitException.UNAVAILABLE ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE _ELEMENT is thrown.	
7	Initialise compareBuffer compareBuffer = 81 11 22 33 44 F5		
	Compare buffers	Result is 00h	
	tag = 06h		
_	compareOffset = 0	Describle 00	
8	Verify current TLV getValueLength()	Result is 06	
9	Initialise compareBuffer		
3	compareBuffer = 81 11 22 33 44 F4		
	Compare buffers with same parameters	Result is +1	
10	Initialise compareBuffer		
	compareBuffer = 81 11 22 33 44 F6		
	Compare buffers with same parameters	Result is -1	
11	Initialise compareBuffer		
	compareBuffer =		
	55 55 81 11 22 33 44 F5 55 55 55 55		
	Compare buffers compareOffset = 2	Result is 00h	
12	Initialise compareBuffer		
12	compareBuffer =		
	55 55 83 81 55 55 55 55 55 55 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
13	Initialise compareBuffer		
	compareBuffer =		
	55 55 83 80 55 55 55 55 55 55 55 55 Compare buffers	Result is +1	
	compareOffset = 2	Result is +1	
14	Initialise compareBuffer		
	compareBuffer =		
	55 55 83 82 55 55 55 55 55 55 55		
	Compare buffers	Result is -1	
4 =	compareOffset = 2		
15	Initialise compareBuffer compareBuffer =		
	83 81 55 55 55 55 55 55 55 55 55		
	Successful call (with tag 02h)	Result is 00h	
	tag = 02h		
	compareBuffer.length = 12		
	compareOffset = 0		
16	Initialise compareBuffer		
	CompareBuffer = 01 02 C4	Dogult in OOh	
	Successful call (with tag B3h) Tag = B3h	Result is 00h	
	CompareBuffer.length = C4		
	CompareOffset = 0		

6.2.4.16.4 Test Coverage

CRR number	Test case number	
N1	6	
N2	8	
N3	7, 11, 12	
N4	9, 13	
N5	10, 14	
N6	15, 16	
P1	1	
P2	2, 3, 4, 5	
C1	Does not apply for	
	EnvelopeHandler	

6.2.4.17 Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference: API_2_ENH_FACRBBS_BSS

6.2.4.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical 0 is returned.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

CRRP4: if an input parameter is not valid (e.g. occurence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.4.17.3 Test Suite files

Test Script: API_2_ENH_FACRBBS_BSS_1.scr
Test Applet: API_2_ENH_FACRBBS_BSS_1.java
Load Script: API_2_ENH_FACRBBS_BSS_1.ldr
Cleanup Script: API_2_ENH_FACRBBS_BSS_1.clr

6.2.4.17.3 Test procedure

Fill the SMS PP with TV. Tag 02 Value 22 44 Tag 32, counger Collable 91 02 findAncCompareBuffer compareBuffer compareBuffer compareBuffer March MullPointerException is thrown compareBuffer MullPointerException ArrayIndexOutOlBoundsException is thrown compareBuffer Length = 6 compareBuffer in thrown in thrown compareBuffer compareBuffer in thrown in thro	lal	Description	API Expectation	ADDII Expectation
Tag 33, Length C4 Value 01 02 1	ld		API Expectation	APDU Expectation
compareBuffer t ag = 06h, occurrence = 1 valueOffset > 0 compareBuffer comp		Tag 33, Length C4 Value 01 02		
cag = 06h, occurrence = 1 valueOffset < 0 compareOffset = 0	1		NullPointerException is thrown	
cag = 06h, occurrence = 1 valueOffset < 0 compareOffset = 0	2	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
compareOffset = 6 compareOffset = 1 3				
compareoffset = 6 compareoffset < 0 compareoffset < 0 compareoffset = 1 valueoffset = 1 compareoffset = 1 compareoffset = 1 compareoffset = 1 compareoffset = 0 compareoffset		valueOffset = 0		
compareLength = 1 compareOffset < 0		<pre>compareBuffer.length = 6</pre>		
CompareDiffect OccupareOffset Occu		-		
compareJuffer Jength = 6 compareJength = 1 compareJength = 0 compareJength = 1 compareJeng				
compareLength = 1 compareLength = 1 compareLength = 1 compareDeffset = 0 compareDeffset = 3 compareDeffset = 3 compareDeffset = 3 compareDeffset = 3 compareDeffset = 0 compareDeffset =	3		ArrayIndexOutOfBoundsExceptio	
d compareLength > compareBuffer.length d compareBuffer length = 5 compareOffset = 0 compareDeffset > compareOffset + compareLength compareBuffer length = 5 compareOffset = 3 compareLength = 3 6			n is thrown	
CompareBuffer CompareBuffe		-		
compareDiffert = 0 compareDiffert = 0 compareDiffert = 0 compareDiffert = 0 compareDiffert + compareDiffert comp	4		A many day day (O) stOfD a una da Cuca matic	
compareOffset = 0 compareLength 5 compareBuffer.length compareBuffer.length sompareBuffer.length sompareLength	4			
compareLength = 6		-	n is thrown	
CompareOffset + compareLength				
ScompareBuffer.length = 5	5		ArrayIndexOutOfBoundsExceptio	
compareBuffer: length = 5 compareLength = 3 compareLength = 3 compareLength = 3 6			· · · · · · · · · · · · · · · · · · ·	
compareLength = 3 compareLength < 0 compareLength = 5 compareLength = 5 compareDeffect = 0 compareLength = 1 compareDeffect = 0 compareDeff				
CompareBuffer Length 5 CompareBuffer 11 Length 12 Length 12 Length 13 CompareBuffer 14 Length 14 Le				
compareButfer length = 5 compareLength = -1				
compareOffset = 0 compareLength = -1	6			
compareLength = -1			n is thrown	
ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown ToolkitException.OUT_OF_TLV_BOUNDARIES is thr		-		
tag = 06h, occurrence = 1 valueOffset = 6 compareBuffer, length = 15 compareDeffset = 0 valueOffset = -1 compareDeffset = 0 compareLength = 1 9			T 11:15 1: 01 T 05 T1 1/	
valueOffset = 6 compareButfer,length = 15 compareOffset = 0 compareIntfer,length = 15 compareDest = -1 compareButfer,length = 15 compareButfer,length = 15 compareButfer,length = 15 compareButfer,length = 15 compareLength = 1 9	/			
compareBuffer.length = 15 compareOffset = 0 valueOffset = -1 compareSuffer.length = 15 compareOffset = 0 valueOffset = -1 compareOffset = 0 compareIngth = 15 compareOffset = 0 compareLength = 1 CompareLength = 1 CompareLength > Value length valueOffset = 0 compareOffset = 0 compa		-	BOUNDARIES IS INFOWN	
compareDefiset = 0 compareDefiset < 0 valueOffset = -1 compareBuffer length = 15 compareDefiset = 0				
CompareLength = 1				
valueOffset = -1 compareBuffer.length = 15 compareOffset = 0 compareLength > Value length valueOffset = 0 compareLength > Value length valueOffset = 0 compareLength = 15 compareOffset = 0 comp		-		
compareBuffer.length = 15 compareDeffset = 0 compareLength > Value length valueOffset = 0 compareBuffer.length = 15 compareDeffset = 0 compareBuffer.length = 15 compareDeffset = 0 compareDeffset = 81 11 22 33 44 F5 Initialise compareBuffer compareDeffset = 0 compareDeffset = 81 11 22 33 44 F4 Verify current TLV getValueDeffset = 81 11 22 33 44 F4 CompareDeffset = 81	8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
compareDeffset = 0 compareLength > Value length valueOffset = 0 compareDeffer.length = 15 compareDeffer.length = 15 compareDeffer.length = 15 compareDeffer.length = 7 10			BOUNDARIES is thrown	
compareLength = 1				
CompareLength > Value length valueOffset = 0 compareOffset = 0 compareOffset = 15 compareOffset = 0 compareOffset = 2 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 5 Initialise compareOffset = 0 compa				
valueOffset = 0 compareBuffer.length = 15 compareOffset = 0 compareLength = 7 10	0		ToolkitEvention OUT OF TIV	
compareBuffer.length = 15 compareIngth = 7 10	9			
compareOffset = 0 compareLength = 7 10			BOONDAINEO IS UNOWIT	
ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown		-		
valueOffset = 2 compareBuffer.length = 15 compareOffset = 0 compareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 CompareOffset = 81 11 22 33 44 F4 Result is 0006 Result is +1 Result is +1				
compareDuffer.length = 15 compareOffset = 0 compareLength = 5 11	10			
compareOffset = 0 compareLength = 5 11			BOUNDARIES is thrown	
CompareLength = 5		-		
Invalid parameter ToolkitException.BAD_INPUT_P ARAMETER is thrown				
Occurrence = 0	11		ToolkitException BAD_INPLIT_P	
Select a TLV (tag 02h) findAndCompareValue() tag = 06h occurrence = 2 ToolkitException.UNAVAILABLE _ELEMENT is thrown	' '	<u>-</u>		
findAndCompareValue() tag = 06h occurrence = 2 Call the getValueLength() method Initialise compareBuffer compareBuffer = 81 11 22 33 44 F5 findAndCompareValue() tag = 06h, occurrence = 1 valueOffset = 0 compareDength = 6 Verify current TLV getValueLength() Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is 00h Result is 00h Result is 0006 Result is 0006	12	Select a TLV (tag 02h)		
tag = 06h occurrence = 2 Call the getValueLength() method Initialise compareBuffer compareBuffer = 81 11 22 33 44 F5 findAndCompareValue() tag = 06h, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 6 Verify current TLV getValueLength() Is initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is 0006 Result is 0006 Result is 0006 Result is 0006	<u> </u>		ToolkitException,UNAVAILABLE	
Call the getValueLength() method				
		occurrence = 2	_	
Initialise compareBuffer compareBuffer = 81 11 22 33 44 F5 findAndCompareValue() tag = 06h, occurrence = 1 valueOffset = 0 compareLength = 6 14 Verify current TLV getValueLength() 15 Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is 0006 Result is 0006 Result is 0006		Call the getValueLength() method	ToolkitException.UNAVAILABLE	
compareBuffer = 81 11 22 33 44 F5 findAndCompareValue() tag = 06h, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 6 14 Verify current TLV getValueLength() 15 Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is 0006			_ELEMENT is thrown.	
findAndCompareValue() tag = 06h, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 6 14	13			
tag = 06h, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 6 14				
valueOffset = 0 compareOffset = 0 compareLength = 6 14			Result is 00h	
compareOffset = 0 compareLength = 6 14		-		
compareLength = 6 14				
14 Verify current TLV getValueLength() 15 Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is 0006 Result is 0006 Result is +1		-		
getValueLength() 15 Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters 16 Initialise compareBuffer	14		Result is 0006	
15 Initialise compareBuffer compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is +1 16 Initialise compareBuffer		=		
compareBuffer = 81 11 22 33 44 F4 Compare buffers with same parameters Result is +1 16 Initialise compareBuffer	15	Initialise compareBuffer		
16 Initialise compareBuffer		compareBuffer = 81 11 22 33 44 F4		
			Result is +1	
compareBuffer = 81 11 22 33 44 F6	16			
		compareBuffer = 81 11 22 33 44 F6	l l	

	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer	1.Count to 1	
''	compareBuffer =		
	55 55 55 22 33 44 F5 55 55 55 55		
	Compare buffers	Result is 00h	
	valueOffset = 2		
	compareOffset = 3		
	compareLength = 4		
18	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 22 33 45 F5 55 55 55 55		
<u> </u>	Compare buffers with same parameters	Result is -1	
19	Initialise compareBuffer		
	compareBuffer =		
<u> </u>	55 55 55 22 33 43 F5 55 55 55 55	Decultie 14	
	Compare buffers with same parameters	Result is +1	
20	Initialise compareBuffer		
	compareBuffer = 83 81 55 55 55 55 55 55 55		
\vdash	findAndCompareValue()	Result is 00h	
	tag = 02h, occurrence = 1	IVESUIT IS ONLI	
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 2		
21	Initialise compareBuffer		
	compareBuffer =		
	22 44 55 55 55 55 55 55 55 55 55		
	findAndCompareValue()	Result is 00h	
	tag = 02h, occurrence = 2		
	valueOffset = 0		
	compared ength = 2		
22	compareLength = 2 Initialise compareBuffer		
~~	compareBuffer =		
	22 45 55 55 55 55 55 55 55 55 55		
	findAndCompareValue()	Result is -1	
	tag = 02h, occurrence = 2		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 2		
23	Initialise compareBuffer		
	compareBuffer =		
<u> </u>	83 81 55 55 55 55 55 55 55 55 55 55 55 55 55	Deput is 00b	
	Successful call (with tag 02h) tag = 02h, occurrence = 1	Result is 00h	
	valueOffset = 0		
	compareBuffer.length = 12		
	compareOffset = 0		
L	compareLength = 2		
24	Initialise compareBuffer		
	compareBuffer = 01 02 C4		
	Successful call (with tag B3h)	Result is 00h	
	tag = B3h, occurrence = 1		
	valueOffset = 0		
	compareBuffer.length = 00C4		
	<pre>compareOffset = 0 compareLength = 00C4</pre>		
25	Successful call, findAndCompareValue with	Result of findAndCompareValue()	
25	length =0	is 00h	
	DstBuffer.length = C4	13 0011	
	DstOffset = C4		
	DstLength = 0		

6.2.4.17.4 Test Coverage

CRR number	Test case number	
N1	12	
N2	14	
N3	13, 17, 20, 21, 25	
N4	19, 15	
N5	16, 18, 22	
N6	23, 24	
P1	1	
P2	2, 3, 4, 5, 6	
P3	7, 8, 9, 10	
P4	11	
C1	Does not apply for EnvelopeHandler	

6.2.5 Class EnvelopeResponseHandler

6.2.5.1 Method getTheHandler

Test Area Reference: API_2_ERH_GTHD

6.2.5.1.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the single system instance of the EnvelopeResponseHandler class.

CRRN2: The EnvelopeResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12])

Parameter errors

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

CRRC2: After the first invocation of the ProactiveHandler.send method the EnvelopeResponseHandler is no more available

6.2.5.1.2 Test suite files

Test Script: API_2_ERH_GTHD_1.scr

Test Applet: API_2_ERH_GTHD_1.java

Load Script: API_2_ERH_GTHD_1.ldr

Cleanup Script: API_2_ERH_GTHD_1.clr

6.2.5.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler twice	The returned objects shall be the	
		same	
2	Verify that getTheHandler returns an	The reference returned shall be an	
	EnvelopeHandler	EnvelopeResponseHandler	
	getTheHandler	(checkcast)	
3	Verify the returned value is not null	The reference returned shall not be	
	getTheHandler	null.	
4	getTheHandler, then send a proactive	ToolkitException	
	command, and then, appendTLV	HANDLER_NOT_AVAILABLE is	
		thrown	

6.2.5.1.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	To be checked in
	Framework tests and
	insert here cross
	reference
C1	To be checked in
	Framework tests and
	insert here cross
	reference
C2	4

6.2.5.2 Method post

Test Area Reference: API_2_ERH_POSTB

6.2.5.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: When the method is called, the toolkit applet can continue it's processing (e.g. prepare a proactive command).

CRRN2: The byte statusType is SW1 of the status.

CRRN3: If the send method is called after a post method, the posted data are the first sent to the ME.

CRRN4: The SIM Toolkit Framework shall take the optional Application Data posted by the triggered toolkit applet if present, secure and send the response packet the SIM Toolkit Framework will return the response APDU defined by the toolkit applet.

Parameter error

No requirements

Context errors

CRRC1: The method shall thrown ToolkitException (HANDLER_NOT_AVAILABLE) if the handler is busy.

6.2.5.2.2 Test suite files

Specific triggering: Call control

Test Script: API_2_ERH_POSTB_1.scr

Test Applet: API_2_ERH_POSTB_1.java

Load Script: API_2_ERH_POSTB_1.ldr

Cleanup Script: API_2_ERH_POSTB_1.clr

6.2.5.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler and then post		9000
	(the handler is empty)		
2	Fill the handler (appendTLV to have bytes		9FFD data are retrieved with GET
	in it)and then post data with status		RESPONSE command
	9F		
3	Verify that after a post the handler is no	ToolkitException	
	more available	HANDLER_NOT_AVAILABL	
	appendTLV, post and then appendTLV	E is thrown on the second	
		appendTLV	
4	construct the response (appendTLV with		9E12 and posted data retrieved by a
	0x10 data) and post it with status 9E and		GET RESPONSE with status 9113
	then send a display text		and display text retrieved by a FETCH
5	Verify that it is possible to send a proactive		91 13 and display text is retrieved by
	command after a post		a FETCH
	getTheHandler and post , then send a		
	display text		
6	Verify it is not possible to post after a	ToolkitException	
	proactive command	HANDLER_NOT_AVAILABL	
	getTheHandler, appendTLV, send a	E is thrown	
	display text, post.		
7	Verify that the handler is no more available		9E12 and posted data retrieved by a
	after a post	HANDLER_NOT_AVAILABL	GET RESPONSE
	getTheHandler, appendTLV, post with	E is thrown	
	status 9E, post with status 9F		

6.2.5.2.4 Test Coverage

CRR number	Test case number	
N1	3, 4, 7	
N2	1, 2, 4, 7	
N3	4, 5	
N4	To be checked in	
	Framework tests and	
	insert here cross	
	reference	
C1	6	

6.2.5.3 Method postAsBERTLV

Test Area Reference: API_2_ERH_POSTBB

6.2.5.3.1 Conformance Requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_void} \begin{array}{c} \text{public void postAsBERTLV(byte statusType,} \\ & \text{byte tag)} \\ & \text{throws ToolkitException} \end{array}$

Normal execution

CRRN1: When the method is called, the toolkit applet can continue it's processing (e.g. prepare a proactive command) the SIM Toolkit Framework will return the response APDU defined by the toolkit applet.

CRRN2: The byte statusType is SW1 of the status

CRRN3: If the send method is called after a postAsBERTLV method, the posted data are the first sent to the ME.

CRRN4: The byte tag is the BER Tag at the beginning of the simple TLV list.

Parameter errors

No requirements

Context errors

 $CRRC1: The \ method \ shall \ thrown \ ToolkitException \ (HANDLER_NOT_AVAILABLE) \ if \ the \ handler \ is \ busy.$

6.2.5.3.2 Test suite files

Specific triggering: Call control

Test Script: API_2_ERH_POSTBB_1.scr

Test Applet: API_2_ERH_POSTBB_1.java

Load Script: API_2_ERH_POSTBB_1.ldr

Cleanup Script: API_2_ERH_POSTBB_1.clr

6.2.5.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler and then postAsBERTLV	/ II - Exposition	9F02 data are retrieved with
'	(the handler is empty)		GET RESPONSE command.
			the tag shall be 33 and the
			length is 00
2	Fill the handler and then postAsBERTLV		9FFF data are retrieved with
-	the data with status 9F, and tag 33		GET RESPONSE command,
	,		the tag shall be 33
3	appendTLV, postAsBERTLV and then	ToolkitException	<u> </u>
	appendTLV	HANDLER_NOT_AVAILABLE is	
		thrown on the second appendTLV	
4	construct the response (appendTLV with		9E14 and posted data
	0x10 data) and postAsBERTLV it with		retrieved by a GET
	status 9E, tag 75 and then send a display		RESPONSE the tag shall be
	text		75 with status 9113 and
			display text retrieved by a
			FETCH
5	getTheHandler and postAsBERTLV, then		9E02 and posted data
	send a display text		retrieved by a GET
			RESPONSE the tag 33 (and
			the length 00) with status
			9113 and display text is
	V K K L	T 1125	retrieved by a FETCH
6	Verify it is not possible to postAsBERTLV	ToolkitException	
	after a proactive command getTheHandler, appendTLV, send a	HANDLER_NOT_AVAILABLE is	
	display text, postAsBERTLV.	thrown on the postAsBERTLV	
7	Verify that the handler is no more available	ToolkitException	9E14 and posted data
	after a postAsBERTLV	HANDLER_NOT_AVAILABLE is	retrieved by a GET
	getTheHandler, appendTLV(with data	thrown on the second postAsBERTLV	RESPONSE the tag shall be
	length = 0x10, postAsBERTLV with status		56 with status 9000
	9E, tag 56, postAsBERTLV with status		
	9F, tag 28		

6.2.5.3.4 Test Coverage

CRR number	Test case number
N1	3, 4, 7
N2	1, 2, 4, 7
N3	4, 5
N4	2, 4, 7
C1	6

6.2.5.4 Method getLength

Test Area Reference: API_2_ERH_GLEN

6.2.5.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public short getLength()

throws ToolkitException

Normal execution

CRRN1: returns the length in bytes of the TLV list.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.5.4.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GLEN_1.scr

Test Applet: API_2_ERH_GLEN_1.java

Load Script: API_2_ERH_GLEN_1.ldr

Cleanup Script: API_2_ERH_GLEN_1.clr

6.2.5.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear the handler	Result of getLength() is 0	
	getLength()		
2	appendTLV with length of 7	Result of getLength() is 9	
	getLength()		
3	Clear the handler and appendTLV with Length	Result of getLength() is 253	
	of 250		
	getLength()		
4	Build a 7Fh Envelope response handler	Result of getLength() is 81h	
	getLength()		
5	Build a 80h Envelope response handler	Result of getLength() is 83h	
	getLength()		

Note: Test case 3 is limited to 253 and not 256 because the current 03.19 [7] is not clear enough on this point. So this test allows the two possible implementations.

6.2.5.4.4 Test Coverage

CRR number	Test case number	
N1	N1 1, 2, 3, 4, 5	
C1	Does not apply for	
	Envelope response	
	handler	

6.2.5.5 Method copy

Test Area Reference: API_2_ERH_COPY_BSS

6.2.5.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: copies the simple TLV list contained in the handler to the destination byte array.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null a NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.

CRRP3: if dstLength is greater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.5.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_COPY_BSS_1.scr
Test Applet: API_2_ERH_COPY_BSS_1.java
Load Script: API_2_ERH_COPY_BSS_1.ldr
Cleanup Script: API_2_ERH_COPY_BSS_1.clr

6.2.5.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with value length of 7		
	NULL as parameter to dstBuffer	NullPointerException is thrown	
2			
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 5		
3	dstLength = 1 dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
٥	dstBuffer.length = 5	n is thrown	
	dstOffset = -1	II is tillowii	
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
5	dstLength = 6 dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
3	dstBuffer.length = 5	n is thrown	
	dstOffset = 3	II is tillowii	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
7	dstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_	
′	dstBuffer.length = 10	BOUNDARIES is thrown	
	dstOffset = 0	BOOMB/ INTEG IS THOWN	
	dstLength = 10		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 9	
	dstBuffer.length = 9		
	<pre>dstOffset = 0 dstLength = 9</pre>		
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 12	
. •	dstBuffer.length = 15		
	dstOffset = 3		
	dstLength = 9		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 9	
	<pre>dstBuffer.length = 15 dstOffset = 3</pre>		
	dstLength = 6		
13	Compare the whole buffer	Result of arrayCompare() is 0	
14	Successful call, copy with length =0	Result of copy() is 15	
	dstBuffer.length = 15		
	dstOffset = 15		
	dstLength = 0		

6.2.5.5.4 Test Coverage

CRR number	Test case number	
N1	9, 11, 13	
N2	8, 10, 12, 14	
P1	1	
P2	2, 3, 4, 5, 6	
P3	7	
C1	Does not apply for	
	Envelope response	
	handler	

6.2.5.6 Method findTLV

Test Area Reference: API_2_ERH_FINDBB

6.2.5.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.

CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.

CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.

CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.

CRRN5: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.6.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FINDBB_1.scr

Test Applet: API_2_ERH_FINDBB_1.java

Load Script: API_2_ERH_FINDBB_1.ldr

Cleanup Script: API_2_ERH_FINDBB_1.clr

6.2.5.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	append the handler with TLVs:		•
	81 03 11 22 33		
	82 02 99 77		
	Invalid input parameter	ToolkitException.BAD_INPUT_P	
	Occurrence = 0	ARAMETER is thrown	
2			
	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 01h		
	Occurrence = 1		
3	Call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
	Tag = 02h		
	Occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	

	m 0.21		
	Tag = 03h		
	Occurrence = 1		
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
1	J. J	ELEMENT is thrown.	
	0 1 1 1 1 1 1 1		
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	Tag = 01h		
	Occurrence = 2		
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
	Jan. 11.1 gerrana – 11. g. 11. () 11. 11. 11. 11. 11. 11. 11. 11. 11. 11	ELEMENT is thrown.	
4.0	A I . TI . /	_LLLIVILIAT IS UITOWIT.	
10	Append a TLV with tag=02h		
	Search the TLV	Result is	
	Tag = 02h	TLV FOUND CR NOT SET	
	Occurrence = 2		
11	Append a TLV with tag=04h		
	Search the TLV	Result is	
	Tag = 04h	TLV FOUND CR NOT SET	
	Occurrence = 1	12V_1 00NB_0N_NO1_021	
12	Search tag 81h	Result is TLV_FOUND_CR_SET	
	Tag = 81h		
	Occurrence = 1		
13	Search tag 84h	Result is	
13	Tag = 84h		
	3	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1		

6.2.5.6.4 Test Coverage

CRR number	Test case number
N1	3, 5
N2	2, 4
N3	10, 11
N4	6, 7,8, 9
N5	12, 13
P1	1
C1	Does not apply for
	Envelope response
	handler

6.2.5.7 Method getValueLength

Test Area Reference: API_2_ERH_GVLE

6.2.5.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.7.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GVLE_1.scr

Test Applet: API_2_ERH_GVLE_1.java

Load Script: API_2_ERH_GVLE_1.ldr

Cleanup Script: API_2_ERH_GVLE_1.clr

6.2.5.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1		AFTEXPECIATION	APDO Expectation
1	appendTLV 02 02 02 02 findTLV with TAG 03		
		TH:4FtiLINIAN/AH ADI F	
	getValueLength()	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown	
2	appendTLV with TAG 0D and length 00		
	Search TLV 0Dh		
	getValueLength()	Result is 00h	
3	Clear the handler and append TLV with TAG		
	0D and length 02		
	Search TLV 0Dh		
	getValueLength()	Result is 02h	
4	Clear the handler and append TLV with TAG		
	0D and length 0x7F		
	Search TLV 0Dh		
	getValueLength()	Result is 7Fh	
5	Clear the handler and append TLV with TAG		
	0D and length 0x80		
	Search TLV 0Dh		
	getValueLength()	Result is 80h	
6	Clear the handler and append TLV with TAG		
	0D and length 0xF1		
	Search TLV 0Dh		
	getValueLength()	Result is F1h	

6.2.5.7.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4, 5, 6
C1	Does not apply for
	EnvelopeResponseHandl
	er
C2	1

6.2.5.8 Method getValueByte

Test Area Reference: API_2_ERH_GVBYS

6.2.5.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_byte} \mbox{public byte getValueByte(short valueOffset)} \\ \mbox{throws ToolkitException}$

Normal execution

CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

Parameter errors

CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.8.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_GVBYS_1.scr

Test Applet: API_2_ERH_GVBYS_1.java

Load Script: API_2_ERH_GVBYS_1.ldr

Cleanup Script: API_2_ERH_GVBYS_1.clr

6.2.5.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV 82 02 81 82, appendTLV 81 03 11		
	22 FE		
	findTLV with TAG 03		
	getValueByte(0)	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
2	Search TLV 01h		
	<pre>getValueByte(3)</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
3	Search TLV 01h		
	getValueByte(2)	Result is FEh	
4	Search TLV 02h		
	getValueByte(0)	Result is 81h	
5	appendTLV with TAG 0D, Length 0x7E, Value: 00, 01,, 7D		
	getValueByte(7D)	Result is 7Dh	
6	clear the handler, appendTLV with TAG 0D, Length 0x80, Value: 00, 01,, 7F		
	getValueByte(7E)	Result is 7Eh	
7	getValueByte(7F)	Result is 7Fh	
8	clear the handler, appendTLV with TAG 0D, Length 0xF1, Value: 00, 01,, F0		
	getValueByte(F0)	Result is F0h	

6.2.5.8.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5, 6, 7, 8
P1	2
C1	Does not apply for EnvelopeResponseHandl
	er
C2	1

6.2.5.9 Method copyValue

Test Area Reference: API_2_ERH_CPYVS_BSS

6.2.5.9.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

Normal execution

CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.9.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CPYVS_BSS_1.scr
Test Applet: API_2_ERH_CPYVS_BSS_1.java

Load Script: API_2_ERH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_ERH_CPYVS_BSS_1.clr

6.2.5.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with TAG: 0D and length 16	AFTEXPECTATION	AF DO Expectation
'	Select Text String TLV		
	copyValue() with a null dstBuffer	Null Dointer Evention in thrown	
2	copyvarue() with a null ustbuller	NullPointerException is thrown	
	detOffeet > detDiiffee leveth	A recyle doy Out Of Doy and Eventio	
	dstOffset ≥ dstBuffer.length dstBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 5	II IS UIIOWII	
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1		
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 0</pre>	n is thrown	
	dstlength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
	aschengen – 1		
7	clear the handler, appendTLV with TAG: 0D		
'	and length 6		
	Select Text String TLV		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 6	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
_	dstLength = 1	Tablitana dia OUT OF TIV	
8	<pre>valueOffset < 0 valueOffset = -1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOUNDARIES IS INIOWN	
	dstOffset = 0		
	dstLength = 1		
9	dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstlength = 7		
10	valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 2	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
11	dstLength = 5 Initialise the handler		
11	copyValue()	ToolkitException.UNAVAILABLE	
	- COP1 * GIAC ()	ELEMENT is thrown	
12	clear the handler, appendTLV with TAG: 0D	LLLIVILIVI IS UITOWII	
'-	and value: 04 00 01 0F		
	Select Text String TLV		
	Successful call	Result of copyValue() is 17	
	<pre>valueOffset = 0</pre>		
	dstBuffer.length = 17		
	dstOffset = 0		
13	dstLength = 17 Compare buffer	Result is 00h	
13	buffer = 04 00 01 0F	IVESUIT IS ON!	
	2011 - 01 00 01 m 0F		
14	initialise dstBuffer		
' '	dstBuffer = 55 55 55		
	Successful call	Result of copyValue() is 15	
	<pre>valueOffset = 2</pre>	, , , ,	
	dstBuffer.length = 20		
	<pre>dstOffset = 3 dstLength = 12</pre>		
l	uschength = 12	ı	

15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
16	Successful call, copyValue with length =0	Result of copyValue() is 20	
	dstBuffer.length = 20		
	dstOffset = 20		
	dstLength = 0		

6.2.5.9.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14, 16
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	EnvelopeResponseHandl
	er
C2	11

6.2.5.10 Method compareValue

Test Area Reference: API_2_ERH_CPRVS_BSS

6.2.5.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Compares the last found TLV element with a buffer:

CRRN1: returns 0 if identical.

CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.

CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.5.10.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CPRVS_BSS_1.scr

Test Applet: API_2_ERH_CPRVS_BSS_1.java

Load Script: API_2_ERH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_ERH_CPRVS_BSS_1.clr

6.2.5.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	appendTLV with TAG: 0D and length 16	·	
	Select Text String TLV		
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	· · · · · · · · · · · · · · · · · · ·	•	
	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 5		
	compareLength = 1		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = -1 compareLength = 1</pre>		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
4	compareBuffer.length = 5	n is thrown	
	compareOffset = 0	II IS UIIOWII	
	compareLength = 6		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5		
	compareOffset = 3		
6	compareLength = 3 compareLength < 0	ArrayIndexOutOfBoundsExceptio	
O	compareBuffer.length = 5	n is thrown	
	compareOffset = 0	II IS UITOWIT	
	compareLength = -1		
7	appendTLV with TAG: 0D and length 6		
	Select Text String TLV		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 6	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 1</pre>		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
0	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15	BOONDAINEO IS UNOWIT	
	compareOffset = 0		
	compareLength = 1		
9	compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareOffset = 0 compareLength = 7		
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	
	length	BOUNDARIES is thrown	
	valueOffset = 2		
	compareBuffer.length = 15		
	compareOffset = 0		
4.	compareLength = 5		
11	Initialise the handler	T 11:15 (1 11)10) (01) (5) 5	
	compareValue()	ToolkitException.UNAVAILABLE ELEMENT is thrown	

	0.5	T T
	0F	
	Select Text String TLV	
	Initialise compareBuffer	
	<pre>compareBuffer =</pre>	
	04 00 01 OF	
	Compare buffers	Result is 00h
	<pre>valueOffset = 0</pre>	
	<pre>compareOffset = 0</pre>	
	compareLength = 17	
13	Initialise compareBuffer	
	compareBuffer =	
	04 00 01 02 03	
	04 05 06 07 08	
	05 0A 0B 0C 0D 0E 10	
		Describie 4
	Compare buffers with same parameters	Result is -1
14	Initialise compareBuffer	
	compareBuffer =	
	03 00 01 0F	D. W. A.
	Compare buffers with same parameters	Result is +1
15	Initialise compareBuffer	
	compareBuffer =	
	55 55 55 01 02	
	03 04 05 06 07 08 09 0A 0B 0C	
	55 55 55 55 55	
	Compare buffers	Result is 00h
	valueOffset = 2	Result is out
	compareOffset = 3	
	compareLength = 12	
16	Initialise compareBuffer	
10	compareBuffer =	
	55 55 55 02 01	
	03 04 05 06 07	
	08 09 0A 0B 0C	
	55 55 55 55	
	Compare buffers with same parameters	Result is -1
17	Initialise compareBuffer	
	compareBuffer =	
	55 55 55 01 02	
	03 04 05 06 07	
	08 09 0A 0A 0D	
	55 55 55 55	
	Compare buffers with same parameters	Result is +1
18	Successful call, compareValue with length =0	Result of compareValue() is 0
	compareBuffer.length = 15	
	<pre>compareOffset = 15</pre>	
	compareLength = 0	

6.2.5.10.4 Test Coverage

CRR number	Test case number	
N1	12, 15, 18	
N2	13, 16	
N3	14, 17	
P1	1	
P2	2, 3, 4, 5, 6	
P3	7, 8, 9, 10	
C1 Does not apply for		
	EnvelopeResponseHandl	
	er	
C2	11	

6.2.5.11 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference: API_2_ERH_FACYB_BS

6.2.5.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.11.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACYB_BS_1.scr

Test Applet: API_2_ERH_FACYB_BS_1.java

Load Script: API_2_ERH_FACYB_BS_1.ldr

Cleanup Script: API_2_ERH_FACYB_BS_1.clr

6.2.5.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	appendTLV with TAG: 0D and length 16 Select Text String TLV		
	<pre>dstOffset ≥ dstBuffer.length tag = 0Dh dstBuffer.length = 20 dstOffset = 20</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>dstOffset < 0 dstBuffer.length = 20 dstOffset = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>dstOffset + length >dstBuffer.length dstBuffer.length = 20 dstOffset = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>length > dstBuffer.length dstBuffer.length = 15</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	

		T	
	dstOffset = 0		
	along the handler appropriately with TAC CC		
6	clear the handler, appendTLV with TAG 02 and Length 02		
-	Select a TLV (tag 02h)		
	findAndCopyValue()	ToolkitEveentien LINAVAII ADLE	
	tag = 03h	ToolkitException.UNAVAILABLEELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
	3. (_ELEMENT is thrown.	
7	appendTLV with TAG: 0D and value: 04 00 01		
	0F		
	Successful call	Result of findAndCopyValue() is	
	Tag = 0Dh	17	
	<pre>dstBuffer.length = 17 dstOffset = 0</pre>		
8	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F	Treating out	
	· · · · · · · · · · · · · · · · · · ·		
9	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue() is	
	dstBuffer.length = 20	19	
	dstOffset = 2		
10	Compare buffer	Result is 00h	
	buffer = 55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
11	clear the handler, appendTLV with TAG: 0D		
	and value: 04 00 01 0F		
	l and - · a· · - · ·		
<u> </u>	append a 2 nd Text String TLV		
	Successful call	Result of findAndCopyValue() is	
	<pre>tag = 0Dh dstBuffer.length = 17</pre>	17	
	dstBuller.length = 1/ dstOffset = 0		
12	Compare buffer	Result is 00h	
L'	buffer = 04 00 01 0F		
13	clear the handler, appendTLV with TAG: 0D		
	and value: 04 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndCopyValue() is	
	tag = 8Dh	17	
	<pre>dstBuffer.length = 17 dstOffset = 0</pre>		
14	Compare buffer	Result is 00h	
' -	buffer = 04 00 01 OF	1 TOOGIL IS OOT	
15	Append tag 0Fh		
L	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndCopyValue() is	
	tag = 8Fh	16	
	dstBuffer.length = 16		
16	dstOffset = 0	Popult is 00h	
16	Compare buffer buffer = 00 01 0F	Result is 00h	
	DUTTET = 00 OT " OL		

6.2.5.11.4 Test Coverage

CRR number	Test case number
N1	8, 10, 12
N2	6
N3	7, 9, 11
N4	13, 14, 15, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.12 Method findAndCopyValue(byte tag, byte occurrence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference: API_2_ERH_FACYBBS_BSS

6.2.5.12.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.12.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACYBBS_BSS_1.scr

Test Applet: API_2_ERH_FACYBBS_BSS_1.java

Load Script: API_2_ERH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_ERH_FACYBBS_BSS_1.clr

6.2.5.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		•
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	appendTLV with TAG: 0D and length 16		
	dstOffset ≥ dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	valueOffset = 0		
	<pre>dstBuffer.length = 5 dstOffset = 5</pre>		
	dstLength = 1		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1	A 1 1 0 10/D 1 5	
4	dstLength >dstBuffer.length dstBuffer.length = 5	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 0	n is thrown	
	dstLength = 6		
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
-	dstLength = 3 dstLength < 0	ArrayladayOutOfPayadaEyaantia	
6	dstBuffer.length = 5	ArrayIndexOutOfBoundsExceptio n is thrown	
	dstOffset = 0	II IS UIIOWII	
	dstLength = -1		
7	appendTLV with TAG: 0D and length 6		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	valueOffset = 6		
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 1		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = -1</pre>	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	<pre>dstOffset = 0 dstLength = 1</pre>		
9	dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	dstBuffer.length = 15		
	dstOffset = 0		
10	dstLength = 7 valueOffset + dstLength > Text String length	ToolkitEveention OUT OF TIV	
10	valueOffset = 2	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstBuffer.length = 15	BOONDAINES IS UNIOWIT	
	dstOffset = 0		
L.,	dstLength = 5		
11	clear the handler, appendTLV with TAG 02 and Length 02		
	Select a TLV (tag 02h)		
	findAndCopyValue()	ToolkitException.UNAVAILABLE	
	tag = 0Dh	ELEMENT is thrown	
	occurrence = 2		
	Call the getValueLength() method	ToolkitException.UNAVAILABLEELEMENT is thrown.	
12	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	<pre>valueOffset = 0 dstBuffer.length = 17</pre>		
<u> </u>	abebarrer.rengen - 1/	1	

	dstOffset = 0		
	dstLength = 17		
13	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
14	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	15	
	valueOffset = 2		
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
16	Append a Text String TLV		
1	tag = 0D		
	buffer = 00 11 22 33 44 55 (no specific		
	DCS byte)		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	valueOffset = 0	' '	
	dstBuffer.length = 20		
	dstOffset = 0		
	dstLength = 17		
17	Compare buffer	Result is 00h	
''	buffer = 04 00 01 0F		
18	Successful call	Result of findAndCopyValue() is	
10	tag = 0Dh, occurrence = 2	6	
	valueOffset = 0		
1	dstBuffer.length = 6		
	dstOffset = 0		
	dstLength = 6		
19	Compare buffer	Result is 00h	
.	buffer = 00 11 22 33 44 55	. 100011	
20	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
		Posult of find And ConvA (alua (Alia	
	Successful call (with tag 8Dh)	Result of findAndCopyValue () is	
	tag = 8Dh	17	
	occurrence = 1		
	<pre>valueOffset = 0 datPuffor longth = 17</pre>		
	<pre>dstBuffer.length = 17 dstOffset = 0</pre>		
1	dstorrset = 0 dstLength = 17		
21	Compare buffer	Result is 00h	
4	buffer = 04 00 01 0F	IVESUIT IS ONLY	
22	Append tag 0Fh	+	
22	huffor = 00 01 05		
	buffer = 00 01 0F	Popult of find And Core Males () :-	
	Successful call (with tag 8Fh)	Result of findAndCopyValue () is	
	<pre>tag = 8Fh occurrence = 1</pre>	16	
	<pre>valueOffset = 0 datPuffor longth = 16</pre>		
	<pre>dstBuffer.length = 16 dstOffset = 0</pre>		
	dstOffset = 0 dstLength = 16		
23	Compare buffer	Result is 00h	
23		VESUIT IS ONLI	
0.4	buffer = 00 01 0F	Deput of find And Compa (-1)	
24	Successful call, findAndCopyValue with	Result of findAndCopyValue () is	
1	length =0	16	
1	dstBuffer.length = 16		
	dstOffset = 16		
	dstLength = 0		

6.2.5.12.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18, 24
N4	20, 21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.13 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference: API_2_ERH_FACRB_BS

6.2.5.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical returns 0.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.13.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACRB_BS_1.scr

Test Applet: API_2_ERH_FACRB_BS_1.java

Load Script: API_2_ERH_FACRB_BS_1.ldr

Cleanup Script: API_2_ERH_FACRB_BS_1.clr

6.2.5.13.3 Test procedure

2	Description appendTLV with TAG: 0D and length 16 findAndCompareValue() with a null dstBuffer and tag 0Dh compareOffset ≥ compareBuffer.length	API Expectation NullPointerException is thrown	APDU Expectation
	findAndCompareValue() with a null dstBuffer and tag 0Dh compareOffset ≥ compareBuffer.length	NullPointerException is thrown	
	tag 0Dh compareOffset ≥ compareBuffer.length	Train Giner Exception to the mi	
	compareOffset ≥ compareBuffer.length		
3		A 1 1 0 101D 1 5 11	
3		ArrayIndexOutOfBoundsExceptio	
3	tag = 0Dh	n is thrown	
3	compareBuffer.length = 20		
- 3	compareOffset = 20	A 1 1 0 10'D 1 5 1:	
"	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>compareBuffer.length = 20 compareOffset = -1</pre>	n is thrown	
		A de de CostOfD - cos de Fore esti-	
4	compareOffset + length >	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length	n is thrown	
	<pre>compareBuffer.length = 20 compareOffset = 5</pre>		
-		A may the day Out Of Day and a Type anti-	
5	<pre>length > compareBuffer.length compareBuffer.length = 15</pre>	ArrayIndexOutOfBoundsExceptio	
	compareOffset = 0	n is thrown	
6	clear the handler, appendTLV with TAG 02		
U	and Length 02		
	Select a TLV (tag 02h)	Tablide and LINIAYAN ABLE	
	<pre>findAndCompareValue() tag = 03h</pre>	ToolkitException.UNAVAILABLE	
	3	_ELEMENT is thrown	
7	Verify current TLV	ToolkitException.UNAVAILABLE	
	getValueLength()	_ELEMENT is thrown.	
8	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 OF		
	Compare buffers	Result is 00h	
	tag = 0Dh		
	compareOffset = 0		
9	Verify current TLV	Result is 17	
	getValueLength()		
10	Initialise compareBuffer		
	compareBuffer =		
	04 00 01 10	D 11: 4	
	Compare buffers with same parameters	Result is -1	
11	Initialise compareBuffer		
	compareBuffer =		
	03 00 01 0F	Docult is 14	
40	Compare buffers with same parameters	Result is +1	
12	Initialise compareBuffer		
	compareBuffer = 55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	compareOffset = 2		
13	append a Text String TLV		
	tag = 0Dh		
	buffer = 00 11 22 33 44 55		
İ	Initialise compareBuffer		
	compareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55	D W. OOL	
	Compare buffers	Result is 00h	

	compareOffset = 2	
14	Initialise compareBuffer	
	compareBuffer =	
	55 55 04 01 01	
	02 03 04 05 06	
	07 08 09 0A 0B	
	OC OD OE OF 55	
	Compare buffers	Result is -1
	compareOffset = 2	
15	Initialise compareBuffer	
	compareBuffer =	
	55 55 04 00 01	
	02 03 04 05 06 07 08 09 0A 0B	
	00 0D 0D 10 55	
	Compare buffers	Result is +1
	compareOffset = 2	Result is +1
16	clear the handler and appendTLV with TAG:	
'	0D and value: 04 00 01 0F	
	Initialise compareBuffer	
	compareBuffer = 04 00 01 0F	
	Successful call (with tag 8Dh)	Result is 00h
	tag = 8Dh	
	compareBuffer.length = 17	
	compareOffset = 0	
17	Append tag 0Fh	
	buffer = 00 01 0F	
	Initialise compareBuffer	
	compareBuffer = 00 01 0F	
	Successful call (with tag 8Fh)	Result is 00h
	tag = 8Fh	
	compareBuffer.length = 16	
	compareOffset = 0	

6.2.5.13.4 Test Coverage

CRR number	Test case number
N1	6,7
N2	7,9
N3	8, 13, 12
N4	10, 14
N5	11, 15
N6	17, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for
	Envelope response
	handler

6.2.5.14 Method findAndCompareValue(byte tag, byte occurrence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference: API_2_ERH_FACRBBS_BSS

6.2.5.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical 0 is returned.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

CRRP4: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD INPUT PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.5.14.3 Test Suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_FACRBBS_BSS_1.scr

Test Applet: API_2_ERH_FACRBBS_BSS_1.java

Load Script: API_2_ERH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_ERH_FACRBBS_BSS_1.clr

6.2.5.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	AFTEXPECTATION	Ar Do Expectation
-	findAndCompareValue() with a null	NullPointerException is thrown	
	compareBuffer	NullPointerException is thrown	
2	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	compareOffset ≥ compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	valueOffset = 0	II IS UIIOWII	
	<pre>compareBuffer.length = 5</pre>		
	compareOffset = 5		
	compareLength = 1	1 1 1 0 10/5	
3	<pre>compareOffset < 0 compareBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	compareOffset = -1	n is thrown	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>compareBuffer.length = 5</pre>	n is thrown	
	compareOffset = 0		
	compareLength = 6	A mouth day Out Of Day and Evantia	
5	compareOffset + compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio n is thrown	
	compareBuffer.length = 5	II IS UIIOWII	
	compareOffset = 3		
	compareLength = 3		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0 compareLength = -1		
	Comparenerigen1		
7	clear the handler and appendTLV with TAG		
_	and length of 6		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	<pre>valueOffset = 6</pre>		
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 1</pre>		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>compareBuffer.length = 15</pre>		
	compareOffset = 0		
9	compareLength = 1 compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
9	valueOffset = 0	BOUNDARIES is thrown	
	compareBuffer.length = 15	BOOND/IRIEO IS UIIOWII	
	<pre>compareOffset = 0</pre>		
4.0	compareLength = 7	T 11:55 .: OUT OF TIME	
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	<pre>length valueOffset = 2</pre>	POONDAKIES IS (IIIOMI)	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 5	T 1125 0 545 0050 5	
11	Invalid parameter	ToolkitException.BAD_INPUT_P	
40	occurrence = 0	ARAMETER is thrown	
12	appendTLV with TAG 02 and length 02 Select a TLV (tag 02h)		
<u> </u>	findAndCompareValue()	ToolkitException.UNAVAILABLE	
	tag = 0Dh	_ELEMENT is thrown	
	occurrence = 2		
13	Verify current TLV	ToolkitException.UNAVAILABLE	
	getValueLength()	_ELEMENT is thrown.	
14	clear the handler and appendTLV with TAG:		
	0D and value: 04 00 01 0F		
	Initialise compareBuffer		
	<pre>compareBuffer = 04 00 01 0F</pre>		
	findAndCompareValue()	Result is 00h	
	tag = 0Dh, occurrence = 1		

		1	I
	<pre>valueOffset = 0</pre>		
	compareOffset = 0		
	compareLength = 17		
15	Verify current TLV	Result is 17	
10	getValueLength()	Trocur to 17	
10	Initialise compareBuffer		
16			
	compareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer		
17	•		
	compareBuffer =		
	03 00 01 OF		
	Compare buffers with same parameters	Result is +1	
18	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers	Result is 00h	
	valueOffset = 2		
	compareOffset = 3		
	compareLength = 12		
40			
19	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 02 01		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
20		TRESULT IS T	
20	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	55 55 55 55		
	Compare buffers with same parameters	Result is +1	
21			
21	append a Text String TLV		
21	append a Text String TLV tag = 0Dh		
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55		
21	append a Text String TLV tag = 0Dh		
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55		
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer		
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F	Result is 00h	
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue()	Result is 00h	
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1	Result is 00h	
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0	Result is 00h	
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0	Result is 00h	
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17	Result is 00h	
21	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer	Result is 00h	
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareOffset = 17 Initialise compareBuffer compareBuffer =	Result is 00h	
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareOffset = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55		
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareOffset = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55	Result is 00h	
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue()		
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2		
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0		
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0		
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6		
	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareLength = 6 Initialise compareBuffer		
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffer = 0		
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffer = 0 compareDuffer = 0 compareDuffer = 0 lnitialise compareBuffer compareBuffer = 00 11 22 33 44 66	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 6 Initialise compareBuffer compareBuffer = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareDuffset = 0 finitialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue()		
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 6 Initialise compareBuffer compareBuffer = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareDuffset = 0 finitialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue()	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffer = 0 compareDuffer = 0 compareDuffer = 0 lnitialise compareBuffer compareBuffer = 00 11 22 33 44 66	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffer = 0 compareBuffer = 0 compareDuffset = 0 compareDuffset = 0 compareDuffer = 0 compareBuffer = 0 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareOffset = 0 compareLength = 6	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 00 11 22 33 44 66 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 clear the handler and appendTLV with TAG:	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 0 compareDuffer = 0 compareDuffer = 0 compareDuffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 Clear the handler and appendTLV with TAG: 0D and value: 04 00 01 0F	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 00 11 22 33 44 66 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 clear the handler and appendTLV with TAG:	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 0 compareDuffer = 0 compareDuffer = 0 compareDuffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 Clear the handler and appendTLV with TAG: 0D and value: 04 00 01 0F	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 00 11 22 33 44 66 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareDuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 Clear the handler and appendTLV with TAG: 0D and value: 04 00 01 0F Initialise compareBuffer compareBuffer = 04 00 01 0F	Result is 00h Result is -1	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareBuffer = 0 compareDuffer = 0 compareBuffer = 0 oompareLength = 6 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareDuffer = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0 compareDuffset = 0 compareBuffer = 04 00 01 0F Successful call (with tag 8Dh)	Result is 00h	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareBuffer compareBuffer compareBuffer = 04 00 01 0F Successful call (with tag 8Dh) tag = 8Dh, occurrence = 1	Result is 00h Result is -1	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareBuffer compareBuffer compareBuffer = 04 00 01 0F Successful call (with tag 8Dh) tag = 8Dh, occurrence = 1 valueOffset = 0	Result is 00h Result is -1	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareDeffset = 0 compareBuffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareDeffset = 04 00 01 0F Successful call (with tag 8Dh) tag = 8Dh, occurrence = 1 valueOffset = 0 compareBuffer.length = 17	Result is 00h Result is -1	
22	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareDuffset = 0 compareOffset = 0 compareDuffset = 0 compareBuffer compareBuffer compareBuffer = 04 00 01 0F Successful call (with tag 8Dh) tag = 8Dh, occurrence = 1 valueOffset = 0	Result is 00h Result is -1	

25	Append tag 0Fh		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result is 00h	
	tag = 8Fh, occurrence = 1		
	<pre>valueOffset = 0</pre>		
	compareBuffer.length = 16		
	<pre>compareOffset = 0</pre>		
	compareLength = 16		
26	Successful call, findAndCompareValue with	Result of findAndCompareValue	
	length =0	() is 00	
	CompareBuffer.length = 16		
	<pre>compareOffset = 16</pre>		
	compareLength = 0		

6.2.5.14.4 Test Coverage

CRR number	Test case number
N1	12,13
N2	15,13
N3	14, 18, 22, 21, 26
N4	16, 19, 23
N5	17, 19
N6	25, 24
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.15 Method appendArray

Test Area Reference: API_2_ERH_APDA_BSS

6.2.5.15.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: appends a buffer into the EditHandler buffer

CRRN2: a successful append does not modify the TLV selected

Parameters error

CRRP1: if buffer is null, a java.lang.NullPointerException is thrown

CRRP2: if offset or length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

 $CRRC2: if the \ Edit Handler \ buffer \ is \ busy, a \ Toolkit Exception \ is \ thrown \ with \ reason \ code \\ HANDLER_NOT_AVAILABLE$

6.2.5.15.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APDA_BSS_1.scr

Test Applet: API_2_ERH_APDA_BSS_1.java

Load Script: API_2_ERH_APDA_BSS_1.ldr

Cleanup Script: API_2_ERH_APDA_BSS_1.clr

6.2.5.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
	Initialize the envelope response handler with a	_	
	TLV of length 1		
1	Null buffer	NullPointerException is thrown	
2	offset ≥ buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 5		
	length = 1		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer.length = 5 offset = -1</pre>	n is thrown	
	length = 1		
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 0		
	length = 6		
5	offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer.length = 5 offset = 3</pre>	n is thrown	
	length = 3		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	buffer.length = 5	n is thrown	
	offset = 0		
	length = -1		
7	Handler overflow	ToolkitException.HANDLER_OV	
	<pre>buffer.length = 256 offset = 0</pre>	ERFLOW is thrown	
	length = 256		
8	append the handler with TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	findTLV 0x81		
	Successful call		
	buffer = FF FE F8		
	offset = 0 length = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
9	Clear the handler		
Ť	Successful call		
	buffer = FF FE F8		
	offset = 0		
	length = 8		
<u> </u>	Call copy() method	D W. OOL	
	Compare handler	Result is 00h	
10	compareBuffer = FF FE F8 Successful call		
10	buffer = 00 01 07		
	offset = 2		
	length = 6		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = FF FE F8 02 03 07		
11	Successful call		

buffer = 11 22 88		
offset = 2		
length = 4		
Call copy() method		
Compare handler	Result is 00h	
compareBuffer = FF FE F8 02 03 07 33		
44 55 66		

6.2.5.15.4 Test Coverage

CRR number	Test case number
N1	9, 10, 11
N2	8
N3	
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.16 Method appendTLV(byte tag, byte value)

Test Area Reference: API_2_ERH_APTLBB

6.2.5.16.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Appends a TLV element to the current TLV list (1-byte element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.16.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBB_1.scr
Test Applet: API_2_ERH_APTLBB_1.java
Load Script: API_2_ERH_APTLBB_1.ldr

Cleanup Script: API_2_ERH_APTLBB_1.clr

6.2.5.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call appendArray()		-
	length = 253		
	Handler Overflow: Call twice the	ToolkitException.HANDLER_OV	
	appendTLV()method	ERFLOW is thrown by one of the	
		two.	
2	append the handler with TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value = 00h		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 01 00		
4	Successful call		
	tag = 01h		
	value = FEh		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 01 00 01 01 FE		

Note: Test case 1 call twice appendTLV because the current 03.19 [7] is not clear enough on this point. So this test allows the two possible implementations.

6.2.5.16.4 Test Coverage

CRR number	Test case number
N1	3, 4
N2	2
C1	1
C2	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.17 Method appendTLV(byte tag, byte value1, byte value2)

Test Area Reference: API_2_ERH_APTLBBB

6.2.5.17.1 Conformance requirements:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Appends a TLV element to the current TLV list (2-byte element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.17.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBBB_1.scr

Test Applet: API_2_ERH_APTL BBB_1.java

Load Script: API_2_ERH_APTL BBB_1.ldr

Cleanup Script: API_2_ERH_APTLBBB_1.clr

6.2.5.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
		AFTEXPECIATION	Ar DO Expectation
1	Call the appendArray with length of 253		
	Handler Overflow: Call the appendTLV()	ToolkitException.HANDLER_OV	
	method	ERFLOW is thrown	
2	clear the handler, append the handler with		
	TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value1 = 00h		
	value2 = 01h		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 02 00 01		
4	Successful call		
	tag = 01h		
	value1 = FEh		
	value2 = FDh		
	Call copy() method		
	Compare handler	Result is 00h	
	compareBuffer = 84 02 00 01 01 02 FE FD		

6.2.5.17.4 Test Coverage

CRR number	Test case number
N1	3, 4
N2	2
C1	1
C2	Does not apply for
	EnvelopeResponseHandl
	er

6.2.5.18 Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)

Test Area Reference: API_2_ERH_APTLB_BSS

6.2.5.18.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Appends a TLV element to the current TLV list (byte-array element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

CRRP1: if value is null, a java.lang.NullPointerException is thrown

CRRP2: if valueoffset or valuelength or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.5.18.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLB_BSS_1.scr

Test Applet: API_2_ERH_APTLB_BSS_1.java

Load Script: API_2_ERH_APTLB_BSS_1.ldr

Cleanup Script: API_2_ERH_APTLB_BSS_1.clr

6.2.5.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value	NullPointerException is thrown	711 DO EXPOSITATION
2	valueOffset ≥ value.length	ArrayIndexOutOfBoundsExceptio	
_	value.length = 5	n is thrown	
	valueOffset = 5		
	valueLength = 1		
3	valueOffset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>value.length = 5 valueOffset = -1</pre>	n is thrown	
	valueLength = 1		
4	valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	<pre>valueOffset = 0</pre>		
	valueLength = 6	A 1 1 0 10/D 1 5	
5	ValueOffset + valueLength > value.length value.length = 5	ArrayIndexOutOfBoundsExceptio	
	valueOffset = 3	n is thrown	
	valueLength = 3		
6	valueLength < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	<pre>valueOffset = 0 valueLength = -1</pre>		
7	Handler overflow	ToolkitException.HANDLER_OV	
'	value.length = 254	ERFLOW is thrown	
	<pre>valueOffset = 0</pre>	2.3. 2011 10 4110 1111	
	valueLength = 254		
8	Bad parameter	ToolkitException.BAD_INPUT_P	
	<pre>value.length = 256 valueOffset = 0</pre>	ARAMETER is thrown	
	valueOffset = 0 valueLength = 256		
9	clear the handler, append the handler with		
	TLVs:		
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Successful call		
	tag = 04 value = FF FE F8		
	valueOffset = 0		
	valueLength = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler		
	Successful call		
	tag = 04 value = FF FE F8		
	valueOffset = 0		
	valueLength = 8		
	Call copy() method		
	Compare handler	Result is 00	
44	CompareBuffer = 04 08 FF FE F8		
11	Successful call tag = 85h		
	value = 00 01 07		
	valueOffset = 2		
L	valueLength = 6		
	Call copy() method		
	Compare handler	Result is 00	
	compareBuffer = 04 08 FF FE F8 85 06 02 03 07		
12	Successful call		
	tag = 01		
	value = 11 22 88		
	valueOffset = 2		
-	valueLength = 4 Call copy() method		
	Can copy() method Compare handler	Result is 00	
	compareBuffer = 04 08 FF FE F8 85 06 02	INGSUILIS OU	
	03 07 01 04 33 44 55 66		
13	Clear the handler		
	Successful call		
	tag = 04		

value = 00 01 7F		
valueOffset = 0		
valueLength = 80h		
Call copy() method		
Compare handler	Result is 00	
compareBuffer = 04 81 80 00 017F		

6.2.5.18.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for
	EnvelopeResponseHandl
	er
C3	8

6.2.5.19 Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length)

Test Area Reference: API_2_ERH_APTLBB_BSS

6.2.5.19.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Appends a TLV element to the current TLV list (1 byte and a byte-array element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

CRRP1: if value2 is null, a java.lang.NullPointerException is thrown

CRRP2: if value2offset or value2length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.5.19.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_APTLBB_BSS_1.scr

Test Applet: API_2_ERH_APTLBB_BSS_1.java

Load Script: API_2_ERH_APTLBB_BSS_1.ldr

Cleanup Script: API_2_ERH_APTLBB_BSS_1.clr

6.2.5.19.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value2	NullPointerException is thrown	
2	value2Offset ≥ value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value20ffset = 5		
3	value2Length = 1 value2Offset < 0	ArrayladayOutOfPayadaEyaantia	
3	value2.length = 5	ArrayIndexOutOfBoundsExceptio n is thrown	
	value20ffset = -1	II IS UIIOWII	
	value2Length = 1		
4	value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	<pre>value2Offset = 0 value2Length = 6</pre>		
5	value2Offset + value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	<pre>value20ffset = 3</pre>		
	value2Length = 3	A 1 1 0 10/D 1 5	
6	<pre>value2Length < 0 value2.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	value20ffset = 0	n is thrown	
	value2Length = -1		
7	Handler overflow	ToolkitException.HANDLER_OV	
	value2.length = 254	ERFLOW is thrown	
	value2Offset = 0		
8	value2Length = 254 Bad parameter	ToolkitException.BAD_INPUT_P	
0	value2.length = 256	ARAMETER is thrown	
	value20ffset = 0	ANAMETER IS UTOWIT	
	value2Length = 256		
9	clear the handler, append the handler with		
	TLVs:		
	81 03 11 22 33 82 02 99 77		
	Select Command Details TLV		
	Successful call		
	tag = 04		
	value1 = 05		
	<pre>value2 = FF FE F8 value20ffset = 0</pre>		
	value2Length = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler		
	Successful call		
	tag = 04		
	value1 = 05 value2 = FF FE F8		
	value2Offset = 0		
	value2Length = 8		
	Call copy() method		
	Compare handler	Result is 00	
11	CompareBuffer = 04 09 05 FF FE F8 Successful call		
11	tag = 85h		
	value1 = 55h		
	value2 = 00 01 07		
	value2Offset = 2		
I	value2Length = 6	1	

	Call copy() method	
	Compare handler	Result is 00
	compareBuffer =	
	04 09 05 FF FE F8	
	85 07 55 02 03 07	
12	Successful call	
	tag = 01	
	value1 = 44h	
	value2 = 11 22 88	
	value20ffset = 2	
	value2Length = 4	
	Call copy() method	
	Compare handler	Result is 00
	CompareBuffer =	
	04 09 05 FF FE F8	
	85 07 55 02 03 07	
	01 05 44 33 44 55 66	
13	Clear the handler	
	Successful call	
	tag = 04	
	value1 = 00	
	value2 = 01 7F	
	value20ffset = 0	
	value2Length = 7Fh	
	Call copy() method	
	Compare handler	Result is 00
	compareBuffer = 04 81 80 00 017F	

6.2.5.19.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for EnvelopeResponseHandl er
C3	8

6.2.5.20 Method clear

Test Area Reference: API_2_ERH_CLER

6.2.5.20.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Clears the TLV list of an EditHandler and resets the current TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.5.20.2 Test suite files

Specific triggering: Unrecognized Envelope

Test Script: API_2_ERH_CLER_1.scr

Test Applet: API_2_ERH_CLER_1.java

Load Script: API_2_ERH_CLER_1.ldr

Cleanup Script: API_2_ERH_CLER_1.clr

6.2.5.20.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	append the handler with TLVs:	Result of getLength() is not null	
	81 03 11 22 33		
	82 02 99 77		
	Select Command Details TLV		
	Call the getLength() method		
	Clear the handler	Result of getLength() is 0	
	Call the getLength() method		
2	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown	

6.2.5.20.4 Test Coverage

CRR number	Test case number	
N1	1, 2	
C1	Does not apply for	
	EnvelopeResponseHandl	
	er	

6.2.6 Class MEProfile

6.2.6.1 Method check (byte index)

Test Area Reference: API_2_MEP_CHECB

6.2.6.1.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

Normal execution

CRRN1: The method checks a facility in the handset profile: returns true if supported and false if not.

Parameters error

CRRP1: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available

Context errors

No requirements

6.2.6.1.2 Test suite files

Specific triggering:

UNFORMATTED_SMS_PP_UPD

No Additional requirements for the GSM personalisation:

Test Script: API_2_MEP_CHECB_1.scr

Test Applet: API_2_MEP_CHECB_1.java

Load Script: API_2_MEP_CHECB_1.ldr (the applet is loaded without INI after the reset (RST)

Cleanup Script: API_2_MEP_CHECB_1.clr

6.2.6.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered	ME_PROFILE_NOT_AVAILABLE	
	Triggered by unformatted SMS	ToolkitException is thrown	
	Index = 1	·	
2	Terminal Profile, Facility is supported	true is returned by the method	
	index = 0		
3	Terminal Profile, Facility is not supported	false is returned by the method	
	Index = 15		

6.2.6.1.4 Test Coverage

CRR number	Test case number	
N1	2,3	
P1	1	

6.2.6.2 Method check (byte [] mask, short offset, short length)

Test Area Reference: API_2_MEP_CHEC_BSS

6.2.6.2.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

Normal execution

CRRN1: The method checks all the facilities corresponding to bits set to 1 in the mask buffer: returns true if they are all supported and false if not.

CRRN2: The method returns true if the length to check is 0.

Parameters error

CRRP1: The method shall throw java.lang.NullPointerException if mask is null.

CRRP2: The method shall throw java.lang.ArrayIndexOutOfBoundsException if offset or length or both would cause access outside array bounds.

CRRP3: The method shall throw ME_PROFILE_NOT_AVAILABLE ToolkitException if Terminal Profile data are not available.

Context errors

No requirements

6.2.6.2.2 Test suite files

Specific triggering:

UNFORMATTED_SMS_PP_UPD

No Additional requirements for the GSM personalisation:

Test Script: API_2_MEP_CHEC_BSS_1.scr

Test Applet: API_2_MEP_CHEC_BSS_1.java

Load Script: API_2_MEP_CHEC_BSS_1.ldr (the applet is loaded without INI after the reset (RST))

Cleanup Script: API_2_MEP_CHEC_BSS_1.clr

6.2.6.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	No Terminal Profile is registered Triggered by unformatted SMS Mask = 0xfffffffffffffffffffffffffffffffffff	ME_PROFILE_NOT_AVAIL ABLE ToolkitException is thrown	
2	NULL as parameter to check mask= NULL	NullPointerException is thrown	
3	Offset > mask.length mask = 0xfffffffffffffffffffffffffffffffffff	ArrayIndexOutOfBoundsExc eption is thrown	
4	Offset < 0 mask = 0xfffffffffffffffffffffffffffffffffff	ArrayIndexOutOfBoundsExc eption is thrown	
5	Length > mask.length mask = 0xfffffffffffffffffffffffffffffffffff	ArrayIndexOutOfBoundsExc eption is thrown	
6	Offset + length > mask.length Mask = 0xfffffffffffffffffffffffffffffffffff	ArrayIndexOutOfBoundsExc eption is thrown	
7	<pre>length = 0 mask = 0xfffffffffffffffffffffffffffffffffff</pre>	true is returned	
8	Check all the Terminal Profile mask = 0xFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFF	false is returned by the method because facility 15 is not supported	
9	Check a part of the Terminal Profile mask = 0xfffffffffffffffffffffffffffffffffff	true is returned by the method: the 16 first facilities except facility 15 have been successfully checked	
10	Check a part of the Terminal Profile mask = 0x0080 Offset = 0 Length = 2	false is returned by the method only facility 15 is checked and not supported.	

6.2.6.2.4 Test Coverage

CRR number	Test case number	
N1	8,9,10	
N2	7	
P1	2	
P2	3,4,5,6	
P3	1	

6.2.7 Class ProactiveHandler

6.2.7.1 Method getTheHandler

Test Area Reference: API_2_PAH_GTHD

6.2.7.1.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall return the single system instance of the ProactiveHandler class.

CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object

Parameter errors

No requirements

Context errors

 $CRRC1: The \ method \ shall \ throw \ Toolkit Exception. HANDLER_NOT_AVAILABLE \ if \ the \ handler \ is \ busy.$

6.2.7.1.2 Test Suite files

Test Script: API_2_PAH_GTHD_1.scr

Test Applet: API_2_PAH_GTHD_1.java

Load Script: API_2_PAH_GTHD_1.ldr

Cleanup Script: API_2_PAH_GTHD_1.clr

6.2.7.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	getTheHandler() twice	The returned objects shall be the	
		same	
2	getTheHandler()	The reference shall be a	
		ProactiveHandler	
3	getTheHandler()	The reference shall not be null	
	- "		

6.2.7.1.4 Test Coverage

CRR number	Test case number	
N1	1, 2, 3	
N2	To be checked in	
	Framework tests and	
	insert here cross	
	reference	
C1	To be checked in	
	Framework tests and	
	insert here cross	
	reference	

6.2.7.2 Method init

Test Area Reference: API_2_PAH_INITBBB

6.2.7.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The init() method initialises the next Proactive command in the ProactiveHandler, with Command details and Device Identities TLV. The source device is always the SIM Card (81h). The Comprehension Required flags are set.

CRRN2: The Command number may take any value between 01h and FEh.

CRRN3: The init() method clears the ProactiveHandler before initialising it.

CRRN4: No TLV is selected after a call to the method.

CRRN5: The handler is not sent to the mobile by the init() method.

Parameter errors

No requirements

Context errors

No requirements

6.2.7.2.2 Test Suite files

Test Script: API_2_PAH_INITBBB_1.scr

Test Applet: API_2_PAH_INITBBB_1.java

Load Script: API_2_PAH_INITBBB_1.ldr

Cleanup Script: API_2_PAH_INITBBB_1.clr

6.2.7.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method	·	-
	type = 01h		
	qualifier = 02h		
	dstDevice = 03h		
	Copy ProactiveHandler in a byte array (source)		
	Compare the byte array	source and reference are	
	reference =	identical	
	81h 03h xxh 01h 02h	lucitical	
	82h 02h 81h 03h		
2	Verify the command number value	01h-FEh	
3	Call the init() method		
	type = FFh		
	qualifier = FEh destination = FDh		
	Copy ProactiveHandler in a byte array		
	(source)		
	Compare the byte array	source and reference are	
	reference =	identical	
	81h 03h xxh FFh FEh	Idontioal	
	82h 02h 81h FDh		
4	Select the 1st TLV in the handler		
	Call the init() method with any value		
	Call the getValueLength() method	UNAVAILABLE_ELEMENT	
		ToolkitException is thrown by	
		getValueLength()	

6.2.7.2.4 Test Coverage

CRR number	Test case number
N1	1, 3
N2	2
N3	3
N4	4
N5	1, 3

6.2.7.3 Method initDisplayText

Test Area Reference: API_2_PAH_INDTBB_BSS

6.2.7.3.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall build a DISPLAY TEXT proactive command in the ProactiveHandler, using qualifier, dcs and buffer parameters. Comprehension required flags are set.

CRRN2: A call to this method clears the handler then initialises it.

CRRN3: No TLV is selected after a call to the method.

CRRN4: The DISPLAY TEXT command is not sent by the method.

CRRN5: The Command Number may take any value between 01h and FEh.

Parameter errors

CRRP1: The method shall throw NullPointerException if buffer is null.

CRRP2: If offset or length or both would cause access outside array bounds, an ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is too small to put the requested data.

6.2.7.3.2 Test Suite files

Test Script: API_2_PAH_INDTBB_BSS_1.scr
Test Applet: API_2_PAH_INDTBB_BSS_1.java
Load Script: API_2_PAH_INDTBB_BSS_1.ldr
Cleanup Script: API_2_PAH_INDTBB_BSS_1.clr

6.2.7.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	•
	buffer = NULL	·	
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer = "Text" offset = 5</pre>	n is thrown	
	length = 0		
3	offset < 0	ArrayIndexOutOfBoundsExceptio	
	buffer = "Text"	n is thrown	
4	offset = -1 length > buffer.length	ArrayIndexOutOfBoundsExceptio	
4	buffer = "Text"	n is thrown	
	offset = 0	The unewit	
	length = 5		
5	<pre>offset + length > buffer.length buffer = "Text"</pre>	ArrayIndexOutOfBoundsExceptio	
	offset = 3	n is thrown	
	length = 2		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer = "Text" offset = 3</pre>	n is thrown	
	length = -1		
7	Successful call, buffer is the whole buffer	No exception is thrown	
	qualifier = 0		
	<pre>dcs = 4 buffer = "TextA"</pre>		
	offset = 0		
	length = 5		
	Verify the command number value	Command number between 01h and FEh	
8	Send the command	and i En	DISPLAY TEXT Proactive
			command
			qualifier = 00h dcs = 4
			Text = "TextA"
9	Succesfull call, buffer is part of a buffer with		DISPLAY TEXT Proactive
	the end part		command
	Send the command		qualifier = 00h
	qualifier = 0 dcs = 4		dcs = 4
	buffer = "12TextB"		Text = "TextB"
	offset = 2		
10	Succesfull call, buffer is part of a buffer with		DISPLAY TEXT Proactive
10	the first part		command
	Send the command		
	qualifier = 0		qualifier = 00h
	<pre>dcs = 4 buffer = "TextC12"</pre>		dcs = 4 Text = "TextC"
	offset = 0		
	length = 5		
11	Succesfull call, buffer is part of a buffer		DISPLAY TEXT Proactive
	Send the command qualifier = 0		command
	dcs = 4		qualifier = 00h
	buffer = "12TextD34"		dcs = 4
	offset = 2 length = 5		Text = "TextD"
12	Succesfull call, qualifier = 81h		DISPLAY TEXT Proactive
	Send the command		command
	qualifier = 81h		1.6.
	<pre>dcs = 4 buffer = "TextE"</pre>		qualifier = 81h dcs = 4
	offset = 0		Text = "TextE"
	length = 5		
13	Succesfull call, DCS=0 (7 bits)		DISPLAY TEXT Proactive
	Send the command		command
	qualifier = 0 dcs = 0		qualifier = 00h
1	<pre>buffer = "TextF"</pre>		dcs = 0
	offset = 0		Text = "TextF"

	length = 5		
14	Succesfull call, DCS=8 (UCS2)		DISPLAY TEXT Proactive
'4	Send the command		command
	qualifier = 0		Command
	dcs = 8		qualifier = 00h
	buffer = "TextG"		dcs = 8
	offset = 0		Text = "TextG"
	length = 5		
15	Call the initDisplayText() method with any		DISPLAY TEXT Proactive
	value		command
	Then build and send a DISPLAY TEXT		0011111111111
	command		qualifier = 00h
	qualifier = 0		dcs = 4
	dcs = 4		Text = "TextHTextH"
	<pre>buffer = "TextHTextH"</pre>		
	offset = 0		
	length = 10		
16	Successful call, text length is null		DISPLAY TEXT Proactive
	Send the command		command
	qualifier = 0		
	dcs = 4		qualifier = 00h
1	<pre>buffer = "" (not null buffer)</pre>		Text String TLV = 8D 00
1	offset = 0		
47	length = 0	LINIAN/ALL ADLE ELEMENT	
17	Select a TLV in the ProactiveHandler	UNAVAILABLE_ELEMENT	
	Call the initDisplayText() method	ToolkitException is thrown by	
1	Call the getValueLength() method	getValueLength()	
<u> </u>	0		DIODI AV TEXT D
18	Successful call, buffer length = 7Eh		DISPLAY TEXT Proactive
1	munlifian - 0		command
	qualifier = 0 dcs = 4		mark Obsider a mark
	buffer = "UUU"		Text String TLV = 8D 7F 04 55 55
	offset = 0		עס /ד על סט אַן עס /דּ עס
1	length = 7Eh		
19	Successful call, buffer length = 7Fh		DISPLAY TEXT Proactive
	,		command
1	qualifier = 0		
	dcs = 4		Text String TLV = 8D 81
	buffer = "UUU"		80 04 55 55
	offset = 0		
	length = 7Fh		DIODI AV TEVT Des a sti
20	Successful call, buffer length = 240		DISPLAY TEXT Proactive
	Oualifier = 0		command
	dcs = 4		Toyt String TIV -
	buffer = "UUU"		Text String TLV = 8D 81 F1 04 55 55
	offset = 0		02 01 11 01 33 33
	length = 240		
21	Call the initDisplayText() method with a too	HANDLER_OVERFLOW	
	long buffer	ToolkitException is thrown	
	qualifier = 0		
	dcs = 4		
	buffer = "XXXX"		
	offset = 0		
	length = 241		NI- managatina
22	Call the initDisplayText() without sending the		No proactive command
	command		shall be sent expected
1			status is '9000'

6.2.7.3.4 Test Coverage

CRR number	Test case number	
N1	8, 9, 10, 11, 12, 13, 14,	
	15, 16, 18, 19, 20	
N2	15	
N3	17	
N4	22	
N5	7	
P1	1	
P2	2, 3, 4, 5, 6	
C1	21	

6.2.7.4 Method initGetInkey

Test Area Reference: API_2_PAH_INGKBB_BSS

6.2.7.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall build a GET INKEY proactive command in the ProactiveHandler, using qualifier, dcs and buffer parameters. Comprehension Required flags are set.

CRRN2: A call to this method clears the handler then initialises it.

CRRN3: No TLV is selected after a call to the method.

CRRN4: The GET INKEY command is not sent by the method.

CRRN5: The Command Number may take any value between 01h and FEh.

Parameter errors

CRRP1: The method shall throw NullPointerException if buffer is null.

CRRP1: If offset or length or both would cause access outside array bounds, a ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is to small to put the requested data.

6.2.7.4.2 Test Suite files

Test Script: API_2_PAH_INGKBB_BSS_1.scr
Test Applet: API_2_PAH_INGKBB_BSS_1.java
Load Script: API_2_PAH_INGKBB_BSS_1.ldr
Cleanup Script: API_2_PAH_INGKBB_BSS_1.clr

6.2.7.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	71 DO Expodiction
	buffer = NULL	Train onto Exception is another	
2	offset > buffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer = "Text"</pre>	n is thrown	
_	offset = 5	A	
3	<pre>offset < 0 buffer = "Text"</pre>	ArrayIndexOutOfBoundsExceptio	
	offset = -1	n is thrown	
4	length > buffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer = "Text"</pre>	n is thrown	
	offset = 0		
5	length = 5 offset + length > buffer.length	ArrayIndexOutOfBoundsExceptio	
3	buffer = "Text"	n is thrown	
	offset = 3	II IS UIIOWII	
	length = 2		
6	length < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>buffer = "Text" offset = 3</pre>	n is thrown	
	length = -1		
7	Successful call, buffer is the whole buffer	No exception is thrown	
	qualifier = 0		
	dcs = 4		
	<pre>buffer = "TextA" offset = 0</pre>		
	length = 5		
	Verify the command number value	Command number between 01h	
	•	and FEh	
8	Send the command		GET INKEY Proactive
			command
			qualifier = 00h
			dcs = 4
			Text = "TextA"
9	Succesfull call, buffer is part of a buffer with		GET INKEY Proactive
	the end part		command
	qualifier = 0 dcs = 4		
	buffer = "12TextB"		qualifier = 00h dcs = 4
	offset = 2		Text = "TextB"
	length = 5		
10	Succesfull call, buffer is part of a buffer with		GET INKEY Proactive
	the first part qualifier = 0		command
	dcs = 4		qualifier = 00h
	buffer = "TextC12"		dcs = 4
	offset = 0		Text = "TextC"
44	length = 5		CET INIVEN Dragative
11	Succesfull call, buffer is part of a buffer Send the command		GET INKEY Proactive command
	qualifier = 0		Continuana
	dcs = 4		qualifier = 00h
	buffer = "12TextD34"		dcs = 4
	offset = 2 length = 5		Text = "TextD"
12	Succesfull call, qualifier = 81h		GET INKEY Proactive
'-	qualifier = 81h		command
	dcs = 4		
	<pre>buffer = "TextE" offget = 0</pre>		qualifier = 81h
	offset = 0 length = 5		dcs = 4 Text = "TextE"
	5		ICAC - ICACE.
13	Succesfull call, DCS=0 (7 bits)		GET INKEY Proactive
	qualifier = 0		command
	dcs = 0		
	buffer = "TextF"		qualifier = 00h
	offset = 0 length = 5		dcs = 0 Text = "TextF"
			IEAU = "IEAUF"
14	Succesfull call, DCS=8 (UCS2)		GET INKEY Proactive
' '	qualifier = 0		command

	dcs = 8		command
	buffer = "TextG"		
	offset = 0		qualifier = 00h
	length = 5		dcs = 8
			Text = "TextG"
15	Call the initGetInkey() method with any value		GET INKEY Proactive
. •	Then build and send a GET INKEY command		command
			Command
	qualifier = 0		3.1.5.1
	dcs = 4		qualifier = 00h
	buffer = "TextHTextH"		dcs = 4
	offset = 0		Text = "TextHTextH"
	length = 10		
16	Successful call, text length is null		GET INKEY Proactive
10			
	Send the command		command
	qualifier = 0		
	dcs = 4		qualifier = 00h
	buffer = ""		Text String TLV = 8D 00
	offset = 0		
	length = 0		
47	-	LINIAN/AU ADI E ELEMENIT	
17	Select a TLV in the ProactiveHandler	UNAVAILABLE_ELEMENT	
	Call the initGetInkey() method	ToolkitException is thrown by	
	Call the getValueLength() method	getValueLength()	
	oun me gerruna zongm(, memeu	gottalaozongin()	
	0 (1 11 (1 1 1 1 1 1 1		
18	Successful call, buffer length = 7Eh		GET INKEY Proactive
			command
	qualifier = 0		
	dcs = 4		Text String TLV =
	buffer = "UUU"		8D 7F 04 55 55
	offset = 0		6D /F 04 55 55
—	length = 7Eh		
19	Successful call, buffer length = 7Fh		GET INKEY Proactive
			command
	qualifier = 0		
	dcs = 4		Text String TLV = 8D 81
	buffer = "UUU"		80 04 55 55
	offset = 0		00 01 33 33
	length = 7Fh		
			OFT BUCEVED
20	Successful call, buffer length = 240		GET INKEY Proactive
			command
	Qualifier = 0		
	dcs = 4		Text String TLV =
	buffer = "UUU"		8D 81 F1 04 55 55
	offset = 0		02 01 11 01 33 33
	length = 240		
04		LIANDI ED OVEDELOW	
21	Call the initGetInkey() method with a too long	HANDLER_OVERFLOW	
	buffer	ToolkitException is thrown	
	qualifier = 0		
	dcs = 4		
	buffer = "XXXX"		
	offset = 0		
<u></u>	length = 241		N
22	Call the initGetInkey() without sending the		No proactive command
	command		shall be sent expected
			status is '9000'
			31a1u3 13 3000

6.2.7.4.4 Test Coverage

CRR number	Test case number
N1	8, 9, 10, 11, 12, 13, 14,
	15, 16, 18, 19, 20
N2	15
N3	17
N4	22
N5	7
P1	1
P2	2, 3, 4, 5, 6
C1	21

6.2.7.5 Method initGetInput

Test Area Reference: API_2_PAH_INGPBB_BSSSS

6.2.7.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method shall build a GET INPUT proactive command in the ProactiveHandler, using qualifier, dcs, buffer, minRespLength and maxRespLength parameters. Comprehension Required flags are set.

CRRN2: A call to this method clears the handler then initialises it.

CRRN3: No TLV is selected after a call to the method.

CRRN4: The GET INPUT command is not sent by the method.

CRRN5: The Command Number may take any value between 01h and FEh.

Parameter errors

CRRP1: The method shall throw NullPointerException if buffer is null.

CRRP2: If offset or length or both would cause access outside array bounds, a ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: A ToolkitException.HANDLER_OVERFLOW shall be thrown if the ProactiveHandler is to small to put the requested data.

6.2.7.5.2 Test Suite files

Test Script: API_2_PAH_INGPBB_BSSSS_1.scr
Test Applet: API_2_PAH_INGPBB_BSSSS_1.java
Load Script: API_2_PAH_INGPBB_BSSSS_1.ldr
Cleanup Script: API_2_PAH_INGPBB_BSSSS_1.clr

6.2.7.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to buffer	NullPointerException is thrown	7 2 C Zapostanon
	buffer = NULL		
2	<pre>offset > buffer.length buffer = "Text" offset = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>offset < 0 buffer = "Text" offset = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>length > buffer.length buffer = "Text" offset = 0 length = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	<pre>offset + length > buffer.length buffer = "Text" offset = 3 length = 2</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
6	<pre>length < 0 buffer = "Text" offset = 3 length = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
7	Successful call, buffer is the whole buffer	No exception is thrown	
	<pre>qualifier = 0 dcs = 4 buffer = "TextA" offset = 0 length = 5 minRespLength = 00h maxRespLength = FFh</pre>	·	
	Verify the command number value	Command number between 01h	
	Send the command	and -FEh	GET INPUT Proactive
8			<pre>command qualifier = 00h dcs = 4 Text = "TextA" Min Length = 00h Max Length = FFh</pre>
9	Succesfull call, buffer is part of a buffer with		GET INPUT Proactive
	the end part Send the command qualifier = 0 dcs = 4 buffer = "12TextB" offset = 2 length = 5 minRespLength = 10h		<pre>command qualifier = 00h dcs = 4 Text = "TextB" Min Length = 10h Max Length = FFh</pre>
10	maxRespLength = FFh		CET INDUE Dropotive
10	Succesfull call, buffer is part of a buffer with the first part Send the command qualifier = 0 dcs = 4 buffer = "TextC12" offset = 0 length = 5 minRespLength = FFh maxRespLength = FFh		GET INPUT Proactive command qualifier = 00h dcs = 4 Text = "TextC" Min Length = FFh Max Length = FFh
11	Succesfull call, buffer is part of a buffer Send the command		GET INPUT Proactive command
	<pre>qualifier = 0 dcs = 4 buffer = "12TextD34" offset = 2 length = 5 minRespLength = 00h maxRespLength = 00h</pre>		qualifier = 00h dcs = 4 Text = "TextD" Min Length = 00h Max Length = 00h
12	Succesfull call, qualifier = 81h qualifier = 81h dcs = 4		GET INPUT Proactive command

	<pre>buffer = "TextE"</pre>		qualifier = 81h
i	offset = 0		dcs = 4
			-
	length = 5		Text = "TextE"
	minRespLength = 00h		Min Length = 00h
	1 2		_
	maxRespLength = 10h		Max Length = 10h
40	0(D00 0 (7)		OFT INDUTED (
13	Succesfull call, DCS=0 (7 bits)		GET INPUT Proactive
	qualifier = 0		command
	1 -		Command
	dcs = 0		
	buffer = "TextF"		qualifier = 00h
	offset = 0		
			dcs = 0
	length = 5		Text = "TextF"
	minRespLength = 10h		Min Length = 10h
	1 3		_
	maxRespLength = 10h		Max Length = 10h
			-
14	Succesfull call, DCS=8 (UCS2)		GET INPUT Proactive
	qualifier = 0		
	1 -		command
	dcs = 8		
	buffer = "TextG"		qualifier = 00h
			_
	offset = 0		dcs = 8
	length = 5		Text = "TextG"
1	minRespLength = 00h		Min Length = 00h
1	maxRespLength = FFh		Max Length = FFh
1	= -		
15	Call the initGetInput() method with any value		GET INPUT Proactive
'			
	Then build and send a GET INPUT command		command
	qualifier = 0		
	1		7.1.5.1
	dcs = 4		qualifier = 00h
	buffer = "TextHTextH"		dcs = 4
	offset = 0		Text = "TextHTextH"
	length = 10		Min Length = 00h
	minRespLength = 00h		Max Length = 10h
			Max Length - 1011
	maxRespLength = 10h		
16	Successful call, text length is null		GET INPUT Proactive
10	l · · · · · · · · · · · · · · · · · · ·		
	Send the command		command
	qualifier = 0		
	-		
	dcs = 4		qualifier = 00h
	buffer = ""		Text String TLV = 8D 00
			_
	offset = 0		Min Length = 00h
	length = 0		Max Length = 10h
	minRespLength = 00h		3
	maxRespLength = 10h		
17	Select a TLV in the ProactiveHandler	UNAVAILABLE ELEMENT	
1 ' '			
	Call the initGetInput() method	ToolkitException is thrown by	
	Call the getValueLength() method		
	Can the getvalueLength() method	getValueLength()	
10			
	Oversandal sell buffer levels 75b		OFT INDUIT Day a still as
18	Successful call, buffer length = 7Eh		GET INPUT Proactive
10	Successful call, buffer length = 7Eh		
10	_		GET INPUT Proactive command
10	qualifier = 0		command
10	qualifier = 0 dcs = 4		
10	qualifier = 0 dcs = 4		<pre>command Text String TLV =</pre>
10	qualifier = 0 dcs = 4 buffer = "UUU"		command Text String TLV = 8D 7F 04 55 55
10	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h
10	qualifier = 0 dcs = 4 buffer = "UUU"		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h
10	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh</pre>		command Text String TLV = 8D 7F 04 55 55
10	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h
10	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU"</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command
	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h
	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU"</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU"</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
20	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h maxRespLength = 00h maxRespLength = 00h maxRespLength = 10h</pre>		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
20	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h maxRespLength = 00h maxRespLength = 00h maxRespLength = 10h</pre>	HANDLER OVERFLOW	command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h maxRespLength = 00h maxRespLength = 10h Call the initGetInput() method with a too long	HANDLER_OVERFLOW	command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
20	<pre>qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h maxRespLength = 00h maxRespLength = 00h maxRespLength = 10h</pre>	HANDLER_OVERFLOW ToolkitException is thrown	command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81
19	qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Eh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 7Fh qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 7Fh minRespLength = 00h maxRespLength = 10h Successful call, buffer length = 236 Qualifier = 0 dcs = 4 buffer = "UUU" offset = 0 length = 236 minRespLength = 00h maxRespLength = 00h maxRespLength = 10h Call the initGetInput() method with a too long		command Text String TLV = 8D 7F 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81 80 04 55 55 Min Length = 00h Max Length = 10h GET INPUT Proactive command Text String TLV = 8D 81

	qualifier = 0	
	dcs = 4	
	<pre>buffer = "XXXX"</pre>	
	offset = 0	
	length = 237	
	minRespLength = 00h	
	maxRespLength = 10h	
22	Call the initGetInput() without sending the	No proactive command
	command	shall be sent expected
		status is '9000'

6.2.7.5.4 Test Coverage

CRR number	Test case number
N1	8, 9, 10, 11, 12, 13, 14,
	15, 16, 18, 19, 20
N2	15
N3	17
N4	22
N5	7
P1	1
P2	2, 3, 4, 5, 6
C1	21

6.2.7.6 Method send

Test Area Reference: API_2_PAH_SEND

6.2.7.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public byte send()

Normal execution

CRRN1: The send() method send the current proactive command to the mobile.

CRRN2: The returned byte is equal to general result of the command (first byte of Result TLV in Terminal Response).

CRRN3: The handler remains unchanged after a call to send() method until the use of initXX() or appendTLV().

CRRN4: There is no invocation of select() or deselect() method.

CRRN5: A pending toolkit applet transaction at the method invocation is aborted.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown is the Result Simple TLV is missing in Terminal Response.

CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the general result byte is missing in the Result Simple TLV in Terminal Response.

6.2.7.6.2 Test Suite files

Test Script: API_2_PAH_SEND_1.scr

Test Applet: API_2_PAH_SEND_1.java

Load Script: API_2_PAH_SEND_1.ldr

Cleanup Script: API_2_PAH_SEND_1.clr

6.2.7.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	7.1. I Expositation	DISPLAY TEXT Proactive
'	qualifier = 00h		command
	dcs = 04h		
	buffer = 'Text'	Described as 10 : 00:	-
2	Terminal Response with General Result = 00	Result of send() is 00h	
	Result TLV = 03 01 00 (command performed		
	successfully)		
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
L.	buffer = 'Text'	D 16 6 10 : 041	
4	Terminal Response with General Result = 01, without Additional information on result	Result of send() is 01h	
	Result TLV = 03 01 01 (command performed		
	with partial comprehension)		
5	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
6	buffer = 'Text' Terminal Response with General Result = 01,	Result of send() is 01h	+
О	with Additional information on result	Result of seria() is offi	
	with Additional information on result		
	Result TLV = 03 02 01 55 (command		
	performed with partial comprehension)		
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
8	buffer = 'Text' Terminal Response with General Result = 02	Result of send() is 02h	+
°	reminai kesponse with General Result = 02	INCOURT OF SCHOOL IS UZII	
	Result TLV = 03 04 02 65 43 21 (Missing		
	information)		
9	Build and send a 7Fh byte command		DISPLAY TEXT Proactive
	(DISPLAY TEXT)		command
	qualifier = 00h		DDD 50111 D0 55
	dcs = 04h buffer = "UUUUU"		BER-TLV = D0 7F Text String TLV = 8D 74
	length = 73h		04 55 55 55
10	Build and send a 80h byte command		DISPLAY TEXT Proactive
-	(DISPLAY TEXT)		command
	qualifier = 00h		
	dcs = 04h		BER-TLV = D0 81 80
	buffer = "UUUUU"		Text String TLV = 8D 75
11	length = 74h Build and send a maximum length command		04 55 55 55 DISPLAY TEXT Proactive
' '	(length of the handler should be 253)		command
	(gii. e. iiie iiaiiaiei eileala se 200)		
	DISPLAY TEXT:		BER-TLV = D0 81 FD
	Qualifier = 0		Text String TLV = 8D 81
	dcs = 4		F1 04 55 55
	<pre>buffer = "UUU" offset = 0</pre>		
	length = 240		
12	Verify that the Proactive Handler is not		
	modified after a send()		
	Build a DISPLAY TEXT command		
	Copy ProactiveHandler to source byte array		
	Send command		
	Copy ProactiveHandler to destination byte		
	array	1	1

	Compare source and destination	Source and destination are identical	
13	Build and send a DISPLAY TEXT command Verify there is no invocation of select() or deselect() method.		DISPLAY TEXT Proactive command
14	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 2 Result TLV	Result of send() is 02h	
	1st Result TLV = 03 02 02 12 2nd Result TLV = 03 03 03 34 56		
15	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE _ELEMENT is thrown by send()	
16	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without general result byte in the Simple TLV Result TLV = 03 00	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown by send()	

6.2.7.6.4 Test Coverage

CRR number	Test case number
N1	1, 3, 5, 7, 9, 10, 11, 12,
	13, 14
N2	2, 4, 6, 8, 14
N3	12
N4	13
N5	To be checked in
	Framework tests and
	insert here cross
	reference
C1	15
C2	16

6.2.7.7 Method getLength

Test Area Reference API_2_PAH_GLEN

6.2.7.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: returns the length in bytes of the TLV list.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.7.7.2 Test Suite files

Test Script: API_2_PAH_GLEN_1.scr

Test Applet: API_2_PAH_GLEN_1.java

Load Script: API_2_PAH_GLEN_1.ldr

Cleanup Script: API_2_PAH_GLEN_1.clr

6.2.7.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear the handler getLength()	Result of getLength() is 0	
2	Call the init() method getLength()	Result of getLength() is 9	
3	Call the initDisplayText() method, with buffer length = 240 getLength()	Result of getLength() is 253	
4	Build a 7Fh Proactive Handler getLength()	Result of getLength() is 7Fh	
5	Build a 80h Proactive Handler getLength()	Result of getLength() is 80h	

6.2.7.7.4 Test Coverage

CRR number	Test case number	
N1	1, 2, 3, 4,5	
C1	Does not apply for	
	Proactive Handler	

6.2.7.8 Method copy

Test Area Reference API_2_PAH_COPY_BSS

6.2.7.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: copies the simple TLV list contained in the handler to the destination byte array.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null a NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.

CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.8.2 Test Suite files

Test Script: API_2_PAH_COPY_BSS_1.scr

Test Applet: API_2_PAH_ COPY_BSS_1.java

Load Script: API_2_PAH_ COPY_BSS_1.ldr

Cleanup Script: API_2_PAH_ COPY_BSS_1.clr

6.2.7.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to dstBuffer	NullPointerException is thrown	•
	•		
2	Call the init() method		
	DstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 6 dstLength = 0</pre>		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1		
4	DstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 0</pre>	n is thrown	
	dstOffset = 0 dstLength = 6		
5	dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
J	dstBuffer.length = 5	n is thrown	
	dstOffset = 3	II lo ullowii	
	dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
7	dstLength > length of the simple TLV list	ToolkitException.OUT_OF_TLV_	
'	dstBuffer.length = 10	BOUNDARIES is thrown	
	dstOffset = 0	BOOTAD/ ITALEO IS ITHOWN	
	dstLength = 10		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 9	
	dstBuffer.length = 9		
	<pre>dstOffset = 0 dstLength = 9</pre>		
9	Compare the buffer	Result of arrayCompare() is 0	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 12	
10	dstBuffer.length = 15	1100dit 01 00py() 10 12	
	dstOffset = 3		
	dstLength = 9		
11	Compare the whole buffer	Result of arrayCompare() is 0	
12	Successful call, dstBuffer is part of a buffer	Result of copy() is 9	
	dstBuffer.length = 15		
	<pre>dstOffset = 3 dstLength = 6</pre>		
13	Compare the whole buffer	Result of arrayCompare() is 0	
13	Compare the whole build	result of arraycompare() is 0	

6.2.7.8.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13
N2	8, 10, 12
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1 Does not apply	
	ProactiveHandler

6.2.7.9 Method findTLV

Test Area Reference API_2_PAH_FINDBB

6.2.7.9.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.

CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.

CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.

CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.

CRRN5: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER NOT AVAILABLE.

6.2.7.9.2 Test Suite files

Test Script: API_2_PAH_FINDBB_1.scr
Test Applet: API_2_PAH_FINDBB_1.java
Load Script: API_2_PAH_FINDBB_1.ldr
Cleanup Script: API_2_PAH_FINDBB_1.clr

6.2.7.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	·	•
	Invalid input parameter	ToolkitException.BAD_INPUT_P	
	Occurrence = 0	ARAMETER is thrown	
2	Call the init() method		
	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	Tag = 01h Occurrence = 1		
3	Call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is TLV_FOUND_CR_SET	
_	Tag = 02h	TROSURIS TEV_T COND_CIT_CET	
	Occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	Tag = 03h		
7	Occurrence = 1 Call the getValueLength() method	To all it Exponding LINIAN/ALL ADLE	
7	Can the getvalueLength() method	ToolkitException.UNAVAILABLE ELEMENT is thrown.	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
0	Tag = 01h	Result is TEV_NOT_FOUND	
	Occurrence = 2		
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown.	
10	Append a TLV with tag=02h		
	Search the TLV	Result is	
	Tag = 02h	TLV_FOUND_CR_NOT_SET	
44	Occurrence = 2		
11	Append a TLV with tag=04h Search the TLV	Result is	
	Tag = 04h	TLV_FOUND_CR_NOT_SET	
	Occurrence = 1	ILV_POUND_CK_NOT_SET	
12	Search tag 81h	Result is TLV_FOUND_CR_SET	
	Tag = 81h		
	Occurrence = 1		
13	Search tag 84h	Result is	
	Tag = 84h	TLV_FOUND_CR_NOT_SET	
1	Occurrence = 1		

6.2.7.9.4 Test Coverage

CRR number	Test case number
N1	3, 5
N2	2, 4
N3	10, 11
N4	6, 7,8, 9
N5	12, 13
P1	1
C1	Does not apply for
	Proactive Handler

6.2.7.10 Method getValueLength

Test Area Reference API_2_PAH_GVLE

6.2.7.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.10.2 Test Suite files

Test Script: API_2_PAH_GVLE_1.scr

Test Applet: API_2_PAH_GVLE_1.java

Load Script: API_2_PAH_GVLE_1.ldr

Cleanup Script: API_2_PAH_GVLE_1.clr

6.2.7.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method		
	getValueLength()	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
2	Call the appendTLV() method		
	tag = 0D		
	valueOffset = 0		
	valueLength = 0		
	Search TLV 0Dh (Text String TLV)		
	<pre>getValueLength()</pre>	Result is 00h	
3	Call the initDisplayText() method		
	length = 1 (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	<pre>getValueLength()</pre>	Result is 02h	
4	Call the initDisplayText() method		
	length = 7Eh (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 7Fh	
5	Call the initDisplayText() method		
	length = 7Fh (+ dcs byte)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 80h	
6	Call the initDisplayText() method		
	length = F0h (maximum text length)		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is F1h	

6.2.7.10.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4, 5, 6
C1	Does not apply for
	Proactive Handler
C2	1

6.2.7.11 Method getValueByte

Test Area Reference API_2_PAH_GVBYS

6.2.7.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

Parameter errors

CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.11.2 Test Suite files

Test Script: API_2_PAH_GVBYS_1.scr

Test Applet: API_2_PAH_GVBYS_1.java

Load Script: API_2_PAH_GVBYS_1.ldr

Cleanup Script: API_2_PAH_GVBYS_1.clr

6.2.7.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the init() method		
	type = FFh		
	qualifier = FEh destination = FDh		
		To alleit Exponetion LINIA VALLADI E	
	getValueByte(0)	ToolkitException.UNAVAILABLE	
2	Course TI V 04h (Commond Dataile TI V)	_ELEMENT is thrown	
	Search TLV 01h (Command Details TLV)	Tablista Cut OF TIV	
	getValueByte(3)	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
_	Course TI V 04h (Commond Dataile TI V)	BOUNDARIES IS thrown	
3	Search TLV 01h (Command Details TLV)	D 11: FEL / 116:)	
	getValueByte(2)	Result is FEh (qualifier)	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 81h (Source)	
	gerr aa.e_y.to(o)	Treesing of the (Country)	
5	initDisplayText()		
	buffer = 00 01 7D		
	length = 7Eh		
	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Dh	
6	initDisplayText()		
	buffer = 00 01 7D 7E		
	length = 7Fh		
	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Dh	
7	getValueByte(7F)	Result is 7Eh	
8	initDisplayText()		
	buffer = 00 01 EF		
	length = F0h		
	Search TLV 0Dh (Text String TLV)	Decott in EEt	
	getValueByte(F0)	Result is EFh	

6.2.7.11.4 Test Coverage

CRR number	Test case number	
N1	3, 4, 5, 6, 7, 8	
P1	2	
C1	Does not apply for	
	Proactive Handler	
C2	1	

6.2.7.12 Method copyValue

Test Area Reference API_2_PAH_CPYVS_BSS

6.2.7.12.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

Normal execution

CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.12.2 Test Suite files

Test Script: API_2_PAH_CPYVS_BSS_1.scr

Test Applet: API_2_PAH_CPYVS_BSS_1.java

Load Script: API_2_PAH_CPYVS_BSS_1.ldr

Cleanup Script: API_2_PAH_CPYVS_BSS_1.clr

6.2.7.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		-
	Select a TLV		
	copyValue() with a null dstBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
_	Select Text String TLV		
	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 6	THO UNIOWIT	
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
	dstLength = 1		
4	dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
_	dstLength = 6	14 1 1 0 10/2 1 5	
5	dstOffset + dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = 3</pre>	n is thrown	
	dstUriset = 3 dstLength = 3		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
0	dstBuffer.length = 5	n is thrown	
	dstOffset = 0	n is thrown	
	dstLength = -1		
7	initDisplayText() with length = 5		
	Select Text String TLV		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 7	BOUNDARIES is thrown	

	1.5.66		
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 0		
8	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	valueOffset < 0	BOUNDARIES is thrown	
	valueOffset = -1		
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 1		
9	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	dstLength > Text String length	BOUNDARIES is thrown	
	valueOffset = 0	BOOTED WILLO IO WILOWIT	
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 7		
10	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
'0	valueOffset + dstLength > Text String	BOUNDARIES is thrown	
	length	BOUNDARIES IS UITOWIT	
	valueOffset = 2		
	dstBuffer.length = 15		
	dstOffset = 0		
	dstLength = 5		
	dschengen – 5		
	L.M.P. al. L. B.		
11	Initialise the handler		
	copyValue()	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown	
12	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F		
	Select Text String TLV		
	Successful call	Result of copyValue() is 17	
	valueOffset = 0	resource sopy rando() to the	
	dstBuffer.length = 17		
	dstOffset = 0		
	dstLength = 17		
13	Compare buffer	Result is 00h	
'	buffer = 04 00 01 0F	recount to con	
	Daller 01 00 01 01		
14	initialise dstBuffer		
14			
-	dstBuffer = 55 55 55	D 1: ()/ 1 (): 45	
	Successful call	Result of copyValue() is 15	
	valueOffset = 2		
	dstBuffer.length = 20		
	dstOffset = 3		
L	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
1	08 09 0A 0B 0C		
1	55 55 55 55		

6.2.7.12.4 Test Coverage

CRR number	Test case number	
N1	13, 15	
N2	12, 14	
P1	1	
P2	2, 3, 4, 5, 6	
P3	7, 8, 9, 10	
C1	Does not apply for	
	Proactive Handler	
C2	11	

6.2.7.13 Method compareValue

Test Area Reference API_2_PAH_CPRVS_BSS

6.2.7.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Compares the last found TLV element with a buffer:

CRRN1: returns 0 if identical.

CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.

CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.7.13.2 Test Suite files

Test Script: API_2_PAH_CPRVS_BSS_1.scr

Test Applet: API_2_PAH_CPRVS_BSS_1.java

Load Script: API_2_PAH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_PAH_CPRVS_BSS_1.clr

6.2.7.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	·	•
	Select a TLV		
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	<pre>initDisplayText() with length = 15 Select Text String TLV</pre>		
	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 6		
	compareLength = 0		
3	<pre>compareOffset < 0 compareBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	compareOffset = -1	n is thrown	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = 0 compareLength = 6</pre>		
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
	>compareBuffer.length	n is thrown	
	compareBuffer.length = 5		
	compareOffset = 3		
6	<pre>compareLength = 3 compareLength < 0</pre>	ArrayIndexOutOfBoundsExceptio	
U	compareBuffer.length = 5	n is thrown	
	compareOffset = 0	I i i i i i i i i i i i i i i i i i i i	
	compareLength = -1		
7	initDisplayText() with length = 5		
	Select Text String TLV		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 7 compareBuffer.length = 15</pre>	BOUNDARIES is thrown	
	compareOffset = 0		
	compareLength = 0		
8	[Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset < 0 valueOffset = -1</pre>	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 1	Tablita Commission Old OF TIM	
9	[Select Text String TLV] compareLength > Text String length	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	valueOffset = 0	BOUNDARIES IS UITOWIT	
	compareBuffer.length = 15		
	compareOffset = 0		
10	compareLength = 7 [Select Text String TLV]	ToolkitException.OUT_OF_TLV_	
10	valueOffset + compareLength > Text String	BOUNDARIES is thrown	
	length		
	valueOffset = 2		
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareLength = 5		
11	Initialise the handler	—	
	compareValue()	ToolkitException.UNAVAILABLE ELEMENT is thrown	
12	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F		
	Select Text String TLV Initialise compareBuffer		
	Select Text String TLV Initialise compareBuffer compareBuffer =		
	Select Text String TLV Initialise compareBuffer compareBuffer = 04 00 01 0F		
	Select Text String TLV Initialise compareBuffer compareBuffer = 04 00 01 0F Compare buffers	Result is 00h	
	Select Text String TLV Initialise compareBuffer compareBuffer = 04 00 01 0F Compare buffers valueOffset = 0	Result is 00h	
	Select Text String TLV Initialise compareBuffer compareBuffer = 04 00 01 0F Compare buffers	Result is 00h	
13	Select Text String TLV Initialise compareBuffer compareBuffer = 04 00 01 0F Compare buffers valueOffset = 0 compareOffset = 0 compareLength = 17	Result is 00h	

1	compareBuffer =		
	04 00 01 02 03		
	04 05 06 07 08		
	05 0A 0B 0C 0D		
	0E 10		
	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer		
17	compareBuffer =		
	03 00 01 OF		
	Compare buffers with same parameters	Result is +1	
	Compare bullers with same parameters	Result is +1	
	1.1.1.11		
15	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers	Result is 00h	
	valueOffset = 2		
	compareOffset = 3		
	compareLength = 12		
16	Initialise compareBuffer		
16	Initialise compareBuffer compareBuffer =		
16			
16	compareBuffer =		
16	compareBuffer = 55 55 55 02 01		
16	compareBuffer = 55 55 55 02 01 03 04 05 06 07		
16	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55	Result is -1	
16	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C	Result is -1	
16	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters	Result is -1	
	compareBuffer =	Result is -1	
	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters	Result is -1	
	compareBuffer =	Result is -1	
	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer compareBuffer = 55 55 55 01 02 03 04 05 06 07	Result is -1	
	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer compareBuffer = 55 55 55 01 02	Result is -1	
	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55		
17	compareBuffer = 55 55 55 02 01 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55 Compare buffers with same parameters Initialise compareBuffer compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters	Result is -1 Result is +1	
	compareBuffer =		
17	CompareBuffer =		
17	compareBuffer =		

6.2.7.13.4 Test Coverage

CRR number	Test case number
N1	12, 15
N2	13, 16
N3	14, 17, 18
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	Proactive Handler
C2	11

6.2.7.14 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference API_2_PAH_FACYB_BS

6.2.7.14.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public short findAndCopyValue(byte tag,

byte[] dstBuffer,
 short dstOffset)
throws java.lang.NullPointerException,
 java.lang.ArrayIndexOutOfBoundsException,
 ToolkitException

Normal execution

CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.14.2 Test Suite files

Test Script: API_2_PAH_FACYB_BS_1.scr

Test Applet: API_2_PAH_FACYB_BS_1.java

Load Script: API_2_PAH_FACYB_BS_1.ldr

Cleanup Script: API_2_PAH_FACYB_BS_1.clr

6.2.7.14.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	•	•
	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	InitDisplayText() with length = 15		
	<pre>dstOffset > dstBuffer.length tag = 0Dh dstBuffer.length = 20 dstOffset = 21</pre>	ArrayIndexOutOfBoundsException is thrown	
3	<pre>dstOffset < 0 dstBuffer.length = 20 dstOffset = -1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>length > dstBuffer.length dstBuffer.length = 15 dstOffset = 0</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>DstOffset + length >dstBuffer.length DstBuffer.length = 20 DstOffset = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
6	initDisplayText()		
	Select a TLV (tag 02h)		
	findAndCopyValue() tag = 03h	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE _ELEMENT is thrown.	

	LANGUAGE TO AN		
7	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F	D 11 (C 1A 1 1/1 C)	
	Successful call	Result of findAndcopyValue() is	
	Tag = 0Dh	17	
	DstBuffer.length = 17		
<u>_</u>	DstOffset = 0	D It : 001	
8	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
9	initialise dstBuffer		
	dstBuffer = 55 55 55		
	Successful call	Result of findAndcopyValue() is	
	dstBuffer.length = 20	19	
	dstOffset = 2		
10	Compare buffer	Result is 00h	
	buffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
44	interior to To 10		
11	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F		-
	append a 2nd Text String TLV	D 14 (6 1A 1 21 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2	
	Successful call	Result of findAndcopyValue() is	
	tag = 0Dh	17	
	dstBuffer.length = 17		
40	dstOffset = 0	Decult is 00h	
12	Compare buffer	Result is 00h	
42	buffer = 04 00 01 0F initDisplayText()		
13	dcs = 4		
	dcs = 4 buffer = 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndcopyValue() is	+
	tag = 8Dh		
	dstBuffer.length = 17	17	
	dstOffset = 0		
14	Compare buffer	Result is 00h	
'-	buffer = 04 00 01 0F	1.Coult is oon	
	201201 - 01 00 01 OF		
15	Append tag 0Fh		
13	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndcopyValue() is	
	tag = 8Fh	16	
	dstBuffer.length = 16	10	
	dstOffset = 0		
16	Compare buffer	Result is 00h	
'0	buffer = 00 01 0F	1.Codit io con	
	241251 - 00 01 01		

6.2.7.14.4 Test Coverage

CRR number	Test case number	
N1	8, 10, 12	
N2	6	
N3	7, 9, 11	
N4	13, 14, 15, 16	
P1	1	
P2	2, 3, 4, 5	
C1	Does not apply for	
	Proactive Handler	

6.2.7.15 Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference API_2_PAH_FACYBBS_BSS

6.2.7.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

Test Applet:

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.15.2 Test Suite files

Test Script: API_2_PAH_FACYBBS_BSS_1.scr

Load Script: API_2_PAH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_PAH_FACYBBS_BSS_1.clr

API_2_PAH_FACYBBS_BSS_1.java

6.2.7.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		·
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
_			
2	initDisplayText() with length = 15	A was the device the Control of Control	
	<pre>dstOffset > dstBuffer.length tag = 0Dh, occurrence = 1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
	valueOffset = 0	II is tillowii	
	<pre>dstBuffer.length = 5</pre>		
	<pre>dstOffset = 6 dstLength = 0</pre>		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = -1		
4	dstLength = 1 dstLength >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
4	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
_	dstLength = 6	A 1 1 0 (0/D 1 5);	
5	<pre>dstOffset + dstLength >dstBuffer.length dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
	dstOffset = 3	II IS UIIOWII	
	dstLength = 3		
6	<pre>dstLength < 0 dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 0	n is thrown	
	dstLength = -1		
7	initDisplayText() with length = 5	TablistEvantion OUT OF TIV	
	<pre>valueOffset > Text String Length tag = 0Dh, occurrence = 1</pre>	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	valueOffset = 7	BOONDAINES IS UNOWN	
	dstBuffer.length = 15		
	<pre>dstOffset = 0 dstLength = 0</pre>		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = -1</pre>	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 1		
9	dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 0 dstBuffer.length = 15</pre>	BOUNDARIES is thrown	
	dstOffset = 0		
	dstLength = 7		
10	valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 2 dstBuffer.length = 15</pre>	BOUNDARIES is thrown	
	dstOffset = 0		
	dstLength = 5		
11	InitDisplayText()		
- ' '	Select a TLV (tag 02h)		
	findAndCopyValue()	ToolkitException.UNAVAILABLE	
	tag = 0Dh	_ELEMENT is thrown	
	call the getValueLength() method	TablistEvantion UNIANAL ADJ.	
	Can the getvalueLength() method	ToolkitException.UNAVAILABLEELEMENT is thrown.	
12	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	<pre>valueOffset = 0</pre>	''	
	dstBuffer.length = 17		
	<pre>dstOffset = 0 dstLength = 17</pre>		
13	Compare buffer	Result is 00h	
-	buffer = 04 00 01 0F		
	1 10 10 10 10		
14	initialise dstBuffer	1	

	dstBuffer = 55 55 55	1	I
	Successful call	Result of findAndcopyValue() is	
	tag = 0Dh, occurrence = 1	15	
	valueOffset = 2		
	dstBuffer.length = 20		
	dstOffset = 3		
	dstLength = 12		
15	Compare buffer	Result is 00h	
	buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C 55 55 55 55 55		
16	Append a Text String TLV		
10	tag = 0D		
	buffer = 00 11 22 33 44 55 (no specific		
	DCS byte)		
	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 1	17	
	valueOffset = 0	1	
	dstBuffer.length = 17		
	dstOffset = 0		
	dstLength = 17		
17	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
18	Successful call	Result of findAndCopyValue() is	
	tag = 0Dh, occurrence = 2	6	
	valueOffset = 0		
	dstBuffer.length = 6		
	dstOffset = 0		
10	dstLength = 6	Result is 00h	
19	Compare buffer buffer = 00 11 22 33 44 55	Result is our	
20	initDisplayText()	+	
20	dcs = 4		
	buffer = 00 01 0F		
	Successful call (with tag 8Dh)	Result of findAndcopyValue() is	
	tag = 8Dh	17	
	occurrence = 1		
	valueOffset = 0		
	dstBuffer.length = 17		
	dstOffset = 0		
04	dstLength = 17	Deput is 00h	
21	Compare buffer	Result is 00h	
	buffer = 04 00 01 0F		
22	Append tag 0Fh	+	
22	buffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result of findAndcopyValue() is	
	tag = 8Fh	16	
	occurrence = 1	10	
	valueOffset = 0		
	dstBuffer.length = 16		
	dstOffset = 0		
	dstLength = 16		
23	Compare buffer	Result is 00h	
	bff 00 01 0E	i	İ
	buffer = 00 01 0F		

6.2.7.15.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18
N4	20, 21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for ProactiveHandler

6.2.7.16 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference API_2_PAH_FACRB_BS

6.2.7.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical returns 0.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.16.2 Test Suite files

Test Script: API_2_PAH_FACRB_BS_1.scr

Test Applet: API_2_PAH_FACRB_BS_1.java

Load Script: API_2_PAH_FACRB_BS_1.ldr

Cleanup Script: API_2_PAH_FACRB_BS_1.clr

6.2.7.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	Exponential	7.1. 2.0 Expositation
-	findAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
2	initDisplayText() with length = 15		
	<pre>compareOffset > compareBuffer.length tag = 0Dh</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
	<pre>compareBuffer.length = 20 compareOffset = 21</pre>		
3	<pre>compareOffset < 0 compareBuffer.length = 20 compareOffset = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
4	<pre>length > compareBuffer.length compareBuffer.length = 15 compareOffset = 0</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
5	compareOffset + length >	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length	n is thrown	
	<pre>compareBuffer.length = 20 compareOffset = 5</pre>		
6	InitDisplayText()		
	Select a TLV (tag 02h)		
	findAndCompareValue() tag = 03h	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE _ELEMENT is thrown.	
7	<pre>initDisplayText() dcs = 4 buffer = 00 01 0F</pre>		
	Initialise compareBuffer		
	compareBuffer = 04 00 01 0F		
	Compare buffers	Result is 00h	
	<pre>tag = 0Dh compareOffset = 0</pre>		
8	Verify current TLV getValueLength()	Result is 17	
9	Initialise compareBuffer		
	compareBuffer = 04 00 01 10		
	Compare buffers with same parameters	Result is -1	
10	Initialise compareBuffer compareBuffer =		
	03 00 01 OF		
	Compare buffers with same parameters	Result is +1	
11	Initialise compareBuffer compareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B 0C 0D 0E 0F 55		
	Compare buffers compareOffset = 2	Result is 00h	
40	annond a Toyl String TIV		
12	append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55		
	Initialise compareBuffer		
	compareBuffer =		
	55 55 04 00 01 02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		

	Compare buffers	Result is 00h	
	compareOffset = 2		
13	Initialise compareBuffer		
13	compareBuffer =		
	-		
	55 55 04 01 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is -1	
	compareOffset = 2		
14	Initialise compareBuffer		
	compareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OD 10 55		
		Decult is 14	
	Compare buffers	Result is +1	
	compareOffset = 2		
15	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	CompareBuffer = 04 00 01 0F		
	Successful call (with tag 8Dh)	Result is 00h	
	tag = 8Dh	result is oon	
	compareBuffer.length = 17		
	compareOffset = 0		
16			
16	Append tag 0Fh		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result is 00h	
	tag = 8Fh		
	compareBuffer.length = 16		
	compareOffset = 0		
17	Initialise compareBuffer		
	compareBuffer = 00 99 01 03 0F		
	Successful call (with tag 8Fh)	Result is +1	
	tag = 8Fh		
	compareBuffer.length = 16		
	compareOffset = 0		
L	1	1	

6.2.7.16.4 Test Coverage

CRR number	Test case number
N1	6
N2	8
N3	7, 11, 12, 17
N4	9, 13
N5	10, 14
N6	15, 16
P1	1
P2	2, 3, 4, 5
C1	Does not apply for Proactive Handler

6.2.7.17 Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference API_2_PAH_FACRBBS_BSS

6.2.7.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public byte findAndCompareValue(byte tag,

byte occurence,
short valueOffset,
byte[] compareBuffer,
short compareOffset,
short compareLength)

throws java.lang.NullPointerException,

java.lang.ArrayIndexOutOfBoundsException,

ToolkitException

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical 0 is returned.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

CRRP4: if an input parameter is not valid (e.g. occurence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.7.17.2 Test Suite files

Test Script: API_2_PAH_FACRBBS_BSS_1.scr

Test Applet: API_2_PAH_FACRBBS_BSS_1.java

Load Script: API_2_PAH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_PAH_FACRBBS_BSS_1.clr

6.2.7.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler		C Exposition
	findAndCompareValue() with a null	NullPointerException is thrown	<u> </u>
	compareBuffer	Train onto Exception is tillowil	l
2	initDisplayText() with length = 15	†	
	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	ı
	valueOffset = 0		ı
	compareBuffer.length = 5		l
	compareOffset = 6		l
3	compareLength = 0 compareOffset < 0	ArrayIndexOutOfBoundsExceptio	<u> </u>
3	compareBuffer.length = 5	n is thrown	
	compareOffset = -1	II IS UIIOWII	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = 0 compareLength = 6</pre>		l
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
J	>compareOffset + compareLength	n is thrown	
	compareBuffer.length = 5		l
	compareOffset = 3		l
	compareLength = 3	American de la Corp.	
6	<pre>compareLength < 0 compareBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	compareOffset = 0	n is thrown	l
	compareLength = -1		l
		<u> </u>	l
7	initDisplayText() with length = 5		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	valueOffset = 7		
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareLength = 0	<u> </u>	l
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	l
	compareBuffer.length = 15		
	<pre>compareOffset = 0 compareLength = 1</pre>		l
9	compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
Ŭ	valueOffset = 0	BOUNDARIES is thrown	l
	compareBuffer.length = 15		l
	compareOffset = 0		l
10	compareLength = 7 valueOffset + compareLength > Text String	ToolkitEvention OUT OF TIV	
10	length	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	valueOffset = 2	POONDAIVIES IS UITOWIT	l
	<pre>compareBuffer.length = 15</pre>		l
	compareOffset = 0		
	compareLength = 5		
11	Invalid parameter	ToolkitEvention BAD INDUT D	
1.1	occurrence = 0	ToolkitException.BAD_INPUT_P ARAMETER is thrown	
	1	ANAMIC LETA 19 HILOMII	l
12	InitDisplayText()		
_ · _	Select a TLV (tag 02h)	<u> </u>	
	findAndCompareValue()	ToolkitException.UNAVAILABLE	
	tag = 0Dh	_ELEMENT is thrown	l
	occurrence = 2		
	Call the getValueLength() method	ToolkitException.UNAVAILABLE	l
	1,551 1 = 10	_ELEMENT is thrown.	
13	initDisplayText()		
	dcs = 4 buffer = 00 01 0F		l
	Initialise compareBuffer	+	<u> </u>
	compareBuffer =		l
	04 00 01 0F		
	findAndCompareValue()	Result is 00h	

	T		
	tag = 0Dh, occurrence = 1		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 17		
14	Verify current TLV	Result is 17	
	getValueLength()		
15	Initialise compareBuffer		
13	compareBuffer =		
	04 00 01 10		
		Result is -1	
	Compare buffers with same parameters	Result is -1	
16	Initialise compareBuffer		
	compareBuffer =		
	03 00 01 0F	1	
	Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 01 02		
1	03 04 05 06 07		
1	08 09 0A 0B 0C		
	55 55 55 55 55	I D. Ivi oci	
	Compare buffers	Result is 00h	
1	valueOffset = 2		
	compareOffset = 3		
	compareLength = 12		
L	1.00.0		
18	Initialise compareBuffer		
	compareBuffer =		
	55 55 55 02 01		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55	D 11: 4	
	Compare buffers with same parameters	Result is -1	
—			
19	Initialise compareBuffer		
19	compareBuffer =		
19	compareBuffer = 55 55 55 01 02		
19	compareBuffer = 55 55 55 01 02 03 04 05 06 07		
19	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D		
19	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55	Describing 4	
19	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters	Result is +1	
19	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer =	Result is +1	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F		
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue()	Result is +1 Result is 00h	
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1		
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0		
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0		
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0		
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17		
	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer		
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer =		
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareDuffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue()		
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareDuffer = 00 compareDuffer = 00 compareDuffer = 00 compareDuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareDuffer = 00 11 22 33 44 55 Initialise compareBuffer compareDuffer = 00 compareDuffer = 00 compareDuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 0	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 55 55 55 55 55 55 55 55	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDuffer = Initialise compareBuffer compareBuffer =	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareLength = 6 Initialise compareBuffer compareDuffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffset = 0 compareDuffset = 0	Result is 00h Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareBuffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 IndAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 6 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue()	Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareLength = 17 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 findAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareDeffset = 0 compareD	Result is 00h Result is 00h	
20	compareBuffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0A 0D 55 55 55 55 55 Compare buffers with same parameters append a Text String TLV tag = 0Dh buffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 04 00 01 0F findAndCompareValue() tag = 0Dh, occurrence = 1 valueOffset = 0 compareOffset = 0 compareBuffer = 00 11 22 33 44 55 Initialise compareBuffer compareBuffer = 00 11 22 33 44 55 IndAndCompareValue() tag = 0Dh, occurrence = 2 valueOffset = 0 compareOffset = 0 compareOffset = 0 compareOffset = 6 Initialise compareBuffer compareBuffer = 00 11 22 33 44 66 findAndCompareValue()	Result is 00h Result is 00h	

			1
	compareLength = 6		
23	initDisplayText()		
	dcs = 4		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	CompareBuffer = 04 00 01 OF		
	Successful call (with tag 8Dh)	Result is 00h	
	tag = 8Dh, occurrence = 1		
	<pre>valueOffset = 0</pre>		
	compareBuffer.length = 17		
	compareOffset = 0		
	compareLength = 17		
24	Append tag 0Fh		
	buffer = 00 01 0F		
	Initialise compareBuffer		
	compareBuffer = 00 01 0F		
	Successful call (with tag 8Fh)	Result is 00h	
	tag = 8Fh, occurrence = 1		
	valueOffset = 0		
	compareBuffer.length = 16		
	compareOffset = 0		
	compareLength = 16		
25	Initialise compareBuffer		
	compareBuffer =0099 02 0F		
	findAndCompareValue()	Result is +1	
	tag = 0Dh, occurrence = 1		
	<pre>valueOffset = 0</pre>		
	compareOffset = 0		
	compareLength = 17		

6.2.7.17.4 Test Coverage

CRR number	Test case number
N1	12
N2	14
N3	13, 17, 20, 21
N4	15, 18, 22
N5	16, 19
N6	23, 24
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1	Does not apply for Proactive Handler

6.2.7.18 Method appendArray

Test Area Reference: API_2_PAH_APDA

6.2.7.18.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: appends a buffer into the Edithandler buffer

CRRN2: a successful append does not modify the TLV selected

Parameters error

CRRP1: if buffer is null, a java.lang.NullPointerException is thrown

CRRP2: if offset or length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.7.18.2 Test suite files

Test Script: API_2_PAH_APDA_1.scr

Test Applet: API_2_PAH_APDA_1.java

Load Script: API_2_PAH_APDA_1.ldr

Cleanup Script: API_2_PAH_APDA_1.clr

6.2.7.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null buffer	NullPointerException is thrown	Al Do Expediation
2	<pre>offset > buffer.length buffer.length = 5 offset = 6 length = 0</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
3	<pre>offset < 0 buffer.length = 5 offset = -1 length = 1</pre>	ArrayIndexOutOfBoundsException is thrown	
4	<pre>length > buffer.length buffer.length = 5 offset = 0 length = 6</pre>	ArrayIndexOutOfBoundsException is thrown	
5	<pre>offset + length > buffer.length buffer.length = 5 offset = 3 length = 3</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
6	<pre>length < 0 buffer.length = 5 offset = 0 length = -1</pre>	ArrayIndexOutOfBoundsExceptio n is thrown	
7	Handler overflow buffer.length = 256 offset = 0 length = 256	ToolkitException.HANDLER_OV ERFLOW is thrown	
8	Initialise handler		
	Select Command Details TLV		
	Successful call buffer = FF FE F8 offset = 0 length = 8 Verify Current TLV: Call getValueLength()	Result is 03h	
9	Clear the handler		
	Successful call		

	buffer = FF FE F8		
	offset = 0		
	length = 8		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8	javacard.framework.Util.arrayCo	
		mpare() is 00h	
10	Successful call		
'	buffer = 00 01 07		
	offset = 2		
	length = 6		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8 02 03 07	javacard.framework.Util.arrayCo	
	oomparebarrer if in it of or or in o,		
	0	mpare() is 00h	
11	Successful call		
	buffer = 11 22 88		
	offset = 2		
	length = 4		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = FF FE F8 02 03 07 33	javacard.framework.Util.arrayCo	
	44 55 66	mpare() is 00h	
12	Clear the handler		
	Successful call		
	buffer = 00 01 FC		
	offset = 0		
	length = 253		
	Call getLength() method	result = 253	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 00 01 FC	javacard.framework.Util.arrayCo	
		mpare() is 00h	
L	L		

6.2.7.18.4 Test Coverage

CRR number	Test case number
N1	9, 10, 11, 12
N2	8
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for ProactiveHandler

6.2.7.19 Method appendTLV(byte tag, byte value)

Test Area Reference: API_2_PAH_APTLBB

6.2.7.19.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

 $\begin{array}{ll} {\rm void\ appendTLV} & {\rm (byte\ tag,\ byte\ value)} \\ & {\rm throws\ ToolkitException} \end{array}$

Normal execution

CRRN1: Appends a TLV element to the current TLV list (1-byte element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code $HANDLER_NOT_AVAILABLE$

6.2.7.19.2 Test suite files

Test Script: API_2_PAH_APTLBB_1.scr
Test Applet: API_2_PAH_APTLBB_1.java
Load Script: API_2_PAH_APTLBB_1.ldr
Cleanup Script: API_2_PAH_APTLBB_1.clr

6.2.7.19.3 Test procedure

Call appendArray() Call appendArray()	
length = 251	
method ERFLOW is thrown 2	
2 Initialise handler Select Command Details TLV Call the appendTLV() method Verify Current TLV: Call getValueLength() 3 Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h	
Select Command Details TLV Call the appendTLV() method Verify Current TLV: Call getValueLength() Result is 03h Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
Select Command Details TLV Call the appendTLV() method Verify Current TLV: Call getValueLength() Result is 03h Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
Call the appendTLV() method Verify Current TLV: Call getValueLength() Result is 03h Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 101 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
Verify Current TLV: Call getValueLength() Result is 03h Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
3 Clear the handler Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
Successful call tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h	
tag = 84h value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
value = 00h Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Call copy() method pareBuffer = 84 01 00 01 01 FE Compare the arrays compareBuffer = 84 01 00 01 01 FE parallel copy() method pa	
Call copy() method Compare the arrays compareBuffer = 84 01 00 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h Result of javacard.framework.Util.arrayCo mpare() is 00h	
Compare the arrays compareBuffer = 84 01 00 4 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE CompareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h	
compareBuffer = 84 01 00 javacard.framework.Util.arrayCo mpare() is 00h 4 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE compareBuffer = 84 01 00 01 01 FE pavacard.framework.Util.arrayCo mpare() is 00h	
mpare() is 00h 4 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE gray Result of javacard.framework.Util.arrayCo mpare() is 00h	
4 Successful call tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE Result of javacard.framework.Util.arrayCo mpare() is 00h	
tag = 01h value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE gavacard.framework.Util.arrayCo mpare() is 00h	
value = FEh Call copy() method Compare the arrays compareBuffer = 84 01 00 01 01 FE pavacard.framework.Util.arrayCo mpare() is 00h	
Compare the arrays compareBuffer = 84 01 00 01 01 FE javacard.framework.Util.arrayCo mpare() is 00h	
compareBuffer = 84 01 00 01 01 FE javacard.framework.Util.arrayCo mpare() is 00h	
mpare() is 00h	
5 Clear the handler	
Call appendArray()	
length = 250 buffer = 00 81 F7 03 04 F9	
Successful call	
tag = 84h	
value = 00h	
Call getLength() method result = 253	
Oall saw A wath all	
Call copy() method	
Compare the array Result of	
compareBuffer = 00 81 F7 03 04 F9 84 01 javacard.framework.Util.arrayCo	
mpare() is 00h	

6.2.7.19.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5
N2	2
C1	1
C2	Does not apply for
	Proactive Handler

6.2.7.20 Method appendTLV(byte tag, byte value1, byte value2)

Test Area Reference: API_2_PAH_APTLBBB

6.2.7.20.1 Conformance requirements:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Appends a TLV element to the current TLV list (2-byte element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER NOT AVAILABLE

6.2.7.20.2 Test suite files

Test Script: API_2_PAH_APTLBBB_1.scr

Test Applet: API_2_PAH_APTLBBB_1.java

Load Script: API_2_PAH_APTLBBB_1.ldr

Cleanup Script: API_2_PAH_APTLBBB_1.clr

6.2.7.20.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Call the initDisplayText()	74 i Exposition	Do Expositation
'	length = 250		
	Handler Overflow: Call the appendTLV()	ToolkitException.HANDLER_OV	
	method	ERFLOW is thrown	
2	Initialise handler		
	Select Command Details TLV		
	Call the appendTLV() method		
	Verify Current TLV: Call getValueLength()	Result is 03h	
3	Clear the handler		
	Successful call		
	tag = 84h		
	value1 = 00h		
-	value2 = 01h	-	
-	Call copy() method	Dogult of	
	Compare the arrays compareBuffer = 84 02 00 01	Result of	
	Comparebuller = 84 02 00 01	javacard.framework.Util.arrayCo	
1	Successful call	mpare() is 00h	
4	tag = 01h		
	value1 = FEh		
	value2 = FDh		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 84 02 00 01 01 02 FE FD	javacard.framework.Util.arrayCo	
		mpare() is 00h	
5	Clear the handler		
	Call appendArray()		
	length = 249		
	buffer = 00 81 F6 03 04 F8		
	Successful call		
	tag = 84h value1 = 00h		
	value2 = 00h		
	Call getLength() method	result = 253	
		100an - 200	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 00 81 F6 03 04 F8 84 02	javacard.framework.Util.arrayCo	
	00 01	mpare() is 00h	
	I.	I I V	

6.2.7.20.4 Test Coverage

CRR number	Test case number
N1	3, 4, 5
N2	2
C1	1
C2	Does not apply for
	Proactive Handler

6.2.7.21 Method appendTLV(byte tag, byte[] value, short valueoffset, short valuelength)

Test Area Reference: API_2_PAH_APTLB_BSS

6.2.7.21.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

void appendTLV (byte tag,

byte[] value,
 short valueoffset,
 short valuelength)
throws java.lang.NullPointerException,
 java.lang.ArrayIndexOutOfBoundsException,
 ToolkitException

Normal execution

CRRN1: Appends a TLV element to the current TLV list (byte-array element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

CRRP1: if value is null, a java.lang.NullPointerException is thrown

CRRP2: if valueoffset or valuelength or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER NOT AVAILABLE

CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.7.21.2 Test suite files

Test Script: API_2_PAH_APTLB_BSS_1.scr

Test Applet: API_2_PAH_APTLB_BSS_1.java

Load Script: API_2_PAH_APTLB_BSS_1.ldr

Cleanup Script: API_2_PAH_APTLB_BSS_1.clr

6.2.7.21.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value	NullPointerException is thrown	•
2	valueOffset > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 6		
	valueLength = 0		
3	valueOffset < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = -1		
	valueLength = 1		
4	valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 0		
	valueLength = 6		
5	valueOffset + valueLength > value.length	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 3		
	valueLength = 3		
6	valueLength < 0	ArrayIndexOutOfBoundsExceptio	
	value.length = 5	n is thrown	
	valueOffset = 0		
<u> </u>	valueLength = -1		
7	Handler overflow	ToolkitException.HANDLER_OV	
	value.length = 254	ERFLOW is thrown	
	valueOffset = 0		

	realwat anoth - 254	T	
0	valueLength = 254	ToolkitEveention DAD INDUT D	
8	Bad parameter value.length = 256	ToolkitException.BAD_INPUT_P ARAMETER is thrown	
	valueOffset = 0	ARAMETER IS thrown	
	valueLength = 256		
9	Initialise handler		
	Select Command Details TLV		
	Successful call		
	tag = 04		
	value = FF FE F8		
	<pre>valueOffset = 0</pre>		
	valueLength = 8		
	Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler		
	Successful call		
	tag = 04		
	<pre>value = FF FE F8 valueOffset = 0</pre>		
	valueLength = 8		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 08 FF FE F8	javacard.framework.Util.arrayCo	
	_	mpare() is 00h	
11	Successful call		
	tag = 85h		
	value = 00 01 07		
	<pre>valueOffset = 2</pre>		
	valueLength = 6		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 08 FF FE F8 85 06 02 03 07	javacard.framework.Util.arrayCo	
4.0		mpare() is 00h	
12	Successful call		
	tag = 01 value = 11 22 88		
	valueOffset = 2		
	valueLength = 4		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 08 FF FE F8 85 06 02	javacard.framework.Util.arrayCo	
	03 07 01 04 33 44 55 66	mpare() is 00h	
13	Clear the handler		
	Successful call		
	tag = 04		
	<pre>value = 00 01 7F valueOffset = 0</pre>		
	valueOffset = 0 valueLength = 80h		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 81 80 00 017F	javacard.framework.Util.arrayCo	
	·	mpare() is 00h	
14	Clear the handler		
	Successful call		
	tag = 04		
	value = 00 01 F9		
	valueOffset = 0		
	valueLength = 250 Call gatLangth() mathod	result = 253	
	Call getLength() method	169011 = 200	
	Call copy() method		
	Can copy() memou		
	Compare handler	Result of	
	compareBuffer = 04 81 FA 00 01F9	javacard.framework.Util.arrayCo	
	<u> </u>	mpare() is 00h	
		1	

6.2.7.21.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13, 14
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for
	Proactive Handler
C3	8

6.2.7.22 Method appendTLV(byte tag, byte value1, byte[] value2, short value2offset, short value2length)

Test Area Reference: API_2_PAH_APTLBB_BSS

6.2.7.22.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

throws java.lang.NullPointerException,

java.lang.ArrayIndexOutOfBoundsException,

ToolkitException

Normal execution

CRRN1: Appends a TLV element to the current TLV list (1 byte and a byte-array element).

CRRN2: A successful append does not modify the TLV selected.

Parameters error

CRRP1: if value2 is null, a java.lang.NullPointerException is thrown

CRRP2: if value2offset or value2length or both would cause access outside the array bounds, or if length is negative, a java.lang.ArrayIndexOutOfBoundsException is thrown.

Context errors

CRRC1: if the EditHandler buffer is too small to append the requested data, a ToolkitException is thrown with reason code HANDLER_OVERFLOW

CRRC2: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

CRRC3: if valuelength is greater than 255, a ToolkitException is thrown with reason code BAD_INPUT_PARAMETER

6.2.7.22.2 Test suite files

Test Script: API_2_PAH_APTLBB_BSS_1.scr

Test Applet: API_2_PAH_APTLBB_BSS_1.java

Load Script: API_2_PAH_APTLBB_BSS_1.ldr

Cleanup Script: API_2_PAH_APTLBB_BSS_1.clr

6.2.7.22.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Null value2	NullPointerException is thrown	, c pootation
2	value2Offset > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	<pre>value20ffset = 6</pre>		
	value2Length = 0	A	
3	<pre>value2Offset < 0 value2.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	value20ffset = -1	n is thrown	
	value2Length = 1		
4	value2Length > value2.length	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value2Offset = 0		
_	value2Length = 6	A de de control D	
5	<pre>value2Offset + value2Length > value2.length value2.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	value20ffset = 3	n is thrown	
	value2Length = 3		
6	value2Length < 0	ArrayIndexOutOfBoundsExceptio	
	value2.length = 5	n is thrown	
	value2Offset = 0		
	value2Length = -1 Handler overflow	ToolkitEveention LIANDLED OV	
7	value2.length = 254	ToolkitException.HANDLER_OV ERFLOW is thrown	
	value20ffset = 0	FIZE FOAM 19 (IIIIOMII	
	value2Length = 254		
8	Bad parameter	ToolkitException.BAD_INPUT_P	
	value2.length = 256	ARAMETER is thrown	
	value20ffset = 0		
9	value2Length = 256 Initialise handler		
9	Select Command Details TLV		
	Successful call		
	tag = 04		
	value1 = 05		
	value2 = FF FE F8		
	value20ffset = 0		
	value2Length = 8 Verify Current TLV: Call getValueLength()	Result is 03h	
10	Clear the handler	result is osii	
	Successful call		
	tag = 04		
	value1 = 05		
	value2 = FF FE F8		
	<pre>value2Offset = 0 value2Length = 8</pre>		
	Call copy() method		
	Compare the arrays	Result of	
	CompareBuffer = 04 09 05 FF FE F8	javacard.framework.Util.arrayCo	
L		mpare() is 00h	
11	Successful call		
	tag = 85h		
	value1 = 55h value2 = 00 01 07		
	value20ffset = 2		
L	value2Length = 6		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer =	javacard.framework.Util.arrayCo	
	04 09 05 FF FE F8 85 07 55 02 03 07	mpare() is 00h	
12	Successful call		
'-	tag = 01		
	value1 = 44h		
	value2 = 11 22 88		
	<pre>value2Offset = 2 value2Length = 4</pre>		
	Call copy() method		
	Compare the arrays	Result of	
	CompareBuffer =	javacard.framework.Util.arrayCo	
	04 09 05 FF FE F8	mpare() is 00h	
L			

	85 07 55 02 03 07		
	01 05 44 33 44 55 66		
13	Clear the handler		
	Successful call		
	tag = 04		
	value1 = 00		
	value2 = 01 7F		
	<pre>value20ffset = 0</pre>		
	value2Length = 7Fh		
	Call copy() method		
	Compare the arrays	Result of	
	compareBuffer = 04 81 80 00 017F	javacard.framework.Util.arrayCo	
		mpare() is 00h	
14	Clear the handler		
1.			
	Successful call		
	tag = 04		
	value1 = 00		
	value2 = 01 F9		
	<pre>value20ffset = 0</pre>		
	value2Length = 249		
	Call getLength() method	result = 253	
	Call copy() method		
	Compare handler	Result of	
	compareBuffer = 04 81 FA 00 01F9	javacard.framework.Util.arrayCo	

6.2.7.22.4 Test Coverage

CRR number	Test case number
N1	10, 11, 12, 13, 14
N2	9
P1	1
P2	2, 3, 4, 5, 6
C1	7
C2	Does not apply for
	Proactive Handler
C3	8

6.2.7.23 Method clear

Test Area Reference: API_2_PAH_CLER

6.2.7.23.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Clears the TLV list of an EditHandler

CRRN2: Resets the current TLV selected.

Parameters error

No requirements

Context errors

CRRC1: if the EditHandler buffer is busy, a ToolkitException is thrown with reason code HANDLER_NOT_AVAILABLE

6.2.7.23.2 Test suite files

Test Script: API_2_PAH_CLER_1.scr

Test Applet: API_2_PAH_CLER_1.java

Load Script: API_2_PAH_CLER_1.ldr

Cleanup Script: API_2_PAH_CLER_1.clr

6.2.7.23.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Initialise the handler	Result of getLength() is not null	
	Select Command Details TLV		
	Call the getLength() method		
	Clear the handler	Result of getLength() is 0	
	Call the getLength() method		
2	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
		_ELEMENT is thrown	

6.2.7.23.4 Test Coverage

CRR number	Test case number
N1	1
N2	2
C1	Does not apply for
	Proactive Handler

6.2.8 Class ProactiveResponseHandler

6.2.8.1 Method copyAdditionalInformation

Test Area Reference: API_2_PRH_CPAI_BSS

6.2.8.1.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The copyAdditionalInformation() method shall copy a part of the additional information field from Result TLV element in dstBuffer, using dstOffset and dstLength.

CRRN2: dstBuffer shall only be modified from <math>dstOffset to (dstOffset + dstLength - 1) (included).

CRRN3: The method returns (dstOffset + dstLength).

CRRN4: If a Result TLV element is available, it becomes the TLV selected after a call to the method.

CRRN5: The method shall copy from the first Result TLV.

Parameter errors

CRRP1: A NullPointerException shall be thrown if dstBuffer is null.

CRRP2: An ArrayIndexOutOfBoundsException shall be thrown if dstOffset or dstLength or both would cause access outside array bounds.

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.

CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if dstLength is greater than the value field of the available TLV.

6.2.8.1.2 Test Suite files

Test Script: API_2_PRH_CPAI_BSS_1.scr

Test Applet: API_2_PRH_CPAI_BSS _1.java

Load Script: API_2_PRH_CPAI_BSS _1.ldr

Cleanup Script: API_2_PRH_CPAI_BSS _1.clr

6.2.8.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 0		command
	dcs = 4		
	buffer = "Text"		
	Terminal Response with 11 additional bytes		
	Result TLV = 03 0C 01 01 23 45 67 89 AB CD EF 01 23 45		
	NULL as parameter to dstBuffer dstBuffer = NULL	NullPointerException is thrown	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 11		
	dstLength = 0	1 1 1 0 10/0 1 5	
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 10 dstOffset = -1</pre>	n is thrown	
	dstOffset = -1 dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
-	dstBuffer.length = 10	n is thrown	
	dstOffset = 0	II IS UII OWII	
	dstLength = 11		
5	dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 6		
	dstLength = 5		
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 10	n is thrown	
	dstOffset = 6		
7	dstLength = -1 Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
′	Build and send a DISPLAT TEXT command		command
	Terminal Response with 5 additional bytes		Command
	Tomma Reopence Will Cadamona Dytes		
	Result TLV = 03 06 01 01 23 45 67 89		
1	Į.	Ĭ.	į

	Successfull call, dstBuffer is the whole buffer	result of	
1	· ·		
	dstBuffer.length = 5	copyAdditionalInformation() is	
	dstOffset = 0	05h.	
	dstLength = 5	0011.	
8	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
_		room or array company() to com	
	src = {01, 23, 45, 67, 89}		
	srcOffset = 00		
1	dest = dstBuffer		
1	destOffset = 0		
1			
1	length = 5		
9	Call the getValueLength() method	Result is 06h.	
9	Can the getvalueLength() method	Result is oon.	
40	Build and send a DISPLAY TEXT command		DIODI AVITEVT Describes
10	Build and Send a DISPLAY TEXT COMMand		DISPLAY TEXT Proactive
			command
			Communa
	Terminal Response with 6 additional bytes		
	'		
	Result TLV = 03 07 01 AB CD EF FE DC BA		
	Successfull call, dstBuffer is part of a buffer	result of	
	dstBuffer.length = 7	copyAdditionalInformation() is	
	dstOffset = 2		
1		07h.	
L	dstLength = 5		
11	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
' '	Compare desibulier demy arraycompare()	result of arraycompare() is out.	
1			
	src = {AB, CD, EF, FE, DC}		
1			
	srcOffset = 00		
	dest = dstBuffer		
1			
1	destOffset = 2		
	length = 5		
40			DIODI AVITEVT D
12	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
			command
			Command
	Terminal Response with 7 additional bytes		
	,		
	Result TLV = 03 08 01 FE DC BA 98 76 54		
	32		
	Successfull call, dstBuffer is part of a buffer	result of	
	dstBuffer.length = 7		
	9	copyAdditionalInformation() is	
	dstOffset = 0	05h.	
	dstLength = 5	0011.	
			<u> </u>
13	Compare dstBuffer using arrayCompare()	result of arrayCompare() is 00h.	
	(FF FG F3 00 FG)		
	$src = {FE, DC, BA, 98, 76}$		
	<pre>srcOffset = 00</pre>		
1			
	dest = dstRuffer		
	dest = dstBuffer		
	<pre>dest = dstBuffer destOffset = 0</pre>		
	destOffset = 0		
	<pre>destOffset = 0 length = 5</pre>		DIODI AVITEVI D
14	destOffset = 0		DISPLAY TEXT Proactive
14	<pre>destOffset = 0 length = 5</pre>		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
14	<pre>destOffset = 0 length = 5</pre>		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77	rocult of	
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer	result of	
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9	copyAdditionalInformation() is	
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2		
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5	copyAdditionalInformation() is 07h.	
14	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2	copyAdditionalInformation() is	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare()	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare()	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44}	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44}	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2	copyAdditionalInformation() is 07h.	
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5	copyAdditionalInformation() is 07h.	command
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5	copyAdditionalInformation() is 07h.	command
	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5	copyAdditionalInformation() is 07h.	command
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes	copyAdditionalInformation() is 07h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method dstBuffer.length = F2h	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h. result of copyAdditionalInformation() is	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method dstBuffer.length = F2h dstOffset = 0	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method dstBuffer.length = F2h	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h. result of copyAdditionalInformation() is	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method dstBuffer.length = F2h dstOffset = 0 dstLength = F2h	result of copyAdditionalInformation() is 07h. result of arrayCompare() is 00h. result of copyAdditionalInformation() is F2h.	DISPLAY TEXT Proactive
15	destOffset = 0 length = 5 Build and send a DISPLAY TEXT command Terminal Response with 8 additional bytes Result TLV = 03 09 01 00 11 22 33 44 55 66 77 Successfull call, dstBuffer is the whole buffer dstBuffer.length = 9 dstOffset = 2 dstLength = 5 Compare dstBuffer using arrayCompare() src = {00, 11, 22, 33, 44} srcOffset = 00 dest = dstBuffer destOffset = 2 length = 5 Build and send a DISPLAY TEXT command Terminal Response with F2h additional bytes Result TLV = 03 81 F3 01 00 01 02 03 Successfull call to the method dstBuffer.length = F2h dstOffset = 0	copyAdditionalInformation() is 07h. result of arrayCompare() is 00h. result of copyAdditionalInformation() is	DISPLAY TEXT Proactive

arcoffeet = 00 dest = datBuffer destOffeet = 0 length = PTh		src = {00, 01, 02, 03, 04}	1	T
deat = datBuffer deatOffSet = 0 length = F2h				
Length = F2h Call the getValueLength() method Result is F3h.				
Result is F3h.		destOffset = 0		
19 Build and send a DISPLAY TEXT command				
Terminal Response with 5 additional bytes Result TLV = 03 06 01 00 11 22 33 44	18	Call the getValueLength() method	Result is F3h.	
Terminal Response with 5 additional bytes Result TLV = 03 06 01 00 11 22 33 44 dstLength > data available dstBuffer .length = 6 dstLength = 10 dstLength = 10 dstLength = 10 dstLength = 10 dstLength = 10 dstLength = 20 dstOffset = 5 dstLength = 5 Compare dstBuffer using arrayCompare() src = { 00h, 01h, 02h, 03h, 04h, 00h, 11h, 22h, 33h, 44h, 03h, 03h, 03h, 04h, 06h, 03h, 0ch, 06h, 06h, 06h, 10h, 11h, 12h, 13h) srcoffset = 0 deat = dstBuffer destOffset = 0 deat = dstBuffer dstBuffer.length = 20 21 Build and send a DISPLAY TEXT command Terminal Response with 2 Result TLV elements lst Result TLV = 03 05 01 01 23 45 67 89 2nd Result TLV = 03 05 01 01 02 23 45 67 89 2nd Result TLV = 03 01 00 Successful call to copyAdditionalInformation() dstBuffer.length = 0 dstLength = 5 Compare dstBuffer using arrayCompare() src = (01, 23, 45, 67, 89) srcoffset = 00 deat = dstBuffer dstoffset = 0 dstoffset = 00 deat = dstBuffer dstoffset = 00 deat = dstBuffer dstoffset = 00 dstoffs				
Terminal Response with 5 additional bytes Result TLV = 03 06 01 00 11 22 33 44	19	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
Result TLV = 03 06 01 00 11 22 33 44				command
dstLength > data available dstBuffer.length = 6 dstCffset = 0 dstLength = 6 dstLength = 5 dstLengt		Terminal Response with 5 additional bytes		
dstLength > data available dstBuffer.length = 6 dstCffset = 0 dstLength = 6 dstLength = 5 dstLengt				
detbuffer_length = 6 detbuffer_length = 6 detbuffer = 6 detbuffer = 6		Result TLV = 03 06 01 00 11 22 33 44		
detbuffer_length = 6 detbuffer_length = 6 detbuffer_length = 6		dati angth - data available	OUT OF THE POLINDADIES	
detLength = 6				
dethength = 6		-	TOOIKILEXCEPTION IS THOWIT	
Terminal Response with 5 additional bytes Result TLV = 03 06 01 00 11 22 33 44		dstLength = 6		
Terminal Response with 5 additional bytes	20	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
Result TLV = 03 06 01 00 11 22 33 44				command
Initialise dstBuffer {00h, 01h, 02h, 03h}		Terminal Response with 5 additional bytes		
Initialise dstBuffer {00h, 01h, 02h, 03h}				
dstBuffer = {00h, 01h, 02h, 03h} Call the copyAdditionalInformation() method dstBuffer.length = 20 dstOffset = 5 dstLength = 5				
Call the copyAdditionalInformation() method dstBuffer length = 20 dstOffset = 5 dstLength = 5 Compare dstBuffer using arrayCompare() src = { 00h, 01h, 02h, 03h, 04h, 00h, 11h, 22h, 33h, 44h, 0Ah, 0Bh, 0Ch, 0Dh, 0Eh, 0Fh, 10h, 11h, 12h, 13h} srcoffset = 0 dest = dstBuffer destOffset = 0 length = 20 21 Build and send a DISPLAY TEXT command Terminal Response with 2 Result TLV elements lst Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 dstOffset = 0 dest = dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcoffset = 00 length = 5 Compare dstBuffer using arrayCompare() est of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} srcoffset = 00 length = 5 Call the getValueLength() method Result is 06h.				
dstBuffer length = 20 dstUength = 5 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h src = { 00h, 01h, 02h, 03h, 04h, 00h, 11h, 22h, 33h, 44h, 00h, 00h, 11h, 12h, 13h} srcoffset = 0 dest = dstBuffer destOffset = 0 length = 20 21 Build and send a DISPLAY TEXT command DISPLAY TEXT Proactif command Terminal Response with 2 Result TLV elements 1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 copyAdditionalInformation() dstBuffer length = 5 dstOffset = 0 dstLength = 5 dstOffset = 0 dstLength = 5 dstOffset = 0 dest = dstBuffer destOffset = 00 dest = dstBuffer destOffset = 00 dest = dstBuffer destOffset = 00 length = 5 dstOffset = 00 length = 5 destOffset = 0 length = 5 dstOffset = 0 length = 5 length =				
dstOffset = 5 dstLength = 5 Compare dstBuffer using arrayCompare() src = { 00h, 01h, 02h, 03h, 04h, 00h, 11h, 22h, 33h, 44h, 0Ah, 08h, 0Ch, 0Dh, 0Eh, 0Fh, 10h, 11h, 12h, 13h} srcOffset = 0 dest = dstBuffer destOffset = 0 length = 20 Ist Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcOffset = 0 dest = dstBuffer destOffset = 0 dstLength = 5 dstOffset = 0 dstLength = 5 dstOffset = 0 dest = dstBuffer destOffset = 0 length = 5 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction DIS				
dstLength = 5 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h				
Compare dstBuffer using arrayCompare() src = {				
00h, 01h, 02h, 03h, 04h, 00h, 11h, 22h, 33h, 44h, 00h, 08h, 0ch, 0ph, 0ph, 0ph, 10h, 11h, 12h, 13h} srcoffset = 0 dest = dstBuffer destOffset = 0 length = 20			result of arrayCompare() is 00h	
00h, 11h, 22h, 33h, 44h, 0Ah, 0Bh, 0Ch, 0Dh, 0Eh, 0Ph, 10h, 11h, 12h, 12h, 13h} src0ffset = 0 dest = dstBuffer destoffset = 0 length = 20				
0Ah, 0Bh, 0Ch, 0Dh, 0Eh, 0Fh, 10h, 11h, 12h, 13h} srcoffset = 0 dest = dstBuffer destoffset = 0 length = 20 21 Build and send a DISPLAY TEXT command Terminal Response with 2 Result TLV elements 1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstoffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcoffset = 00 dest = dstBuffer destoffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction DISPLAY TEXT Proaction Result is 06h.				
OPh, 10h, 11h, 12h, 13h} src0ffset = 0 0 dest = dstBuffer destOffset = 0 length = 20 DISPLAY TEXT command DISPLAY TEXT Proactive Terminal Response with 2 Result TLV elements lst Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 result of CopyAdditionalInformation() dstBuffer.length = 5 05h. dstOffset = 0 05h. dstLength = 5 05h. 22 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} result of arrayCompare() is 00h. srcoffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. 24 Build and send a DISPLAY TEXT command DISPLAY TEXT Proactive				
srcOffset = 0 dest = dstBuffer destOffset = 0 length = 20 21 Build and send a DISPLAY TEXT command Terminal Response with 2 Result TLV elements lst Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction DISPLAY TEXT Proact				
destOffset = 0 length = 20 21 Build and send a DISPLAY TEXT command Terminal Response with 2 Result TLV elements 1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction DISPLAY TEXT P				
length = 20		dest = dstBuffer		
DISPLAY TEXT Proactive command DISPLAY TEXT Proactive command				
Terminal Response with 2 Result TLV elements 1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} srcoffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction	24	,		DICDLAY TEXT Drop of the
Terminal Response with 2 Result TLV elements 1st Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proaction	21	Build and Send a DISPLAT TEXT command		
elements lst Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proactive DISPLAY T		Terminal Pesnense with 2 Pesult TI V		Command
lst Result TLV = 03 06 01 01 23 45 67 89 2nd Result TLV = 03 01 00 Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proactive DISPLAY TEXT Proactive Result of copyAdditionalInformation() is copyAdditionalInformation() is copyAdditionalInformation() is 05h. Result of arrayCompare() is 00h.				
Successfull call to CopyAdditionalInformation() CopyAddition		elements		
Successfull call to copyAdditionalInformation() dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22 Compare dstBuffer using arrayCompare() src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. result of copyAdditionalInformation() is 05h. result of arrayCompare() is 00h.		1st Result TLV = 03 06 01 01 23 45 67 89		
copyAdditionalInformation() dstBuffer.length = 5 copyAdditionalInformation() is 05h. 05h. 22 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} srcoffset = 00 dest = dstBuffer destOffset = 0 length = 5 Result is 06h. 24 Build and send a DISPLAY TEXT command		2nd Result TLV = 03 01 00		
dstBuffer.length = 5 dstOffset = 0 dstLength = 5 22				
dstoffset = 0 dstLength = 5 22			1	
dstLength = 5 22			05h.	
22 Compare dstBuffer using arrayCompare() result of arrayCompare() is 00h. src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proactive				
src = {01, 23, 45, 67, 89} srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23	22		result of arravCompare() is 00h	
srcOffset = 00 dest = dstBuffer destOffset = 0 length = 5 23 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proactive				
dest = dstBuffer destOffset = 0 length = 5 23				
destOffset = 0 length = 5 Call the getValueLength() method Result is 06h. DISPLAY TEXT Proactive Display Text Pr				
length = 5 Call the getValueLength() method Result is 06h. 24 Build and send a DISPLAY TEXT command DISPLAY TEXT Proactive DISPLAY				
23 Call the getValueLength() method Result is 06h. 24 Build and send a DISPLAY TEXT command DISPLAY TEXT Proactive				
24 Build and send a DISPLAY TEXT command DISPLAY TEXT Proactive	23		Result is 06h.	
command	24	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
				command
Terminal Response without Result Simple ToolkitException.UNAVAILABLE				
TLVELEMENT is thrown by send()		TLV	_ELEMENT is thrown by send()	
B. C. B. C. B. C. T. H. H. T. C. T. C.		B		
ProactiveResponseHandler, getTheHandler ToolkitException.UNAVAILABLE				
call copyAdditionalInformation() _ELEMENT is thrown		call copyAdditionalInformation()	ELEMENI is thrown	

6.2.8.1.4 Test Coverage

CRR number	Test case number
N1	8, 11, 13, 15, 17, 20, 22
N2	20
N3	7, 10, 12, 14, 16, 21
N4	9, 18, 23
N5	21, 22, 23
P1	1
P2	2, 3, 4, 5, 6
C1	24
C2	19

6.2.8.2 Method copyTextString

Test Area Reference: API_2_PRH_CPTS_BS

6.2.8.2.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The copyTextString() method copies the text string value from the first Text String TLV element, using dstBuffer and dstOffset.

CRRN2: If a Text String TLV element is available, it becomes the TLV selected.

CRRN3: The method returns (dstOffset + length of copied value).

Parameter errors

CRRP1: A NullPointerException shall be thrown if dstBuffer is null.

CRRP2: A ArrayIndexOutOfBoundsException shall be thrown if dstOffset or dstOffset + (length of the TextString to be copied, without the Data Coding Scheme included), as specified for the returned value, would cause access outside array bounds.

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.

6.2.8.2.2 Test Suite files

Test Script: API_2_PRH_CPTS_BS_1.scr

Test Applet: API_2_PRH_CPTS_BS_1.java

Load Script: API_2_PRH_CPTS_BS_1.ldr

Cleanup Script: API_2_PRH_CPTS_BS_1.clr

6.2.8.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a GET INPUT command	Al I Expediation	GET INPUT Proactive
1	qualifier = 00h		command
	dcs = 04h		Command
	buffer = 'Text'		
	minRespLength = 00h		
	maxRespLength = FFh		
	Terminal Response		
	•		
	Text String TLV = 0D 02 04 41		
	ProactiveResponseHandler.getTheHandler();	NullPointerException is thrown	
	call the copyTextString() method with a null		
	dstBuffer		
	1.5.66		
	dstBuffer = null		
2	dstOffset = 0 Build and send a GET INPUT command		GET INPUT Proactive
	Build and Send a GET INFOT Command		command
			Proactive
	Terminal Decrease		Proactive
	Terminal Response		
	Text String TLV = 0D 04 04 "ABC"		
	dstOffset + text length > dstBuffer.length	ArrayIndexOutOfBoundsException	
	and the state of t	is thrown	
	dstBuffer.length = 04h		
	dstOffset = 02h		
3	dstOffset < 0	ArrayIndexOutOfBoundsException	
		is thrown	
	dstBuffer.length = 04h		
	dstOffset = -1		DIODI AV TEVT
4	Build and send a DISPLAY TEXT command		DISPLAY TEXT
	qualifier = 00h dcs = 04h		Proactive command
	buffer = 'Text'		
	Terminal Response without Text String TLV		
	3		
	ProactiveResponseHandler.getTheHandler();	UNAVAILABLE ELEMENT	
	call the copyTextString() method	ToolkitException is thrown	
5	Build and send a GET INPUT command		GET INPUT Proactive
			command
			Proactive
	Terminal Response with a null Text String TLV		
	Text String TLV = 0D 00		
	Initialise dstBuffer		
	dstBuffer = {F00h, F01h, F02h, F03h}	Docult of constant Christin (A) in CO	
	Call the copyTextString() method	Result of copyTextString() is 02h	
	dstBuffer.length = 04h		
	dstOffset = 02h		
6	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {0F0h, 0F1h, 0F2h, 0F3h}		
	<pre>srcOffset = 00h dest = dstBuffer</pre>		
	dest = dstBuffer destOffset = 00h		
	length = 04h		
7	Build and send a GET INPUT command		GET INPUT Proactive
			command
			Proactive
	Terminal Response with text length = 01h		
	2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1		
	Text String TLV = 0D 02 04 41		
	Initialise dstBuffer		
	dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyTextString() method	Result of copyTextString() is 01h	
	dstBuffer.length = 04h		
l	dstOffset = 00h	I	I

8	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {41h, 01h, 02h, 03h}		
	srcOffset = 00h		
	<pre>dest = dstBuffer destOffset = 00h</pre>		
	length = 04h		
9	Build and send a GET INPUT command		GET INPUT Proactive
			command Proactive
	Terminal Response with text length = 02h		Proactive
	_		
	Text String TLV = 0D 03 04 42 43 Initialise dstBuffer		
	dstBuffer = {00h, 01h, 02h, 03h}		
	Call the copyTextString() method	Result of copyTextString() is 04h	
	dstBuffer.length = 04h		
	dstOffset = 02h		
10	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {00h, 01h, 42h, 43h}		
	srcOffset = 00h		
	<pre>dest = dstBuffer destOffset = 00h</pre>		
	length = 04h		
11	Call the getValueLength() method	Result is 03h	
12	Build and send a GET INPUT command		GET INPUT Proactive
L			command
	Terminal Response with text length = 7Eh		
	Text String TLV = 0D 7F 04 01 02 7E		
	Initialise dstBuffer		
	dstBuffer = {00h, 00h 00h}	D	
	Call the copyTextString() method	Result of copyTextString() is 7Eh	
	dstBuffer.length = 7Eh		
13	dstOffset = 00h Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
10		recount of array compare() to com	
	<pre>src = {01h,, 7Eh} srcOffset = 00h</pre>		
	dest = dstBuffer		
	<pre>destOffset = 00h length = 7Eh</pre>		
14	Call the getValueLength() method	Result is 7Fh	
15	Build and send a GET INPUT command		GET INPUT Proactive
	Terminal Response with text length = 7Fh		command
	Text String TLV = 0D 81 80 04 01 027F Initialise dstBuffer		
	dstBuffer = {00h, 01h FFh}		
	Call the copyTextString() method	Result of copyTextString() is 8Fh	
	dstBuffer.length = FFh		
	dstOffset = 10h		
16	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {00h, 01h, 0Fh,		
	01h,7Fh, 8Fh, FFh} srcOffset = 00h		
	dest = dstBuffer		
	destOffset = 00h		
17	length = FFh Build and send a GET INPUT command	1	GET INPUT Proactive
			command
	Terminal Response with text length = EFh		
	Text String TLV = 0D 81 F0 04 01 02 EF		
	Initialise dstBuffer		

	dstBuffer = {00h, 00h 00h}	1	
	,	D 1: (T :0:) : FFI	
	Call the copyTextString() method	Result of copyTextString() is EFh	
	1 - 2 - 6 - 1 1		
	<pre>dstBuffer.length = FFh dstOffset = 00h</pre>		
4.0		D 11 (0 (): 001	
18	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
	src = {01h,EFh, 00h 00h }		
	src = {UIN,EFN, UUN UUN } srcOffset = 00h		
	dest = dstBuffer		
	destOffset = 00h		
	length = FFh		
19	Build and send a GET INPUT command		GET INPUT Proactive
13	Bana ana sena a GET na GT commana		command
	Townsing Doomongo with two Toys Chrise TV		Command
	Terminal Response with two Text String TLV		
	1-t mout Charles mil 7 - 0D 03 04 40 43		
	1st Text String TLV = 0D 03 04 42 43 2nd Text String TLV = 0D 02 04 44		
	Initialise dstBuffer		
	dstBuffer = {00h, 01h, 02h, 03h}		
		Description of the Total Observation of the Oak	
	Call the copyTextString() method	Result of copyTextString() is 04h	
	dstBuffer.length = 04h		
	dstOffset = 02h		
20	Compare dstBuffer using arrayCompare()	Result of arrayCompare() is 00h	
20	Compare distribution using arrayCompare()	Result of arrayCompare() is our	
	src = {00h, 01h, 42h, 43h}		
	srcOffset = 00h		
	dest = dstBuffer		
	destOffset = 00h		
	length = 04h		
21	Call the getValueLength() method	Result is 03h	
-			

6.2.8.2.4 Test Coverage

CRR number	Test case number
N1	6, 8, 10, 13, 16, 18, 20
N2	11, 14, 21
N3	5, 7, 9, 12, 15, 17, 19
P1	1
P2	2, 3
C1	4

6.2.8.3 Method getAdditionalInformationLength

Test Area Reference: API_2_PRH_GTIL

6.2.8.3.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: This method returns the length of the additional information field from the first Result TLV in the ProactiveResponseHandler.

CRRN2: After a successful execution of the method, the Result TLV becomes the selected TLV of the ProactiveResponseHandler.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.

6.2.8.3.2 Test Suite files

Test Script: API_2_PRH_GTIL_1.scr

Test Applet: API_2_PRH_GTIL_1.java

Load Script: API_2_PRH_GTIL_1.ldr

Cleanup Script: API_2_PRH_GTIL_1.clr

6.2.8.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
	qualifier = 00h		command
	dcs = 04h		
	buffer = 'Text' Terminal Response without additional		
	information		
	ProactiveResponseHandler.getTheHandler();	Result is 00h	
	call the getAdditionalInformationLength() method		
2	Call the getValueLength() method	Result is 01h	
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT ProactiveProactive command
	Terminal Response with 1 additional byte		
	Result TLV = 03 02 02 55	Decole in Oak	
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 01h	
4	Call the getValueLength() method	Result is 02h	
5	Build and send a DISPLAY TEXT command		DISPLAY TEXT ProactiveProactive command
	Terminal Response with 7Eh additional bytes		
	Result TLV = 03 7F 02 55 55 55		
	ProactiveResponseHandler.getTheHandler();	Result is 7Eh	
	call the getAdditionalInformationLength()		
6	method Call the getValueLength() method	Result is 7Fh	
О	Can the getvalueLength() method	Result is 7Fff	
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 7Fh additional bytes		
	Result TLV = 03 81 80 02 55 55 55 ProactiveResponseHandler.getTheHandler();	Result is 7Fh	
	call the getAdditionalInformationLength() method	Result is 7F11	
8	Call the getValueLength() method	Result is 80h	
9	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 80h additional bytes		
	Result TLV = 03 81 81 02 55 55 55		
	ProactiveResponseHandler.getTheHandler();	Result is 80h	

	call the getAdditionalInformationLength() method		
10	Call the getValueLength() method	Result is 81h	
11	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with F2h additional bytes Result TLV = 03 81 F3 02 55 55 55		
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is F2h	
12	Call the getValueLength() method	Result is F3h	
13	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 2 Result TLV 1st Result TLV = 03 03 02 01 23 2nd Result TLV = 03 01 00		
	ProactiveResponseHandler.getTheHandler(); call the getAdditionalInformationLength() method	Result is 02h	
14	Call the getValueLength() method	Result is 03h	
15	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE_E LEMENT is thrown by send()	
	Get ProactiveResponseHandler		
	Call the getAdditionalInformationLength() method	ToolkitException.UNAVAILABLE_E LEMENT is thrown by getAdditionalInformationLength ()	

6.2.8.3.4 Test Coverage

CRR number Test case number	
N1	1, 3, 5, 7, 9, 11, 13
N2	2, 4, 6, 8, 10, 12, 14
C1	15

6.2.8.4 Method getGeneralResult

Test Area Reference: API_2_PRH_GTGR

6.2.8.4.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: This method returns the general result of a proactive command.

CRRN2: After a successful execution of the method, the Result TLV becomes the selected TLV of the ProactiveResponseHandler.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Result TLV element.

CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the general result byte is missing in the Result Simple TLV.

6.2.8.4.2 Test Suite files

Test Script: API_2_PRH_GTGR_1.scr

Test Applet: API_2_PRH_GTGR_1.java

Load Script: API_2_PRH_GTGR_1.ldr

Cleanup Script: API_2_PRH_GTGR_1.clr

6.2.8.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command qualifier = 00h dcs = 04h		DISPLAY TEXT Proactive command
	buffer = 'Text' Terminal Response with General Result = 00		
	(command performed successfully)		
	ProactiveResponseHandler.getTheHandler()	Result of getGeneralResult() is 00h	
	Call the getGeneralResult() method	recount of gotoonorum toount() to con-	
2	Call the getValueLength() method	Result is 01h	
3	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 01, without Additional information on result (command performed with partial comprehension)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 01h	
4	Call the getValueLength() method	Result is 01h	
5	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 01, with Additional information on result Result TLV = 03 02 01 55 (command performed with partial comprehension)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 01h	
6	Call the getValueLength() method	Result is 02h	
7	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with General Result = 02		
	Result TLV = 03 04 02 65 43 21 (Missing information)		
	ProactiveResponseHandler.getTheHandler() Call the getGeneralResult() method	Result of getGeneralResult() is 02h	
8	Call the getValueLength() method	Result is 04h	
9	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive

		T	
			command
	Terminal Response with 7Fh additional bytes		
	Result TLV = 03 81 80 02 55 55 55		
	ProactiveResponseHandler.getTheHandler();	Result is 02h	
	call the getGeneralResult() method	Troodit to 0211	
10	Call the getValueLength() method	Result is 80h	
11	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response with 2 Result TLV		
	1st Result TLV = 03 02 02 12 2nd Result TLV = 03 03 03 34 56		
	<pre>ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method</pre>	Result is 02h	
12	Call the getValueLength() method	Result is 02h	
13	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without Result Simple TLV	ToolkitException.UNAVAILABLE_E LEMENT is thrown by send()	
	ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method	UNAVAILABLE_ELEMENT ToolkitException is thrown	
14	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without General Result Byte in Result Simple TLV	ToolkitException.UNAVAILABLE_E LEMENT is thrown by send()	
	<pre>ProactiveResponseHandler.getTheHandler(); call the getGeneralResult() method</pre>	OUT_OF_TLV_BOUNDARIES ToolkitException is thrown	
	Result TLV = 03 00		

6.2.8.4.4 Test Coverage

CRR number Test case number	
N1	1, 3, 5, 7, 9, 11
N2	2, 4, 6, 8, 10, 12
C1	13
C2	14

6.2.8.5 Method getItemIdentifier

Test Area Reference: API_2_PRH_GTII

6.2.8.5.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The method returns the item identifier byte value from the first Item Identifier TLV element.

CRRN2: If an Item Identifier TLV element is available, it becomes the TLV selected.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Item Identifier TLV element.

CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the item identifier byte is missing in the Item Identifier Simple TLV.

6.2.8.5.2 Test Suite files

Test Script: API_2_PRH_GTII_1.scr

Test Applet: API_2_PRH_GTII_1.java

Load Script: API_2_PRH_GTII_1.ldr

Cleanup Script: API_2_PRH_GTII_1.clr

6.2.8.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive
			command
	Terminal Response (no Item Identifier TLV		
	available)		
	Call to getItemIdentifier() with unavailable	UNAVAILABLE_ELEMENT	
	Item Identifier TLV	ToolkitException is thrown	
2	Build and send a SELECT ITEM command with 2 items (ID=01, 02)		SELECT ITEM Proactive command
	Terminal Response with Item 1 selected		
	Item Identifier TLV = 10 01 01		
	Call the getItemIdentifier() method	Result is 01h	
3	Call the getValueByte() method valueOffset = 00h	Result is 01h	
4	Build and send a SELECT ITEM command with 3 items (ID=03, 05, 07)		SELECT ITEM Proactive command
	Terminal Response with Item 5 selected		
	Item Identifier TLV = 10 01 05		
	Call the getItemIdentifier() method	Result is 05h	
5	Call the getValueByte() method valueOffset = 00h	Result is 05h	
6	Build and send a SELECT ITEM command with 3 items (ID=FDh, FEh, FFh)		SELECT ITEM Proactive command
	Terminal Response with Item FFh selected		
	Item Identifier TLV = 10 01 FF		
	Call the getItemIdentifier() method	Result is FFh	
7	Call the getValueByte() method valueOffset = 00h	Result is FFh	
8	Build and send a SELECT ITEM command with 3 items (ID=FDh, FEh, FFh)		SELECT ITEM Proactive command

	Terminal Response with 2 Item Identifier TLV		
	1st Item Identifier TLV = 10 01 FFh		
	2nd Item Identifier TLV = 10 01 FEh		
	Call the getItemIdentifier() method	Result is FFh	
9	Call the getValueByte() method	Result is FFh	
	valueOffset = 00h		
10	Build and send a DISPLAY TEXT command		DISPLAY TEXT Proactive command
	Terminal Response without item identifier in the Item Identifier Simple TLV		
	Item Identifier TLV = 10 00		
	Call to getItemIdentifier()	OUT_OF_TLV_BOUNDARIES ToolkitException is thrown	

6.2.8.5.4 Test Coverage

CRR number Test case number	
N1	2, 4, 6, 8
N2	3, 5, 7, 9
C1	1
C2	10

6.2.8.6 Method getTextStringCodingScheme

Test Area Reference: API_2_PRH_GTCS

6.2.8.6.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: This method returns the data coding scheme from the first Text String TLV element.

CRRN2: If a Text String TLV element is available, it becomes the TLV selected.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.

CRRC2: A ToolkitException.OUT_OF_TLV_BOUNDARIES shall be thrown if the Text String TLV is present with a length of 0.

6.2.8.6.2 Test Suite files

Test Script: API_2_PRH_GTCS_1.scr

Test Applet: API_2_PRH_GTCS_1.java

Load Script: API_2_PRH_GTCS_1.ldr

Cleanup Script: API_2_PRH_GTCS_1.clr

6.2.8.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	7.1. Exposition	DISPLAY TEXT Proactive
			command
	Terminal Response (no Text String TLV		
	element available)		
	Call to getTextStringCodingScheme() with	UNAVAILABLE_ELEMENT	
	unavailable Text String TLV	ToolkitException is thrown	
2	Build and send a GET INPUT command	Tochkie Acoption is thown	GET INPUT Proactive
			command
	Terminal Response with a null Text String TLV		
	Theret Obsides TIM OF OR		
	Text String TLV = 0D 00 Call the getTextStringCodingScheme()	OUT_OF_TLV_BOUNDARIES	
	method	ToolkitException is thrown	
	momou	Toolkite Xooption to tillown	
3	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response with text length = 01h, DCS		
	= 04h		
	Text String TLV = 0D 02 04 "A"		
	Call the getTextStringCodingScheme() method	Result is 04h	
4	Call the getValueLength() method	Result is 02h	
5	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response with text length = 02h,		
	DCS = 00h		
	Text String TLV = 0D 03 00 "BB"		
	Call the getTextStringCodingScheme() method	Result is 00h	
	method		
6	Call the getValueLength() method	Result is 03h	
	oun me gerrane-engin() memou	1.000.11.000.11	
7	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response with text length = 7Eh, DCS = 08h		
	DC2 = 0011		
	Text String TLV = 0D 7F 08 01 02 7E		
	Call the getTextStringCodingScheme()	Result is 08h	
	method		
\vdash	Call the weth about a mostly A th	Deput in 75k	
8	Call the getValueLength() method	Result is 7Fh	
9	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response with text length = 7Fh,		
	DCS = 04h		
\vdash	Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringCodingScheme()	Result is 04h	
	method	TROGUIL IO OTIT	
10	Call the getValueLength() method	Result is 80h	

11	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = EFh, DCS = 08h		
	Text String TLV = 0D 81 F0 08 01 02 EE EF		
	Call the getTextStringCodingScheme() method	Result is 08h	
12	Call the getValueLength() method	Result is F0h	
13	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with 2 Text String TLV		
	1st Text String TLV = 0D 02 04 41 2nd Text String TLV = 0D 03 08 42 43		
	Call the getTextStringCodingScheme() method	Result is 04h	
14	Call the getValueLength() method	Result is 02h	

6.2.8.6.4 Test Coverage

CRR number	Test case number
N1	3, 5, 7, 9, 11, 13
N2	4, 6, 8, 10, 12, 14
C1	1
C2	2

6.2.8.7 Method GetTextStringLength

Test Area Reference: API_2_PRH_GTTL

6.2.8.7.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The getTextStringLength() method returns the text string length value from the first Text String TLV element.

CRRN2: If a Text String TLV element is available, it becomes the TLV selected.

Parameter errors

No requirements

Context errors

CRRC1: A ToolkitException.UNAVAILABLE_ELEMENT shall be thrown in case of unavailable Text String TLV element.

6.2.8.7.2 Test Suite files

Test Script: API_2_PRH_GTTL_1.scr

Test Applet: API_2_PRH_GTTL_1.java

Load Script: API_2_PRH_GTTL_1.ldr

Cleanup Script: API_2_PRH_GTTL_1.clr

6.2.8.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a DISPLAY TEXT command	AFI Expectation	DISPLAY TEXT Proactive
'	Build and Send a DISPEAT TEXT Command		command
	Terminal Response (no Text String TLV		
	element available)		
	Call to getTextStringLength() with unavailable	UNAVAILABLE_ELEMENT	
	Text String TLV	ToolkitException is thrown	
2	Build and send a GET INPUT command	Toolkit Zeedptier to thrown	GET INPUT Proactive
			command
	Terminal Response with a null Text String TLV Text String TLV = 0D 00		
	Call the getTextStringLength() method	Result is 00h	
	gan and got oxioning conguity mountain	Troodit is con	
3	Call the getValueLength() method	Result is 00h	
4	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response with text length = 01h,		
	DCS = 04h		
	Text String TLV = 0D 02 04 "A"		
	Call the getTextStringLength() method	Result is 01h	
5	Call the getValueLength() method	Result is 02h	
	oun the getvaluezength() method	result is ozii	
6	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Township Doomongo with toy t longth 02h		
	Terminal Response with text length = 02h, DCS = 00h		
	200 = 0011		
	Text String TLV = 0D 03 00 "BB"		
	Call the getTextStringLength() method	Result is 02h	
7	Call the getValueLength() method	Result is 03h	
′	Jan the getvaluetenguith method	Tresult is USIT	
8	Build and send a GET INPUT command		GET INPUT Proactive
			command
	Torminal Dognanas with toyt langth - 75h		
	Terminal Response with text length = 7Eh, DCS = 08h		
	Text String TLV = 0D 7F 08 01 02 7E		
	Call the getTextStringLength() method	Result is 7Eh	
	0.11.41	D 11: 751	
9	Call the getValueLength() method	Result is 7Fh	
10	Build and send a GET INPUT command		GET INPUT Proactive
.0			command
	Terminal Response with text length = 7Fh,		
	DCS = 04h		
	Text String TLV = 0D 81 80 04 01 02 7F Call the getTextStringLength() method	Result is 7Fh	
ı	Jan and got roxtoaning conguity mounds	1 NOSUICIO 71 II	I

11	Call the getValueLength() method	Result is 80h	
12	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with text length = EFh, DCS = 04h		
	Text String TLV = 0D 81 F0 04 01 02 EE EF		
	Call the getTextStringLength() method	Result is EFh	
13	Call the getValueLength() method	Result is F0h	
14	Build and send a GET INPUT command		GET INPUT Proactive command
	Terminal Response with 2 Text String TLV 1st Text String TLV = 0D 02 04 41 2nd Text String TLV = 0D 03 08 42 43		
	Call the getTextStringLength() method	Result is 01h	
15	Call the getValueLength() method	Result is 02h	

6.2.8.7.4 Test Coverage

CRR number	Test case number	
1	2, 4, 6, 8, 10, 12, 14	
2	3, 5, 7, 9, 11, 13, 15	
3	1	

6.2.8.8 Method getTheHandler

Test Area Reference: API_2_PRH_GTHD

6.2.8.8.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

 $\label{eq:public_static} \mbox{ ProactiveResponseHandler getTheHandler()} \\ \mbox{ throws ToolkitException}$

Normal execution

CRRN1: The method shall return the single system instance of the ProactiveHandler class.

CRRN2: The EnvelopeHandler is a Temporary JCRE Entry Point Object

Parameter errors

No requirements

Context errors

CRRC1: The method shall throw ToolkitException.HANDLER_NOT_AVAILABLE if the handler is busy.

6.2.8.8.2 Test Suite files

Test Script: API_2_PRH_GTHD_1.scr

Test Applet: API_2_PRH_GTHD_1.java

Load Script: API_2_PRH_GTHD_1.ldr

Cleanup Script: API_2_PRH_GTHD_1.clr

6.2.8.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a Proactive Command		Proactive Command
	Terminal Response		
	getTheHandler() twice	The returned objects shall be the	
		same	
2	getTheHandler()	The reference shall be a	
		ProactiveResponseHandler	
3	getTheHandler()	The reference shall not be null	

6.2.8.8.4 Test Coverage

CRR number	Test case number	
N1	1, 2, 3	
N2	To be checked in	
	Framework tests and	
	insert here cross	
	reference	
C1	To be checked in	
	Framework tests and	
	insert here cross	
	reference	

6.2.8.9 Method getLength

Test Area Reference API_2_PRH_GLEN

6.2.8.9.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: returns the length in bytes of the TLV list.

Parameter errors

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException.HANDLER_NOT_AVAILABLE.

6.2.8.9.2 Test Suite files

Test Script: API_2_PRH_GLEN_1.scr

Test Applet: API_2_PRH_GLEN_1.java

Load Script: API_2_PRH_GLEN_1.ldr

Cleanup Script: API_2_PRH_GLEN_1.clr

6.2.8.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Build and send a Display Text command		DISPLAY TEXT Proactive
			command
	Terminal Response without additional		
	information in General Result TLV		
	ProactiveResponseHandler.getTheHandler()	Result of getLength() is 12	
	getLength()		
2	Build and send a Display Text command		DISPLAY TEXT Proactive
			command
	Terminal Response with F2h additional		
	information in General Result TLV		
	ProactiveResponseHandler.getTheHandler()	Result of getLength() is FFh	
	getLength()		

6.2.8.9.4 Test Coverage

CRR number	Test case number
N1	1, 2
C1	Does not apply for
	Proactive Response
	Handler

6.2.8.10 Method copy

Test Area Reference API_2_PRH_COPY_BSS

6.2.8.10.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: copies the simple TLV list contained in the handler to the destination byte array.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null a NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative, an ArrayIndexOutOfBoundsException is thrown.

CRRP3: if dstLength is grater than the length of the simple TLV List, an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException. OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.10.2 Test Suite files

Test Script: API_2_PRH_COPY_BSS_1.scr

Test Applet: API_2_PRH_COPY_BSS_1.java

Load Script: API_2_PRH_COPY_BSS_1.ldr

Cleanup Script: API_2_PRH_COPY_BSS_1.clr

6.2.8.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a DISPLAY TEXT command	•	DISPLAY TEXT Proactive
			command
	Terminal Response without Additional		
	Information in General Result TLV:		
	81 03 01 21 00 02 02 82 81 03 01 00		
	01 03 01 21 00 02 02 02 03 03 04 00		
	ProactiveResponseHandler.getTheHandler()	NullPointerException is thrown	
	copy() with NULL as parameter to dstBuffer	I valii olitter Exception is tillown	
	copy() with NOLE as parameter to dstburier		
2	dotOffoot > dotDuffor longth	A recyled a vOut Of Bounda Evacetic	
-	<pre>dstOffset > dstBuffer.length dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsExceptio	
	dstOffset = 6	n is thrown	
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
"	dstBuffer.length = 5	n is thrown	
	dstOffset = -1	II lo till owi!	
	dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
	dstLength = 6		
5	dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 3		
L_	dstLength = 3	1 1 1 0 10/0	
6	dstLength < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	<pre>dstOffset = 0 dstLength = -1</pre>		
7	dstLength > length of the simple TLV list	ToolkitEveention OUT OF TIV	
'	dstBuffer.length = 13	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
	dstOffset = 0	BOUNDARIES IS UIIOWII	
	dstLength = 13		
8	Successful call, dstBuffer is the whole buffer	Result of copy() is 12	
	dstBuffer.length = 12		
	dstOffset = 0		
	dstLength = 12		
9	Compare the buffer with buffer:	Result of arrayCompare() is 0	
	81 03 01 21 00 02 02 82 81 03 01 00	·	
10	Successful call, dstBuffer is part of a buffer	Result of copy() is 15	
	dstBuffer.length = 20		
	dstOffset = 3		
44	dstLength = 12	D	
11	Compare the whole buffer	Result of arrayCompare() is 0	
	Reference = 00 01 02		
	81 03 01 21 00		
	02 02 82 81		
	03 01 00		
	OF 10 11 12 13		
12	Initialize dstBuffer		
	dstBuffer = 00h 01h 02h 13h		
	Successful call, dstBuffer is part of a buffer	Result of copy() is 12	
	dstBuffer.length = 20		
	dstOffset = 3		
40	dstLength = 9	Decult of owner Occurrence () in C	
13	Compare the whole buffer	Result of arrayCompare() is 0	
	Reference = 00 01 02		
	81 03 01 21 00		
	02 02 82 81		
	0C 0D 0E		
	OF 10 11 12 13		

6.2.8.10.4 Test Coverage

CRR number	Test case number
N1	9, 11, 13
N2	8, 10, 12
P1	1
P2	2, 3, 4, 5, 6
P3	7
C1	Does not apply for
	Proactive Response
	Handler

6.2.8.11 Method findTLV

Test Area Reference API_2_PRH_FINDBB

6.2.8.11.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of the TLV list (handler buffer):

CRRN1: the method is successful if the required occurrence exists then the corresponding TLV becomes current.

CRRN2: if the method is successful then it returns TLV_FOUND_CR_SET when Comprehension Required flag is set.

CRRN3: if the method is successful then it returns TLV_FOUND_CR_NOT_SET when Comprehension Required flag is not set.

CRRN4: if the required occurrence of the TLV element does not exist, the current TLV is no longer defined and TLV_NOT_FOUND is returned.

CRRN5: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if an input parameter is not valid (e.g. occurrence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.11.2 Test Suite files

Test Script: API_2_PRH_FINDBB_1.scr

Test Applet: API_2_PRH_FINDBB_1.java

Load Script: API_2_PRH_FINDBB_1.ldr

Cleanup Script: API_2_PRH_FINDBB_1.clr

6.2.8.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a DISPLAY TEXT command	7ti i Expositation	DISPLAY TEXT Proactive
			command
	Terminal Response with 2 General Result TLV		
	81 03 01 21 00		
	82 02 82 81		
	03 01 00 03 02 01 12		
	findTLV() with Invalid input parameter	ToolkitException.BAD_INPUT_P	
	occurrence = 0	ARAMETER is thrown	
2	Search 1st TLV	Result is TLV_FOUND_CR_SET	
	tag = 01h		
3	call the getValueLength() method	Result is 03h	
4	Search 2nd TLV	Result is USII	
4	tag = 02h	Result is TEV_FOOND_CR_SET	
	occurrence = 1		
5	Call the getValueLength() method	Result is 02h	
6	Select a TLV (tag 02h)		
	Search a wrong tag	Result is TLV_NOT_FOUND	
	tag = 04h occurrence = 1		
7	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
'	can the gerralaciongin() method	_ELEMENT shall be thrown	
8	Search a tag with wrong occurrence	Result is TLV_NOT_FOUND	
	tag = 01h		
	occurrence = 2		
9	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
10	Search 3rd TLV	_ELEMENT shall be thrown. Result is	
10	tag = 03h	TLV_FOUND_CR_NOT_SET	
	occurrence = 1	TEV_FOUND_CK_NOT_SET	
11	Call the getValueLength() method	Result is 01h	
12	Search 3rd TLV	Result is	
	tag = 03h	TLV_FOUND_CR_NOT_SET	
13	occurrence = 2 Call the getValueLength() method	Result is 02h	
13	Jan the getvalueLength() method	INGGUILIO UZII	
14	Search tag 83h	Result is	
	Tag = 83h	TLV_FOUND_CR_NOT_SET	
L	Occurrence = 1		
15	Search tag 82h	Result is TLV_FOUND_CR_SET	
	Tag = 82h Occurrence = 1		
	00001101100 - 1	1	J

6.2.8.11.4 Test Coverage

CRR number	Test case number
N1	3, 5, 11, 13
N2	2, 4
N3	10, 12
N4	6, 7,8, 9
N5	14,15
P1	1
C1	Does not apply for
	Proactive Response
	Handler

6.2.8.12 Method getValueLength

Test Area Reference API_2_PRH_GVLE

6.2.8.12.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: gets and returns the binary length of the value field for the last TLV element which has been found in the handler.

Parameter errors

No requirements

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.12.2 Test Suite files

Test Script: API_2_PRH_GVLE_1.scr
Test Applet: API_2_PRH_GVLE_1.java
Load Script: API_2_PRH_GVLE_1.ldr
Cleanup Script: API_2_PRH_GVLE_1.clr

6.2.8.12.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	•	GET INPUT Proactive
			command
	Terminal Response		
	Text String TLV = 0D 00		
	ProactiveResponseHandler.getTheHandler()	ToolkitException.UNAVAILABLE	
	GetValueLength()	_ELEMENT is thrown	
2	Search TLV 0Dh		
	getValueLength()	Result is 00h	
3	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response		
	Text String TLV = 0D 02 04 41		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 02h	
4	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 7Eh		
	Text String TLV = 0D 7F 04 01 02 7E		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is 7Fh	
5	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 7Fh		
	Text String TLV = 0D 81 80 04 01 02 7E		
	7F		
	Search TLV 0Dh (Text String TLV)	Description 00h	
	getValueLength()	Result is 80h	
6	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = EFh		
	Text String TLV = 0D 81 F0 04 01 02 EF		
	Search TLV 0Dh (Text String TLV)		
	getValueLength()	Result is F0h	

6.2.8.12.4 Test Coverage

CRR number	Test case number
N1	2, 3, 4, 5, 6
C1	Does not apply for
	Proactive Response
	Handler
C2	1

6.2.8.13 Method getValueByte

Test Area Reference API_2_PRH_GVBYS

6.2.8.13.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: Gets a byte from the last TLV element which has been found in the handler and returns its value (1 byte).

Parameter errors

CRRP1: if valueOffset is out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.13.2 Test Suite files

Test Script: API_2_PRH_GVBYS_1.scr

Test Applet: API_2_PRH_GVBYS_1.java

Load Script: API_2_PRH_GVBYS_1.ldr

Cleanup Script: API_2_PRH_GVBYS_1.clr

6.2.8.13.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 7Eh Text String TLV = 0D 7F 04 01 02 7E		
	ProactiveResponseHandler.getTheHandler()		
	getValueByte(0)	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
2	Search TLV 01h (Command Details TLV)		
	getValueByte(3)	ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown	
3	Search TLV 01h (Command Details TLV)		
	getValueByte(2)	Result is 00h (qualifier)	
4	Search TLV 02h (Device Identities TLV)		
	getValueByte(0)	Result is 82h (Source)	
5	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Eh	
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = EFh Text String TLV = 0D 81 F0 04 01 02 7E 7F EF		
	Search TLV 0Dh (Text String TLV)		
	getValueByte(7E)	Result is 7Eh	
7	GetValueByte(7F)	Result is 7Fh	
8	GetValueByte(EF)	Result is EFh	
	1		

6.2.8.13.4 Test Coverage

CRR number	Test case number	
N1	3, 4, 5, 6, 7, 8	
P1	2	
C1	Does not apply for	
	Proactive Response	
	Handler	
C2	1	

6.2.8.14 Method copyValue

Test Area Reference API_2_PRH_CPYVS_BSS

6.2.8.14.1 Conformance requirement

The method with following header shall be compliant with its definition in the API.

Normal execution

CRRN1: copies a part of the last TLV element which has been found, into a destination. buffer.

CRRN2: returns dstOffset + dstLength.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException is thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException is thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.14.2 Test Suite files

Test Script: API_2_PRH_CPYVS_BSS_1.scr
Test Applet: API_2_PRH_CPYVS_BSS_1.java
Load Script: API_2_PRH_CPYVS_BSS_1.ldr
Cleanup Script: API_2_PRH_CPYVS_BSS_1.clr

6.2.8.14.3 Test procedure

Terminal Response, Text String length = 5 Proxt Striag TUV = 00 to 04 to 02 = 05 ProactiveResponseHandler, getTheHandler() Select Text String TUV Copyvalue() with a null datauffer MullPointerException is thrown ArrayIndexOutOfBoundsExceptio is thrown distingties length = 5 datoOffset > datSUffset 0 datLength = 0 detLength = 1 detLength = 0 detLength = 0 detLength = 1 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 1 detLength = 0 detLe	ld	Description	API Expectation	APDU Expectation
Terminal Response, Text String length = 5 Text String TW = 00 8 6 04 01 02 = 05 ProactiveResponseAndler, getTheHandler() Select Text, String TW Copyvalue() with a mult detentfer detable for length = 5 detable for length = 5 detable for length = 5 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 5 detable for length = 6 detable for length = 5 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 7 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 6 detable for length = 7 detable for length = 8 8 8 detable for length = 8 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 detable for length = 16 Text String tw = 00 to 1 04 00 01 = 00 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17 Detable for length = 17	_		All Expodution	
ProactiveResponseHandler_getTheHandler() Select Text String TLV Select Text String Into				
ProactiveResponseHandler,getTheHandler() Select Text String TV CopyValue() with a mult detEntifer NullPointerException is thrown				
Select Text String TTV CopyValue() with a mult datEquifer NullPointerException is thrown				
CopyValue() with a mult datBuffer NullPointerException is thrown datBuffer Length = 6 datLength = 0 datCoffset 6 datLength = 0 datBuffer Length = 5 datCoffset - 1 datLength = 1 datLength datBuffer Length = 5 datCoffset - 2 datLength datBuffer Length = 5 datCoffset - 3 datLength datBuffer Length - 5 datCoffset - 3 datLength datBuffer Length - 5 datCoffset - 3 datLength				
detBaffer.length = 5 detStoffset = 6 detLength = 0 detLength = 0 detStuffget.length = 5 detCoffset = 1 detStuffget.length = 5 detCoffset = 0 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 0 d			NullPointerException is thrown	
detBaffer.length = 5 detStoffset = 6 detLength = 0 detLength = 0 detStuffget.length = 5 detCoffset = 1 detStuffget.length = 5 detCoffset = 0 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 1 detStuffget = 0 d				
datoffset = 6 datsutfer.length = 5 datoffset = -1 datsutfer.length = 5 datoffset = -1 datsutfer.length = 5 datoffset = -1 datsutfer.length = 5 datoffset = -1 datsutfer.length = 5 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 6 datoffset = 0 dattergib = 0 dattergib = 1 7 valueOffset > Text String Length valueOffset = 0 dattergib	2		I	
distDiffeet < 0 distBuffeer length = 1 distLength > distBuffeer length = 1 distDiffeet length = 1 distDiffeet length = 1 distDiffeet length l			n is thrown	
datBatfer .length = 5 datLength > 1 datLength = 1 datLength > 3 datLength > 5 datDiffset + datLength > 5 datDiffset = 0 datLength = 5 datLength > 6 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 5 datLength = 1 7 valueOffset = 0 datLength = 15 datCiffset = 0 datLength = 7 ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown ToolkitException.OUT_OF_				
dstOffset = -1 dstLength >dstBuffer.length dstBuffer.length = 5 dstUngth = 5 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 6 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 7 dstUngth = 10 dstUngth = 1	3			
d stLength = 1 4			n is thrown	
dstBuffer.length = 5 dstCffset + of dstLength > dstBuffer.length dstCffset + of dstLength > dstBuffer.length dstOffset + of dstLength > dstBuffer.length dstOffset = 3 dstLength = 5 dstCffset = 3 dstLength = 5 dstCffset = 0 dstLength = 1 7				
dstOffset = 0 dstLength = 6 5	4		ArrayIndexOutOfBoundsExceptio	
distLength = 6 StatLength > distBuffer.length distOffset = distLength = 5 distOffset = 1 distLength = 3 distLength = 3 distLength = 3 distLength = 5 distLength = 5 distLength = 5 distLength = 5 distLength = 1 distLengt			n is thrown	
Solution Solution				
dstLength = 3 dstLength = 3 dstLength = 3 dstLength = 5 dstDeffer length = 5 dstDeffer length = 5 dstDeffer length = 5 dstDeffer length = 15 Text String TLV = 00 11 04 00 01 0F ProactiveResponseHandler .getTheHandler Copyvalue() ToolkitException.OUT_OF_TLV_ BOUNDARIES is thrown GET INPUT Proactive command GET INPUT Proactive command Terminal Response, Text String length = 16 Text String TLV = 00 11 04 00 01 0F ProactiveResponseHandler .getTheHandler Copyvalue() Successful call ValueOffset = 0 DstBuffer .length = 17 DstDeffer length = 18 DstDeffer length = 18	5		ArrayIndexOutOfBoundsExceptio	
dstLength = 3 6 dstLength = 0 dstLength = 5 dstLength = 0 dstLength = 1 dstLength = 1 7 valueOffset > Text String Length valueOffset = 7 dstBuffer.length = 15 dstOffset = 0 dstLength = 0 dstLength = 0 dstLength = 1 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 dstDuffer.length = 15 dstOffset = 0 dstLength = 1 dstDuffer.length = 15 dstOffset = 0 dstLength = 1 7 ValueOffset = 0 dstLength = 1 Doubtlength = 1 10 ValueOffset + ostlength = 15 DstOffset = 0 DstLength = 5 Successful call ValueOffset = 0 DstLength = 5 Successful call Compare buffer Buffer = 04 00 01 0F Buffer = 04 00 01 0F Successful call Result of copyValue() is 15 Successful call Result of copyValue() is 15 Successful call Result of copyValue() is 15				
distBuffer length = 5 distLength < 0 distLength = 5 distOffset = 0 distLength = -1 distBuffer length = 1 distBuffer length = 15 distBuffer length = 10 distBuffer length = 15 distBuffer length = 16 distBuffer length = 17 distBuffer length = 18 di				
dstEuffer.length = 5 dstOffset = 0 dstLength = -1 7	6	9	ArrayIndexOutOfBoundsExceptio	
dstLength = -1 valueOffset > Text String Length valueOffset = 7 dstBuffer.length = 15 dstOffset = 0 dstLength = 0 dstLength = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength = 15 dstOffset = 0 dstLength > ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown 9		dstBuffer.length = 5		
7				
valueOffset = 7 dstBuffer.length = 15 dstDepth = 0 dstLength = 0 dstLength = 0 dstLength = 0 dstLength = 1 dstBuffer.length = 15 dstDeffset = 0 dstLength = 1 dstOffset = 0 dstLength = 1 dstOffset = 0 dstLength = 7 dstDeffset = 0 dstLength = 7 dstDeffset = 0 dstLength = 7 dstDeffset = 0 dstLength = 15 DstDeffset = 0 DstLength = 5 DstDeffset = 0 DstLength = 5 DstDeffset = 0 DstLength = 15 DstOffset = 0 DstLength = 15 DstOffset = 0 DstLength = 15 DstOffset = 0 DstLength = 16 Text String Ibn = 16 Text String Ibn = 16 Text String Ibn = 0 ToolkitException.UNAVAILABLE ELEMENT is thrown 11		uschengen – -1		
detBuffer_length = 15 detDffeet = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 0 detLength = 1 detBuffer_length = 15 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 1 detBuffer_length = 15 detDffeet = 0 detLength = 1 detBuffer_length = 15 detDffeet = 0 detLength = 7 detBuffer_length = 15 detDffeet = 0 detLength = 7 detBuffer_length = 15 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 5 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 16 detDffeet = 0 detLength = 17 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 15 detDffeet = 0 detLength = 17 detDffeet = 0	7	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
dstDffset = 0 dstLength = 0 dstLength = 0			BOUNDARIES is thrown	
8		-		
valueOffset = -1 dstBuffer.length = 15 dstOffset = 0 dstLength = 1 dstDuffset = 0 dstLength > Text String length valueOffset = 0 dstLength = 1 dstOffset = 0 dstLength = 1 dstOffset = 0 dstLength = 7 dstLength = 7 dstLength = 1 DstOffset = 0 DstLength = 5 DstBuffer.length = 1 DstOffset = 0 DstLength = 5 DstDuffset = 0 DstLength = 1 DstOffset = 0 DstDuffset = 0 DstDuffset = 0 DstBuffer.length = 1 DstOffset = 0 DstBuffer.length = 1 DstOffset = 0 DstDuffset = 0 DstD				
dstBuffer.length = 15 dstLength = 7 9	8			
dstOffset = 0 dstLength > Text String length valueOffset = 0 dstBuffer.length = 15 dstOffset = 0 dstBuffer.length = 15 dstOffset = 0 DstLength = 5 10 ValueOffset + dstLength > Text String length ValueOffset + dstLength > Text String length ValueOffset = 2 DstBuffer.length = 15 DstOffset = 0 DstLength = 5 11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Successful call ValueOffset = 0 DstLength = 17 DstOffset = 0 DstLength = 17 Compare buffer Buffer = 04 00 01 0F Result of copyValue() is 17 Result is 00h Result of copyValue() is 15 Successful call Result of copyValue() is 15			BOUNDARIES is thrown	
Select Text String TLV Select Text String TLV Select Text String TLV Select Text String TLV Select Text String TLV Successful call Select Text String TLV Select Text String Text		_		
valueOffset = 0 dstBuffer.length = 15 dstOffset = 0 dstLength = 7 10		-		
dstBuffer.length = 15 dstOffset = 0 dstLength = 7 10 ValueOffset + dstLength > Text String length ValueOffset = 2 DstBuffer.length = 15 DstOffset = 0 DstLength = 5 11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Result is 00h Result is 00h Result of copyValue() is 15	9			
dstLength = 7			BOUNDARIES IS INFOWN	
ValueOffset + dstLength > Text String length ValueOffset = 2 DstBuffer.length = 15 DstOffset = 0 DstLength = 5				
ValueOffset = 2 DstBuffer.length = 15 DstOffset = 0 DstLength = 5 11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Select Text String TLV ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Buffer = 04 00 01 0F Result of copyValue() is 15 Result is 00h Result of copyValue() is 15	10		ToolkitEveention OUT OF TIV	
DstBuffer.length = 15 DstOffset = 0 DstLength = 5 11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE ELEMENT is thrown 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15	10			
DstLength = 5 11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15				
11 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15				
Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result of copyValue() is 15 Result is 00h		 nacrendru = 2		
Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h Result is 00h 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15	11	Send a GET INPUT command		GET INPUT Proactive
Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler CopyValue() ToolkitException.UNAVAILABLE _ELEMENT is thrown 12 Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15				command
ProactiveResponseHandler.getTheHandler CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Result of copyValue() is 17 Compare buffer Buffer = 04 00 01 0F Result is 00h result is 00h Result of copyValue() is 15				
CopyValue() Select Text String TLV Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 Result of copyValue() is 17 Compare buffer Buffer = 04 00 01 0F Result is 00h Result is 00h Result of copyValue() is 15				
ELEMENT is thrown 12			ToolkitException.UNAVAILABLE	
Successful call ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result of copyValue() is 17 Result of copyValue() is 17 Result is 00h				
ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F Result is 00h initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15	12		Decide of an artist of the control o	
DstBuffer.length = 17 DstOffset = 0 DstLength = 17 13 Compare buffer Buffer = 04 00 01 0F 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15			Result of copyvalue() is 17	
DstLength = 17				
Compare buffer Buffer = 04 00 01 0F Result is 00h initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15				
Buffer = 04 00 01 0F 14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15	12		Result is 00h	
14 initialise dstBuffer dstBuffer = 55 55 55 Successful call Result of copyValue() is 15	13		1 Count is out	
dstBuffer = 55 55 55 Successful call Result of copyValue() is 15				
Successful call Result of copyValue() is 15	14			
	<u> </u>		Result of copy\/alue() is 15	
			Transfer of copy value() to 10	

	DstBuffer.length = 20		
	DstOffset = 3		
	DstLength = 12		
15	Compare buffer	Result is 00h	
	Buffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		

6.2.8.14.4 Test Coverage

CRR number	Test case number
N1	13, 15
N2	12, 14
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for Proactive Response
	Handler
C2	11

6.2.8.15 Method compare Value

Test Area Reference API_2_PRH_CPRVS_BSS

6.2.8.15.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Compares the last found TLV element with a buffer:

CRRN1: returns 0 if identical.

CRRN2: returns -1 if the first miscomparing byte in simple TLV List is less than that in compareBuffer.

CRRN3: returns 1 if the first miscomparing byte in simple TLV List is greater than that in compareBuffer.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

CRRC2: in case of unavailable TLV element an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException UNAVAILABLE_ELEMENT.

6.2.8.15.2 Test Suite files

Test Script: API_2_PRH_CPRVS_BSS_1.scr

Test Applet: API_2_PRH_CPRVS_BSS_1.java

Load Script: API_2_PRH_CPRVS_BSS_1.ldr

Cleanup Script: API_2_PRH_CPRVS_BSS_1.clr

6.2.8.15.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler()		
	Select Text String TLV		
	compareValue() with a null compareBuffer	NullPointerException is thrown	
2	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = 6 compareLength = 0</pre>		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
3	compareBuffer.length = 5	n is thrown	
	compareOffset = -1	II is tillowii	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
_	compareLength = 6	A 1 1 0 10'D 1 5 1:	
5	compareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
	>compareBuffer.length compareBuffer.length = 5	n is thrown	
	compareOffset = 3		
	compareLength = 3		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
	compareLength = -1		
7	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 7 compareBuffer.length = 15</pre>	BOUNDARIES is thrown	
	compareOffset = 0		
	compareLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
9	compareLength = 1	ToolkitEveention OUT OF TIV	
9	compareLength > Text String length valueOffset = 0	ToolkitException.OUT_OF_TLV_BOUNDARIES is thrown	
	compareBuffer.length = 15	BOUNDARIES IS UIIOWII	
	compareOffset = 0		
	compareLength = 7		
10	valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	
	length	BOUNDARIES is thrown	
	valueOffset = 2		
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareLength = 5		
11	Send a GET INPUT command		GET INPUT Proactive
' '	Cond a CE1 iiii O1 Communa		command
	Terminal Response, Text String length = 16		
I	1	1	I

	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	CompareValue()	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
12	Select Text String TLV		
	Initialise compareBuffer		
	CompareBuffer =		
	04 00 01 0F		
	Compare buffers	Result is 00h	
	ValueOffset = 0		
	CompareOffset = 0 CompareLength = 17		
	Comparehengen = 17		
13	Initialise compareBuffer		
.0	CompareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
14	Initialise compareBuffer		
	CompareBuffer =		
	03 00 01 0F		
	Compare buffers with same parameters	Result is +1	
15	Initialise compareBuffer		
	CompareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0B 0C 55 55 55 55 55		
	Compare buffers	Result is 00h	
	ValueOffset = 2	Result is oon	
	CompareOffset = 3		
	CompareLength = 12		
16	Initialise compareBuffer		
	CompareBuffer =		
	55 55 55 02 01 03 04 05 06 07		
	08 09 0A 0B 0C		
	55 55 55 55		
	Compare buffers with same parameters	Result is -1	
17	Initialise compareBuffer		
• •	CompareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07		
	08 09 0A 0A 0D		
	Compare buffers with same parameters	Result is +1	
	Compare bullers with same parameters	INCOURTS TT	

6.2.8.15.4 Test Coverage

CRR number	Test case number	
N1	12, 15	
N2	13, 16	
N3	14, 17	
P1	1	
P2	2, 3, 4, 5, 6	
P3	7, 8, 9, 10	
C1	Does not apply for	
Proactive Resp		
	Handler	
C2	11	

6.2.8.16 Method findAndCopyValue(byte tag, byte[] dstBuffer, short valueOffset)

Test Area Reference API_2_PRH_FACYB_BS

6.2.8.16.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the first occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + length of the copied value is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.16.2 Test Suite files

Test Script: API_2_PRH_FACYB_BS_1.scr

Test Applet: API_2_PRH_FACYB_BS_1.java

Load Script: API_2_PRH_FACYB_BS_1.ldr

Cleanup Script: API_2_PRH_FACYB_BS_1.clr

6.2.8.16.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()		
	FindAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
		-	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh	n is thrown	
	dstBuffer.length = 20		
	dstOffset = 21		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 20	n is thrown	
	dstOffset = -1		
4	length > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 15	n is thrown	
	dstOffset = 0		
5	dstOffset + length >dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 20	n is thrown	

	dstOffset = 5		
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)		
	findAndCopyValue() tag = 04h	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE _ELEMENT is thrown.	
7	Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0	Result of findAndcopyValue() is 17	
8	Compare buffer Buffer = 04 00 01 0F	Result is 00h	
9	initialise dstBuffer dstBuffer = 55 55 55		
	Successful call DstBuffer.length = 20 DstOffset = 2	Result of findAndcopyValue() is 19	
10	Compare buffer Buffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0E 0F 55	Result is 00h	
11	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, with 2 Text String TLV 0D 11 04 00 01 0F 0D 02 04 41		
	ProactiveResponseHandler.getTheHandler()		
	Successful call Tag = 0Dh DstBuffer.length = 17 DstOffset = 0	Result of findAndcopyValue() is 17	
12	Compare buffer Buffer = 04 00 01 0F	Result is 00h	
13	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F Proactive Perponse Handler (act The Handler)		
	ProactiveResponseHandler.getTheHandler() Successful call (with tag 8Dh) Tag = 8Dh DstBuffer.length = 17 DstOffset = 0	Result of findAndcopyValue() is 17	
14	Compare buffer Buffer = 04 00 01 0F	Result is 00h	
		l .	

6.2.8.16.4 Test Coverage

CRR number	Test case number	
N1	8, 10, 12	
N2	6	
N3	7, 9, 11	
N4	13, 14	
P1	1	
P2	2, 3, 4, 5	
C1	Does not apply for	
	Proactive Response	
	Handler	

6.2.8.17 Method findAndCopyValue(byte tag, byte occurence, short valueOffset, byte[] dstBuffer, short dstOffset, short dstLength)

Test Area Reference API_2_PRH_FACYBBS_BSS

6.2.8.17.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: looks for the indicated occurrence of a TLV element from the beginning of a TLV list and copy its value into a destination buffer.

CRRN2: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN3: if the method is successful then the corresponding TLV becomes current and dstOffset + dstLength is returned.

CRRN4: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if dstBuffer is null NullPointerException shall be thrown.

CRRP2: if dstOffset or dstLength or both would cause access outside array bounds, or if dstLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, dstLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.17.2 Test Suite files

Test Script: API_2_PRH_FACYBBS_BSS_1.scr

Test Applet: API_2_PRH_FACYBBS_BSS_1.java

Load Script: API_2_PRH_FACYBBS_BSS_1.ldr

Cleanup Script: API_2_PRH_FACYBBS_BSS_1.clr

6.2.8.17.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	Expodution	GET INPUT Proactive
			command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()	NiviliDelinterFree C. C. C.	
	findAndCopyValue() with a null dstBuffer	NullPointerException is thrown	
2	dstOffset > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh, occurrence = 1	n is thrown	
	<pre>valueOffset = 0 dstBuffer.length = 5</pre>		
	dstBuller.length = 5 dstOffset = 6		
	dstLength = 0		
3	dstOffset < 0	ArrayIndexOutOfBoundsExceptio	
	<pre>dstBuffer.length = 5 dstOffset = -1</pre>	n is thrown	
	dstOffset = -1 dstLength = 1		
4	dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
	dstBuffer.length = 5	n is thrown	
	dstOffset = 0		
5	dstLength = 6 dstOffset + dstLength > dstBuffer.length	ArrayIndexOutOfBoundsExceptio	
J	dstBuffer.length = 5	n is thrown	
	dstOffset = 3	2	
	dstLength = 3	Amerika day Oy 10/D	
6	<pre>dstLength < 0 dstBuffer.length = 5</pre>	ArrayIndexOutOfBoundsException is thrown	
	dstOffset = 0	n is thrown	
	dstLength = -1		
7	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05 ProactiveResponseHandler.getTheHandler()		
	valueOffset > Text String Length	ToolkitException.OUT_OF_TLV_	
	tag = 0Dh, occurrence = 1	BOUNDARIES is thrown	
	<pre>valueOffset = 7</pre>		
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstOffset = 0 dstLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstollset = 0 dstLength = 1		
9	dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 0</pre>	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstOffset = 0 dstLength = 7		
10	valueOffset + dstLength > Text String length	ToolkitException.OUT_OF_TLV_	
	<pre>valueOffset = 2</pre>	BOUNDARIES is thrown	
	<pre>dstBuffer.length = 15 dstOffset = 0</pre>		
	dstLength = 5		
11	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)		
	Sciect a ILV (tay UZII)	1	

	findAndCopyValue() tag = 0Dh	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE _ELEMENT is thrown.	
12	Successful call Tag = 0Dh, occurrence = 1 ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17	Result of findAndCopyValue() is 17	
13	Compare buffer Buffer = 04 00 01 0F	Result is 00h	
14	initialise dstBuffer dstBuffer = 55 55 55 Successful call Tag = 0Dh, occurrence = 1 ValueOffset = 2 DstBuffer.length = 20 DstOffset = 3	Result of findAndcopyValue() is 15	
15	DstLength = 12 Compare buffer Buffer = 55 55 55 01 02 03 04 05 06 07 08 09 0A 0B 0C 55 55 55 55 55	Result is 00h	
16	Send a GET INPUT command Terminal Response, with 2 Text String TLV OD 11 04 00 01 02 OF OD 06 00 11 22 33 44 55 (no specific DCS		GET INPUT Proactive command
	byte) ProactiveResponseHandler.getTheHandler()		
	Successful call Tag = 0Dh, occurrence = 1 ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17	Result of findAndCopyValue() is 17	
17	Compare buffer Buffer = 04 00 01 0F	Result is 00h	
18	Successful call Tag = 0Dh, occurrence = 2 ValueOffset = 0 DstBuffer.length = 6 DstOffset = 0 DstLength = 6	Result of findAndCopyValue() is 6	
19	Compare buffer Buffer = 00 11 22 33 44 55	Result is 00h	
20	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler()		
	Successful call (with tag 8Dh) Tag = 8Dh, occurrence = 1 ValueOffset = 0 DstBuffer.length = 17 DstOffset = 0 DstLength = 17	Result of findAndcopyValue() is 17	
21	Compare buffer Buffer = 04 00 01 0F	Result is 00h	

6.2.8.17.4 Test Coverage

CRR number	Test case number
N1	13, 15, 17, 19
N2	11
N3	12, 14, 16, 18
N4	20, 21
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
C1	Does not apply for
	Proactive Response
	Handler

6.2.8.18 Method findAndCompareValue(byte tag, byte[] compareBuffer, short compareOffset)

Test Area Reference API_2_PRH_FACRB_BS

6.2.8.18.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

Normal execution

Looks for the first occurrence of a TLV element from beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical returns 0.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer returns -1.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer returns 1.

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset would cause access outside array bounds ArrayIndexOutOfBoundsException shall be thrown.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.18.2 Test Suite files

Test Script: API_2_PRH_FACRB_BS_1.scr

Test Applet: API_2_PRH_FACRB_BS_1.java

Load Script: API_2_PRH_FACRB_BS_1.ldr

Cleanup Script: API_2_PRH_FACRB_BS_1.clr

6.2.8.18.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	711 Expodution	GET INPUT Proactive
•			command
	Terminal Response, Text String length = 15		
	Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()		
	FindAndCompareValue() with a null dstBuffer	NullPointerException is thrown	
2	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	tag = 0Dh	n is thrown	
	<pre>compareBuffer.length = 20 compareOffset = 21</pre>		
3	compareOffset < 0	ArrayIndexOutOfBoundsExceptio	
Ū	compareBuffer.length = 20	n is thrown	
	compareOffset = -1		
4	length > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>	n is thrown	
5	CompareOffset + length >	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length	n is thrown	
	<pre>CompareBuffer.length = 20 CompareOffset = 5</pre>		
	COMPATENTIBER - 3		
6	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)	ToolkitEveenties LINAVALLABLE	
	findAndCompareValue() tag = 04h	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	Call the getValueLength() method	ToolkitException.UNAVAILABLE	
	Jan the gerralaceonging memod	_ELEMENT is thrown.	
7	Initialise compareBuffer		
	CompareBuffer = 04 00 01 OF		
	Compare buffers	Result is 00h	
	Tag = 0Dh	Troodit is oon	
	CompareOffset = 0		
8	Verify current TLV	Result is 17	
9	GetValueLength() Initialise compareBuffer		
9	CompareBuffer =		
	04 00 01 10		
	Compare buffers with same parameters	Result is -1	
10	Initialise compareBuffer		
-	CompareBuffer =		
	03 00 01 0F	Describie of	
	Compare buffers with same parameters	Result is +1	
11	Initialise compareBuffer		
	CompareBuffer =		
	55 55 04 00 01 02 03 04 05 06		
	02 03 04 05 06 07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	CompareOffset = 2		
12	Send a GET INPUT command		GET INPUT Proactive
14	Sena a GET INFOT COMMINANA		command
	Terminal Response, with 2 Text String TLV		
	OD 11 04 00 01 OF		

	OD 06 00 11 22 33 44 55		
	ProactiveResponseHandler.getTheHandler()		
	Initialise compareBuffer		
	CompareBuffer =		
	55 55 04 00 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is 00h	
	CompareOffset = 2		
13	Initialise compareBuffer		
	CompareBuffer =		
	55 55 04 01 01		
	02 03 04 05 06		
	07 08 09 0A 0B		
	OC OD OE OF 55		
	Compare buffers	Result is -1	
	CompareOffset = 2		
1			
14	Initialise compareBuffer		
14	Initialise compareBuffer CompareBuffer =		
14			
14	CompareBuffer =		
14	CompareBuffer = 55 55 04 00 01		
14	CompareBuffer = 55 55 04 00 01 02 03 04 05 06		
14	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B	Result is +1	
14	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55	Result is +1	
14	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers	Result is +1	
14	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers	Result is +1	GET INPUT Proactive
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2	Result is +1	GET INPUT Proactive command
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler()	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F	Result is +1	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F Compare buffers (with tag 8Dh)	Result is +1 Result is 00h	
	CompareBuffer = 55 55 04 00 01 02 03 04 05 06 07 08 09 0A 0B 0C 0D 0D 10 55 Compare buffers CompareOffset = 2 Send a GET INPUT command Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F		

6.2.8.18.4 Test Coverage

CRR number	Test case number
N1	6
N2	8
N3	7, 11, 12
N4	9, 13
N5	10, 14
N6	15
P1	1
P2	2, 3, 4, 5
C1	Does not apply for
	Proactive Response
	Handler

6.2.8.19 Method findAndCompareValue(byte tag, byte occurence, short valueOffset, byte[] compareBuffer, short compareOffset, short compareLength)

Test Area Reference API_2_PRH_FACRBBS_BSS

6.2.8.19.1 Conformance requirement

The method with following header shall be compliant to its definition in the API.

public byte findAndCompareValue(byte tag,

byte occurence,
short valueOffset,
byte[] compareBuffer,
short compareOffset,
short compareLength)

throws java.lang.NullPointerException,

java.lang.ArrayIndexOutOfBoundsException,

ToolkitException

Normal execution

Looks for the indicated occurrence of a TLV element from the beginning of a TLV list and compare its value with a buffer:

CRRN1: if no TLV element is found, the UNAVAILABLE_ELEMENT exception is thrown and the current TLV is no longer defined.

CRRN2: if the method is successful then the corresponding TLV becomes current.

CRRN3: if identical 0 is returned.

CRRN4: if the first miscomparing byte in simple TLV is less than that in compareBuffer -1 is returned.

CRRN5: if the first miscomparing byte in simple TLV is greater than that in compareBuffer 1 is returned

CRRN6: The search method is comprehension required flag independent.

Parameter errors

CRRP1: if compareBuffer is null NullPointerException shall be thrown.

CRRP2: if compareOffset or compareLength or both would cause access outside array bounds, or if compareLength is negative ArrayIndexOutOfBoundsException shall be thrown.

CRRP3: if valueOffset, compareLength or both are out of the current TLV an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException OUT_OF_TLV_BOUNDARIES.

CRRP4: if an input parameter is not valid (e.g. occurence = 0) an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException BAD_INPUT_PARAMETER.

Context errors

CRRC1: if the handler is busy an instance of ToolkitException shall be thrown. The reason code shall be ToolkitException HANDLER_NOT_AVAILABLE.

6.2.8.19.2 Test Suite files

Test Script: API_2_PRH_FACRBBS_BSS_1.scr

Test Applet: API_2_PRH_FACRBBS_BSS_1.java

Load Script: API_2_PRH_FACRBBS_BSS_1.ldr

Cleanup Script: API_2_PRH_FACRBBS_BSS_1.clr

6.2.8.19.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Send a GET INPUT command	AiTExpectation	GET INPUT Proactive
'	Send a GET INFOT Command		command
	Terminal Response, Text String length = 15		Command
	Text String TLV = 0D 10 04 01 02 0F		
	ProactiveResponseHandler.getTheHandler()		
	findAndCompareValue() with a null	NullPointerException is thrown	
	compareBuffer	Null officerException is thown	
	Comparebatier		
2	compareOffset > compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
_	tag = 0Dh, occurrence = 1	n is thrown	
	valueOffset = 0		
	compareBuffer.length = 5		
	compareOffset = 6		
3	compareLength = 0 compareOffset < 0	ArrayIndayOutOfPaundaEyeantia	
3	compareBuffer.length = 5	ArrayIndexOutOfBoundsException is thrown	
	compareOffset = -1	III IS UIIOWII	
	compareLength = 1		
4	compareLength >compareBuffer.length	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	compareOffset = 0		
5	compareLength = 6 CompareOffset + compareLength	ArrayIndexOutOfBoundsExceptio	
э	compareOnset + compareLength compareBuffer.length		
	CompareBuffer.length = 5	n is thrown	
	CompareOffset = 3		
	CompareLength = 3		
6	compareLength < 0	ArrayIndexOutOfBoundsExceptio	
	compareBuffer.length = 5	n is thrown	
	<pre>compareOffset = 0 compareLength = -1</pre>		
7	Send a GET INPUT command		GET INPUT Proactive
			command
	Terminal Response, Text String length = 5		
	Text String TLV = 0D 06 04 01 02 05		
	ProactiveResponseHandler.getTheHandler()		
	valueOffset ≥ Text String Length	ToolkitException.OUT_OF_TLV_	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 7</pre>	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 0		
8	valueOffset < 0	ToolkitException.OUT_OF_TLV_	
	valueOffset = -1	BOUNDARIES is thrown	
	<pre>compareBuffer.length = 15 compareOffset = 0</pre>		
	compareUffset = 0 compareLength = 1		
9	compareLength > Text String length	ToolkitException.OUT_OF_TLV_	
	valueOffset = 0	BOUNDARIES is thrown	
	compareBuffer.length = 15		
	compareOffset = 0		
10	compareLength = 7 valueOffset + compareLength > Text String	ToolkitException.OUT_OF_TLV_	
10	length	BOUNDARIES is thrown	
	valueOffset = 2	BOONDAINES IS UITOWIT	
	compareBuffer.length = 15		
	compareOffset = 0		
	compareLength = 5		
1.4	Invalid parameter	ToolkitEveention DAD INDUT D	
11	Invalid parameter Occurrence = 0	ToolkitException.BAD_INPUT_P ARAMETER is thrown	
	Joseph John John John John John John John Joh	ANAIVIETEK IS (IIIOWI)	
12	Send a GET INPUT command		GET INPUT Proactive
'-	55 2 5E1 51 5511111and		command
	Terminal Response, Text String length = 16		
	Text String TLV = 0D 11 04 00 01 0F		
	ProactiveResponseHandler.getTheHandler()		
	Select a TLV (tag 02h)		
			_

	findAndCompareValue() tag = 0Dh	ToolkitException.UNAVAILABLE _ELEMENT is thrown	
	occurrence = 2 Call the getValueLength() method		
		ToolkitException.UNAVAILABLE _ELEMENT is thrown.	
13	Initialise compareBuffer CompareBuffer =		
	04 00 01 0F findAndCompareValue()	Result is 00h	
	<pre>tag = 0Dh, occurrence = 1 valueOffset = 0</pre>		
	<pre>compareOffset = 0 compareLength = 17</pre>		
14	Verify current TLV	Result is 17	
	GetValueLength()	TROOM TO TY	
15	Initialise compareBuffer compareBuffer =		
	04 00 01 10	Describie A	
	Compare buffers with same parameters	Result is -1	
16	Initialise compareBuffer compareBuffer =		
	Compare buffers with same parameters	Result is +1	
17	Initialise compareBuffer		
	compareBuffer = 55 55 55 01 02		
	03 04 05 06 07 08 09 0A 0B 0C		
	55 55 55 55 55 Compare buffers	Result is 00h	
	<pre>valueOffset = 2 compareOffset = 3</pre>		
	compareLength = 12		
18	Initialise compareBuffer compareBuffer =		
	55 55 55 02 01 03 04 05 06 07		
	08 09 0A 0B 0C 55 55 55 55		
	Compare buffers with same parameters	Result is -1	
19	Initialise compareBuffer compareBuffer =		
	55 55 55 01 02		
	03 04 05 06 07 08 09 0A 0A 0D		
	Compare buffers with same parameters	Result is +1	
20	Send a GET INPUT command		GET INPUT Proactive command
	Terminal Response, with 2 Text String TLV		
	OD 06 00 11 22 33 44 55		
	ProactiveResponseHandler.getTheHandler() Initialise compareBuffer		
	compareBuffer = 04 00 01 0F		
	<pre>findAndCompareValue() tag = 0Dh, occurrence = 1</pre>	Result is 00h	
	<pre>valueOffset = 0 compareOffset = 0</pre>		
	compareLength = 17		
21	Initialise compareBuffer		
	compareBuffer = 00 11 22 33 44 55	D. W. OOL	
	findAndCompareValue()	Result is 00h	

	tag = 0Dh, occurrence = 2		
	valueOffset = 0		
	compareOffset = 0		
	compareLength = 6		
22	Initialise compareBuffer		
	compareBuffer =		
	00 11 22 33 44 66		
	findAndCompareValue()	Result is -1	
	tag = 0Dh, occurrence = 2		
	<pre>valueOffset = 0</pre>		
	compareOffset = 0		
	compareLength = 6		
23	Send a GET INPUT command		GET INPUT Proactive
23	Send a GET INPUT command		GET INPUT Proactive command
23	Send a GET INPUT command Terminal Response, Text String length = 16		
23			
23	Terminal Response, Text String length = 16		
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F		
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler()		
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer		
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer =	Result is 00h	
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F Compare buffers (with tag 8Dh) tag = 8Dh, occurrence = 1	Result is 00h	
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F Compare buffers (with tag 8Dh)	Result is 00h	
23	Terminal Response, Text String length = 16 Text String TLV = 0D 11 04 00 01 0F ProactiveResponseHandler.getTheHandler() Initialise compareBuffer CompareBuffer = 04 00 01 0F Compare buffers (with tag 8Dh) tag = 8Dh, occurrence = 1	Result is 00h	

6.2.8.19.4 Test Coverage

CRR number	Test case number
N1	12
N2	14
N3	13, 17, 20, 21
N4	15, 18, 22
N5	16, 19
N6	23
P1	1
P2	2, 3, 4, 5, 6
P3	7, 8, 9, 10
P4	11
C1 Does not apply for	
	Proactive Response
	Handler

6.2.9 Class ToolkitRegistry

6.2.9.1 Method allocateTimer

Test Area Reference: API_2_TKR_ATIM

6.2.9.1.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public byte allocateTimer() throws ToolkitException

Normal execution

CRRN1: the returned timer identifier shall be between 01 and 08 inclusive.

CRRN2: the returned timer identifier shall be different from a previously allocated but not released one.

CRRN3: The SIM Toolkit Framework shall trigger the applet when receiving an ENVELOPE(TIMER EXPIRATION) command for the allocated timer.

CRRN4: A call to isEventSet() method for EVENT_TIMER_EXPIRATION should return true if the applet has at least one timer allocated.

Parameters error

No requirements

Context errors

CRRC1: Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE if all the timers are allocated.

CRRC2: Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE if the maximum number of timers have been allocated to this applet according to installation parameter.

6.2.9.1.2 Test suite files

Test Script: API_2_TKR_ATIM_1.scr

Test Applet: API_2_TKR_ATIM_1.java

API_2_TKR_ATIM_2.java

API_2_TKR_ATIM_3.java

Installation parameters:

For this test procedure the non-volatile memory of each instance is 200 (Hexa).

The maximum timer parameter value is as follows for each applet:

- applet 1 (API_2_TKR_ATIM_1): 8 timers

applet 2 (API_2_TKR_ATIM_2): 4 timers

- applet 3 (API_2_TKR_ATIM_3): 0 timer

Load Script: API_2_TKR_ATIM_1.ldr

The load script installs the 6 instances.

Cleanup Script: API_2_TKR_ATIM_1.clr

6.2.9.1.3 Test procedure

ld	Description	API Expectation	APDU Expectation
2	Allocates up to 8 timers (applet 1) 8 * allocateTimer(). Allocate timers more than the maximum (applet 1) The applet 1 allocates 1 more timer.	No exception shall be thrown. Timer ID returned shall be between 01 and 08 inclusive. It shall be different after each call. Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE.	
3	Check applet is Triggered by ENVELOPE(TIMER_EXPIRATION) command (applet1) Send ENVELOPE(TIMER EXPIRATION) with all timers id (not in an increase order). Calls releaseTimer(id) each time a timer expires.	Shall trigger each time an ENVELOPE(TIMER EXPIRATION) is sent to the SIM, for Timer ID = '01' to '08'.	
	Allocate up to 4 timers (applet 2) 4 * allocateTimer().	No exception shall be thrown. Each time, the returned timer identifier shall be between '01' and '08' inclusive. It shall be different after each call.	
5	Allocate timers more than the maximum (applet 3) The applet 3 allocates 1 more timer.	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE.	

6.2.9.1.4 Test Coverage

CRR number	Test case number
N1	1,4
N2	1,4
N3	3
N4	1
C1	2
C2	5

6.2.9.2 Method changeMenuEntry

Test Area Reference: API_2_TKR_CMETB_BSSBZBS

6.2.9.2.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: The SIM Toolkit Framework shall dynamically update the menu stored in the ME by issuing a SET UP MENU proactive command. The later will reflect the changes done for the entry. The SIM Toolkit Framework shall use the data of the EF sume file in order to build the SET UP MENU command.

CRRN2: The default state of the changed menu entry is 'enabled'.

CRRN3: a call to isEventSet() method on EVENT_MENU_SELECTION shall return true before and after the call.

CRRN4: if helpSupported was true then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return true.

CRRN5: if helpSupported was true then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM for this entry, then the SIM Toolkit framework shall trigger the applet.

CRRN6: if help supported was true, the SIM Toolkit Framework shall issue a SETUP MENU command with command qualifier = '80'

CRRN7: if helpSupported was false and if no entries is supporting help then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return false .

CRRN8: if helpSupported was false and if no entries is supporting help then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM, then the SIM Toolkit framework shall not trigger the applet.

CRRN9: The SIM Toolkit Framework shall supply in the SET UP MENU command with the icon identifier provided in the icon identifier list within the item icon identifier list Simple TLV if all the applets registered to the EVENT_MENU_SELECTION provide it.

CRRN10: The SIM Toolkit Framework shall set in the SET UP MENU command with the Icon list qualifier transmitted to the ME as 'icon is not self explanatory' if one of the applet registered prefers this qualifier.

CRRN11: If Next Action Indicator was different from '00', the SIM Toolkit Framework shall issue a SETUP MENU proactive command containing an Items Next Action Indicator simple TLV with the comprehension flag set to 0 as defined in GSM 11.14 [4].

Parameters error

CRRP1: Shall throw java.lang.NullPointerException - if menuEntry is null

CRRP2: Shall throw java.lang.ArrayIndexOutOfBoundsException - if offset would cause access outside array bounds

CRRP3: Shall throw java.lang.ArrayIndexOutOfBoundsException - if length would cause access outside array bounds

CRRP4: Shall throw java.lang.ArrayIndexOutOfBoundsException - if both offset and length would cause access outside array bounds

Context errors

CRRC1: Shall throw a ToolkitException with MENU_ENTRY_NOT_FOUND reason if the Menu Identifier isn't associated to the calling applet instance.

CRRC2: Shall throw ALLOWED_LENGTH_EXCEEDED if the menu entry string is bigger than the allocated space.

6.2.9.2.2 Test suite files

Additional requirements for the GSM personalisation:

- content of EF sume shall be:

- Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_CMETB_BBSSBZBS_1.scr

Test Applet: API_2_TKR_CMETB_BBSSBZBS_1.java

- entry '01' is "Init1"

- entry '02' is "Init2"

Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 2

- Position / Identifier for each menu entry: '01'/'01','02'/'02'

Load Script: API_2_TKR_CMETB_BBSSBZBS_1.ldr

Cleanup Script: API_2_TKR_CMETB_BBSSBZBS_1.clr

6.2.9.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Applet changes the entry's title by menuEntry buffer, with a greater length than the initial length		
	1- ChangeMenuEntry()with parameters:		
	Id = '02' MenuEntry = "UseAllBuffer" Offset = 0	1- No exception shall be thrown.	
	Length = menuEntry.length NextAction = 0	2- shall return true.	
	HelpSupported = false IconQualifier = 0 IconIdentifier = 0.	3- shall return false.	
	2- isEventSet(EVENT_MENU_SELECTION).		The SIM shall issue a
	3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST).		SETUP MENU proactive command which contains the new text for entry ID '02'.
2	Changing the title with part of menuEntry buffer		
	1- changeMenuEntry()with parameters:		
	<pre>Id = '01' MenuEntry = "UsePartOfBuffer"</pre>	1- No exception shall be thrown.	
	Offset = 3 Length = 12	2- Shall return true.	
	<pre>NextAction = 0 HelpSupported = false IconQualifier = 0 IconIdentifier = 0.</pre>	3- Shall return false.	
	2- isEventSet(EVENT_MENU_SELECTION).		The SIM shall issue a
	3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)		SETUP MENU proactive command which contains the new text for entry ID '01'.
3	Length = 0		
	1- changeMenuEntry() for entry '01' and entry '02', with parameters:		
	<pre>Id = '01'/'02' MenuEntry = "LengthEquals0" Offset = 0 Length = 0</pre>	1- No exception shall be thrown.	
	NextAction = 0 HelpSupported = false	2- Shall return true.	
	IconQualifier = 0 IconIdentifier = 0.	3- shall return false.	
	2- isEventSet(EVENT_MENU_SELECTION).		
	3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST).		The SIM shall issue a SETUP MENU proactive command which contains for entry '01'and entry '02', no text part.
4	Setting a next action indicator != 0		
	1- changeMenuEntry()with parameters: Id = '02' MenuEntry = "NextActionIndic" Offset = 0	 No exception shall be thrown. Shall return true. Shall return false. 	The SIM shall issue a SETUP MENU proactive command which contains an
	Length = menuEntry.length NextAction = '10' (SETUP CALL) HelpSupported = false		Items Next Action Indicator list and which contains a command qualifier '80'.

	IconQualifier = 0		
	IconIdentifier = 0		
	2- isEventSet(EVENT_MENU_SELECTION).		
	, ,		
	3-		
	<pre>isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST).</pre>		
	4- changeMenuEntry()with parameters:		
	Id = '02'		
	MenuEntry = "NextActionIndic"		
	Offset = 0 Length = menuEntry.length		
	NextAction = '10' (SETUP CALL)		
	HelpSupported = true		
	<pre>IconQualifier = 0 IconIdentifier = 0</pre>		
_	Charling applet is triggered by a		
5	Checking applet is triggered by a MENU SELECTION HELP REQUEST	Applet is trigged by a	
	MENO_OLLEGIION_HELI _KEQOLOT	MENU_SELECTION_HELP_REQU	
		EST and the Item Identifier is 02	
6	with Item Identifier = '02' help supported=true		
	noip supported—true		
	1- changeMenuEntry()with parameters:		
	changementality () with parameters.		
	<pre>Id = '01' MenuEntry = "HelpSupported"</pre>	4. No expension about he shrows	
	Offset = 0	1- No exception shall be thrown.	
	Length = menuEntry.length	2- Shall return true.	
	NextAction = 0 HelpSupported = true		
	IconQualifier = 0	3- Shall return true.	
	<pre>IconIdentifier = 0</pre>		
	2- isEventSet(EVENT_MENU_SELECTION).		
			The SIM shall issue a
	3- isEventSet(EVENT_MENU_SELECTION_HELP_R		SETUP MENU proactive
	EQUEST).		command which contains a
7	Chapting applet is triggered by a		command qualifier '80'.
7	Checking applet is triggered by a MENU_SELECTION_HELP_REQUEST	Applet is trigged by a	
		MENU_SELECTION_HELP_REQU	
	<pre>Send ENVELOPE(MENU_SELECTION_HELP_REQUEST) with Item Identifier = '01'</pre>	EST and the Item Identifier is 01	
8	Setting icons, help supported = false		
	<pre>1- changeMenuEntry() for entries '01','02', with parameters:</pre>		
	<pre>Id = '01'/'02' MenuEntry = "IconQualifier"</pre>		
	Offset = 0	1- No exception shall be thrown.	
	Length = menuEntry.length NextAction = 0	·	
	HelpSupported = false	2- Shall return true.	
	<pre>IconQualifier = '01'</pre>	3- Shall return false.	
	<pre>IconIdentifier = '02' / '01'</pre>	Jo- Onan retuill laise.	
	2- isEventSet(EVENT_MENU_SELECTION).		
	3-		The SIM shall issue a
	isEventSet(EVENT_MENU_SELECTION_HELP_R		SETUP MENU proactive
	EQUEST).		command which contains an
9	MenuEntry is disabled	1- No exception shall be thrown.	Icon Identifier List.
9	menuLini y is disabled	11- 140 exception shall be tillown.	The SIM shall issue a
	1- disableMenuEntry('01').	2- No exception shall be thrown.	SETUP MENU proactive
	2- changeMenuEntry()with parameters:	O Ob all mater	command which contains
1		3- Shall return true.	the entry. Without Icon
			identifier List Simple TLV

	Id = '01'	4- Shall return false.	
	MenuEntry = "EnableEntry"	4- Shall return laise.	
	Offset = 0		
	Length = menuEntry.length		
	NextAction = 0		
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0		
	<pre>3- isEventSet(EVENT_MENU_SELECTION).</pre>		
	4-		
	isEventSet(EVENT_MENU_SELECTION_HELP_R		
	EQUEST).		
10	MenuEntry is null		
		Chall throw	
	changeMenuEntry()with:	Shall throw	
	MenuEntry = NULL	java.lang.NullPointerException.	
11	Offset causes access outside array bounds		
1	oncor caucoc accoc calciac analy beanac		
	Id = '01'		
	MenuEntry = "Violation"	Shall throw	
	Offset = menuEntry.length +1		
	Length = 0	java.lang.ArrayIndexOutOfBoundsE	
	NextAction = 0	xception.	
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0		
12	Big Offset causes access outside array bounds		
'-	ng onoct sauces access calcius array bearias		
	Id = '01'		
	MenuEntry = "Violation"	Chall throw	
	Offset = 255	Shall throw	
	Length = 1	java.lang.ArrayIndexOutOfBoundsE	
	NextAction = 0	xception.	
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0		
13	Offset < 0 causes access outside array bounds		
13	Onset < 0 causes access outside array bounds		
	Id = '01'		
	MenuEntry = "Violation"		
	Officet - 1	Shall throw	
	Length = 1	java.lang.ArrayIndexOutOfBoundsE	
	NextAction = 0	xception.	
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0		
	· · · · · · · · · · · · · · · · · · ·		
14	Length causes access outside array bounds		
14	Length causes access outside allay boulids		
	Id = '01'		
	MenuEntry = "Violation"	Shall throw	
	Offset = 0	java.lang.ArrayIndexOutOfBoundsE	
	Length = MenuEntry.length + 1	xception.	
	NextAction = 0	Acoption.	
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0.		
15	Length < 0 causes access outside array		
	bounds		
	201100		
	Id = '01'		
	MenuEntry = "Violation"	Shall throw	
	Offset = 0	java.lang.ArrayIndexOutOfBoundsE	
	Length = -1	xception.	
	NextAction = 0	1	
	HelpSupported = false		
	IconQualifier = 0		
	IconIdentifier = 0.		

		I	
16	Both offset and length causes access outside		
	array bounds		
	TA _ 1011		
	<pre>Id = '01' MenuEntry = "Violation"</pre>	Shall throw	
	Offset ∈ [1, MenuEntry.length]	java.lang.ArrayIndexOutOfBoundsE	
	Length = MenuEntry.length	xception.	
	NextAction = 1	Acoption:	
	HelpSupported = false	•	
	<pre>IconQualifier = 0</pre>		
	<pre>IconIdentifier = 0</pre>		
17	Invalid ID used		
	Id = '00'		
	MenuEntry = contains text, != null	Chall throws a Tablist Evaportion with	
	Offset = 0	Shall throw a ToolkitException with	
	Length = menuEntry.length < 16	MENU_ENTRY_NOT_FOUND	
	NextAction = 0	reason code.	
	HelpSupported = false		
	<pre>IconQualifier = 0 IconIdentifier = 0</pre>		
10	ID isn't allocated to a menu entry of this applet		
10	instance		
	instance		
	Id = '0A'		
	MenuEntry = contains text, != null	Shall throw a ToolkitException with	
	Offset = 0	reason code:	
	Length = menuEntry.length < 16	MENU_ENTRY_NOT_FOUND.	
	NextAction = 0 HelpSupported = false	IMENG_ENTRY ENGIGE	
	IconQualifier = 0		
	IconIdentifier = 0		
19	The text is bigger than the allocated space		
	Id = '02'		
	MenuEntry = contains text, != null		
	Offset = 0	Shall throw a ToolkitException with	
	Length = menuEntry.length > 15	reason code:	
	NextAction = 0 HelpSupported = false	ALLOWED_LENGTH_EXCEEDED.	
	IconOualifier = 0		
	IconIdentifier = 0		
20	With a smaller text length than the initial length		
	1. changeMenuEntry()with parameters:		
	Id = '02'		
	MenuEntry = "Init"	1 No expension shall be through	
	Offset = 0	No exception shall be thrown.	
	Length = menuEntry.length	2. Shall return true.	
	NextAction = 0 HelpSupported = false	2. Shan retuin tide.	
	IconQualifier = 0	3. Shall return false.	The SIM shall issue a
	IconIdentifier = 0	o. Shall return raise.	SETUP MENU proactive
	2. isEventSet(EVENT_MENU_SELECTION)		command which contains the new text for entry ID
	3.		'02'.
	isEventSet(EVENT_MENU_SELECTION_HELP_R		, o <u>z</u> .
	EQUEST)		

6.2.9.2.4 Test Coverage

CRR number	Test case number	
N1	1,2,3,4,6,8,9,20	
N2	9	
N3	1,2,3,4,6,8,9,20	
N4	6	
N5	7,5	
N6	6	
N7	1,2,3,4,8,9,20	
N8	To be checked in framework	
	tests and insert cross reference	
	here	
N9	8, 9	
N10	8	
N11	4	
P1	10	
P2	11,12,13	
P3	14,15	
P4	16	
C1	17,18	
C2	19	

6.2.9.3 Method clearEvent

Test Area Reference: API_2_TKR_CEVTB

6.2.9.3.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: A call to isEventSet() method for a cleared event should return false after a call to clearEvent.

CRRN2:The SIM Toolkit Framework shall not trigger the applet on the occurrence of the cleared event anymore.

CRRN3: if event was EVENT_CALL_CONTROL_BY_SIM and after the call, no applet is registered to it, The SIM Toolkit Framework shall allow an applet to register to this event.

CRRN4: if event was EVENT_CALL_CONTROL_BY_SIM and one applet is still registered to these event, The SIM Toolkit Framework shall not allow an applet to register to this event.

CRRN5: if event was EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM and after the call, no applet is registered to it, The SIM Toolkit Framework shall allow an applet to register to this event.

CRRN6: if event was EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM and one applet is still registered to these event, The SIM Toolkit Framework shall not allow an applet to set this event.

Parameters error

CRRP1: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT MENU SELECTION.

CRRP2: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_MENU_SELECTION_HELP_REQUEST.

CRRP3: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_TIMER_EXPIRATION.

CRRP4: Shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED if event was EVENT_STATUS_COMMAND.

Context errors

No requirements

6.2.9.3.2 Test suite files

Test Script: API_2_TKR_CEVTB_1.scr

Test Applet: API_2_TKR_ CEVTB _1.java

As default but applet registers to an event list which contains all defined events in GSM 03.19 [7] excepted those that aren't allowed or supported by setEvent().

Load Script: API_2_TKR_ CEVTB_1.ldr

Cleanup script: API_2_TKR_ CEVTB_1.clr

6.2.9.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Clear ALLOWED unregistered events For events ranging from 1 to 127 excepted those that aren't allowed (EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND), the applet calls: 1- clearEvent() method 2- isEventSet() method	1- No exception is thrown each time. 2- Shall return false each time.	
2	Clear registered events 1- For each ALLOWED and SUPPORTED events, the applet calls setEvent() method. 2- For events ranging from 1 to 127 excepted those that aren't allowed, the applet calls: 2.1- clearEvent() method 2.2- isEventSet() method	 No exception shall be thrown. No exception shall be thrown. Shall return false. 	
3	Clearing NOT ALLOWED events For each event among: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND 1- The applet calls clearEvent(event) method.	1- Each time, clearEvent shall throw a Toolkit Exception with reason EVENT_NOT_ALLOWED.	
4	Checking applet isn't triggered by an ENVELOPE(SMS-PP DOWNLOAD) command 1 - reset and initialise the card 2 - An ENVELOPE(SMS-PP DOWNLOAD) is sent with a TAR referencing applet.	Applet is not trigged by an ENVELOPE(SMS-PP DOWNLOAD) command	

6.2.9.3.4 Test Coverage

CRR number	Test case number	
N1	1,2	
N2	4	
N3	Framework	
N4	Framework	
N5	Framework	
N6	Framework	
P1	3	
P2	3	
P3	3	
P4	3	

6.2.9.4 Method disableMenuEntry

Test Area Reference: API_2_TKR_DMETB

6.2.9.4.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: A call to isEventSet() method on EVENT_MENU_SELECTION shall return the same result before and after the call to disableMenuEntry() method.

CRRN2: A call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST shall return the same result before and after the call to disableMenuEntry() method.

CRRN3: After invocation of this method the SIM Toolkit Framework shall dynamically update the menu stored in the ME .

CRRN4: After invocation of this method, if there is no more enabled menu entries then the SIM Toolkit framework shall issue a SETUP MENU proactive command containing Item Data Object for Item 1 TLV with a length of zero and no value part.

Parameters error

No requirements

Context errors

CRRC1: shall throw a ToolkitException with reason = ENTRY_NOT_FOUND if the menu entry doesn't exist for this applet

6.2.9.4.2 Test suite files

Additional requirements for the GSM personalisation:

- content of EF sume shall be:

- Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_DMETB_1.scr

Test Applet: API_2_TKR_DMETB_1.java

Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 2

- Position / Identifier for each menu entry: '01'/'01', '02'/'02'

Load Script: API_2_TKR_DMETB_1.ldr

Cleanup script: API_2_TKR_DMETB_1.clr

6.2.9.4.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Check the menu state before disabling a previously enabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- reset and initialise the card 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	1- Shall return true2- Shall return false	1- The SIM shall issue a SET UP MENU proactive command with entry '01' and '02'.
2	Check the menu state after disabling a previously enabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- disableMenuEntry('01') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_REQUEST)	 No exception shall be thrown. Shall return true. Shall return false. 	3- The SIM shall issue a SET UP MENU proactive command with entry '02' only.
3	Check the menu before disabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- change Menu Entry '02' to indicate help supported 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	Shall return true Shall return true	3- The SIM shall issue a SET UP MENU proactive command with entry '02', indicating help supported.
4	Check the menu after disabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- disableMenuEntry('02') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	 No exception shall be thrown. Shall return true. Shall return true. 	3- The SIM shall issue a SET UP MENU proactive command with 1st Item TLV with a length of 0.
5	Disabling invalid entries For ID ranging from '00' to 'FF' except '01' and '02', the applet calls disableMenuEntry(ID) method.	Each time a Toolkit Exception with MENU_ENTRY_NOT_FOUND reason code shall be thrown.	

6.2.9.4.4 Test Coverage

CRR number	Test case number	
N1	1,2,3,4	
N2	1,2,3,4	
N3	2,4	
N4	4	
C1	5	

6.2.9.5 Method enableMenuEntry

Test Area Reference: API_2_TKR_EMETB

6.2.9.5.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: A call to isEventSet() method on EVENT_MENU_SELECTION shall return the same result before and after the call to enableMenuEntry() method.

CRRN2: A call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST shall return the same result before and after the call to enableMenuEntry() method.

CRRN3:The SIM Toolkit Framework shall dynamically issue a SETUP MENU proactive command which does contain an ITEM SIMPLE TLV object for this entry.

Parameters error

No requirements

Context errors

CRRC1: shall throw a ToolkitException with reason = MENU_ENTRY_NOT_FOUND if the menu entry doesn't exist for this applet

6.2.9.5.2 Test suite files

Additional requirements for the GSM personalisation:

- content of EF sume shall be:

- Title Alpha Identifier: "TOOLKIT TEST"

Test Script: API_2_TKR_EMETB_1.scr

Test Applet: API_2_TKR_EMETB_1.java

Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 2

- Position / Identifier for each menu entry: '01'/'01', '02'/'02'

Load Script: API_2_TKR_EMETB_1.ldr

Cleanup script: API_2_TKR_EMETB_1.clr

6.2.9.5.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Check menu state before enabling a previously disabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- isEventSet(EVENT_MENU_SELECTION) 2- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST) 3- disableMenuEntry('01')	 Shall return true Shall return false No exception shall be thrown. 	3- The SIM shall issue a SET UP MENU proactive command with entry '02' only.
2	Check menu state after enabling a previously disabled entry not registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- enableMenuEntry('01') 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	 No exception shall be thrown. Shall return true. Shall return false. 	3- The SIM shall issue a SET UP MENU proactive command with entry '01' and '02'.
3	Check menu state before enabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- change Menu Entry '02' to indicate help supported 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST) 4- disableMenuEntry('02')	Shall return true Shall return true No exception shall be thrown	4- The SIM shall issue a SET UP MENU proactive command with entry '01'. The help information available flag.is not verified
4	Check menu state after enabling a previously enabled entry registered to EVENT_MENU_SELECTION_HELP_REQUEST 1- enableMenuEntry('02'). 2- isEventSet(EVENT_MENU_SELECTION) 3- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	 No exception shall be thrown. Shall return true. Shall return true. 	3- The SIM shall issue a SET UP MENU proactive command with entries '01' and '02' indicating help supported.
5	Enabling invalid entries For ID ranging from '00' to 'FF' except '01' and '02', the applet calls enableMenuEntry(ID) method.	Each time a Toolkit Exception with MENU_ENTRY_NOT_FOUND reason code shall be thrown.	

6.2.9.5.4 Test Coverage

CRR number	Test case number	
N1	1,2,3,4	
N2	1,2,3,4	
N3	1,2,3,4	
C1	5	

6.2.9.6 Method getEntry

Test Area Reference: API_2_TKR_GETY

6.2.9.6.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: returns a reference to the applet ToolkitRegistry object of the calling applet.

CRRN2: Each successive call to getEntry() method shall return the same object.

Parameters error

No requirements

6.2.9.6.2 Test suite files

Test Script: API_2_TKR_GETY_1.scr

Test Applet: API_2_TKR_GETY_1.java

Load Script: API_2_TKR_GETY_1.ldr

Cleanup script: API_2_TKR_GETY_1.clr

6.2.9.6.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Installalation	Returns a not null ToolkitRegistry	
	In the constructor, the applet instance calls the getEntry() method.	instance.	
2	Check it returns the same entry	Returns the same ToolkitRegistry	
	The applet calls the getEntry() method again.	instance as for test case 1.	

6.2.9.6.4 Test Coverage

CRR number	Test case number
N1	1
N2	2

6.2.9.7 Method getPollInterval

Test Area Reference: API_2_TKR_GPOL

6.2.9.7.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public short getPollInterval()

Normal execution

CRRN1: shall return a value between 1 and 15300 if applet is registered to EVENT_STATUS_COMMAND event.

CRRN2: shall return POLL_NO_DURATION value (0) if the toolkit applet is not registered to EVENT_STATUS_COMMAND event.

Parameters error

No requirements.

Context errors

No requirements.

6.2.9.7.2 Test suite files

Test Script: API_2_TKR_GPOL_1.scr

Test Applet: API_2_TKR_ GPOL_1.java

Load Script: API_2_TKR_ GPOL_1.ldr

Cleanup script: API_2_TKR_ GPOL_1.clr

6.2.9.7.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Applet isn't registered to EVENT_STATUS_COMMAND		
	getPollInterval().	Shall return 0.	
2	Requesting max duration		
	1- requestPollInterval(15300)	1- No exception shall be thrown.	
	2- Reset and initialize the card	3- Shall return a value between 1	
	3- getPollInterval()	and 15300.	
3	Requesting System Duration		
	1-		
	requestPollInterval(POLL_SYSTEM_DURATION)	1- No exception shall be thrown.	
	2- Reset and initialize the card	3- Shall return a value between 1 and 15300.	
	3- getPollInterval().		
4	Requesting no Duration		
	1- requestPollInterval(POLL_NO_DURATION)		
	2- Reset and initialize the card	1- No exception shall be thrown.3- Shall return 0.	
	3- getPollInterval().		

6.2.9.7.4 Test Coverage

CRR number	Test case number	
N1	2,3	
N2	1,4	

6.2.9.8 Method initMenuEntry

Test Area Reference: API_2_TKR_IMET_BSSBZBS

6.2.9.8.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

- CRRN1: The SIM Toolkit Framework shall automatically update the menu stored in the ME by issuing a SETUP MENU proactive command. The later will reflect the changes done for the entry. The SIM Toolkit Framework shall use the data of the EF sume file in order to build the SET UP MENU command.
- CRRN2: a call to isEventSet() method on EVENT_MENU_SELECTION shall return true after the 1st successful call (without an exception).
- CRRN3: if helpSupported was true then a following call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return true.
- CRRN4: if helpSupported was true then after the completion of the SETUP MENU command, if an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command is received by the SIM for this entry, then the SIM Toolkit framework shall trigger the applet.
- CRRN5: if help supported was true, the SIM Toolkit Framework shall issue a SETUP MENU command with command qualifier = '80'
- CRRN6: if helpSupported was false and there isn't any menu entry supporting help then a call to isEventSet() method on EVENT_MENU_SELECTION_HELP_REQUEST event shall return false.
- CRRN7: The SIM Toolkit Framework shall supply in the SET UP MENU command with the icon identifier provided in the icon identifier list within the item icon identifier list Simple TLV if all the applets registered to the EVENT_MENU_SELECTION provide it.
- CRRN8: The SIM Toolkit Framework shall set in the SET UP MENU command with the Icon list qualifier transmitted to the ME as 'icon is not self explanatory' if one of the applet registered prefers this qualifier.
- CRRN9: If Next Action Indicator was different from '00', the SIM Toolkit Framework shall issue a SETUP MENU proactive command containing an Items Next Action Indicator simple TLV with the comprehension flag set to 0.
- CRRN10: After the completion of the SETUP MENU command, if an ENVELOPE (MENU_SELECTION) command is received by the SIM for this identifier, then the SIM Toolkit framework shall trigger the applet.

Parameters error

- CRRP1: Shall throw java.lang.NullPointerException if menuEntry is null
- CRRP2: Shall throw java.lang.ArrayIndexOutOfBoundsException if offset would cause access outside array bounds
- $CRRP3: Shall\ throw\ java.lang. ArrayIndexOutOfBounds Exception-if\ length\ would\ cause\ access\ outside\ array\ bounds$
- CRRP4:Shall throw java.lang.ArrayIndexOutOfBoundsException if both offset and length would cause access outside array bounds

Context errors

CRRC1: Shall throw ALLOWED_LENGTH_EXCEEDED if the menu entry string is bigger than the allocated space

CRRC2: Shall throw REGISTRY_ERROR if the menu entry cannot be initialised (eg no more item data in applet loading parameter)

6.2.9.8.2 Test suite files

Additional requirements for the GSM personalisation:

- content of EF sume shall be:
 - Title Alpha Identifier: "TOOLKIT TEST"
 - Test case trigger: 1- Applet instantiation
 - 2- Menu selection
 - 3- Menu selection Help Supported

Test Script: API_2_TKR_IMET_BSSBZBS_1.scr

Test Applet: API_2_TKR_IMET_BSSBZBS_1.java

Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 6

- Position / Identifier for each menu entry: '01'/'01', '02'/'02', '03'/'03', '04'/'04', '05'/'05', and '06'/'06'

Load Script: API_2_TKR_IMET_BSSBZBS_1.ldr

Cleanup script: API_2_TKR_IMET_BSSBZBS_1.clr

6.2.9.8.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	NULL as parameter to menuEntry	Shall throw a	
	MenuEntry = NULL	java.lang.NullPointerException.	
2	Offset > menuEntry.length MenuEntry = "ToolkitTest" Offset = 12 Length = 0	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
3	Offset < 0 MenuEntry = "ToolkitTest" Offset = -1 Length = 11	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
4	Offset = 255 MenuEntry = "ToolkitTest" Offset = 255 Length = 11	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
5	<pre>Length = menuEntry.length+1 MenuEntry = "ToolkitTest" Offset = 0 Length = 12</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
6	<pre>Length < 0 MenuEntry = "ToolkitTest" Offset = 0</pre>	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	

1	Length = -1		
7	Offset + length > menuEntry.length MenuEntry = "ToolkitTest" Offset = 11 Length = 1	Shall throw java.lang.ArrayIndexOutOfBoundsE xception.	
8	MenuEntry.length > size allocated at loading for each menu entry MenuEntry = "ToolkitTest impossible" Offset = 0 Length = 16	ALLOWED_LENGTH_EXCEEDED ToolkitException is thrown.	
9	Successful call, menuEntry is the whole buffer		
	1- initMenuEntry() MenuEntry = "TOOLKIT TEST 1" Offset = 0 Length = 14 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0 2- isEventSet(EVENT_MENU_SELECTION)	1- No exception shall be thrown, Shall return ID '01'.2- Shall return true.	
10	Successful call, menuEntry part of a buffer 1- initMenuEntry() MenuEntry = "1234567TOOLKIT TEST 2" Offset = 7 Length = 14 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0 2- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	1- No exception shall be thrown,Shall return ID '02'. 2- Shall return false.	
11	Successful call, menuEntry with help supported 1- initMenuEntry() MenuEntry = "TOOLKIT TEST 3" Offset = 0 Length = 14 NextAction = '00' HelpSupported = true IconQualifier = '00' IconIdentifier = 0 2- isEventSet(EVENT_MENU_SELECTION_HELP_R EQUEST)	1- No exception shall be thrown, Shall return ID '03' 2- Shall return true.	
12	Successful call, menuEntry with an Icon MenuEntry = "TOOLKIT TEST 4" Offset = 0 Length = 14 NextAction = '00' HelpSupported = false IconQualifier = '01' [icon not self explanatory] IconIdentifier = 1 Successful call,	 No exception shall be thrown. Shall return ID '04' No exception shall be thrown. 	
13		17 No avacation aboli ha through	

	menuEntry with a next action indication	2- Shall return ID '05'	
	-		
	MenuEntry = "TOOLKIT TEST 5" Offset = 0		
	Length = 14		
	NextAction = '24' [Select Item] HelpSupported = false		
	IconQualifier = '00'		
	IconIdentifier = 0		
14	Successful call,	No exception shall be thrown, Shall return ID '06'.	
	length = 0	return ib 06.	
	initMenuEntry()		
	MenuEntry = "ToolkitTest"		
	Offset = 0		
	Length = 0 NextAction = '00'		
	HelpSupported = false		
	IconQualifier = '00'		
	<pre>IconIdentifier = 0</pre>		
15	Initialize more entry than allocated at loading		
	MenuEntry = "ToolkitTest"	REGISTRY_ERROR	
	Offset = 0	ToolkitException is thrown.	
	Length = 11		
16	Dynamic update of the menu stored by the ME		Card shall Send a SetUpMenu Proactive
			command:
	Fetch		[CommandQualifier]=help
			supported
			[Alphald]="TOOLKIT TEST"
			[ItemId=1] = "TOOLKIT TEST 1"
			[ItemId=2] = "TOOLKIT
			TEST 2"
			[ItemId=3] = "TOOLKIT
			TEST 3"
			[ItemId=4] = "TOOLKIT TEST 4"
			[ItemId=5] = "TOOLKIT
			TEST 5"
			[ItemId=6] = ""
			[ItemsNextAction]=0600000
			0002400
17	Check Applet is triggered by		
	ENVELOPE(MENU_SELECTION) command	Applet is trigged by an ENVELOPE(MENU_SELECTION)	
	Menu Entry ID = '01'	command & Menu Entry ID = '01'	
18	Check Applet is triggered by	-	
10	ENVELOPE(MENU_SELECTION) command	Applet is trigged by an	
	,	ENVELOPE(MENU_SELECTION)	
	Menu Entry ID = '02'	command & Menu Entry ID = '02'	
19	Check Applet is triggered by		
	ENVELOPE(MENU_SELECTION) command	Applet is trigged by an	
	Menu Entry ID = '03'	ENVELOPE(MENU_SELECTION) command & Menu Entry ID = '03'	
		Command & Mend Entry ID = 03	
20	Check Applet is triggered by ENVELOPE(MENU_SELECTION) command	Applot in trigged by an	
	LIVELOFE(IVIENO_SELECTION) COMMINAND	Applet is trigged by an ENVELOPE(MENU_SELECTION)	
	Menu Entry ID = '04'	command & Menu Entry ID = '04'	
21	Check Applet is triggered by		
21	ENVELOPE(MENU_SELECTION) command	Applet is trigged by an	
		ENVELOPE(MENU_SELECTION)	
	Menu Fntry ID = '05'	command & Menu Entry ID = '05'	
L	Menu Entry ID = '05'	zzmana a mona zmry ib = 00	<u> </u>

	Check Applet is triggered by ENVELOPE (MENU_SELECTION_HELP_REQUEST) command		
	Menu Entry ID = '03'	Applet is trigged by an ENVELOPE(MENU_SELECTION_HELP_REQUEST) command & Menu Entry ID = '03'	
23	Check Applet is triggered by		
	ENVELOPE(MENU_SELECTION) command	Applet is trigged by an	
		ENVELOPE(MENU_SELECTION)	
	Menu Entry ID = '06'	command & Menu Entry ID = '06'	

6.2.9.8.4 Test Coverage

CRR number	Test case number	
N1	16	
N2	9	
N3	11	
N4	22	
N5	11,16	
N6	10	
N7	12,16	
N8	12,16	
N9	13,16	
N10	9,10,11,12,13,14,17,18,19,20,2	
	1,23	
P1	1	
P2	2,3,4	
P3	5,6	
P4	7	
C1	8	
C2	14	

6.2.9.9 Method is Event Set

Test Area Reference: API_2_TKR_IEVSB

6.2.9.9.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

public boolean isEventSet(byte event)

Normal execution

CRRN1: shall return true if the event is set in the Toolkit Registry for the applet

CRRN2: shall return false if the event isn't set in the Toolkit Registry for the applet

Parameters error

No requirements.

Context errors

No requirements

6.2.9.9.2 Test suite files

Test Script: API_2_TKR_IEVSB_1.scr

Test Applet: API_2_TKR_ IEVSB _1.java

Installation parameter:

Same as default applet but with:

- Maximum text length for a menu entry: 15

- Maximum number of menu entries: 1

- Position / Identifier for each menu entry: '01'/'01'

- Maximum number of timers: 1

Load Script: API_2_TKR_ IEVSB_1.ldr

Cleanup script: API_2_TKR_ IEVSB_1.clr

6.2.9.9.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Events aren't set Applet calls isEventSet() for each events ranging from 1 to 127 excepted EVENT_FORMATTED_SMS_PP_ENV and EVENT_MENU_SELECTION.	Shall return false each time.	
2	For EVENT_FORMATTED_SMS_PP_ENV isEventSet (EVENT_FORMATTED_SMS_PP_ENV)	Shall return true.	
3	For EVENT_MENU_SELECTION isEventSet (EVENT_MENU_SELECTION)	Shall return true	
4	After clearing EVENT_FORMATTED_SMS_PP_ENV 1- clearEvent(EVENT_FORMATTED_SMS_PP_ENV) 2- isEventSet(EVENT_FORMATTED_SMS_PP_ENV) .	1- No exception shall be thrown.2- Shall return false.	
5	Setting events 1- For each SUPPORTED and ALLOWED events for setEvent(), applet calls: 1.1- setEvent() method 1.2- isEventSet() method.	1.1- No exception shall be thrown.1.2- Shall return true each time.	
6	For EVENT_MENU_SELECTION_HELP_REQUEST 1- isEventSet (EVENT_MENU_SELECTION_HELP_REQUEST) 2- call changeMenuEntry with help supported 3- isEventSet (EVENT_MENU_SELECTION_HELP_REQUEST)	1- Shall return false.3- Shall return true	
7	For EVENT_TIMER_EXPIRATION 1- isEventSet(EVENT_TIMER_EXPIRATION)	1- Shall return false.	

	2- call allocateTimer() 3- isEventSet(EVENT_TIMER_EXPIRATION)	3- Shall return true	
8	For EVENT_STATUS_COMMAND		
	1- isEventSet(EVENT_STATUS_COMMAND) 2- call requestPollInterval(POLL_SYSTEM_DURATION) 3- isEventSet(EVENT_STATUS_COMMAND)	1- Shall return false.3- Shall return true	

6.2.9.9.4 Test Coverage

CRR number	Test case number	
N1	2,3,4,5,6,7,8	
N2	1,5,6,7,8	

6.2.9.10 Method releaseTimer

Test Area Reference: API_2_TKR_RTIMB

6.2.9.10.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: if it was the last allocated timer for the applet then a following call to isEventSet() method for EVENT_TIMER_EXPIRATION should return false.

CRRN2: if applet has timers allocated then a call to isEventSet(EVENT_TIMER_EXPIRATION) shall return true.

CRRN3: After invocation of the method the indicated timer shall be released and available for reallocation.

CRRN4: The applet is deregistered of the EVENT_TIMER_EXPIRATION for the indicated Timer Identifier.

Parameters error

CRRP1: shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer identifier isn't between 1 and 8.

Context errors

CRRC1: shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer is valid but isn't allocated to this applet.

6.2.9.10.2 Test suite files

Test Script: API_2_TKR_RTIMB_1.scr

Test Applet: API_2_TKR_RTIMB_1.java

Installation parameter:

As Default, except max timer which is set to 8.

Load Script: API_2_TKR_RTIMB_1.ldr

Cleanup script: API_2_TKR_RTIMB_1.clr

6.2.9.10.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Releasing not allocated timers For each timer ID ranging from '00' to 'FF', applet calls releaseTimer(ID).	Each time, method shall throw a ToolkitException with reason code INVALID_TIMER_ID.	
2	Releasing allocated timers		
	1- 8 * allocateTimer() .	1- No exception shall be thrown.2- Each time, no exception shall be	
	2- 7 * releaseTimer(id).	thrown.	
	3- isEventSet(EVENT_TIMER_EXPIRATION)	3- Shall return true	
3	Releasing invalid timer ID	1- Shall throw a ToolkitException	
	1- releaseTimer('FF') method	with INVALID_TIMER_ID reason code.	
	2- isEventSet(EVENT_TIMER_EXPIRATION)	2- Shall return true.	
4	Releasing last timer		
	1- releaseTimer(last timer allocated)	1- No exception shall be thrown.	
	2- isEventSet(EVENT_TIMER_EXPIRATION)	2- Shall return false.	
5	Checking we can allocate timers after they have been released		
	8 * allocateTimer().	No exception shall be thrown.	
6	Releasing all timers.		
	For 1 to 8, releaseTimer(id).	No exception shall be thrown.	
7	Checking applet isn't triggered by ENVELOPE(TIMER_EXPIRATION) command	Applet is not trigged by an ENVELOPE(TIMER_EXPIRATION) command	
1	Send ENVELOPE(TIMER_EXPIRATION)	oommand .	

6.2.9.10.4 Test Coverage

CRR number	Test case number	
N1	4	
N2	2,3	
N3	5,6	
N4	7	
P1	1,3	
C1	Framework	

6.2.9.11 Method requestPollInterval

Test Area Reference: API_2_TKR_RPOLS

6.2.9.11.1 Conformance requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: If duration is between 1 and 15300 or equal to POLL_SYSTEM_DURATION, the applet registers to EVENT_STATUS_COMMAND.

CRRN2: If duration is POLL_NO_DURATION, the applet is deregistered from EVENT_STATUS_COMMAND.

Parameters error

CRRP1: the method should throw a ToolkitException with REGISTRY_ERROR reason if duration is > 15300 or is < -1 (POLL_SYSTEM_DURATION).

Context errors

No Requirements.

6.2.9.11.2 Test suite files

Test Script: API_2_TKR_RPOLS_1.scr

Test Applet: API_2_TKR_RPOLS_1.java

Load Script: API_2_TKR_RPOLS_1.ldr

Cleanup script: API_2_TKR_RPOLS_1.clr

6.2.9.11.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Requesting a value between 1 and 15300 s		
	1- isEventSet(EVENT_STATUS_COMMMAND)	1- Shall return false.	
	2- For duration ranging from 1 to 15300, requestPollInterval(duration).	2- No exception shall be thrown.	
	3- isEventSet(EVENT_STATUS_COMMAND).	3- Shall return true.	
2	1- Check Applet is triggered by a STATUS commandreset and card initialisation 2- Send STATUS command	2- Applet is trigged by a STATUS command	
3	Requesting POLL SYSTEM DURATION		
	1- isEventSet(EVENT_STATUS_COMMMAND).	1- Shall return true.	
	2- RequestPollInterval(POLL_SYSTEM_DURATI	2- No exception shall be thrown.	
	ON).	3- Shall return true.	
	3- IsEventSet(EVENT_STATUS_COMMAND).	5- Shall return true.	
4	Check Applet is triggered by a STATUS command 1- reset and card initialisation 2- Send STATUS command	2- Applet is trigged by a STATUS command	
5	Requesting invalid duration	Each time a TaalkitEvaantian with	
	For duration ranging from 15301 to 15305, -2 requestPollInterval(duration)	Each time, a ToolkitException with REGISTRY_ERROR reason code, shall be thrown.	
6	Requesting POLL NO DURATION		
	1- isEventSet(EVENT_STATUS_COMMMAND)	1- Shall return true.	
	2- requestPollInterval(POLL_NO_DURATION)	2- No exception shall be thrown.	
	3- isEventSet(EVENT_STATUS_COMMAND)	3- Shall return false.	

7	reset and card initialisation	2- Applet is not trigged by a STATUS command	

6.2.9.11.4 Test Coverage

CRR number	Test case number
N1	1,2,3,4
N2	6,7
P1	5

6.2.9.12 Method setEvent

Test Area Reference: API_2_TKR_SEVTB

6.2.9.12.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: a following call to isEventSet() method with the same event id shall answer true for the applet.

CRRN2: the SIM Toolkit Framework shall trigger the applet if an occurrence of the set event happens.

CRRN3: this method shall accept all the events defined in GSM 0319 excepted: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND

Parameters error

CRRP1: shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason if event is 0.

CRRP2: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_MENU_SELECTION.

CRRP3: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_MENU_SELECTION_HELP_REQUEST.

CRRP4: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_TIMER_EXPIRATION.

CRRP5: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if event is EVENT_STATUS_COMMAND.

Context errors

CRRC1: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if event is EVENT_CALL_CONTROL_BY_SIM but another applet is already registered to it.

CRRC2: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if event is EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM but another applet is already registered to it.

6.2.9.12.2 Test suite files

Test Script: API_2_TKR_SEVTB_1.scr

Test Applet: API_2_TKR_SEVTB_1.java

API_2_TKR_SEVTB_2.java

Load Script: API_2_TKR_SEVTB_1.ldr

The load script installs the 2 instances.

Cleanup script: API_2_TKR_SEVTB_1.clr

6.2.9.12.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Applet 1 is triggered by ENVELOPE(SMS_	•	•
	PP_FORMATTED) command.		
	TT_TORMATTED) Communic.	Applet 1 shall be triggered	
	Cand FM7FI ODF (CMC DD FORMATTED)	Applet 1 shall be triggered	
	Send ENVELOPE(SMS_PP_FORMATTED)		
2	Setting ALLOWED and SUPPORTED events		
-	Setting ALLOWED and SUFFORTED events		
	1- For all events defined in GSM 0319		
	(from 1 to 19) and allowed:		
	EVENT_PROFILE_DOWNLOAD,		
	EVENT_FORMATTED_SMS_PP_ENV,		
	EVENT_FORMATTED_SMS_PP_UPD,		
	EVENT_UNFORMATTED_SMS_PP_ENV,		
	EVENT_UNFORMATTED_SMS_PP_UPD,		
	EVENT_UNFORMATTED_SMS_CB,		
	EVENT_CALL_CONTROL_BY_SIM,	1.1- No exception shall be thrown.	
	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM,	1.1- No exception shall be tillown.	
	EVENT_EVENT_DOWNLOAD_MT_CALL,	1.2- Shall return false.	
	EVENT_EVENT_DOWNLOAD_CALL_CONNECTED,	1.2- Shall return false.	
1	EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED,		
	EVENT_EVENT_DOWNLOAD_LOCATION_STATUS,	1.3- No exception shall be thrown.	
	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY,		
	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABL	1.4- Shall return true.	
	E, EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS		
	EVENT_EVENT_DOWNDOAD_CARD_READER_STATOS	1.5- No exception shall be thrown.	
	1.1- clearEvent(event)	·	
	The Grant Commercial C		
	1.2- isEventSet(event)		
	1.3- setEvent(event)		
	1.4- isEventSet(event)		
	1.1- ISEVENESEC (EVENC)		
	1.5- clearEvent(event)		
3	Event 0		
		Shall throw a ToolkitException with	
	Call setEvent(0)	EVENT_NOT_SUPPORTED reason	
		code.	
		· · · · · · · · · · · · · · · · · · ·	
4	Setting EVENT_MENU_SELECTION		
'		Shall throw a ToolkitException with	
1	Call setEvent(EVENT_MENU_SELECTION)	EVENT_NOT_ALLOWED reason	
1	_ ,	code.	
1		0000.	
5	Setting		
3	EVENT_MENU_SELECTION_HELP_REQUEST		
	LVENT_WENO_OLLEGION_HELF_NEWOEST	Shall throw a ToolkitException with	
	Call	EVENT_NOT_ALLOWED reason	
	setEvent(EVENT_MENU_SELECTION_HELP_REQUES	code.	
	T)		
6	Setting EVENT_TIMER_EXPIRATION		
1		Shall throw a ToolkitException with	
	Call setEvent(EVENT_TIMER_EXPIRATION)	EVENT_NOT_ALLOWED reason	
		code.	
7	Setting EVENT_STATUS_COMMAND		
'		Shall throw a ToolkitException with	
	1	S With a residence with	

	Call setEvent(EVENT_STATUS_COMMAND)	EVENT_NOT_ALLOWED reason code.	
		code.	
8	Setting EVENT_CALL_CONTROL_BY_SIM		
	Call setEvent(EVENT_CALL_CONTROL_BY_SIM)	No Exception shall be thrown	
9	Setting		
_	EVENT_MO_SHORT_MESSAGE_CONTROL_B		
	Y_SIM		
	Call	No Exception shall be thrown	
	setEvent(EVENT_MO_SHORT_MESSAGE_CONTROL_B		
	Y_SIM)		
10	Check applet is triggered by an	Applet is trigged by an	
	ENVELOPE(CALL_CONTROL_BY_SIM)	ENVELOPE(CALL_CONTROL_BY_	
	Trigger the applet	SIM)	
11	Check applet is triggered by an ENVELOPE(MO_SHORT_MESSAGE_CONTRO	Applet is trigged by an	
	L BY SIM)	ENVELOPE(MO_SHORT_MESSAG	
	Trigger the Applet	E_CONTROL_BY_SIM)	
12	Applet 2 is triggered by ENVELOPE(SMS_	Applet 2 is trigged by an	
	PP_DOWNLOAD) command.	ENVELOPE(SMS_	
	Trigger the applet 2	PP_DOWNLOAD) command	
13	Applet 2 registers to	Shall throw a ToolkitException with	
	CALL_CONTROL_BY_SIM but it is already assigned	EVENT_ALREADY_REGISTERED reason code.	
	setEvent(EVENT_CALL_CONTROL_BY_SIM)	reason code.	
14	Applet 2 registers to	Shall throw a ToolkitException with	
	MO_MESSAGE_CONTROL_BY SIM	EVENT_ALREADY_REGISTERED	
	but it is already assigned setEvent(EVENT_MO_SHORT_MESSAGE_CONTROL_B	reason code.	
	Y_SIM)		
	U	II.	

6.2.9.12.4 Test Coverage

CRR number	Test case number
N1	2
N2	1, 8,9,10, 11, 12
N3	2,4,5,6,7
P1	3
P2	4
P3	5
P4	6
P5	7
C1	13
C2	14

6.2.9.13 Method setEventList

Test Area Reference: API_2_TKR_SEVL_BSS

6.2.9.13.1 Conformance Requirement:

The method with following header shall be compliant to its definition in the API.

Normal execution

CRRN1: for all events set successfully by this method, a call to isEventSet() method should return true.

CRRN2: the SIM Toolkit Framework shall trigger the applet if an occurrence of one of the successfully registered events happens.

CRRN3: this method shall accept all the events defined in GSM 0319 excepted: EVENT_MENU_SELECTION, EVENT_MENU_SELECTION_HELP_REQUEST, EVENT_TIMER_EXPIRATION, EVENT_STATUS_COMMAND.

Parameters error

CRRP1: shall throw a java.lang.NullPointerException if eventList is null.

CRRP2: shall throw a java.lang.ArrayIndexOutOfBoundsException if offset would cause access outside array bounds.

CRRP3: shall throw a java.lang.ArrayIndexOutOfBoundsException if length would cause access outside array bounds.

CRRP4: shall throw a java.lang.ArrayIndexOutOfBoundsException if both offset and length would cause access outside array bounds.

CRRP5: shall throw a ToolkitException with EVENT_NOT_SUPPORTED reason if event is 0.

CRRP6: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_MENU_SELECTION.

CRRP7: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_MENU_SELECTION_HELP_REQUEST.

CRRP8: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_TIMER_EXPIRATION.

CRRP9: shall throw a ToolkitException with EVENT_NOT_ALLOWED reason if eventList contains EVENT_STATUS_COMMAND.

Context errors

CRRC1: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if eventList contains EVENT_CALL_CONTROL_BY_SIM but another applet is already registered to it.

CRRC2: shall throw a ToolkitException with EVENT_ALREADY_REGISTERED if eventList contains EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM but another applet is already registered to it.

6.2.9.13.2 Test suite files

Test Script: API_2_TKR_SEVL_BSS_1.scr

Test Applet: API_2_TKR_ SEVL_BSS _1.java

API_2_TKR_ SEVL_BSS _2.java

Load Script: API_2_TKR_ SEVL_BSS_1.ldr

The load script installs the 2 instances.

Cleanup script: API_2_TKR_ SEVL_BSS_1.clr

6.2.9.13.3 Test Procedure

Ī	ld	Description	API Expectation	APDU Expectation
ſ	1	Applet 1 Registering all eventList buffer	1- No exception shall be thrown.	
		EventList = all allowed events defined in	2- No exception shall be thrown.	

	GSM 0319: EVENT_PROFILE_DOWNLOAD, EVENT_FORMATTED_SMS_PP_ENV, EVENT_FORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB, EVENT_CALL_CONTROL_BY_SIM, EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_EVENT_DOWNLOAD_CALL_CONNECTED, EVENT_EVENT_DOWNLOAD_LOCATION_STATUS, EVENT_EVENT_DOWNLOAD_USER_ACTIVITY, EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE, EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS 1- For each event in EventList clearEvent(event) 2- setEventList(eventList) Offset = 0 Length = eventList.lentgh	3- Each time shall return true.4- No exception shall be thrown.	
	<pre>3- For all events in eventList isEventSet(event) 4- For each event in EventList clearEvent(event)</pre>		
2	Registering part of eventList buffer		
	<pre>EventList = all allowed events defined in GSM 0319 (see test case 1). 1- For each event in EventList clearEvent(event) 2- setEventList(eventList, offset, length) Offset > 0 Length = eventList.lentgh - offset 3- For all events in eventList: isEventSet(event) 4- For each event in EventList: clearEvent(event)</pre>	 No exception shall be thrown. No exception shall be thrown. Each time shall return true for events ranging from offset to offset+length else shall return false. No exception shall be thrown. 	
3	Null buffer EventList = null	Shall throw a java.lang.NullPointerException Exception	
<u> </u>	Out of bosonda affect		
4	Out of bounds offset Offset = eventList.length Length = 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
5	Out of bounds and big offset Offset = 255 Length = 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
6	Offset < 0 Offset = -1 Length = 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	
7	Out of bounds length Offset = 0 Length = eventList.length + 1	Shall throw a java.lang.ArrayIndexOutOfBounds Exception	

8	Out of bounds and big length		
°	Out of bounds and big length	Shall throw a	
	Offset = 0	java.lang.ArrayIndexOutOfBounds	
	Length = 255	Exception	
	1		
9	Length < 0	Shall throw a	
	Offset = 0	java.lang.ArrayIndexOutOfBounds	
	Length = -1	Exception	
		·	
10	Out of bounds offset + Length	Shall throw a	
	Offset + length > eventList.length + 1	java.lang.ArrayIndexOutOfBounds	
	origon / rengen / cvenozibe/rengen / r	Exception	
11	Event 0		
		Shall throw a ToolkitException with	
	Call setEventList(eventList) with eventList indicating event 0	EVENT_NOT_SUPPORTED	
	eventhist indicating event o	reason code.	
40	EVENT MENU CELECTION		
12	EVENT_MENU_SELECTION	Shall throw a ToolkitException with	
	Call setEventList(eventList) with	reason code	
	eventList indicating EVENT_MENU_SELECTION	EVENT_NOT_ALLOWED.	
13	EVENT_MENU_SELECTION_HELP_REQUEST		
	Call confirmentiation (amount in) 112	Shall throw a ToolkitException with	
	Call setEventList(eventList) with eventList indicating	reason code	
	EVENT_MENU_SELECTION_HELP_REQUEST	EVENT_NOT_ALLOWED.	
14	EVENT_TIMER_EXPIRATION		
14	EVENT_TIMEN_EXPINATION	Shall throw a ToolkitException with	
	Call setEventList(eventList) with	reason code	
	eventList indicating	EVENT_NOT_ALLOWED.	
	EVENT_TIMER_EXPIRATION		
15	EVENT_STATUS_COMMAND		
	Call setEventList(eventList) with	Shall throw a ToolkitException with	
	eventList indicating EVENT_STATUS_COMMAND	reason code EVENT_NOT_ALLOWED.	
	,	EVENT_NOT_ALLOWED.	
16	Setting EVENT_CALL_CONTROL_BY_SIM		
	0		
	gotEventligt/ligt 0 2) with ligh	Shall not throw an exception	
	<pre>setEventList(List, 0, 2) with List containing</pre>		
	EVENT_CALL_CONTROL_BY_SIM &		
	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM		
17	Check applet is triggered by an	Applet is trigged by an	
	ENVELOPE(CALL_CONTROL_BY_SIM) Reset and initialise the card	ENVELOPE(CALL_CONTROL_BY	
	Trigger the applet	_SIM)	
18	Check applet is triggered by an	Applet is trigged by an	
	ENVELOPE(MO_SHORT_MESSAGE_CONTROL	ENVELOPE(MO_SHORT_MESSA	
	_ BY_SIM) Trigger the applet	GE_CONTROL_BY_SIM)	
19	Applet 2 registers to CALL CONTROL BY SIM		
•	but it is already assigned	Shall throw a ToolkitException with	
	setEventList(MonoEventList,0,1) with	EVENT_ALREADY_REGISTERED	
	MonoEventList containing	reason code.	
	EVENT_CALL_CONTROL_BY_SIM		
20	Applet 2 registers to		
	MO_SHORT_MESSAGE_CONTROL_BY_SIM		
	but it is already assigned	Shall throw a ToolkitException with	
	setEventList(MonoEventList,0,1) with	EVENT_ALREADY_REGISTERED	
	MonoEventList containing	reason code.	
	EVENT_MO_SHORT_MESSAGE_CONTROL_BY		
	_SIM		
1			

6.2.9.13.4 Test Coverage

CRR number	Test case number	
N1	1,2	
N2	16,17,18	
N3	1,2,11, 12,13,14,15	
P1	3	
P2	4,5,6	
P3	7,8,9	
P4	10	
P5	11	
P6	12	
P7	13	
P8	14	
P9	15	
C1	19	
C2	20	

6.2.10 Class ViewHandler

It is not possible to test the methods provided by this class as it is declared 'abstract'; it will be done in the class inheriting it: EditHandler, EnvelopeHandler, ProactiveResponseHandler, ProactiveHandler.

6.2.11 Class ToolkitException

6.2.11.1 Exception Constants

Test Area Reference: API_2_TKE_CONS

6.2.11.1.1 Conformance requirement:

There is no API, only constants.

Normal execution

CRRN1: The Constants of the class ToolkitException shall all have the same name and value defined in the GSM03.19.

Parameters error

No requirements

Context errors

No requirements

6.2.11.1.2 Test suite files

None

6.2.11.1.3 Test procedure

The constants in Java are resolved at compilation time, therefore a runtime test is not useful. No test of constants will be performed.

6.2.11.2 Constructor ToolkitException

Test Area Reference: API_2_TKE_COORS

6.2.11.2.1 Conformance requirement:

The constructor with following headershall compliant to its definition in the API.

public ToolkitException(short reason)

Normal execution

CRRN1: Construct a ToolkitException instance with the specified reason.

Parameters error

No requirements

Context errors

No requirements

6.2.11.2.2 Test suite files

No additional requirements for the GSM personalisation:

Test Script: API_2_TKE _COORS_1.scr

Test Applet: API_2_TKE _COORS_1.java

Load Script: API_2_TKE _COORS_1.ldr

Cleanup script: API_2_TKE _COORS_1.clr

6.2.11.2.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	reason = (short) 19	ToolkitException.getReason() =	
		(short)19	

6.2.11.2.4 Test Coverage

CRR number	Test case number
N1	1

6.2.11.3 Method throwlt

Test Area Reference: API_2_TKE_THITS

6.2.11.3.1 Conformance requirement:

The method with following header shall compliant to its definition in the API.

Normal execution

CRRN1: Throws the JCRE instance of the ToolkitException class with the specified reason.

CRRN2: extends javacard.framework.CardRuntimeException

Parameters error

No requirements

Context errors

No requirements

6.2.11.3.2 Test suite files

No additional requirements for the GSM personalisation:

Test Script: API_2_TKE_ THITS_1.scr

Test Applet: API_2_TKE_ THITS_1.java

Load Script: API_2_TKE_THITS_1.ldr

Cleanup Script: API_2_TKE_THITS_1.clr

6.2.11.3.3 Test procedure

ld	Description	API Expectation	APDU Expectation
1	Throws the JCRE instance of ToolkitException	Reason = 0	
	with the specified reason		
2	Throws the JCRE instance of ToolkitException	Reason = 1	
	with the specified reason		
3	Throws the JCRE instance of ToolkitException	Reason = 15	
	with the specified reason		
4	ToolkitException extends	Reason = 0	
	javacard.framework.CardRuntimeException		
5	ToolkitException extends	Reason = 1	
	javacard.framework.CardRuntimeException		
6	ToolkitException extends	Reason = 15	
	javacard.framework.CardRuntimeException		

6.2.11.3.4 Test Coverage

CRR number	Test case number
N1	1, 2, 3
N2	4, 5, 6

6.3 SIM Toolkit Framework

6.3.1 Minimum Handler Availability

This test area tests the rules that define the minimum requirements for the availability of the system handlers.

6.3.1.1 ProactiveHandler

Test Area Reference: FWK_MHA_PAHD

6.3.1.1.1 Conformance Requirement

Normal Execution

CRRN1: If a proactive session is not ongoing the ProactiveHandler is available from the invocation to the termination of the processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_STATUS_COMMAND

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_PROFILE_DOWNLOAD

6.3.1.1.2 Test Suite Files

Test Script: FWK_MHA_ PAHD_1.scr

Test Applet: FWK_MHA_PAHD_1.java

FWK_MHA_PAHD_2.java

Load Script: FWK_MHA_ PAHD_1.ldr

Cleanup Script: FWK_MHA_ PAHD_1.clr

Parameter File: FWK_MHA_ PAHD_1.par

6.3.1.1.3 Test Procedure

ld	Description	API /Framework Expectation	APDU Expectation
1	Applets registration to all events and Proactive Handler availability with EVENT_PROFILE_DOWNLOAD		
	Applet1 is registered to all events defined in [7]. Using the methods initMenuEntry () for EVENT_MENU_SELECTION, requestPollInterval () for EVENT_STATUS_COMMAND, allocateTimer () for EVENT_TIMER_EXPIRATION and setEventList () for the rest of the events.		
	Applet2 is registered to all events defined in [7], except EVENT_CALL_CONTROL_BY_SIM and EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM. Using the methods initMenuEntry () for	1- Applet1 is triggered	
	EVENT_MENU_SELECTION, requestPollInterval () for EVENT_STATUS_COMMAND, allocateTimer () for EVENT_TIMER_EXPIRATION and setEventList () for the rest of the events.	2- No exception is thrown.3- Applet2 is triggered	
	The priority of applet1 is higher than priority of applet2 1-Terminal Profile command is sent to SIM without the facility of SET_EVENT_LIST, POLL_INTERVAL,SET UP IDLE MODE TEXT and SET UP MENU.	4- No exception is thrown	
	2-Applet1 gets the Proactive Handler Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD		
	3-Applet2 gets the Proactive Handler Applet2 is deregistered to EVENT_PROFILE_DOWNLOAD		
2	Proactive Handler availability with EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported	1- Applet1 is triggered	
	1-Envelope menu selection with help request is sent to the SIM	2- No exception is thrown	
	2-Applet1 gets the Proactive Handler	3- Applet2 is triggered	
	3-Envelope menu selection with help request is sent to the SIM	4- No exception is thrown	
	4-Applet2 gets the Proactive Handler		

ld	Description	API /Framework Expectation	APDU Expectation
3	Proactive Handler availability with	·	•
	EVENT_MENU_SELECTION		
		1- Applet1 is triggered	
	1-Envelope menu selection is sent to the SIM		
	 	2- No exception is thrown.	
	O annichi note the present and	110 CACCPROTES BITOWIT.	
	2-Applet1 gets the Proactive Handler		
		3- Applet2 is triggered	
	3-Envelope menu selection is sent to the		
	SIM		
		4- No exception is thrown.	
<u></u>	4-Applet2 gets the Proactive Handler	. 110 CACCPROTT IS UTIOWIT.	
4	Proactive Handler availability with		
	EVENT_FORMATTED_SMS_PP_ENV	1- Applet1 is triggered	
	1-Envelope dataDownLoad formatted is sent	i Appieti is tilggered	
	to the SIM		
		2- No exception is thrown.	
	2-Applet1 gets the Proactive Handler		
		3- Applet2 is triggered	
	3-Envelope dataPownLoad formatted is sent		
	3-Envelope dataDownLoad formatted is sent to the SIM		
		4- No exception is thrown.	
	4-Applot2 gots the Presetive Handley		
	4-Applet2 gets the Proactive Handler		
5	Proactive Handler availability with		
	EVENT_UNFORMATTED_SMS_PP_ENV		
		1- Applet1 is triggered	
	1-Envelope dataDownLoad unformatted is sent to the SIM		
	Beur co cue and		
		2- No exception is thrown.	
	2-Applet1 gets the Proactive Handler	1.10 Oxoophori is unlown.	
		3- Applet2 is triggered	
		A No support	
	3-Applet2 gets the Proactive Handler	4- No exception is thrown.	
6	Proactive Handler availability with		
	EVENT_UNFORMATTED_CELL BROADCAST		
		1- Applet1 is triggered	
	1-Envelope cell broadcast unformatted is sent to the SIM		
	Bene to the Biri	2- No exception is thrown	
	2-Applet1 gets the Proactive Handler	2- NO evcebuou is miromii	
		3- Applet2 is triggered	
		4- No exception is thrown	
7	3-Applet2 gets the Proactive Handler Proactive Handler availability with		
'	EVENT_TIMER_EXPIRATION		
		1- Applet1 is triggered	
	1-Timer Id =1 Envelope Timer Expiration is sent to the		
	SIM	2- No exception is thrown.	
	2-Applet1 gets the Proactive Handler	3- Applet2 is triggered	
	3-Timer id=2		
	Envelope Timer Expiration is sent to the		
	SIM	4- No exception is thrown	
	4- Applet2 gets the Proactive Handler		
8	Proactive Handler availability with		
	EVENT_CALL_CONTROL_BY_SIM		

ld	Description	API /Framework Expectation	APDU Expectation
		1- Applet1 is triggered	•
	1-Envelope call control by SIM is sent to the SIM		
	CHC DIM	2- No exception is thrown.	
	2-Applet1 gets the Proactive Handler	2 110 Oxoophori is unown.	
9	Proactive Handler availability with		
	EVENT_MO_SHORT_MESSAGE_CONTROL		
	1 Tanadana wa shash wasana sashasil las GTM	1- Applet1 is triggered	
	1-Envelope mo short message control by SIM is sent to the SIM		
	2-Applet1 gets the Proactive Handler	2- No exception is thrown	
10	Proactive Handler availability with		
	EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent	1- Applet1 is triggered	
	to the SIM		
	2-Applet1 gets the Proactive Handler	2- No exception is thrown.	
		3- Applet2 is triggered	
	2 Applet 2 gots the December Wordler	4- No exception is thrown	
	3-Applet2 gets the Proactive Handler		
11	Proactive Handler availability with		
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT		
	ED		
	1-Envelope event download call connected		
	is sent to the SIM	1- Applet1 is triggered	
	2-Applet1 gets the Proactive Handler	2- No exception is thrown.	
		·	
	2 Applet 2 gots the Dreastive Handler	3- Applet2 is triggered	
	3-Applet2 gets the Proactive Handler	4- No exception is thrown	
12	Proactive Handler availability with		
	EVENT_EVENT_DOWNLOAD_CALL_DISCONN		
	ECTED	1- Applet1 is triggered	
	1-Envelope event download call	. Applet to triggered	
	disconnected is sent to the SIM	2- No exception is thrown.	
		3- Applet2 is triggered	
	2-Applet1 gets the Proactive Handler	O- Applete is triggered	
		4- No exception is thrown.	
	3-Applet2 gets the Proactive Handler		
13	Applets triggering with		
	EVENT_EVENT_LOCATION_STATUS	1- Applet1 is triggered	
	1-Envelope event download location status		
	is sent to the SIM	O Na assauli i ii	
	2-Applet1 gets the Proactive Handler	2- No exception is thrown.	
		3- Applet2 is triggered	
	3-Applet 2 gets the Presetive Mandler	4- No exception is thrown	
	3-Applet2 gets the Proactive Handler		
14	Proactive Handler availability with		
	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Envelope event download user activity is		
	. TITE I . TITS GOMMITOGG ADOL GOOLVILY ID	I	

sent to SIM 2-Applet1 gets the Proactive Handler 2-Applet2 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 4- No exception is thrown 3-Applet2 is triggered 4- No exception is thrown 4- Applet1 is triggered 4- No exception is thrown 3- Applet2 gets the Proactive Handler 4- No exception is thrown 3- Applet2 gets the Proactive Handler 4- No exception is thrown 4- Applet1 is triggered 4- No exception is thrown 4- Applet1 is triggered 4- No exception is thrown 4- Applet1 is triggered 4- No exception is thrown 4- Applet2 is triggered 4- No exception is thrown 5- Applet2 gets the Proactive Handler 7- Proactive Handler availability with	ld	Description	API /Framework Expectation	APDU Expectation
2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 4- No exception is thrown 4- Applet1 is triggered 4- Applet1 gets the Proactive Handler 4- No exception is thrown 3- Applet2 gets triggered 4- No exception is thrown 4- Applet1 is triggered 4- Applet1 is triggered 4- No exception is thrown 5- Applet1 is triggered 4- No exception is thrown 4- No exception is thrown 5- Applet2 is triggered 4- No exception is thrown 5- Applet1 is triggered 6- Applet1 is triggered 7- Applet1 is triggered 8- Applet2 is triggered 9- No exception is thrown 9- Applet1 is triggered 1- Applet1 i				
3-Applet2 gets the Proactive Handler 15			The state of the s	
3-Applet2 gets the Proactive Handler 15		2-Applet1 gets the Presetive Handler		
3-Applet2 gets the Proactive Handler 15		2-Appleti gets the Ploattive Handler	2- No exception is thrown	
3-Applet2 gets the Proactive Handler 15				
4- No exception is thrown Proactive Handler availability with EVENT_EVENT_DOWNLOAD IDLE_SCREEN AVAILABLE 1Envelope event download idle screen available is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 16 Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS 1Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 1- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 1- Status command is sent to the SIM 2- Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown.			3- Applet2 is triggered	
4- No exception is thrown Proactive Handler availability with EVENT_EVENT_DOWNLOAD IDLE_SCREEN AVAILABLE 1Envelope event download idle screen available is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 16 Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS 1Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 1- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 1- Status command is sent to the SIM 2- Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown.		3-Applet2 gets the Proactive Handler		
Proactive Handler availability with EVENT_EVENT_DOWNLOAD IDLE_SCREEN AVAILABLE 1-Applet1 gets the Proactive Handler 1- Applet1 is triggered		5 Applet2 gets the Frontier Handler	4- No exception is thrown	
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Envelope event download idle screen available is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 3- Applet2 gets the Proactive Handler 4- No exception is thrown 1- Applet1 is triggered 4- No exception is thrown 1- Applet1 is triggered 4- No exception is thrown 1- Applet1 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet1 is triggered	15	Proactive Handler availability with		
1-Envelope event download idle screen available is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 3- Applet2 gets the Proactive Handler 16 Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER STATUS 1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown.				
1- Envelope event download idle screen available is sent to the SIM 2-Applet1 gets the Proactive Handler 3- Applet2 gets the Proactive Handler 16		AVAILABLE		
available is sent to the SIM 2-Applet1 gets the Proactive Handler 3- Applet2 gets the Proactive Handler 4- No exception is thrown 16		1 8	1- Applet1 is triggered	
2-Applet1 gets the Proactive Handler 3- Applet2 gets the Proactive Handler 16				
3- Applet2 is triggered 4- No exception is thrown 16		avariable is sens to one bin		
3- Applet2 is triggered 4- No exception is thrown 16			2- No exception is thrown.	
4- No exception is thrown 16 Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS 1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1-Applet1 is triggered 4- No exception is thrown 2- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 1- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown.		2-Applet1 gets the Proactive Handler		
3- Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS 1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown.			3- Applet2 is triggered	
3- Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS 1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown.				
16 Proactive Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS 1-Envelope event download card reader status is sent to the SIM		3- Applet2 gets the Proactive Handler	4- No exception is thrown	
1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1- Applet1 is triggered 1- Applet1 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet3 is triggered 1- Applet4 is triggered 1- Applet4 is triggered 1- Applet5 is triggered 1- Applet6 is triggered 1- Applet7 is triggered 1- Applet8 is triggered 1-		5 Applet2 gets the Houetive Handler		
1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1- Applet1 is triggered 1- Applet1 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet3 is triggered 1- Applet4 is triggered 1- Applet4 is triggered 1- Applet5 is triggered 1- Applet6 is triggered 1- Applet7 is triggered 1- Applet8 is triggered 1-				
1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown 1- Applet1 is triggered 1- Applet1 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet2 is triggered 1- Applet3 is triggered 1- Applet4 is triggered 1- Applet4 is triggered 1- Applet5 is triggered 1- Applet6 is triggered 1- Applet7 is triggered 1- Applet8 is triggered 1-	16	Proactive Handler availability with		
1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 2-Applet2 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 4- No exception is thrown 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown. 3- Applet1 is triggered 4- No exception is thrown.	10			
1-Envelope event download card reader status is sent to the SIM 2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 4- No exception is thrown 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler 19 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2- Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered				
2-Applet1 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 3-Applet2 gets the Proactive Handler 4- No exception is thrown 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 1- Applet1 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet1 is triggered 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered			1- Applet1 is triggered	
2-Applet1 gets the Proactive Handler 3- Applet2 is triggered 4- No exception is thrown 17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered		-		
3- Applet2 gets the Proactive Handler 3- Applet2 gets the Proactive Handler 4- No exception is thrown 1- Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet1 is triggered 1- Applet1 is triggered 1- Applet1 is triggered 3- Applet2 is triggered		status is sent to the SIM	O No superstine in the same	
3-Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered		2-Applet1 gets the Proactive Handler	2- No exception is thrown.	
3-Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered			3- Applet2 is triggered	
3-Applet2 gets the Proactive Handler Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered 1- Applet1 is triggered 1- Applet1 is triggered 1- Applet1 is triggered 3- Applet2 is triggered			o rippioi is miggores	
17 Proactive Handler availability with EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown. 1- Applet1 is triggered			4- No exception is thrown	
EVENT_STATUS_COMMAND 1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered 2- No exception is thrown. 3- Applet2 is triggered	4-			
1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered	17			
1-Status command is sent to the SIM 2-Applet1 gets the Proactive Handler 3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered 3- Applet2 is triggered		EVENT_STATUS_COMMAND	1- Applet1 is triggered	
3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 2- No exception is thrown.		1-Status command is sent to the SIM	T-Applett is triggered	
3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 2- No exception is thrown.				
3- Applet2 is triggered 4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 2- No exception is thrown.		2-Applot1 gots the Presstive Handler	2- No exception is thrown.	
4- No exception is thrown. 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 1- Applet1 is triggered 2- No exception is thrown. 3- Applet2 is triggered		2 Apprect dece the broadcrive Handler		
3- Applet2 gets the Proactive Handler 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered			3- Applet2 is triggered	
3- Applet2 gets the Proactive Handler 18 Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered			1. No exception is thrown	
Proactive Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered	1	3- Applet2 gets the Proactive Handler	T- NO exception is tillown.	
UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered				
UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered				
UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered	18	Proactive Handler availability with		
1-An unrecognized Envelope (BER TLV Tag unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered		UNRECOGNIZED_ENVELOPE		
unrecognized) is sent to the SIM 2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered		1 2 4 7 7 7 7 7 7	1- Applet1 is triggered	
2-Applet1 gets the Proactive Handler 2- No exception is thrown. 3- Applet2 is triggered				
2-Applet1 gets the Proactive Handler 3- Applet2 is triggered			2- No exception is thrown	
		2-Applet1 gets the Proactive Handler	o.copuon lo unown.	
3-Applet2 gets the Proactive Handler 4- No exception is thrown			3- Applet2 is triggered	
3-Applet2 gets the Proactive Handler 4- No exception is thrown				
		3-Applet2 gets the Proactive Handler	4- No exception is thrown	

6.3.1.1.4 Test Coverage

CRR Number	Test Case Number	
------------	------------------	--

CRR Number	Test Case Number
CRRN1	1,2,3,4,5,6,7,8,9,10,11,12,
	13,14,15,16,17,18

6.3.1.2 ProactiveResponseHandler

Test Area Reference: FWK_MHA_ PRHD

6.3.1.2.1 Conformance Requirement

Normal Execution

CRRN1: The ProactiveResponseHandler is available after the first call to the ProactiveHandler.send method to the termination of the processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

 ${\tt EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE}$

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_STATUS_COMMAND

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_PROFILE_DOWNLOAD

6.3.1.2.2 Test Suite Files

Test Script: FWK_MHA_ PRHD_1.scr

Test Applet: FWK_MHA_ PRHD_1.java

FWK_MHA_PRHD_2.java

Load Script: FWK_MHA_ PRHD_1.ldr

Cleanup Script: FWK_MHA_ PRHD_1.clr

Parameter File: FWK_MHA_ PRHD_1.par

6.3.1.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to all events and Proactive Response Handler availability with		
	EVENT_PROFILE_DOWNLOAD		
	1- Applet1 is registered to all events defined in [7], applet2 is registered to all events defined in [7] except EVENT_CALL_CONTROL_BY_SIM and		
	EVENT_MO_SMS_CONTROL_BY_SIM. Using the methods initMenuEntry() for EVENT_MENU_SELECTION,		
	requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.		
	1-Terminal Profile command is sent to the SIM without the facility of SET_EVENT_LIST and POLL_INTERVAL, ,SET UP IDLE MODE TEXT and SET UP MENU.	1-Applet1 is triggered No exception is thrown	
	Applet1 builds a proactive command DISPLAY TEXT. 2- ProactiveHandler.send() method is		2- The proactive command DISPLAY TEXT is fetched
	called	3- No exception is thrown	TERMINAL RESPONSE
	3- ProactiveResponseHandler.getTheHandler() method is called		
	Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD	4- Applet2 is triggered	
	Applet1 execution is finished		
	Applet2 builds a proactive command DISPLAY TEXT.		5 The constitution of the
	4- ProactiveHandler.send() method is called		5- The proactive command DISPLAY TEXT is fetched
		6- No exception is thrown	TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called		
	Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD		

ld	Description	API/Framework Expectation	APDU Expectation
2	Proactive Response Handler availability with	·	
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported		
	1-Envelope menu selection with help request is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3- ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL RESPONSE
	Applet1 execution is finished Envelope menu selection with help request is sent to the SIM		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	
3	Proactive Response Handler availability with EVENT_MENU_SELECTION		
	1-Envelope menu selection is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	. I I I I I I I I I I I I I I I I I I I
	Applet1 execution is finished		
	4-Envelope menu selection is sent to the SIM	4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	5- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
	6-ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	TERMINAL RESPONSE

ld	Description	API/Framework Expectation	APDU Expectation
4	Proactive Response Handler availability with		
	EVENT_FORMATTED_SMS_PP_ENV		
	1-Envelope dataDownLoad formatted is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called Applet1 execution is finished	3- No exception is thrown	TERMINAL RESPONSE
	4-Envelope dataDownLoad formatted is sent to the SIM	4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	5-ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
	method is called	6- No exception is thrown	TERMINAL RESPONSE
5	Proactive Response Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-Envelope dataDownLoad unformatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2- ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3- ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	Applet1 execution is finished	4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
6	Proactive Response Handler availability with		•
	EVENT_UNFORMATTED_SMS_CB		
	1-Envelope call broadcast unformatted is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called.	3- No exception is thrown	TERMINAL RESPONSE
		4- Applet2 is triggered	
	Applet1 execution is finished		
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	
7	Proactive Response Handler availability with EVENT_TIMER_EXPIRATION		
	Timer id=1 1-Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	Applet1 execution is finished	o no exception is thrown	
	Timer id=2 Envelope Timer Expiration is sent to the SIM		
	Applet builds a proactive command DISPLAY TEXT	4- Applet2 is triggered	
	4-ProactiveHandler.send() method is called		
		5- No exception is thrown	
	5-ProactiveResponseHandler.getTheHandler() method is called		6- A proactive command
	Applet2 execution is finished		DISPLAY TEXT is fetched TERMINAL RESPONSE
			TERMINAL ILLOI ONGE

ld	Description	API/Framework Expectation	APDU Expectation
8	Proactive Response Handler availability with EVENT_CALL_CONTROL_BY_SIM		
	1-Envelope call control by sim is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		
			2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	<pre>3-ProactiveResponseHandler.getTheHandler() method is called</pre>	3- No exception is thrown	
9	Proactive Response Handler availability with _ MO_SHORT_MESSAGE_CONTROL_BY_SIM		
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	Applet1 execution is finished		

ld	Description	API/Framework Expectation	APDU Expectation
10	Proactive Response Handler availability with	-	-
	EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called.		TERMINAL RESPONSE
	Applet1 execution is finished	3- No exception is thrown	
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is		
	called		5- A proactive command DISPLAY TEXT is fetched
	5-		TERMINAL RESPONSE
	ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	
11			
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT		
	ED		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
	3-ProactiveResponseHandler.getTheHandler() method is called		TERMINAL RESPONSE
	Applet1 execution is finished	3- No exception is thrown	
		4- Applet2 is triggered	
	Applet builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	
	moditor to cuttor	I.	I

ld	Description	API/Framework Expectation	APDU Expectation
12	Proactive Response Handler availability with	P	
	EVENT_EVENT_DOWNLOAD_CALL_DISCONN		
	ECTED		
	1-Envelope event download call	1- Applet1 is triggered	
	disconnected is sent to the SIM	The process of the state of the	
	Applet1 builds a proactive command DISPLAY		
	TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command
			DISPLAY TEXT is fetched
			TERMINAL RESPONSE
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler()		
	method is called	3- No exception is thrown	
	Applet1 execution is finished		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	IEAI		
	4- ProactiveHandler.send() method is		
	called		
			5- A proactive command
	5-		DISPLAY TEXT is fetched
	ProactiveResponseHandler.getTheHandler()		TEDMINIAL DECRONCE
	method is called	6- No exception is thrown	TERMINAL RESPONSE
13	Proactive Response Handler availability with	o No exception is thown	
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT		
	ED		
	1-Envelope event download location status	1- Applet1 is triggered	
	is sent to the SIM	The process of the stage of the	
	Applet1 builds a proactive command DISPLAY TEXT		
	1EX1		
	2-ProactiveHandler.send() method is called		2-A proactive command
			DISPLAY TEXT is fetched
			TERMINAL RESPONSE
			. LANIMARIA INCOLONOL
	3-ProactiveResponseHandler.getTheHandler()		
	method is called	3- No exception is thrown	
	Applet1 execution is finished		
	11 11 11 11 11 11 11 11 11 11 11 11 11		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY		
	TEXT		
	4- ProactiveHandler.send() method is		
	called		5- A proactive command DISPLAY TEXT is fetched
			DISPLAT TEXT IS TETCHED
			TERMINAL RESPONSE
	5-	6- No exception is thrown	
	ProactiveResponseHandler.getTheHandler() method is called		
<u> </u>		I	I

ld	Description	API/Framework Expectation	APDU Expectation
14	Proactive Response Handler availability with	•	·
	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Envelope event download user activity is	1 Applet1 is triggered	
	sent to the SIM	T- Applet is triggered	
	Applet1 builds a proactive command DISPLAY		
	TEXT		
	2-ProactiveHandler.send() method is called		
			2-A proactive command
			DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	TERMINAL REGI GROE
		-	
	Applet1 execution is finished	4 Applot2 is triggored	
		4- Applet2 is triggered	
	Applet 2 builds a projective command DIGDIAN		
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is		5- A proactive command
	called		DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5-		
	ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	
15	Proactive Response Handler availability with		
	EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_		
	AVAILABLE		
	1-Envelope event download idle screen	1- Applet1 is triggered	
	available is sent to the SIM		
	Applet1 builds a proactive command DISPLAY		
	TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command
	2 123351Venanate1.56na() method 15 called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler()	3- No exception is thrown	
	method is called		
	Applot1 organism is finished		
	Applet1 execution is finished		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY		
	TEXT		5- A proactive command
	4- ProactiveHandler.send() method is		DISPLAY TEXT is fetched
	called		TERMINAL RESPONSE
			I EKIVIIINAL KEOPUNOE
	5-		
	ProactiveResponseHandler.getTheHandler()	6- No exception is thrown	
	method is called		

ld	Description	API/Framework Expectation	APDU Expectation
16	Proactive Response Handler availability with		-
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2-A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	Applet1 execution is finished		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		
	4- ProactiveHandler.send() method is called		5- A proactive command DISPLAY TEXT is fetched
	carred		TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler()		
17	method is called Proactive Response Handler availability with	6- No exception is thrown	
17	EVENT_STATUS_COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	Applet1 builds a proactive command DISPLAY TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called	3- No exception is thrown	
	Applet1 execution is finished		
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		5- A proactive command
	4- ProactiveHandler.send() method is called		DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	5- ProactiveResponseHandler.getTheHandler() method is called	6- No exception is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
18	Proactive Response Handler availability with UNRECOGNIZED_ENVELOPE 1-An unrecognized Envelope is sent to the	1- Applet1 is triggered	·
	SIM Applet1 builds a proactive command DISPLAY		
	TEXT		
	2-ProactiveHandler.send() method is called		2- A proactive command DISPLAY TEXT is fetched
			TERMINAL RESPONSE
	3-ProactiveResponseHandler.getTheHandler() method is called Applet1 execution is finished	3- No exception is thrown	
		4- Applet2 is triggered	
	Applet2 builds a proactive command DISPLAY TEXT		5- A proactive command DISPLAY TEXT is fetched
	4- ProactiveHandler.send() method is called		TERMINAL RESPONSE
		6- No exception is thrown	
	5-		
	ProactiveResponseHandler.getTheHandler() method is called		

6.3.1.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2,3,4,5,6,7,8,9,10,11,12,13,
	14,15,16,17,18

6.3.1.3 EnvelopeHandler

Test Area Reference: FWK_MHA_ ENHD

6.3.1.3.1 Conformance Requirement

Normal Execution

CRRN1: The EnvelopeHandler and its content are available for all toolkit applets triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_UNRECOGNIZED_ENVELOPE

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

Context Errors

CRRC1: The EnvelopeHandler and its content are not available for any toolkit applet triggered from the invocation to the termination of their processToolkit method for the following events:

EVENT_STATUS_COMMAND

EVENT_PROFILE_DOWNLOAD

6.3.1.3.2 Test Suite Files

Test Script: FWK_MHA_ ENHD_1.scr

Test Applet: FWK_MHA_ENHD_1.java

FWK_MHA_ENHD_2.java

Load Script: FWK_MHA_ ENHD_1.ldr

Cleanup Script: FWK_MHA_ENHD_1.clr

Parameter File: FWK_MHA_ ENHD_1.par

6.3.1.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet1 and Applet2 registration and Envelope Handler availability with EVENT_PROFILE_DOWNLOAD		•
	1- Applet1 is registered to all events defined [7]. Using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.		
	Applet2 is registered to all events defined [7] except EVENT_CALL_CONTROL_BY_SIM and EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM. Using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer for EVENT_TIMER_EXPIRATION and setEventList for the rest of the events.	1- No exception is thrown	
	2-Terminal Profile command is sent to SIM without the facility of SET_EVENT_LIST ,SETUP_IDLE_MODE_TEXT ,POLL_INTERVAL and SETUP MENU	2- Applet1 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD	3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
	4-EnvelopeHandler.getTheHandler() method is called by Applet2 Applet2 is deregistered to EVENT_PROFILE_DOWNLOAD	4- Applet2 is triggered 5- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
2	Envelope Handler availability with EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported		
	Envelope menu selection with help request is sent to the SIM		
	Applet1 is triggered 1-EnvelopeHandler.getTheHandler() method is called by Applet1	1- No exception is thrown. Applet1 finalizes.	
	2-Envelope menu selection with help request is sent to the SIM	2- Applet2 is triggered	
3	3-EnvelopeHandler.getTheHandler() method is called by Applet2 Envelope Handler availability with EVENT_MENU_SELECTION	3- No exception is thrown.	
	1-Envelope menu selection is sent to the SIM	1- Applet1 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
	3-Envelope menu selection is sent to the SIM	3- Applet2 is triggered	
	4-EnvelopeHandler.getTheHandler() method is called by Applet2	4- No exception is thrown.	

Id Description API/Framework Expectation 4 Envelope Handler availability with	APDU Expectation
EVENT_FORMATTED_SMS_PP_ENV	
1-A EVENT_FORMATTED_SMS_PP_ENV envelope is 1- Applet1 is triggered	
sent to the SIM	
2-EnvelopeHandler.getTheHandler() method 2- No exception is thrown.	
is called by Applet1	
Applet1 finalizes.	
3-A EVENT_FORMATTED_SMS_PP_ENV envelope is 3- Applet2 is triggered	
sent to the SIM	
4-EnvelopeHandler getTheHandler() method 4- No exception is thrown.	
4-EnvelopeHandler.getTheHandler() method is called by Applet2	
5 Envelope Handler availability with	
EVENT_UNFORMATTED_SMS_PP_ENV	
1-An unformatted sms pp envelope is sent 1. Applet1 is triggered	
1-An unformatted sms pp envelope is sent to the SIM	
2-EnvelopeHandler.getTheHandler() method	
is called by Applet1 2- No exception is thrown.	
Applet1 finalizes.	
3- Applet2 is triggered	
3-EnvelopeHandler.getTheHandler() method is called by Applet2 4- No exception is thrown.	
6 Envelope Handler availability with	
EVENT_UNFORMATTED_CB	
1 Franchisco and have described for the describe	
1-Envelope cell broadcast unformatted is sent to the SIM	
1- Applet1 is triggered	
2-EnvelopeHandler.getTheHandler() method	
is called by Applet1	
Applet1 finalizes. 2- No exception is thrown	
3- Applet2 is triggered	
3-EnvelopeHandler.getTheHandler() method	
4- No exception is thrown	
7 Envelope Handler availability with EVENT_TIMER_EXPIRATION	
Timer id=1	
1-Envelope Timer Expiration is sent to the SIM	
2-EnvelopeHandler.getTheHandler() method 2- No exception is thrown.	
is called by Applet1	
Applet1 finalizes.	
Timer id=2	
3-Envelope Timer Expiration is sent to the SIM	
3- Applet2 is triggered	
4-EnvelopeHandler.getTheHandler() method	
is called by Applet2	
Applet2 finalizes. 4- No exception is thrown.	
8 Envelope Handler availability with	
EVENT_CALL_CONTROL_BY_SIM	
1-Envelope call control by sim is sent to 1- Applet1 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	the SIM		
	2-EnvelopeHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	

ld	Description	API/Framework Expectation	APDU Expectation
9	Envelope Handler availability with EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM	·	·
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1.	2- No exception is throw	
10	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
	THE TANKE TO SEE THE TOTAL TO SEE THE TOTAL TOTA	3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	4- No exception is thrown.	
11	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- Applet2 is triggered	
12	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_CALL_DISCONE CTTED	4- No exception is thrown.	
	1-Envelope event download call disconnected is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 finalizes.	2- No exception is thrown. 3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method	4- No exception is thrown.	
13	is called by Applet2 Envelope Handler availiability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS	·	
	1-Envelope event download location status is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	3- Applet2 is triggered	
	TO CATTOA DY APPIECZ	4- No exception is thrown.	

ld	Description	API/Framework Expectation	APDU Expectation
14	Envelope Handler availiability with	·	-
	EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Envelope event download user activity is	1- Applet1 is triggered	
	sent to the SIM	pp.occ to triggorou	
	2-EnvelopeHandler.getTheHandler() method		
	is called by Applet1	2- No exception is thrown	
	Applet1 finalizes.		
		3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method		
45	is called by Applet2	4- No exception is thrown	
15	Envelope Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_		
	AVAILABLE		
	1 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5 5		
	1-Envelope event download idle screen available is sent to the SIM	1- Applet1 is triggered	
		. Applet to the	
	2-EnvelopeHandler.getTheHandler() method		
	is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
	APPIECI IIIIAIIIZES.	o.kopkon lo unown.	
		3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method		
	is called by Applet2	4- No exception is thrown.	
16	Envelope Handler availiability with		
	EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS		
	_		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method		
	is called by Applet1	2- No exception is thrown.	
	Applet1 finalizes.	3- Applet2 is triggered	
	3-EnvelopeHandler.getTheHandler() method	4. No evention is thrown	
	is called by Applet2	4- No exception is thrown.	
17	Envelope Handler availaibility with EVENT STATUS COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method	2- A Toolkit exception	
	is called by Applet1	HANDLER_NOT_AVAILABLE is	
		thrown	
		3- Applet2 is triggered	
		, applote is triggered	
	3-EnvelopeHandler.getTheHandler() method	4- A Toolkit exception	
	is called by Applet2	HANDLER_NOT_AVAILABLE is thrown	
18	Envelope Handler availiability with EVENT_		
'	UNRECOGNIZED_ENVELOPE		
	1-An unrecognized Envelope is sent to the	A Appleted in this pro-	
	I-An unrecognized Envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeHandler.getTheHandler() method	O No superii i ii	
	is called by Applet1 Applet1 finalizes.	2- No exception is thrown.	
		3- Applet2 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	3-EnvelopeHandler.getTheHandler() method is called by Applet2	4- No exception is thrown.	

6.3.1.3.4 Test Coverage

CRR Number Test Case Number	
CRRN1	14,15,16,17,18,19,20,21
CRRN2	14,15,16,17,18,19,20,21
CRRC1	1,2,3,4,5,6,7,8,9,10,11,12,13

6.3.1.4 EnvelopeResponseHandler

Test Area Reference: FWK_MHA_ ERHD

6.3.1.4.1 Conformance Requirement

Normal Execution

CRRN1: The handler is available for all triggered toolkit applets from the invocation of the processToolkit method of the toolkit applet until a toolkit applet has posted an envelope response or the first invocation of the ProactiveHandler.send method for the following events:.

EVENT_FORMATTED_SMS_PP_ENV

EVENT_UNFORMATTED_SMS_PP_ENV

EVENT_CALL_CONTROL

EVENT_SMS_MO_CONTROL

EVENT_UNRECOGNIZED_ENVELOPE

CRRN2: After a call to the post method the handler is not longer available

CRRN3: After a call to the send method the handler is not longer available

Context Errors

CRRC1: The handler is not available for the following events:

EVENT_UNFORMATTED_SMS_CB

EVENT_MENU_SELECTION

EVENT_MENU_SELECTION_HELP_REQUEST

EVENT_TIMER_EXPIRATION

EVENT_EVENT_DOWNLOAD_MT_CALL

EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

EVENT_STATUS_COMMAND

EVENT_PROFILE_DOWNLOAD

6.3.1.4.2 Test Suite Files

Test Script: FWK_MHA_ ERHD_1.scr

Test Applet: FWK_MHA_ ERHD_1.java

FWK_MHA_ERHD_2.java

Load Script: FWK_MHA_ ERHD_1.ldr

Cleanup Script: FWK_MHA_ ERHD_1.clr

Parameter File: FWK_MHA_ ERHD_1.par

6.3.1.4.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Toolkit Applet1 and Toolkit Applet2 registration and Envelope Response Handler availability with EVENT_PROFILE_DOWNLOAD		
	1- Applet1 Toolkit 1 is registered to all events defined in [7]. Using the methods initMenuEntry() for EVENT_MENU_SELECTION, requestPollInterval() for EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList() for the rest of the events.		
	Applet2 Toolkit 2 is registered to EVENT_UNFORMATTED_SMS_PP_ENV and EVENT_UNRECOGNIZED_ENVELOPE.	1- No exception is thrown	
	2-Terminal Profile command is sent to SIM without the facility of SET_EVENT_LIST ,SETUP_IDLE_MODE_TEXT, SETUP_MENU and POLL_INTERVAL.	2- Applet1 is triggered	
	Applet1 is triggered 32-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 Applet1 is deregistered to EVENT_PROFILE_DOWNLOAD	3- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
2	Envelope Response Handler availiability with		-
	EVENT_MENU_SELECTION_HELP_REQUEST		
	Perform SIM initialization with all the facilities supported		
	1-Envelope menu selection with help request is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is	
		thrown	
3	Envelope Response Handler availiability with EVENT_MENU_SELECTION		
	1-A envelope menu selection is sent to the SIM	1- Applet1 is triggered	
	The Applet1 is triggered		
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
4	Envelope Response Handler availability with		
	EVENT_UNFORMATTED_CB		
	1-Envelope cell broadcast unformatted is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
5	Envelope Response Handler availiability with EVENT_TIMER_EXPIRATION		
	1-Envelope Timer Expiration is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
6	Envelope Response Handler availiability with EVENT_EVENT_DOWNLOAD_MT_CALL		
	1-Envelope event download mt call is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2 -A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
7	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED		
	1-Envelope event download call connected is sent to the SIM	1- Applet1 is triggered.	

ld	Description	API/Framework Expectation	APDU Expectation
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
8	Envelope Response Handler availiability with EVENT_EVENT_DOWNLOAD_CALL_DISCONN ECTED 1-Envelope event download call disconnected is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
9	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_LOCATION_STA TUS 1-Envelope event download location status is sent to the SIM	1- Applet1 is triggered.	
	2-Applet A obtains the Envelope Response Handler	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
10	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_USER_ACTIVITY 1-Envelope event download user activity is sent to the SIM		
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
11	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE 1-Envelope event download idle screen available is sent to the SIM	1- Applet1 is triggered.	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
12	Envelope Response Handler availability with EVENT_EVENT_DOWNLOAD_CARD_READERSTATUS		
	1-Envelope event download card reader status is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is	

ld	Description	API/Framework Expectation	APDU Expectation
	·	thrown	
13	Envelope Response Handler availability with EVENT_STATUS_COMMAND		
	1-Status command is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
14	Envelope Response Handler availability with EVENT_FORMATTED_SMS_PP_ENV		
	1-A formatted sms pp envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 3-Applet1 builds an additional information for response packet and it calls the post		3- The response packet is
	method 4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method) The Applet1 finalizes	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	sent
	5-A EVENT_FORMATTED_SMS_PP_ENV envelope is sent to the SIM	5- Applet1 is triggered	
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	6- No Exception is thrown	7. The proceeding command
	7-Applet1 builds a proactive command and it calls the send() method		7- The proactive command is sent
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	8- Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
15	Envelope Response Handler availability with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-A unformatted sms pp envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the envelope response and it calls the post() method 4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	3- The envelope response is sent
	The Applet1 finalizes	5- Applet2 is triggered.	
	5-EnvelopeResponseHandler.getTheHandler()		

ld	Description	API/Framework Expectation	APDU Expectation
	method is called	6- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown.	
	Applet2 finalizes.	unown.	
	6-A unformatted sms pp envelope is sent to the SIM	7- Applet1 is triggered	
	method is carred.	8- No exception is thrown.	
	8-Applet1 builds a proactive command and it calls the send() method		9- The proactive command is fetched and the Terminal
	9-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	10- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	response is issued.
		Applet1 finalizes	
		11- Applet2 is triggered	
	10-EnvelopeResponseHandler.getTheHandler() method is called by Applet2	12- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown	
16	Envelope Response Handler availability with EVENT_CALL_CONTROL_BY_SIM		
	1-Envelope call control by sim is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		
	3-Applet1 builds the envelope response and it calls the postAsBERTLV() method	2- No exception is thrown.	3- The envelope response is sent
	4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	4- Toolkit exception	
	The Applet1 finalizes	HANDLER_NOT_AVAILABLE is thrown for each method	
	5-Envelope call control by sim is sent to the SIM		
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	5- Applet1 is triggered	
	7-Applet1 builds a proactive command and it calls the send() method	6- No Exception is thrown	7- The proactive command
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)		is fetched and the Terminal response is issued
		8- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
17	Envelope Response Handler availability with EVENT_MO_SHORT_MESSAGE_CONTROL_B Y SIM		
	1-Envelope mo short message control by sim is sent to the SIM	1- Applet1 is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the envelope response and it calls the postAsBERTLV() method		3-The envelope response is sent
	4-Applet1 calls all methods of the Envelope Response Handler (including the inherited method) The Applet1 finalizes	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
	5-Envelope mo short message control by \sin is sent to the SIM	5- Applet1 is triggered	
	6-EnvelopeResponseHandler.getTheHandler() method is called by Applet1		
	7-Applet1 builds a proactive command and it calls the send method	6- No exception is thrown	7- The proactive command
	8-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	8- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	is fetched and the Terminal Response is issued
18	Envelope Response Handler availability with EVENT_UNRECOGNIZED_ENVELOPE		
	1-An unrecognized Envelope is sent to the SIM	1- Applet1 is triggered	
	2-EnvelopeResponseHandler.getTheHandler() method is called by Applet1	2- No exception is thrown.	
	3-Applet1 builds the envelope response and it calls the postAsBERTLV() or post method		3- The envelope response is sent
	4-Applet1 calls all methods of Envelope Response Handler (including the inherited method) The Applet1 finalizes	4- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	Some
	The Appreci Tinatizes	5- Applet2 is triggered.	
	54-EnvelopeResponseHandler.getTheHandler() method is called	6- A Toolkit exception HANDLER_NOT_AVAILABLE is	
	Applet2 finalizes	thrown for each method	
	6-An unrecognized Envelope is sent to the SIM	7- Applet1 is triggered.	
	7-EnvelopeResponseHandler.getTheHandler() method is called	8- No exception is thrown.	
	8-Applet1 builds a proactive command and it calls the send() method		9- The proactive command
	9-Applet1 calls all methods of the Envelope Response Handler (including the inherited method)	10- A Toolkit exception HANDLER_NOT_AVAILABLE is	is fetched and the Terminal response is issued
		thrown for each method Applet1 finalizes	
		11- Applet2 is triggered	
	10-EnvelopeResponseHandler.getTheHandler() method is called by Applet2	33	

ld	Description	API/Framework Expectation	APDU Expectation
		12- A Toolkit exception HANDLER_NOT_AVAILABLE is thrown for each method	
19	The envelope response is sent when a proactive session is ongoing		
	1-A formatted SMS PP envelope is sent to the SIM.	1- Applet1 is triggered.	
	2-Proactive command DISPLAY TEXT is built and it calls the send() method.		2- 91 XX
	3-A call control by sim envelope is sent to the SIM.	3- Applet1 is triggered	
	4-EnvelopeResponseHandler.getTheHandler() method is called by Applet1 5-Applet1 builds the envelope response and it calls the postAsBERTLV	4- No exception is thrown	
	Te carib the posthobaria		5-The envelope response is sent 9F YY
			GET RESPONSE Data 91 XX Fetch DISPLAY TEXT
			Terminal Response DISPLAY TEXT

Note: Due to an inconsistency in [7] specification it is not possible to cover the test case when an applet try to post data in multitriggering.

6.3.1.4.4 Test Coverage

CRR Number	Test Case Number
CRRN1	14,15,16,17,18, 19
CRRN2	14,15,16,17,18, 19
CRRN3	14,15,16,17,18, 19
CRRC1	1,2,3,4,5,6,7,8,9,10,11,12,13

6.3.2 Handler Integrity

6.3.2.1 ProactiveHandler

Test Area Reference: FWK_HIN_ PAHD

6.3.2.1.1 Conformance Requirement

Normal Execution

CRRN1: At the processToolkit invocation the TLV-List is cleared.

CRRN2: After a call to ProactiveHandler.send method the handler will remain unchanged until the ProactiveHandler.init or appendTLV method are called.

6.3.2.1.2 Test Suite Files:

Test Script: FWK_HIN_ PAHD_1.scr

Test Applet: FWK_HIN_PAHD_1.java

FWK_HIN_PAHD_2.java

Load Script: FWK_HIN_PAHD_1.ldr

Cleanup Script: FWK_HIN_PAHD_1.clr

Parameter File: FWK_HIN_PAHD_1.par

6.3.2.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	At the processToolkit invocation the TLV-List is cleared		
	Applet1 and Applet2 are registered to EVENT_UNFORMATTED_SMS_PP_ENV.		
	1-An envelope containing an unformatted sms pp is sent to the SIM	1- Applet1 is triggered.	
	2-ProactiveHandler.getLength() method is called by Applet1	2- The return value is 0	
2	TLV-List change after the init method invocation		
	ProactiveHandler.init() method is called by Applet1		
	1-ProactiveHandler.getLength() method is called by Applet1	1- The return value is 9	
3	The TLV-List remains unchanged after the send() method invocation		
	1-ProactiveHandler.send() method is called by Applet1		1- The proactive command is fetched and the terminal response is issued.
	2-ProactiveHandler.getLength() method is called by Applet1	2- The return value is 9, and its contents is the same than before the calling to send method	
	It's checked that the content is the same than before the calling to send method using ProactiveHandler.copyValue and Util.arrayCompare methods		
	Applet1 finalizes		

ld	Description	API/Framework Expectation	APDU Expectation
4	At the processToolkit invocation the TLV-List		
	is cleared	1- Applet2 is triggered	
	1-ProactiveHandler.getLength() method is called by Applet2	2- The return value is 0	
	2-ProactiveHandler.getValueLength() method is called by Applet2	3- ToolkitException UNAVAILABLE_ELEMENT is thrown	

6.3.2.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3, 4
CRRN2	3

6.3.2.2 ProactiveResponseHandler

Test Area Reference: FWK_HIN_ PRHD

6.3.2.2.1 Conformance Requirement

Normal Execution

CRRN1: The ProactiveResponseHandler content is changed after the call to ProactiveHandler.send method and remains unchanged until next call to the ProactiveHandler.send method.

CRRN2: The ProactiveResponseHandler may not be available before the first call to ProactiveHandler.send method, if available the content is cleared.

6.3.2.2.2 Test Suite Files

Test Script: FWK_HIN_ PRHD_1.scr

Test Applet: FWK_HIN_PRHD_1.java

Load Script: FWK_HIN_PRHD_1.ldr

Cleanup Script: FWK_HIN_PRHD_1.clr

Parameter File: FWK_HIN_PRHD_1.par

6.3.2.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration and ProactiveResponseHandler obtaining		
	1-Applet is registered to all events defined in [7].		
	Using the methods initMenuEntry for		
	EVENT_MENU_SELECTION, requestPollInterval() for		
	<pre>EVENT_STATUS_COMMAND, allocateTimer() for EVENT_TIMER_EXPIRATION and setEventList()</pre>	1- No exception is thrown	
	for the rest of the events.	·	
	Terminal Profile command is sent to the		
	SIM without the facility of SET_EVENT_LIST ,SETUP_IDLE_MODE_TEXT, SETUP_MENU and		
	POLL_INTERVAL.	2- Applet is triggered.	
	For each event:	3- Behaviour 1:	
	2-ProactiveResponseHandler.getTheHandler()	Toolkit Exception HANDLER_NOT_AVAILABLE is	
	is called	thrown.	
		Behaviour 2:	
	T5	No exception is thrown, the return	
	<pre>If handler is available, ProactiveResponseHandler.getLength() is</pre>	value is 0	
	called		
2	The ProactiveResponseHandler remains		
	unchange after send method invocation until next send method invocation		
	1-Applet builds a proactive command	1- The ProactiveResponseHandler	2- A proactive command is
	ProactiveHandler.send() method is called	contains the terminal response	fetched
			The terminal response is sent with length 12
	2-ProactiveResponseHandler.getLength() method is called	3- The return value is 12	
	3-ProactiveHandler.init() method is called	4- No exception is thrown and the Proactive Response Handler remains unchanged	
		Tomanio unonangou	
		5- The ProactiveResponseHandler	
	4-ProactiveHandler.send() method is called	contains the terminal response of	6- A proactive command is fetched
		the second proactive command	The terminal response is sent
			with length 15
	E Duna shina Da su su su di an su su di an su di	7- The return value is 15	
	5-ProactiveResponseHandler.getLength() method is called		

6.3.2.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2
CRRN2	1

6.3.2.3 EnvelopeHandler

Test Area Reference: FWK_HIN_ ENHD

6.3.2.3.1 Conformance Requirement

Normal Execution

Parameter File:

CRRN1: The EnvelopeHandler and its content are available for all triggered toolkit applets, from the invocation to the termination of their processToolkit method.

CRRN2: The SIM Toolkit Framework guarantees that all triggered toolkit applets receive the data.

6.3.2.3.2 Test Suite Files

Test Script: FWK_HIN_ENHD_1.scr

Test Applet: FWK_HIN_ENHD_1.java

Load Script: FWK_HIN_ENHD_1.ldr

Cleanup Script: FWK_HIN_ENHD_1.clr

FWK_HIN_ENHD_1.par

6.3.2.3.3 Test Procedure

k	Description	API/Framework Expectation	APDU Expectation
	Applet initialization and Envelope Handler integrity checks with EVENT_MENU_SELECTION_HELP_REQUEST		
	1- Applet is registered to all events defined in [7] except EVENT_PROFILE_DOWNLOAD and EVENT_STATUS_COMMAND. Using the methods initMenuEntry() for EVENT_MENU_SELECTION, allocateTimer()for EVENT_TIMER_EXPIRATION, and setEventList() for the rest of the events. Perform SIM initialization with all the facilities supported	1-No exception is thrown	
	2-Envelope menu selection with help request is sent to the SIM	2- Applet is triggered	
	3-EnvelopeHandler.getTheHandler() method is called	3- No exception is thrown.	
	4-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	4- No exception is thrown	
	The EnvelopeHandler.findTLV() method is called with TAG_HELP_REQUEST		
	5-A proactive command DISPLAY TEXT is sent	6- Applet is triggered	5- 91 xx.
	6-Envelope call control by sim is sent to SIM	o- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	7- It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	7- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		

ld	Description	API/Framework Expectation	APDU Expectation
			A proactive command
			Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	Check that the TAG_HELP_REQUEST is the TLV selected	8- The contents of the envelope	
	8-The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	handler shall be the same as stored in buffer 1	
2	Envelope Handler integrity checks with EVENT_MENU_SELECTION		
	1-An envelope menu selection is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ITEM_IDENTIFIER		
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to		4- 91 XX
	SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6- It's checked the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ITEM_IDENTIFIER is the TLV selected		uio Olivi
	7- The contents of EnvelopeHandler are		
	compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
3	Envelope Handler integrity checks with EVENT_FORMATTED_SMS_PP_ENV		
	1-A formatted sms pp envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_SMS_TPDU		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_SMS_TPDU is the TLV selected		are onvi
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
4	Envelope Handler integrity checks with EVENT_UNFORMATTED_SMS_PP_ENV		
	1-A unformatted sms pp envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV method is called with TAG_DEVICE_IDENTITIES		4 04 VV
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the		

Id	Description	ADI/Eromowork Expostation	ADDII Expectation
ld	Description envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	API/Framework Expectation 6- No exception is thrown and the handler contains the envelope call control by SIM	APDU Expectation
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES	Solition by Chin	
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	
5	Envelope Handler integrity checks with EVENT_UNFORMATTED_SMS_CB		
	1-A unformatted cellbroadcast envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_CELLBROADCAST_PAGE		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent 5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called	6- No exception is thrown and the	
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_CELLBROADCAST_PAGE is the TLV selected 7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	

ld	Description	API/Framework Expectation	APDU Expectation
6	Envelope Handler integrity checks with EVENT_TIMER_EXPIRATION	·	
	1-A timer expiration envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_TIMER_ID		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_TIMER_IDE is the TLV selected		
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
7	Envelope Handler integrity checks with EVENT_CALL_CONTROL_BY_SIM		
	1-A call control envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is		

ld	Description	API/Framework Expectation	APDU Expectation
	called	,	μ
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the		
	TLV selected 7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
8	Envelope Handler integrity checks with EVENT_		
	MO_SHORT_MESSAGE_CONTROL_BY_SIM		
	1-A mo short message control by sim envelope is sent to SIM	1- Applet is triggered	
	envelope is sent to sim		
		2. No expension is thrown	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		4- 91 ^^
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
	It's checked that the TAG ADDRESS is the		The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1.	
·	<u> </u>	ı	

ld	Description	API/Framework Expectation	APDU Expectation
	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_MT_CALL	Ar // Tamework Expectation	Ar Do Expectation
	1-A event download mt call envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
	It's checked that the TAG_ADDRESS is the TLV selected		Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
10	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_CALL_CONNECTED		
	1-A event download call connected envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		4- 91 XX
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	EnvelopeHandler.getTheHandler() method is		
	called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the TLV selected		
	7- The contents of EnvelopeHandler are compared with bufferl using	7- The contents of the envelope handler shall be the same as stored in buffer 1.	
11	Util.arrayCompare() Envelope Handler integrity checks with EVENT_		
	EVENT_DOWNLOAD_CALL_DISCONNECTED		
	1-A event download call disconnected envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_ADDRESS		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_ADDRESS is the TLV selected		
	7- The contents of EnvelopeHandler are compared with bufferl using	7- The contents of the envelope handler shall be the same as stored	

ld	Description	API/Framework Expectation	APDU Expectation
	Util.arrayCompare()	in buffer 1.	
12	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_LOCATION_STATUS		
	1-A event download location status envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_LOCATION_STATUS		
	4-A proactive command DISPLAY TEXT is sent		4-91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_LOCATION_STATUS is the TLV selected		and only
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_USER_ACTIVITY	AFI/Framework Expectation	APDO Expectation
	1-A event download user activity envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	4-A proactive command DISPLAY TEXT is sent		4- 91 XX
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
14	Envelope Handler integrity checks with EVENT_ EVENT_DOWNLOAD_IDLE_SCREEN_AVAILAB LE		
	1-A event download idle screen available envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4- 91 XX
	EnvelopeHandler.getTheHandler() method is called		

ld	Description	API/Framework Expectation	APDU Expectation
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES Call Control execution is finished.	6- No exception is thrown and the handler contains the envelope call control by SIM	
	It's checked that the TAG_DEVICE_IDENTITIES is the TLV selected		Proactive command Display Text is fetched The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with buffer1 using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	
15	Envelope Handler integrity checks with EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS		
	1-A event download card reader status envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy() The EnvelopeHandler.findTLV() method is called with TAG_CARD_READER_STATUS	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent		4 04 VV
	5-Envelope call control by sim is sent to SIM	5- Applet is triggered	4- 91 XX
	EnvelopeHandler.getTheHandler() method is called 6-It's checked that the contents of the		
	envelope handler is the envelope call control using EnvelopeHandler.copy and Util.arrayCompare() methods The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES	6- No exception is thrown and the handler contains the envelope call control by SIM	
			Dragative command Diaplay
			Proactive command Display Text is fetched
	It's checked that the TAG_CARD_READER_STATUS is the TLV selected		The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are compared with bufferl using Util.arrayCompare()	7- The contents of the envelope handler shall be the same as stored in buffer 1	

ld	Description	API/Framework Expectation	APDU Expectation
16	Envelope Handler integrity checks with UNRECOGNIZED_ENVELOPE		
	1-A unrecognized envelope is sent to SIM	1- Applet is triggered	
	2-EnvelopeHandler.getTheHandler() method is called	2- No exception is thrown.	
	3-Copy the contents of the envelope handler in buffer 1 using EnvelopeHandler.copy()	3- No exception is thrown.	
	4-A proactive command DISPLAY TEXT is sent		
	5-Envelope call control by sim is sent to SIM		4- 91 XX
		5- Applet is triggered	
	EnvelopeHandler.getTheHandler() method is called The EnvelopeHandler.getValueLength() is called		
	6-It's checked that the contents of the envelope handler is the envelope call control using EnvelopeHandler.copy() and Util.arrayCompare() methods	6- No exception is thrown and the handler contains the envelope call control by SIM	
	The EnvelopeHandler.findTLV() method is called with TAG_DEVICE_IDENTITIES		
	Call Control execution is finished.		
			Proactive command Display Text is fetched
			The terminal Response of DISPLAY TEXT is sent to the SIM
	7- The contents of EnvelopeHandler are	7- The contents of the envelope	
	compared with buffer1 using Util.arrayCompare()	handler shall be the same as stored in buffer 1	

6.3.2.3.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1,2,3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	
CRRN2	1,2,3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 16	

6.3.3 Applet Triggering

6.3.3.1 EVENT_PROFILE_DOWNLOAD

Test Area Reference: FWK_APT_EPDW

6.3.3.1.1 Conformance Requirement

Normal Execution

CRRN1: Upon the reception of Terminal Profile command by the SIM, the STF stores the ME Profile and then triggers the registered toolkit applets.

CRRN2: The applet is not triggered by the EVENT_PROFILE_DOWNLOAD once it has deregistered from this event.

CRRN3: The STF shall not reply busy to a Terminal Profile command

6.3.3.1.2 Test Suite Files

Test Script: FWK_APT_EPDW_1.scr

Test Applet: FWK_APT_EPDW_1.java

FWK_APT_EPDW_2.java

FWK_APT_EPDW_3.java

Load Script: FWK_APT_EPDW_1.ldr

Cleanup Script: FWK_APT_EPDW_1.clr

Parameter File: FWK_APT_EPDW_1.par

6.3.3.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_PROFILE_DOWNLOAD and triggering		
	Applet1 is registered to the EVENT_PROFILE_DOWNLOAD		
	Applet2 is registered to the EVENT_PROFILE_DOWNLOAD		
	Applet3 is not registered to the EVENT_PROFILE_DOWNLOAD and is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-Terminal Profile command is sent to SIM	1- Applet1 is triggered	
	Applet1 execution is finished		
		2- Applet2 is triggered	
	Applet2 execution is finished		
		3- Applet3 is not triggered	
2	The STF shall not reply busy to a Terminal Profile command		
	1-Formatted sms pp envelope is sent to SIM	1- Applet3 is triggered by the EVENT_FORMATTED_SMS_PP_ENV	
	Applet3 builds a REFRESH proactive command in sim initialization mode 2-ProactiveHandler.send() method is called by applet3		2- A proactive command is
		Applet3 is suspended until the terminal response	sent
	3-Terminal Profile command is sent to SIM	3- Applet1 is triggered by EVENT_PROFILE_DOWNLOAD	
	Applet1 calls Toolkit Registry.clearEvent(EVENT_PROFILE_DOWNLOAD)		
		4- Applet2 is triggered by EVENT_PROFILE_DOWNLOAD	

ld	Description	API/Framework Expectation	APDU Expectation
	4-Applet2 calls Toolkit Registry.clearEvent(EVENT_PROFILE_DOWNLOAD) ToolkitRegistry.setEvent(EVENT_PROFILE_DOW NLOAD) method is called		The terminal Response of the proactive command is sent
	Applet3 execution finish		
3	Deregistered applets are not triggered		
		Applet3 is triggered (Applet1 and Applet2 are not triggered)	

6.3.3.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1
CRRN2	3
CRRN3	2

6.3.3.2 EVENT_MENU_SELECTION

Test Area Reference: FWK_APT_EMSE

6.3.3.2.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_MENU_SELECTION when an Envelope Menu Selection is received with the item identifier of a menu entry of this applet if no proactive session is ongoing.

6.3.3.2.2 Test Suite Files

Test Script: FWK_APT_EMSE_1.scr

Test Applet: FWK_APT_EMSE_1.java

FWK_APT_EMSE_2.java

Load Script: FWK_APT_EMSE_1.ldr

Cleanup Script: FWK_APT_EMSE_1.clr

Parameter File: FWK_APT_EMSE_1.par

6.3.3.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_MENU_SELECTION and triggering		
	ToolkitRegistry.initMenuEntry() method is called in the constructor of applet1 and Applet2.		
	For applet1: MenuEntry="Applet1" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0	1- The method must return true.	
	For applet2: MenuEntry="Applet2" Offset=0 Length=menuEntry.length HelpSupported=false IconQualifier=0 IconIdentifier=0		
	event= EVENT_MENU_SELECTION 1-ToolkitRegistry.isEventSet() is called in constructor.		
	Perform SIM initialization the facility SET UP MENU and without the facilities SET EVENT LIST and POLL INTEVAL features		
	2-Item Identifier = 1 Event Menu Selection envelope is sent to the SIM with the item identifier of a menu entry of applet	2- Applet1 is triggered and applet2 is not triggered	
	3-Item Identifier = 2 Event Menu Selection envelope is sent to the SIM with the item identifier of a menu entry of applet	3- Applet2 is triggered and applet1 is not triggered	

6.3.3.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1

6.3.3.3 EVENT_MENU_SELECTION_HELP_REQUEST

Test Area Reference: FWK_APT_EMSH

6.3.3.3.1 Conformance Requirement

Normal Execution

CRRN1: If and ENVELOPE (MENU_SELECTION_HELP_SUPPORTED) command is received for one entry supporting help, then STF shall trigger the corresponding applet.

6.3.3.3.2 Test Suite Files

Test Script: FWK_APT_EMSH_1.scr

Test Applet: FWK_APT_EMSH_1.java

 $FWK_APT_EMSH_2.java$

Load Script: FWK_APT_EMSH_1.ldr

Cleanup Script: FWK_APT_EMSH_1.clr

Parameter File: FWK_APT_EMSH_1.par

6.3.3.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to		
	EVENT_MENU_SELECTION_HELP_REQUEST		
	and triggering		
	ToolkitRegistry.InitMenuEntry() method is		
	called in the constructor of applet1 and		
	Applet2.		
	For Applet1:		
	MenuEntry="Applet1"		
	Offset=0		
	Length=menuEntry.length		
	HelpSupported=true		
	IconQualifier=0		
	IconIdentifier=0		
	7 7 1 10		
	For Applet2:		
	MenuEntry="Applet2"		
	Offset=0		
	Length=menuEntry.length HelpSupported=true		
	IconQualifier=0		
	IconIdentifier=0		
	event= EVENT_MENU_SELECTION_HELP_REQUEST		
	1-ToolkitRegistry.isEventSet() is called	1-The command must return true.	
	in constructor.		
	Perform SIM initialization the facility		
	SET UP MENU and without the facilities SET		
	EVENT LIST and POLL INTEVAL features		
	0 7 1 1 1 5 1	2 Applet4 is triggered and explot2	
	2-Item identifier =1 Menu Selection Help Request envelope is	2- Applet1 is triggered and applet2	
	sent to the SIM with the item identifier	is not triggered	
	of a menu entry of applet		
	or a mena enery or apprec		
	3-Item identifier =2		
	Menu Selection Help Request envelope is	3- Applet2 is triggered and applet1	
	sent to the SIM with the item identifier	is not triggered	
	of a menu entry of applet	332 22	
			1

6.3.3.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1

6.3.3.4 EVENT_FORMATTED_SMS_PP_ENV

Test Area Reference: FWK_APT_EFSE

6.3.3.4.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_FORMATTED_SMS_PP_ENV once it has registered to this event and Formatted Envelope DataDownLoad with the corresponding TAR, defined at the applet loading, is received and no proactive session is ongoing

CRRN2: The applet is not triggered by the EVENT_FORMATTED_SMS_PP_ENV once it has deregistered from this event.

6.3.3.4.2 Test Suite Files

Test Script: FWK_APT_EFSE_1.scr

Test Applet: FWK_APT_EFSE_1.java

Load Script: FWK_APT_EFSE_1.ldr

Cleanup Script: FWK_APT_EFSE_1.clr

Parameter File: FWK_APT_EFSE_1.par

6.3.3.4.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT FORMATTED_SMS_PP_ENV and triggering Applet is registered to EVENT_FORMATTED_SMS_PP_ENV and EVENT_UNRECOGNIZED_ENVELOPE		
	1-An Envelope EVENT_FORMATTED_SMS_PP_ENV is sent to the SIM.	1- Applet is triggered	
2	Applet deregistration		
	ToolkitRegistry.clearEvent() method is called for EVENT_FORMATTED_SMS_PP_ENV 2-A formatted sms pp envelope is sent to the SIM. An unrecognized envelope is sent to the sim	2- Applet is not triggered	
	ToolkitRegistry.setEvent() method is called for EVENT_FORMATTED_SMS_PP_ENV 3-An Envelope FORMATTED_SMS_PP_ENV is sent to the SIM	3- Applet is triggered	

6.3.3.4.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1
CRRN2	2

6.3.3.5 EVENT_UNFORMATTED_SMS_PP_ENV

Test Area Reference: FWK_APT_EUSE

6.3.3.5.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_ UNFORMATTED_SMS_PP_ENV once it has registered to this event and an Unformatted Envelope DataDownLoad is received if no proactive session is ongoing

CRRN2: The applet is not triggered by the EVENT_ UNFORMATTED_SMS_PP_ENV once it has deregistered from this event.

6.3.3.5.2 Test Suite Files

Test Script: FWK_APT_EUSE_1.scr

Test Applet: FWK_APT_EUSE_1.java

Load Script: FWK_APT_EUSE_1.ldr

Cleanup Script: FWK_APT_EUSE_1.clr

Parameter File: FWK_APT_EUSE_1.par

6.3.3.5.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_UNFORMATTED_SMS_PP_ENV and triggering		
	Applet is registered to the EVENT_UNFORMATTED_SMS_PP_ENV and ENVENT_FORMATTED_SMS_PP_ENV. 1-Toolkit Registry.isEventSet() method is		
	called for EVENT_UNFORMATTED_SMS_PP_ENV	1- The method returns true	
	2-An Envelope UNFORMATTED_SMS_PP_ENV is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Toolkit Registry.clearEvent()method is called for EVENT_UNFORMATTED_SMS_PP_ENV 1-An unformatted sms pp envelope is sent to the SIM.	1- Applet isn't triggered	
	A formatted sms pp envelope is sent to the sim		
	Toolkit Registry.setEvent() method is called for EVENT_UNFORMATTED_SMS_PP_ENV		
	2-An Envelope UNFORMATTED_SMS_PP_ENV is sent to the SIM	2- Applet is triggered	

6.3.3.5.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1
CRRN2	2

6.3.3.6 EVENT_CALL_CONTROL_BY_SIM

Test Area Reference: FWK_APT_ECCN

6.3.3.6.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_CALL_CONTROL_BY_SIM once it has registered to this event and an Envelope Call Control is received

CRRN2: The applet is not triggered by the EVENT_CALL_CONTROL_BY_SIM once it has deregistered from this event.

6.3.3.6.2 Test Suite Files

Test Script: FWK_APT_ECCN_1.scr

Test Applet: FWK_APT_ECCN_1.java

Load Script: FWK_APT_ECCN_1.ldr

Cleanup Script: FWK_APT_ECCN_1.clr

Parameter File: FWK_APT_ECCN_1.par

6.3.3.6.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_CALL_CONTROL_BY_SIM and triggering Applet1 is registered to EVENT_CALL_CONTROL_BY_SIM.		
	Applet2 is registered to EVENT_FORMATTED_SMS_PP_ENV 1-An Envelope Call control by SIM is sent to SIM Applet1 execution is finished	1- Applet1 is triggered	
2	Applet deregistration and registration of the third applet to EVENT_CALL-CONTROL_BY_SIM.		
	1-An Envelope Formatted SMS PP envelope is sent to SIM	1-Applet2 is triggered by EVENT_FORMATTED_SMS_PP_ENV.	
	Applet2 contructs a DISPLAY TEXT proactive command.		
	2-ProactiveHandler.send() method is called 3-An Envelope Call control by SIM envelope is sent to SIM		2- A proactive command DISPLAY TEXT is sent and applet is suspended until the terminal response

ld	Description	API/Framework Expectation	APDU Expectation
	ToolkitRegistry.clearEvent() is called for EVENT_CALL_CONTROL_BY_SIM. Applet1 finalizes.		TERMINAL RESPONSE of DISPLAY TEXT is sent to the SIM
	ToolkitRegistry.setEvent() method is called for EVENT_CALL_CONTROL_BY_SIM.		
	Applet2 is finished		
3	Applet triggering		
	An Envelope Call control by SIM envelope is sent ot SIM	Applet2 is triggered. (Applet1 is not triggered)	

6.3.3.6.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2
CRRN2	3

6.3.3.7 EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM

Test Area Reference: FWK_APT_EMCN

6.3.3.7.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM once it has registered to this event and an Envelope MO Short Message Control.

CRRN2: The applet is not triggered by the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM once it has deregistered from this event.

6.3.3.7.2 Test Suite Files

Test Script: FWK_APT_EMCN_1.scr

Test Applet: FWK_APT_EMCN_1.java

FWK_APT_EMCN_2.java

Load Script: FWK_APT_EMCN_1.ldr

Cleanup Script: FWK_APT_EMCN_1.clr

Parameter File: FWK_APT_EMCN_1.par

6.3.3.7.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM and triggering		
	Applet1 is reggistered to EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		
	Applet2 is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-An Envelope MO short message envelope is sent to SIM	1- Applet1 is triggered.	
	Applet1 execution is finished		
2	Applet deregistration and registration of the		
_	third applet to		
	EVENT_MO_SHORT_MESSAGE_CONTROL_B		
	Y_SIM. The STF shall not reply busy to a call control		
	envelope		
	-		
	1-An Envelope formatted SMS PP envelope is sent to SIM.	1- Applet2 is triggered.	
	Applet2 builds a DISPLAY TEXT proactive command.		2- A Proactive command
	2-ProactiveHandler.send() method is called.		DISPLAY TEXT is sent and applet is suspended until the terminal response
	3-An Envelope MO Short message envelope is sent to SIM	3- Applet1 is triggered.	
	ToolkitRegistry.clearEvent() for EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM. Applet1 finalizes.		TERMINAL DEODONOS. (
			TERMINAL RESPONSE of DISPLAY TEXT is sent to the SIM
	ToolkitRegistry.setEvent() method is called for		
	EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.		
	Applet2 execution finished.		
3	Applet3 triggering		
	An Envelope MO SMS control by SIM envelope is sent ot SIM	Applet2 is triggered. (Applet1 is not triggered)	

6.3.3.7.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2
CRRN2	3

6.3.3.8 EVENT_TIMER_EXPIRATION

Test Area Reference: FWK_APT_ETEX

6.3.3.8.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_TIMER_EXPIRATION once it has been registered to this event and an Envelope Timer Expiration with a Timer Identifier of the applet is received if no proactive session is ongoing.

CRRN2: The applet is not triggered by the EVENT_TIMER_EXPIRATION once it has been deregistered from this event.

6.3.3.8.2 Test Suite Files

Test Script: FWK_APT_ETEX_1.scr

Test Applet: FWK_APT_ETEX_1.java

Load Script: FWK_APT_ETEX_1.ldr

Cleanup Script: FWK_APT_ETEX_1.clr

Parameter File: FWK_APT_ETEX_1.par

6.3.3.8.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_TIMER_EXPIRATION and triggering Applet is registered to the EVENT_TIMER_EXPIRATION using the allocateTimer() method and to EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>event= EVENT_TIMER_EXPIRATION 1-Toolkit Registry.isEventSet() method is called.</pre>	1- The method returns true	
	2-An Envelope TIMER_EXPIRATION is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration Timer id=1 Toolkit Registry.ReleaseTimer() method is called 1-An Envelope timer expiration is sent to the SIM. An Envelope formated sms pp envelope is sent to the sim Toolkit Registry.AllocateTimer() method is called	1- Applet isn't triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	2-An Envelope TIMER_EXPIRATION is sent to the SIM.	2- Applet is triggered	

6.3.3.8.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1
CRRN2	2

6.3.3.9 EVENT_UNFORMATTED_SMS_CB

Test Area Reference: FWK_APT_EUCB

6.3.3.9.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_UNFORMATTED_SMS_CB once it has registered to this event and an Envelope Cell Broadcast DownLoad is received.

CRRN2: The applet is not triggered by the EVENT_UNFORMATTED_SMS_CB once it has deregistered from this event.

6.3.3.9.2 Test Suite Files

Test Script: FWK_APT_EUCB_1.scr

Test Applet: FWK_APT_EUCB_1.java

Load Script: FWK_APT_EUCB_1.ldr

Cleanup Script: FWK_APT_EUCB_1.clr

Parameter File: FWK_APT_EUCB_1.par

6.3.3.9.3 Test Procedure

ld	Description	API Expectation	APDU Expectation
1	Applet registration to EVENT_UNFORMATTED_SMS_CB and triggering		
	Applet is registered to the EVENT_UNFORMATTED_SMS_CB and EVENT_FORMATTED_SMS_PP_ENV.		
	<pre>event= EVENT_UNFORMATTED_SMS_CB 1-Toolkit Registry.isEventSet() method is called.</pre>	1- Method returns true.	
	2-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Toolkit Registry.ClearEvent()method is called for EVENT_UNFORMATTED_SMS_CB		
	1-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	1- Applet isn't triggered	

ld	Description	API Expectation	APDU Expectation
	An Envelope formatted sms pp envelope is sent to the sim		
	event= EVENT_UNFORMATTED_SMS_CB		
	Toolkit Registry.setEvent() method is called for EVENT_UNFORMATTED_SMS_CB		
	2-An Envelope UNFORMATTED_SMS_CB is sent to the SIM.	2- Applet is triggered	

6.3.3.9.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.10 EVENT_EVENT_DOWNLOAD_MT_CALL

Test Area Reference: FWK_APT_EDMC

6.3.3.10.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_MT_CALL once it has registered to this event and an Envelope Event DownLoad MT Call is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_MT_CALL once it has deregistered from this event.

6.3.3.10.2 Test Suite Files

Test Script: FWK_APT_EMSE_1.scr

Test Applet: FWK_APT_EMSE_1.java

Load Script: FWK_APT_EMSE_1.ldr

Cleanup Script: FWK_APT_EMSE_1.clr

Parameter File: FWK_APT_EMSE_1.par

6.3.3.10.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_MT_CALL and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_MT_CALL and to EVENT_FORMATTED_SMS_PP_ENV.		
	event= EVENT_EVENT_DOWNLOAD_MT_CALL 1-Toolkit Registry.isEventSet() method is called.	1- The method returns true	
	2-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	event= EVENT_EVENT_DOWNLOAD_MT_CALL Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim		
	event= EVENT_EVENT_DOWNLOAD_MT_CALL Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_MT_CALL is sent to the SIM.	2- Applet is triggered	

6.3.3.10.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.11 EVENT_EVENT_DOWNLOAD_CALL_CONNECTED

Test Area Reference: FWK_APT_EDCC

6.3.3.11.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED once it has registered to this event and an Envelope Event DownLoad Call Connected is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED once it has deregistered from this event.

6.3.3.11.2 Test Suite Files

Test Script: FWK_APT_EDCC_1.scr

Test Applet: FWK_APT_EDCC_1.java

Load Script: FWK_APT_EDCC_1.ldr

Clean-up Script: FWK_APT_EDCC_1.clr

6.3.3.11.3 Test Procedure

1	Applet registration to		
	EVENT_EVENT_DOWNLOAD_CALL_CONNECT ED and triggering		
	LD and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CALL_CONNECTED and to EVENT_FORMATTED_SMS_PP_ENV.		
	event= EVENT_EVENT_DOWNLOAD_CALL_CONNECTED 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_CALL_CONNECTED is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
	Applet delegistration		
	event=EVENT_EVENT_DOWNLOAD_CALL_CONNECTED Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-A call connected event dowload is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim		
	<pre>Event= EVENT_EVENT_DOWNLOAD_CALL_CONNECTED Toolkit Registry.setEvent() method is called</pre>		
	Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_CALL_CONNECTED is sent to the SIM.	2- Applet is triggered	

6.3.3.11.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.12 EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED

Test Area Reference: FWK_APT_EDCD

6.3.3.12.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED once it has registered to this event and an Envelope Event DownLoad Call Disconnected is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED once it has deregistered from this event.

6.3.3.12.2 Test Suite Files

Test Script: FWK_APT_EDCD_1.scr
Test Applet: FWK_APT_EDCD_1.java
Load Script: FWK_APT_EDCD_1.ldr
Cleanup Script: FWK_APT_EDCD_1.clr
Parameter File: FWK_APT_EDCD_1.par

6.3.3.12.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_CALL_DISCONN ECTED and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED and to EVENT_FORMATTED_SMS_PP_ENV.		
	Event=EVENT_EVENT_DOWNLOAD_CALL_DISCONNECT ED 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_CALL_DISCONNECTED is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim.		
	Event= EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED is	2- Applet is triggered	

ld	Description	API/Framework Expectation	APDU Expectation
	sent to the SIM.		

6.3.3.12.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.13 EVENT_EVENT_DOWNLOAD_LOCATION_STATUS

Test Area Reference: FWK_APT_EDLS

6.3.3.13.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS once it has registered to this event and an Envelope Event DownLoad Location Status is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS once it has deregistered from this event.

6.3.3.13.2 Test Suite Files

Test Script: FWK_APT_EDLS_1.scr

Test Applet: FWK_APT_EDLS_1.java

Load Script: FWK_APT_EDLS_1.ldr

Cleanup Script: FWK_APT_EDLS_1.clr

Parameter File: FWK_APT_EDLS_1.par

6.3.3.13.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_LOACTION_STA TUS and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_LOCATION_STATUS and to EVENT_FORMATTED_SMS_PP_ENV.		
	Event=EVENT_EVENT_DOWNLOAD_LOCATION_STATUS 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	2- Applet is triggered.	
2	Applet deregistration		
	Event=EVENT_EVENT_DOWNLOAD_LOCATION_STATUS Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported		

ld	Description	API/Framework Expectation	APDU Expectation
	1-An Envelope EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_LOCATION_STATUS Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_LOCATION_STATUS is sent to the SIM.	2- Applet is triggered	

6.3.3.13.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.14 EVENT_EVENT_DOWNLOAD_USER_ACTIVITY

Test Area Reference: FWK_APT_EDUA

6.3.3.14.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY once it has registered to this event and an Envelope Event DownLoad User Activity is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY once it has deregistered from this event.

6.3.3.14.2 Test Suite Files

Test Script: FWK_APT_EDUA_1.scr

Test Applet: FWK_APT_EDUA_1.java

Load Script: FWK_APT_EDUA_1.ldr

Cleanup Script: FWK_APT_EDUA_1.clr

Parameter File: FWK_APT_EDUA_1.par

6.3.3.14.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_USER_ACTIVITY and triggering Applet is registered to the EVENT_EVENT_DOWNLOAD_USER_ACTIVITY and to		
	EVENT_FORMATTED_SMS_PP_ENV. Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY		
	1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_USER_ACTIVITY Toolkit Registry.setEvent() method is called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_USER_ACTIVITY is sent to the SIM.	2- Applet is triggered	

6.3.3.14.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.15 EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE

Test Area Reference: FWK_APT_EDIS

6.3.3.15.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE once it has registered to this event and an Envelope Event DownLoad Idle Screen Available is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE once it has deregistered from this event.

6.3.3.15.2 Test Suite Files

Test Script: FWK_APT_EDIS_1.scr

Test Applet: FWK_APT_EDIS_1.java

Load Script: FWK_APT_EDIS_1.ldr

Cleanup Script: FWK_APT_EDIS_1.clr

FWK_APT_EDIS_1.par

6.3.3.15.3 Test Procedure

Parameter File:

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_ AVAILABLE and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE and to EVENT_FORMATTED_SMS_PP_ENV Event= EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE 1-Toolkit Registry.isEventSet() method is	1- Method retuns true	
	called. 2-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event=EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVA ILABLE		
	Toolkit Registry.clearEvent()method is called Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE		
	Toolkit Registry.setEvent() method is		
	called Perform SIM initialization with all the facilities supported		
	2-An Envelope EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE is sent to the SIM.	2- Applet is triggered	

6.3.3.15.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.16 EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS

Test Area Reference: FWK_APT_EDCR

6.3.3.16.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS once it has registered to this event and Envelope Event DownLoad Card Reader Status is received.

CRRN2: The applet is not triggered by the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS once it has deregistered from this event.

6.3.3.16.2 **Test Suite Files**

Test Script: FWK_APT_EDCR_1.scr Test Applet: FWK_APT_EDCR_1.java Load Script: FWK_APT_EDCR_1.ldr Cleanup Script: FWK_APT_EDCR_1.clr Parameter File:

FWK_APT_EDCR_1.par

6.3.3.16.3 **Test Procedure**

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_EVENT_DOWNLOAD_CARD_READER _STATUS and triggering		
	Applet is registered to the EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS and to EVENT_FORMATTED_SMS_PP_ENV		
	Event=EVENT_EVENT_DOWNLOAD_CARD_READER_STA TUS 1-Toolkit Registry.isEventSet() method is called.	1- Method returns true	
	2-An Envelope EVENT_DOWNLOAD_CARD_READER_STATUS is sent to the SIM.	2- Applet is triggered	
2	Applet deregistration		
	Event= EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS		
	Toolkit Registry.clearEvent()method is called		
	Perform SIM initialization with all the facilities supported		
	1-An Envelope EVENT_DOWNLOAD_CARD_READER_STATUS is sent to the SIM.	1- Applet isn't triggered	
	An Envelope formatted sms pp envelope is sent to the sim		
	<pre>Event= EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS</pre>		
	Toolkit Registry.setEvent() method is called		
	Perform SIM initialization with all the facilities supported		

ld	Description	API/Framework Expectation	APDU Expectation
	2-An Envelope	2- Applet is triggered	
	EVENT_DOWNLOAD_CARD_READER_STATUS is sent		
	to the SIM.		

6.3.3.16.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.17 EVENT_UNRECOGNIZED_ENVELOPE

Test Area Reference: FWK_APT_EUEV

6.3.3.17.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_UNRECOGNIZED_ENVELOPE once it has registered to this event and an Unrecognized Envelope is received.

CRRN2: The applet is not triggered by the EVENT_UNRECOGNIZED_ENVELOPE once it has deregistered from this event.

6.3.3.17.2 Test Suite Files

Test Script: FWK_APT_EUEN_1.scr

Test Applet: FWK_APT_EUEN_1.java

Load Script: FWK_APT_EUEN_1.ldr

Cleanup Script: FWK_APT_EUEN_1.clr

Parameter File: FWK_APT_EUEN_1.par

6.3.3.17.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet registration to EVENT_UNRECOGNIZED_ENVELOPE and triggering		
	Applet is registered to the EVENT_UNRECOGNIZED_ENVELOPE and to EVENT_FORMMATTED_SMS_PP_ENV Event= EVENT_UNRECOGNIZED_ENVELOPE 1-Toolkit Registry.isEventSet() method is called. 2-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	1- Method returns true2- Applet is triggered	
2	Applet deregistration		
	Event= EVENT_UNRECOGNIZED_ENVELOPE Toolkit Registry.clearEvent()method is called		
	1-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	1- Applet isn't triggered	
	a formatted sms pp envelope is sent to the sim		
	Event= EVENT_UNRECOGNIZED_ENVELOPE Toolkit Registry.setEvent() method is called		
	2-An Envelope UNRECOGNIZED_ENVELOPE is sent to the SIM.	2- Applet is triggered	

6.3.3.17.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	2	

6.3.3.18 EVENT_STATUS_COMMAND

Test Area Reference: FWK_APT_ESTC

6.3.3.18.1 Conformance Requirement

Normal Execution

CRRN1: The applet is triggered by the EVENT_STATUS_COMMAND once it has registered to this event and a Status Command is received.

CRRN2: The applet is not triggered by the EVENT_STATUS_COMMAND once it has deregistered from this event.

6.3.3.18.2 Test Suite Files

Test Script: FWK_APT_ESTC_1.scr

Test Applet: FWK_APT_ESTC_1.java

FWK_APT_ESTC_2.java

FWK_APT_ESTC_3.java

Load Script: FWK_APT_ESTC_1.ldr

Cleanup Script: FWK_APT_ESTC_1.clr

Parameter File: FWK_APT_ESTC_1.par

6.3.3.18.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applets registration to EVENT_STATUS_COMMAND and triggering		
	Applet1 is registered to EVENT_STATUS_COMMAND using the requestPollInterval() command.		
	Applet2 is registered to EVENT_STATUS_COMMAND using the RequestPollInterval() command.		
	Applet3 is registered to EVENT_FORMATTED_SMS_PP_ENV.		
	1-A status command is sent to SIM	1- Applet1 is triggered.	
	Applet1 execution is finished	2- Applet2 is triggered.	
	Applet2 execution is finished	3- Applet 3 is not triggered	
2	Applet deregistration and registration of the third applet to EVENT_STATUS_COMMAND. The STF shall not reply busy to a call control envelope		
	1-A formatted sms pp envelope is sent to SIM	1- Applet3 is triggered.	
	Applet3 builds a DISPLAY TEXT. 2- ProactiveHandler.send() is called		2- A proactive command DISPLAY TEXT is sent and applet is suspended until the terminal response
	3-A status command is sent to SIM.	3- Applet1 is triggered.	
	requestPollInteval with POLL_NO_DURATION is called		
	Applet1 finalized		
	requestPollInteval with POLL_NO_DURATION is called Applet2 finalized	4- Applet2 is triggered.	
	requestPollInterval() method is called.		
			5- TERMINAL RESPONSE of DISPLAY TEXT is sent to the SIM
	Applet3 execution finished.		

Description	API/Framework Expectation	APDU Expectation
Applet3 triggering		
Perform SIM initialization with all the facilities supported		
Status command is sent to SIM.	Applet3 is triggered. (Applet1 and Applet2 are not triggered)	
-	Applet3 triggering Perform SIM initialization with all the facilities supported	Applet3 triggering Perform SIM initialization with all the facilities supported Status command is sent to SIM. Applet3 is triggered. (Applet1 and Applet2 are not

6.3.3.18.4 Test Coverage

CR Number	Test Case Number
CRRN1	1,2
CRRN2	3

6.3.4 Proactive Command Sending by the STF

6.3.4.1 System Proactive Commands

Test Area Reference: FWK_PCS_SPCO

6.3.4.1.1 Conformance Requirements

Normal Execution

CRRN1: When a toolkit applet changes a menu entry of its registry object, the SIM Toolkit Framework shall dynamically* update the menu stored in the ME during the current card session

CRRN2: The STF shall use the data of the EFsume file when issuing the SET UP MENU proactive command.

CRRN3: For all EVENT_EVENT_DOWNLOAD_*: When a toolkit applet changes one or more of these requested events of its registry object, the STF shall dynamically* update the event list stored in the ME during the current card session by SET UP EVENT LIST proactive command.

*The STF shall send its system proactive command as soon as no proactive session is pending and all the applets registered to the current events have been triggered and have returned from the processToolkit method invocation.

6.3.4.1.2 Test Suite Files

Test Script: FWK_PCS_SPCO_1.scr

Test Applet: FWK_PCS_SPCO_1.java

Load Script: FWK_PCS_SPCO_1.ldr

Cleanup Script: FWK_PCS_SPCO_1.clr

Parameter File: FWK_PCS_SPCO_1.par

6.3.4.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
	Install Applet 1, Registered to the EVENT_EVENT_DOWNLOAD_MT_CALL and EVENT_EVENT_DOWNLOAD_LOCATION_STATUS Perform SIM initialization with EVENT DOWNLOAD facilities supported		setEventList proactive command [Event list]= '19020003' or '99020003'
2	Trigger the applet by ENVELOPE		1. DISPLAY TEXT

ld	Description	API/Framework Expectation	APDU Expectation
	(SMS_FORMATTED_PP) command Clear the events and build a display text		Proactive command
	command		2. SET UP EVENT LIST Proactive command
			[CommandQualifier]= 00h

6.3.4.1.4 Test Coverage

CRR number	Test case number
N1	see:
	chapter 6.2.9.2, CRRN1,
	chapter 6.2.9.4, CRRN3,
	chapter 6.2.9.5 CRRN4,
	chapter 6.2.9.8 CRRN1
N2	see:
	chapter 6.2.9.2 CRRN1,
	chapter 6.2.9.8 CRRN1
N3	1,2

6.3.4.2 Interaction with GSM commands

Test Area Reference: FWK_PCS_IGCO

6.3.4.2.1 Conformance Requirements

Normal Execution

CRRN1: The STF shall process a GSM command even when a proactive command is pending (before and after the FETCH command until the terminal response). The STF shall answer with the SW1 and SW2 described in [3] and [4].

6.3.4.2.2 Test Suite Files

Test Script: FWK_PCS_IGCO_1.scr

Test Applet: FWK_PCS_IGCO_1.java

Load Script: FWK_PCS_IGCO_1.ldr

Cleanup Script: FWK_PCS_IGCO_1.clr

Parameter File: FWK_PCS_IGCO_1.par

6.3.4.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Interaction with GSM Commands after		
	TERMINAL PROFILE in connection with FETCH		
	and TERMINAL RESPONSE		
	Applet is registered to Menu Selection		
	RST		
	TERMINAL PROFILE		
	(Profile: supports all facilities except:		
	SET UP EVENT LIST, POLL INTERVAL and		
	POLLING OFF)		
	1- System issues a proactive command		4 04
	SETUP_MENU		1- 91xx
	2- SELECT MF		
	3- GET RESPONSE (6 Bytes)		

ld	Description	API/Framework Expectation	APDU Expectation
	4- Failed SELECT File 5- FETCH 6- SELECT MF 7- GET RESPONSE (6 Bytes) 8- TERMINAL RESPONSE		2- 9Fxx 3- 91xx 4- 9404 5- Proactive Command: SETUP MENU 6- 9Fxx 7- 9000 8- 9000
2	Interaction with GSM Commands after ENVELOPE (MENU SELECTION) in connection with FETCH and TERMINAL RESPONSE		
	Menu Entry ID = 0x01 1- SELECT MF 2- GET RESPONSE (6 Bytes) 3- Failed SELECT File 4- FETCH 5- SELECT MF		1- 9FXX 2- 91XX 3- 9404 4- Proactive Command: DISPLAY TEXT
	6- GET RESPONSE (6 Bytes) 7- TERMINAL RESPONSE		5- 9FXX 6- 9000 7- 9000
3	Interaction with GSM Commands after TERMINAL RESPONSE in proactive command session in connection with FETCH and TERMINAL RESPONSE		
	Menu Entry ID = 0x02 1- SELECT MF 2- GET RESPONSE (6 Bytes) 3- FETCH 4- SELECT MF 5- GET RESPONSE (6 Bytes) 6- Failed SELECT File 7- TERMINAL RESPONSE		1- 9FXX 2- 91XX 3- Proactive Command: DISPLAY TEXT 4- 9FXX 5- 9000 6- 9404 7- 9000
	8- SELECT MF 9- GET RESPONSE (6 Bytes) 10-Failed SELECT File 11-FETCH 12-SELECT MF 13-GET RESPONSE (6 Bytes) 14-TERMINAL RESPONSE		8- 9FXX 9- 91XX 10-9404 11-Proactive Command: DISPLAY TEXT 12-9FXX 13-9000 14-9000

6.3.4.2.4 Test Coverage

CRR number	Test case number
N1	1,2,3

6.3.5 Exception Handling

6.3.5.1 Hide Exceptions from the ME

Test Area Reference: FWK_EXH_HEME

6.3.5.1.1 Conformance Requirements

Normal Execution

CRRN1: A toolkit applet may throw an exception, but this error will not be sent to the ME.

* Because the behaviour of the SIM is not exactly defined for the above CRRN, there are no tests defined here yet.

6.3.5.2 Interaction with Multiple Triggering

Test Area Reference: FWK_EXH_IMTG

6.3.5.2.1 Conformance Requirements

Normal Execution:

CRRN1: An exception thrown by a toolkit applet, will not influence toolkit applets registered to the same event

6.3.5.2.2 Test Suite Files

Test Script: FWK_EXH_IMTG _1.scr

Test Applet: : FWK_EXH_IMTG _1.java

Load Script: : FWK_EXH_IMTG_1.ldr

Cleanup Script: FWK_EXH_IMTG_1.clr

Parameter File: FWK_EXH_IMTG_1.par

6.3.5.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	Load/install 2 toolkit applets registered to EVENT_STATUS_COMMAND, EVENT_PROFILE_DOWNLOAD, EVENT_UNRECOGNISED_ENVELOPE, EVENT_EVENT_DOWNLOAD_MT_CALL, EVENT_UNFORMATTED_SMS_PP_ENV, EVENT_UNFORMATTED_SMS_PP_UPD, EVENT_UNFORMATTED_SMS_CB applet1: Priority= 0x01, applet2: Priority= 0x02,		
	(i.e. applet1 is triggered before applet2)		
1	Status_Command is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
2	Profile_Download is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
3	Unrecognised_Envelope is sent		

ld	Description	API/Framework Expectation	APDU Expectation
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
4	Event_Download_MT_Call is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
5	Unformatted_SMS_PP_Env is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
6	Unformatted_SMS_PP_Upd is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		
7	Unformatted_SMS_CB is sent		
	Applet 1 is triggered	Applet1: NullPointerException is thrown	
	Applet 2 is triggered		

6.3.5.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2,3,4,5,6,7

6.3.6 Framework Security Management

Security Parameters

The table that follows contains the security parameters that shall be used when the 03.48 security is required in the test cases developed in the current section.

Parameter Value in hexadecin	
KIC	11
KID	11
CNTR	00 00 00 00 01
Key for ciphering	01 41 42 7F DA E8 91 A7
Key for RC/CC/DS	01 23 45 67 89 AB CD EF

If a parameter is not listed explicitly in the above table, the default values of section 4.7.3.1 apply.

6.3.6.1 Input Data

Test Area Reference: FWK_FWS_INDA

6.3.6.1.1 Conformance Requirements

Normal Execution

CRRN1: If the SIM receives an envelope APDU containing an SMS_DATADOWNLOAD BER TLV formatted according to GSM03.48, the SIM Toolkit Framework shall verify the GSM03.48 security of the SMS TPDU.

CRRN2: The toolkit applet will only be triggered if the TAR is known and the security verified.

6.3.6.1.2 Test Area Files

Test Script: FWK_FWS_INDA_1.scr

Test Applet: FWK_FWS_INDA_1.java

FWK_FWS_INDA_2.java

Load Script: FWK_FWS_INDA_1.ldr

Cleanup Script: FWK_FWS_INDA_1.clr

Parameter File: FWK_FWS_INDA_1.par

6.3.6.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Framework checks the Cryptographic checksum and deciphers the data Applet1 is loaded and installed 1-Envelope(SMS-PP) 03.48 formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; Data = 01	The applet is triggered.	
2	Framework checks the Cryptographic checksum and deciphers the data Applet2 is installed 1-Envelope(SMS-PP) 03.48 formatted is sent to the SIM with this features: Ciphering; Cryptographic checksum; No proof of receipt; TAR of Applet 1 Data = 02 2-Envelope(SMS-PP) 03.48 formatted is sent to the SIM with this features:	This Envelope(SMS-PP) triggers Applet 1 This Envelope(SMS-PP) triggers Applet 2	The SIM answers to the Envelope with status words 9000
	No ciphering; No cryptographic checksum; No proof of receipt; TAR of Applet 2 Data = 03		The SIM answers to the Envelope with status words 9000
3	Envelope(SMS-PP) 03.48 formatted with wrong cryptographic checksum No ciphering; Wrong Cryptographic checksum; No proof of receipt;	No applet is triggered	The SIM answers to the Envelope with status words 9000

ld	Description	API/Framework Expectation	APDU Expectation
	TAR of Applet 1		
	Data = 04		

6.3.6.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2,3
CRRN2	3

6.3.6.2 Output Data

Test Area Reference: FWK_FWS_OUDA

6.3.6.2.1 Conformance Requirements

Normal Execution

CRRN1: The SIM Toolkit Framework shall secure and send the response packet.

6.3.6.2.2 Test Area Files

Test Script: FWK_FWS_OUDA_1.scr

Test Applet: FWK_FWS_OUDA_1.java

Load Script: FWK_FWS_OUDA_1.ldr

Cleanup Script: FWK_FWS_OUDA_1.clr

Parameter File: FWK_FWS_OUDA_1.par

6.3.6.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Envelope(SMS-PP) 03.48 formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt Data in plain text = "APPLET1"	The applet is triggered and sends a "Display Text" proactive command with the data received in the Envelope.	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has no application data. The SIM answers to the Get Response command with status words 91xx to issue a Display Text "APPLET1".
2	Envelope(SMS-PP) 03.48 formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt Data in plain text = "APPLET1"	The applet posts application data. It does not call the ProactiveHandler.send() method	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the application data posted by the application. The SIM answers to the Get Response command with status words 9000.
3	Envelope(SMS-PP) 03.48 formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receipt	The applet posts application data and calls the ProactiveHandler.send() method to send a "Display Text" proactive command with the data received in	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the

ld	Description	API/Framework Expectation	APDU Expectation
	Data in plain text = "TEST"	the Envelope.	application data posted by the application. The SIM answers to the Get Response command with status words 91xx to issue the Display Text "TEST".
4	Envelope(SMS-PP) 03.48 formatted Ciphering; Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; proof of receipt shall be ciphered Data in plain text = "TEST"	The applet posts application data and calls the ProactiveHandler.send() method to send a "Display Text" proactive command with the data received in the Envelope.	The SIM answers to the Envelope with status words 9Fxx and a PoR is retrieved with a GetResponse command. The PoR has the application data posted by the application. The SIM answers to the Get Response command with status words 91xx to issue the Display Text "TEST".
5	Envelope(SMS-PP) 03.48 formatted The Terminal Profile command shall be issued with the facility "'9EXX' response code for SIM data download error" enabled The Envelope(SMS-PP) formatted has to be issued with the following features: No ciphering; Wrong Cryptographic checksum; proof of receipt response shall be sent using SMS-Deliver-Report; no security applied to proof of receiptData in plain text = "TEST"	No applet is triggered	The SIM answers to the Envelope with status words 9Exx and a PoR is retrieved with a GetResponse command. The Response Status Code Octet shall be '01'.

6.3.6.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1,2,3,4,5

6.3.7 Envelope Response Posting

6.3.7.1 EVENT_CALL_CONTROL_BY_SIM

Test Area Reference: FWK_ERP_ECCN

6.3.7.1.1 Conformance Requirements

Normal Execution

CRRN1: The SIM Toolkit Framework can't reply busy when an Envelope(Call Control) is sent to the SIM.

6.3.7.1.1 Test Area Files

Test Script: FWK_ERP_ECCN_1.scr

Test Applet: FWK_ERP_ECCN_1.java

FWK_ERP_ECCN_2.java

FWK_ERP_ECCN_3.java

Load Script: FWK_ERP_ECCN_1.ldr

Cleanup Script: FWK_ERP_ECCN_1.clr

Parameter File: FWK_ERP_ECCN_1.par

6.3.7.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet 1 is registered on the EVENT_CALL_CONTROL_BY_SIM, Applet2 is registered and triggered on the EVENT_MENU_SELECTION.		
	1-Applet2 invokes the method send() and no fetch is performed 2-Envelope(Call Control) is sent to the SIM 3-Applet1 calls the method EnvelopeResponseHandler.postASBERTLV() to change any incoming dialling number into	Applet2 is suspended Applet1 is triggered.	The SIM answer 9Fxx to the
	+11 22 33 44.		Envelope(Call Control) The dialling number is retrieved with a GetResponse command. The SIM answers to the Get Response command with status words 91xx.
	4-A Fetch command is sent to the SIM 5-A Terminal Response command is sent to the SIM	Analato) a quantina aball acatinus	
	6-Delete applet1 & applet2	Applet2's execution shall continue.	
	7-Install applet3		

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet 3 is registered on both the events EVENT_CALL_CONTROL_BY_SIM and EVENT_MENU_SELECTION.		, , , , , ,
	1-Envelope Menu Selection is sent to the SIM.	Applet3 is triggered on the EVENT_MENU_SELECTION	
	2-Applet3 invokes the method send()and no fetch is performed)	Applet3 is suspended on the send() method	
	3-Envelope(Call Control) is sent to the SIM	Applet3 is triggered on the EVENT_CALL_CONTROL_BY_SI	
	4-Applet3 calls the method EnvelopeResponseHandler.postASBERTLV() to change any incoming dialling number into +11 22 33 44.	M.	The SIM answer 9Fxx to the Envelope(Call Control)
			The dialling number is retrieved with a GetResponse command.
			The SIM answers to the Get Response command with status words 91xx.
	5-A Fetch command is sent to the SIM		
	6-A Terminal Response command is sent to the SIM		
		The Applet3's execution shall continue.	

6.3.7.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1.2

6.3.7.2 EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM

Test Area Reference: FWK_ERP_EMCN

6.3.7.2.1 Conformance Requirements

Normal Execution

CRRN1: The SIM Toolkit Framework can't reply busy when an Envelope(MO-Short Message Control) is sent to the SIM.

6.3.7.2.2 Test Area Files

Test Script: FWK_ERP_EMCN_1.scr

Test Applet: FWK_ERP_EMCN_1.java

FWK_ERP_EMCN_2.java

FWK_ERP_EMCN_3.java

Load Script: FWK_ERP_EMCN_1.ldr

Cleanup Script: FWK_ERP_EMCN_1.clr

Parameter File: FWK_ERP_EMCN_1.par

6.3.7.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Applet 1 is registered on the EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM; Applet2 is registered and triggered on the EVENT_MENU_SELECTION.		
	2-Envelope(MO-SM control) is sent to the SIM	Applet2 is suspended Applet 1 is triggered.	The SIM answers 9Fxx to the Envelope(MO-Short Message Control)
	4-A Fetch command is sent to the SIM		The TP_Destination_Address is retrieved with a GetResponse command. The SIM answers to the Get Response command with status words 91xx.
	5-A Terminal Response command is sent to the SIM		
	6-Delete applet1 & applet2 7-Install applet3	The Applet's execution shall continue.	

ld	Description	API/Framework Expectation	APDU Expectation
2	Applet 3 is registered on both the events EVENT_MO_SHORT_MESSAGE_CONTROL_B Y_SIM and EVENT_MENU_SELECTION.		
	1-Applet3 invokes the method send()and no fetch is performed)	Applet 3 is suspended on the send() method	
	2-Envelope(MO-SM control) is sent to the SIM 3-Applet3 calls the method EnvelopeResponseHandler.postASBERTLV() to	Applet3 is triggered on the EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM.	
	change any incoming TP_Destination_Address and any RP_Destination_Address of the Service Center into +11 22 33 44.		The SIM answers 9Fxx to the Envelope(MO-Short Message Control)
			The TP_Destination_Address is retrieved with a GetResponse command.
			The SIM answers to the Get Response command with status words 91xx.
	4-A Fetch command is sent to the SIM		
	5-A Terminal Response command is sent to the SIM		
		The Applet3's execution shall continue.	

6.3.7.2.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1.2

6.3.7.3 EVENT_UNRECOGNIZED_ENVELOPE

Test Area Reference: FWK_ERP_EUEN

6.3.7.3.1 Conformance Requirements

Normal Execution

 $CRRN1: The\ Envelope Response Handler\ is\ available\ for\ the\ EVENT_UNRECOGNIZED_ENVELOPE.$

6.3.7.3.2 Test Area Files

Test Script: FWK_ERP_EUEN_1.scr

Test Applet: FWK_ERP_EUEN_1.java

Load Script: FWK_ERP_EUEN_1.ldr

Cleanup Script: FWK_ERP_EUEN_1.clr

Parameter File: FWK_ERP_EUEN_1.par

6.3.7.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	An applet triggered on the	The post() method returns no	The SIM answers to the
	EVENT_UNRECOGNIZED_ENVELOPE calls the	exception	Envelope with status words
	EnvelopeResponseHandler.post() method		9Fxx. The data retrieved
			with the GetResponse
			command are the ones
			posted by the applet.

6.3.7.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1

6.3.8 Toolkit Installation

6.3.8.1 Timers Allocation

Test Area Reference: FWK_TIN_TMAL

6.3.8.1.1 Conformance Requirements

Normal execution

CRRN1: One toolkit applet can register to several timers, but a timer can only be allocated to one toolkit applet.

Context errors

CRRC1 : Allocated timers shall not exceed the maximum number of timers allowed for this applet instance defined during installation.

CRRC2: The total number of timers allocated for all the applets shall not exceed 8.

6.3.8.1.2 Test suite files

Test Script: FWK_TIN_TMAL_1.scr

Test Applet: FWK_TIN_TMAL_1.java

FWK_TIN_TMAL_2.java

FWK_TIN_TMAL_3.java

Load Script: FWK_TIN_TMAL_1.ldr

Cleanup Script: FWK_TIN_TMAL_1.clr

Parameter File: FWK_TIN_TMAL_1.par

6.3.8.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	More than 8 timers at the instantiation of applet1: check that applet1 is not installed or that it is not possible to allocate more than 8 timers. Install for install of applet1 with maximum 9 timers allocated. applet1 is triggered: we allocate 9 timers applet1 is selected	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE only on the 9 th allocateTimer()	The SIM answers to the Envelope with status words 90 00 2 behaviours may be expected: 1. applet1 is not found, status word 6X XX 2. applet1 has been installed and only 8 timers
	Reset the card and delete instance of applet1		are allocated
2	Good installation of applet2 Install for install of applet2 (maximum 4 timers allocated).		The SIM answers to the Envelope with status words 90 00
3	Allocate 4 timers Applet2	No exception shall be thrown.	
4	Allocate one more timer Applet2	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE	
5	Good installation of applet3 Install for install of applet3 (maximum 8 timers allocated).		The SIM answers to the Envelope with status words 90 00
6	Allocate 4 timers Applet3	No exception shall be thrown.	
7	Allocate one more timer Applet3	Shall throw a ToolkitException with reason NO_TIMER_AVAILABLE	
8	Check that each timerId (allocated by applet2 and applet3) is between 1 and 8 and is different from each other		

6.3.8.1.4 Test Coverage

CRR number	Test case number
N1	2,3,8
C1	1, 7
C2	4,5,6

6.3.8.2 Item Identifier

Test Area Reference: FWK_TIN_ITID

6.3.8.2.1 Conformance Requirements

Normal execution

CRRN1: If the requested item identifier in the range [1-127] is not already allocated, then this item identifier shall be allocated to the current applet.

CRRN2: If the requested item identifier is '00', the card shall take the first free value in the range [128,255].

Parameters error

CRRP1: If the requested item identifier is in the range [128,255], then the card shall reject the install command.

Context errors

CRRC1: If the requested item identifier in the range [1-127] is already allocated, then the card shall reject the install command.

6.3.8.2.2 Test suite files

Test Script: FWK_TIN_ITID_1.scr

Test Applet: FWK_TIN_ITID _1.java

FWK_TIN_ITID _2.java

FWK_TIN_ITID _3.java

Load Script: FWK_TIN_ITID _1.ldr

Cleanup Script: FWK_TIN_ITID _1.clr

Parameter File: FWK_TIN_ITID _1.par

6.3.8.2.3 Test Procedure

Id	Description	API/Framework Expectation	APDU Expectation
1	Bad installation of applet1		
	Install for install of applet1. The following parameters item Id equal to 128		applet1 is not found, status word 6X XX
	applet1 is selected		
2	Good installation of applet1		
	Install for install of applet1. item Id = 1 for the first menu and 127 for the second one		The SIM answers to the Envelope with status words 91xx to send back to the ME the 2 new menus.
	A Terminal Profile is sent to the card with only PROFILE_DOWNLOAD, SMS_PP_DOWNLOAD, MENU_SELECTION,		The menus are

Id	Description	API/Framework Expectation	APDU Expectation
	SET_UP_MENU and COMMAND_RESULT facilities.		(position/itemId/text)
	facilities.		01/01/menu11
			02/127/menu12
3	Bad installation of applet2		
	Item identifier already allocated		
	Install for install of applet2.		
	item Id = 127		
	applet2 is selected		
			applet2 is not found, status
			word 6X XX
4	Good installation of applet2		
	Install for install of applet2.		The SIM answers to the Envelope with status words
	item Id = 0		91xx to send back to the
			ME the 3 menus.
			The menus are
			01/01/menu11
			02/127/menu12
			03/128/menu21
5	Good installation of applet3		
	Install for install of applet3.		The SIM answers to the Envelope with status words
	item Id = 0		91xx to send back to the
			ME the 4 menus.
			The menus are
			01/01/menu11
			02/127/menu12
			03/128/menu21
			04/129/menu31
6	Good delete and installation of applet2		
			The SIM answers to the

Id	Description	API/Framework Expectation	APDU Expectation
	Delete instance of applet2		Terminal Profile with status words 91xx to send back to the ME the 3 menus.
	Perform a RESET and a Terminal Profile with the facilities of PROFILE_DOWNLOAD, SMS-PP_DATA_DOWNLOAD, MENU_SELECTION, COMMAND_RESULT and SET_UP_MENU		
			The menus are
			01/01/menu11
			02/127/menu12
			03/129/menu31
	Install for install of applet2.		
	item Id = 0		The SIM answers to the Envelope with status words 91xx to send back to the ME the 4 menus.
			The menus are
			01/01/menu11
			02/127/menu12
			03/128/menu21
			04/129/menu31
			0 1/12// Heliu 31

6.3.8.2.4 Test Coverage

CRR number	Test case number
N1	2

CRR number	Test case number
N2	4,5,6
P1	1
C1	3

6.3.8.3 Item Position

Test Area Reference: FWK_TIN_ITPO

6.3.8.3.1 Conformance Requirements

Normal execution

CRRN1: The position of the new menu entries is an absolute position among the existing ones.

CRRN2: If the position identifier is 00h, the menu shall have the last position

6.3.8.3.2 Test suite files

Test Script: FWK_TIN_ITPO_1.scr

Test Applet: FWK_TIN_ ITPO _1.java

FWK_TIN_ ITPO _2.java

FWK_TIN_ ITPO _3.java

Load Script: FWK_TIN_ ITPO _1.ldr

Cleanup Script: FWK_TIN_ ITPO _1.clr

Parameter File: FWK_TIN_ ITPO _1.par

6.3.8.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Installation of applet1		
	Perform Install for install of applet1.Position/ItemId 01/01 02/02		The menus are
	A Terminal Profile is sent to the card		(position/itemId/text) 01/01/menu11 02/02/menu12
2	Installation of applet2 Perform Install for install of applet2. Position/ItemId 03/03 04/04		The SIM answers to the Envelope with status words 91xx to send back to the ME the 4 menus. The menus are (position/itemId/text) 01/01/menu11 02/02/menu12 03/03/menu21 04/04/menu22
3	Installation of applet3 Perform Install for install of applet3. Position/ItemId 00/05		The SIM answers to the Envelope with status words 91xx to send back to the ME the 5 menus. The menus are (position/itemId/text) 01/01/menu11 02/02/menu12 03/03/menu21 04/04/menu22 05/05/menu31

6.3.8.3.4 Test Coverage

Note: As Item Position management is not fully specified in the [7] or [8] all possible tests cannot be performed.

CRR number	Test case number
N1	1,2
N2	3

6.3.8.4 Maximum Text Length for a menu entry

Test Area Reference: FWK_TIN_MLME

6.3.8.4.1 Conformance Requirements

Normal execution

CRRN1: The maximum length of item text string is defined at the installation of the toolkit applet.

Parameters errors

CRRP1: If initMenuEntry length parameter is greater than the allocated space (Maximum Text Length for a menu entry), then a ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown.

CRRP2: If changeMenuEntry length parameter is greater than the allocated space (Maximum Text Length for a menu entry), then a ToolkitException ALLOWED_LENGTH_EXCEEDED is thrown.

6.3.8.4.2 Test suite files

Test Script: FWK_TIN_MLME_1.scr

Test Applet: FWK_TIN_MLME_1.java

Load Script: FWK_TIN_MLME_1.ldr

Cleanup Script: FWK_TIN_MLME_1.clr

Parameter File: FWK_TIN_MLME_1.par

6.3.8.4.3 Test Procedure

ld	Description	API / Framework Expectation	APDU Expectation
1	Installation of applet with 2 menus not		
	exceeding the maximum text length		
	Install one applet with 2 menu entries		
	allowed and max. text length equal to 10.		
	initMenuEntry defined at the install		
	(install) command		
	MenuEntry = "MenuEntry1", "MenuEntry2"		
	Offset = 0		
	Length = 10		
	NextAction = '00'		
	HelpSupported = false		
	<pre>IconQualifier = '00'</pre>		
_	IconIdentifier = 0		
2	initMenuEntry with a too large length	ToolkitException	
		ALLOWED_LENGTH_EXCEEDED	
	initMenuEntry with length equal to 11	is thrown	
	MenuEntry = " MenuEntry03"		
	Offset = 0		
	Length = 11		
	NextAction = '00'		
	HelpSupported = false		
	IconQualifier = '00'		
	<pre>IconIdentifier = 0</pre>		
3	in is Name. Control with a windst language		- OFT LID MENUL (O Harran)
3	initMenuEntry with a right length		a SET UP MENU (2 items)
			is issued with TLV item
	initMenuEntry with length parameter equal		length equal to 11 (Identifier
	to 10		+ Text string of item)
	MenuEntry = " MenuEntry3" Offset = 0		
	Length = 10		
	NextAction = '00'		
	HelpSupported = false		
	IconQualifier = '00'		
	IconIdentifier = 0		
4	changeMenuEntry with a right length		a SET UP MENU (2 items)
	ondingoniona Entry With a right forigati		is issued with TLV item
	Applet1 is triggered by a		length equal to 11 (Identifier
	EVENT MENU SELECTION.		+ Text string of item)
	changeMenuEntry of menu 1, with length		+ rext string or item)
	parameter equal to 10		
	Id = '01'		
	MenuEntry = "MenuEntry4"		
	Offset = 0		
	Length = menuEntry.length		
	NextAction = 0		
	HelpSupported = false		
	<pre>IconQualifier = 0</pre>		
	<pre>IconIdentifier = 0</pre>		
	Dotumn from progoggeo-11-4-		
5	Return from processToolkit	ToolkitEvaantian	Chall not receive a CET UD
5	changeMenuEntry with a too large length	ToolkitException	Shall not receive a SET UP
	3	·== • · · · == === · · · · · ·== · · · ·	MENU different from the
	Applet1 is triggered by a	is thrown	previous one
	EVENT_MENU_SELECTION.		

ld	Description	API / Framework Expectation	APDU Expectation
	ChangeMenuEntry of menu 1, with length		
	parameter equal to 11		
	Id = '02'		
	MenuEntry = "MenuEntry05"		
	Offset = 0		
	Length = menuEntry.length		
	NextAction = 0		
	HelpSupported = false		
	<pre>IconQualifier = 0</pre>		
	<pre>IconIdentifier = 0</pre>		
	Return from processToolkit		

6.3.8.4.4 Test Coverage

CRR number	Test case number
CRRN1	1, 3, 4
CRRP1	2
CRRP2	5

6.3.8.5 Maximum number of menu entries

Test Area Reference: FWK_TIN_NBME

6.3.8.5.1 Conformance Requirements

Normal execution

CRRN1: The maximum number of menu entries is defined at the installation of the toolkit applet and can be the maximum number of successful invocations of the method initMenuEntry .

Parameters errors

CRRP1: If the menu entry cannot be initialised (e.g. no more item data in applet loading parameter), a ToolkitException with the REGISTRY_ERROR reason code is thrown.

6.3.8.5.2 Test suite files

Test Script: FWK_TIN_NBME_1.scr

Test Applet: FWK_TIN_NBME_1.java

FWK_TIN_NBME_2.java

Load Script: FWK_TIN_NBME_1.ldr

Cleanup Script: FWK_TIN_NBME_1.clr

Parameter File: FWK_TIN_NBME_1.par

6.3.8.5.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Installation of applet with 3 menus	No Exception is thrown	
	Install (install) applet with max. number		
	of menu entry is `3', defined at the		
	install (install) command.		
	initMenuEntry for each menu entry allowed		
	(3 times)		
	MenuEntry = "menu1", "menu2", "menu3"		
	Offset = 0		
	Length = 5		

ld	Description	API/Framework Expectation	APDU Expectation
	NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0		
2	<pre>init of a 4th menu initMenuEntry one more time MenuEntry = "menu4" Offset = 0 Length = 5 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>	ToolkitException REGISTRY_ERROR is thrown	SET UP MENU (3 items) is issued with TLV item length equal to 6 (Identifier + Text string of item)
3	<pre>Installation of 2nd applet with 0 menu Install (install) another applet, with max. number of menu entry is '0', defined at the install (install) command. initMenuEntry once MenuEntry = "menul" Offset = 0 Length = 5 NextAction = '00' HelpSupported = false IconQualifier = '00' IconIdentifier = 0</pre>	ToolkitException REGISTRY_ERROR is thrown	Shall not receive a SET UP MENU different from the previous one

6.3.8.5.4 Test Coverage

CRR number	Test case number
CRRN1	1
CRRP1	2, 3

6.3.8.6 Access Domain

Test Area Reference: FWK_TIN_ACDO

6.3.8.6.1 Conformance Requirements

Normal execution

CRRN1: The Access Domain parameter indicates the mechanism used to control the applet instance access to the GSM file System ('00' means full access to the GSM File System, 'FF' means no access to the GSM File System).

Parameters errors

CRRP1: If the Access Domain Parameter requested is not supported, the card shall return the Status Word '6A80', incorrect parameters in data field, to the Install(Install) command.

CRRP2: If an applet with Access Domain Parameter 'FF' (i.e. No Access to the GSM File System) tries to access a GSM file (e.g. invoke the updateBinary(..) method) the framework shall throw a SIMViewException with a AC_NOT_FULFILLED reason.

6.3.8.6.2 Test suite files

Test Script: FWK_TIN_ACDO_1.scr

Test Applet: FWK_TIN_ACDO_1.java

FWK_TIN_ACDO_2.java

FWK_TIN_ACDO_3.java

Load Script: FWK_TIN_ACDO_1.ldr

Cleanup Script: FWK_TIN_ACDO_1.clr
Parameter File: FWK_TIN_ACDO_1.par

6.3.8.6.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	Install (install) applet1 with:	-	-
	- Length of Access Domain field value is		
	- Access Domain Parameter value is '00' (full access to the GSM File System)		
	Install (install) applet2 with: - Length of Access Domain field value is		
	'1' - Access Domain Parameter value is 'FF' (No access to the GSM File System)		
	Install (install) applet3 with: - Length of Access Domain field value is		
	- Access Domain Parameter value is '00' (full access to the GSM File System)		
1	readBinary/readRecord method with full Access Domain Parameter	1 to 4- no exception is thrown	
	<pre>1- Select EF-TARU file whose Read access condition is ALWAYS Perform the readBinary method: fileOffset = 0 resp = abRead[] respOffset = 0 respLength = 3</pre>	5- SIMViewException AC_NOT_FULFILLED is thrown	
	2- Select EF-SMS file whose Read access condition is CHVI Perform the readRecord method:		
	recNumber = 1 mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0 resp = abRead[] respOffset = 0 respLength = 3		
	3- Select EF-TRAC file whose Read access condition is CHV2 Perform the readBinary method: fileOffset = 0 resp = abRead[] respOffset = 0 respLength = 3		
	4- Select EF-SUME file Read access condition is ADM0 Perform the readBinary method: fileOffset = 0 resp = abRead[] respOffset = 0 respLength = 3		
	5- Select EF-TNR file whose Read access condition is NEVER Perform the readBinary method: fileOffset = 0 resp = abRead[] respOffset = 0 respLength = 3		
2	updateBinary/updateRecord method with full Access Domain Parameter	1 to 4- no exception is thrown	
Ь			

ld	Description	API/Framework Expectation	APDU Expectation
	For each case, send an Envelope that	5- SIMViewException	
	triggers the applet with the	AC_NOT_FULFILLED is thrown	
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	1- Select EF-TNR file whose Update access		
	condition is ALWAYS		
	Perform the updateBinary method:		
	fileOffset = 0		
	resp = abUpdate[FFFFFF] respOffset = 0		
	respLength = 3		
	2- Select EF-SMS file whose Update access		
	condition is CHV1		
	Perform the updateRecord method: recNumber = 1		
	mode = REC_ACC_MODE_ABSOLUTE_CURRENT		
	recOffset = Oresp = abUpdate[]		
	respOffset = 0		
	respLength = 3		
	3- Select EF-FDN file whose Update access		
	condition is CHV2		
	Perform the updateBinary method:		
1	recNumber = 1		
	<pre>mode = REC_ACC_MODE_ABSOLUTE_CURRENT recOffset = 0</pre>		
	resp = abUpdate[]		
1	respOffset = 0		
	respLength = 3		
	4_ Coloat EF_CIME file medata acces		
	4- Select EF-SUME file Update access condition is ADM0		
	Perform the updateBinary method:		
	fileOffset = 0		
	resp = abUpdate[]		
1	respOffset = 0 respLength = 3		
1			
1	5- Select EF-TNU file whose Update access		
	condition is NEVER		
	Perform the updateBinary method: fileOffset = 0		
	resp = abUpdate[]		
1	respOffset = 0		
	respLength = 3		
3	invalidate method with full Access Domain	1 to 4- no exception is thrown	
	Parameter	1 to 1 no exception is unlown	
		5- SIMViewException	
	1- Select EF-TNR file whose Invalidate	AC_NOT_FULFILLED is thrown	
	access condition is ALWAYS Perform the invalidate method		
	2- Select EF-TIAC file whose Invalidate		
	access condition is CHV1		
	Perform the invalidate method		
1	3- Select EF-ADN file whose Invalidate		
	access condition is CHV2		
	Perform the invalidate method		
1	4- Select EF-SUME file Invalidate access		
	condition is ADM0		
1	Perform the invalidate method		
	5- Select EF-CNIV file whose Invalidate access condition is NEVER		
	Perform the invalidate method		
L			
4	rehabilitate method with full Access Domain	1 to 4- no exception is thrown	
	Parameter	-	
	1- Select EF-TNR file whose Rehabilitate	5- SIMViewException	
	1- Select EF-INR life whose Rehabilitate	AC_NOT_FULFILLED is thrown	

ld	Description	API/Framework Expectation	APDU Expectation
	access condition is ALWAYS		•
	Perform the rehabilitate method		
	2- Select EF-IMSI file whose Rehabilitate access condition is CHV1		
	Perform the rehabilitate method		
	3- Select EF-ADN file whose Rehabilitate		
	access condition is CHV2		
	Perform the rehabilitate method		
	4- Select EF-SUME file Rehabilitate access		
	condition is ADMO Perform the rehabilitate method		
	5- Select EF-CNRI file whose Rehabilitate		
	access condition is NEVER		
	Perform the rehabilitate method		
5	increase method with full Access Domain	1 to 4- no exception is thrown	
	Parameter		
		5- SIMViewException	
	1- Select EF-CNU file whose Increase access condition is ALWAYS	AC_NOT_FULFILLED is thrown	
	Perform the increase method:		
	<pre>incr = abIncreaseValue[] incrOffset = 0</pre>		
	resp = abRead[]		
	respOffset = 0		
	2- Select EF-ACM file whose Increase		
	access condition is CHV1 Perform the increase method:		
	incr = abIncreaseValue[]		
	<pre>incrOffset = 0 resp = abRead[]</pre>		
	respOffset = 0		
	3- Select EF-CIAC file whose Increase		
	access condition is CHV2		
	Perform the increase method: incr = abIncreaseValue[]		
	incrOffset = 0		
	resp = abRead[] respOffset = 0		
	4- Select EF-CIAA file Increase access condition is ADM0		
	Perform the increase method:		
	<pre>incr = abIncreaseValue[] incrOffset = 0</pre>		
	resp = abRead[]		
	respOffset = 0		
	5- Select EF-CNR file whose Increase		
	access condition is NEVER Perform the increase method		
	Applet1 finalizes		
6		SIMViewException	
	Parameter	AC_NOT_FULFILLED is thrown	
	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Select EF-TARU file whose Read access condition is ALWAYS		
	Perform the readBinary method:		
	fileOffset = 0 resp = abRead[]		
	respOffset = 0		
	respLength = 3 t		
	updateRecord method with no Access Domain	SIMViewException	

		API/Framework Expectation	APDU Expectation
	Parameter	AC_NOT_FULFILLED is thrown	
S	Send an Envelope that triggers the applet		
	with the EVENT_UNFORMATTED_SMS_PP_ENV		
	event.		
g	Select EF-SMS file whose Update access		
	condition is CHV1		
	Perform the updateRecord method:		
	<pre>fileOffset = 0 resp = abUpdate[]</pre>		
	respOffset = 0		
	respLength = 3		
8	invalidate method with no Access Domain Parameter	SIMViewException AC_NOT_FULFILLED is thrown	
S	Send an Envelope that triggers the applet		
w	rith the EVENT_UNFORMATTED_SMS_PP_ENV		
e ⁻	event.		
S	Select EF-ADN file whose Invalidate access		
C	condition is CHV2		
P	Perform the invalidate method		
9	rehabilitate method with no Access Domain	SIMViewException	
		AC_NOT_FULFILLED is thrown	
	Send an Envelope that triggers the applet with the EVENT_UNFORMATTED_SMS_PP_ENV		
	event.		
	Select EF-SUME file Rehabilitate access		
	Perform the rehabilitate method		
10	increase method with no Access Domain	SIMViewException	
	Parameter	AC_NOT_FULFILLED is thrown	
S	Send an Envelope that triggers the applet		
w	rith the EVENT_UNFORMATTED_SMS_PP_ENV		
e ⁻	event.		
S	Select EF-CNR file whose Increase access		
C	condition is NEVER		
P	Perform the increase method		
A;	applet2 finalizes		
A	applet3 restore EF-SUME		

6.3.8.6.4 Test Coverage

Note: As Item Position management is not fully specified in the [7] or [8] all possible tests cannot be performed.

CRR number	Test case number	
CRRN1	1, 2, 3, 4, 5	
CRRP1	Not tested	
CRRP2	6, 7, 8, 9, 10	

6.3.8.7 Priority Level

Test Area Reference: FWK_TIN_PRLV

6.3.8.7.1 Conformance Requirements

Normal execution

CRRN1: The priority specifies the order of activation of an applet compared to the other applet registered to the same event ('01': Highest priority level, 'FF': Lowest priority level)

CRRN2: If two or more applets are registered to the same event and have the same priority level, the applets are activated according to their installation date (i.e. the most recent applet is activated first)

6.3.8.7.2 Test suite files

Test Script: FWK_TIN_PRLV_x.scr, x from 1 to 12

Test Applet: FWK_TIN_PRLV_x.java, x from 1 to 12, 8A, 8B, 9A, 9B, 10A, 10B

Load Script: FWK_TIN_PRLV_x.ldr, x from 1 to 12

Cleanup Script: FWK_TIN_PRLV_x.clr, x from 1 to 12

Parameter File: FWK_TIN_PRLV_x.par, x from 1 to 12, , 8A, 8B, 9A, 9B, 10A, 10B

6.3.8.7.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	All applets are registered on an		
	EVENT_UNFORMATTED_SMS_PP_ENV event		
1	Trigger 2 applets with 2 different maximum		
	Priority Levels		
	Turk 33 (in the 33)		
	<pre>Install (install) applet1 with priority level '2' and applet2 with priority level</pre>		
	'1', from package fwk_tin_prlv_1.		
	i / iiom pachage iwi_ciii_piiv_i.		
	Send an Envelope that triggers the 2		
	applets with the	A static variable is used to validate	
	EVENT_UNFORMATTED_SMS_PP_ENV event.	triggering order: applet2 is	
		triggered before applet1	
	Delete applets instances and packages		
2	Trigger 2 applets with 2 different maximum		
	Priority Levels		
	Install (install) applet1 with priority		
	level '1' and applet2 with priority level		
	'2', from package fwk tin prlv 2.		
	, , , , , , , , , , , , , , , , , , , ,		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.	A static vericle is used to velidate	
		A static variable is used to validate	
	Delete applets instances and packages	triggering order: applet1 is	
3		triggered before applet2.	
3	Trigger 2 applets with 2 different Priority Levels		
	Levels		
	Install (install) applet1 with priority		
	level '80' and applet2 with priority level		
	'7F', from package fwk_tin_prlv_3.		
	Cond on Broad on that it is a second		
	Send an Envelope that triggers the 2 applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	BVBN1_ONI ONNATIED_OND_II_BNV CVCIIC.		
	Delete applets instances and packages	A static variable is used to validate	
		triggering order: applet2 is	
		triggered before applet1	

4	Trigger 2 applets with 2 different Priority	T I	
4	Levels		
	Install (install) applet1 with priority level '7F' and applet2 with priority level '80', from package fwk_tin_prlv_4.		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A static variable is used to validate triggering order: applet2 is triggered before applet1	
5	Delete applets instances and packages Trigger 3 applets with the same Priority Level	33	
	<pre>Install (install) applet 1, 2, 3 in this order with same priority level from package fwk_tin_prlv_5.</pre>		
	Send an Envelope that triggers the 3 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages.	A static variable is used to validate triggering order: applet3 is triggered before applet2, and applet2 is triggered before applet1.	
6	Trigger 2 applets from 2 classes, with 2 different Priority Level		
	Install (install) applet1 from class A with priority level '2' Install (install) applet2 from class B		
	with priority level `1'		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances and packages	A static variable is used to validate triggering order: applet2 is triggered before applet1	
7	Trigger 2 applets from 2 classes, with the same Priority Level		
	Install (install) applet1 from class A with priority level `1'		
	<pre>Install (install) applet2 from class B with priority level `1'</pre>		
	Send an Envelope that triggers the 2 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event. Delete applets instances and packages		
	before appreed instances and passages	A static variable is used to validate triggering order: applet2 is triggered before applet1	
8	Trigger 2 applets from 2 packages, with 2 different Priority Level		
	<pre>Install package fwk_tin_prlv_8. Install (install) applet1 from package fwk_tin_prlv_8A with priority level `2'</pre>		
	Install (install) applet2 from package fwk_tin_prlv_8B with priority level '1'		

_		,	
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applets instances ad packages	A static variable is used to validate	
	Delete appreces impediates an paomages	triggering order: applet2 is	
		triggered before applet1	
9	Trigger 2 applets from 2 packages, with the		
	same Priority Level		
	-		
	Install package fwk_tin_prlv_9.		
	Install (install) applets 1 from package		
	fwk_tin_prlv_9A and applet2 from package		
	<pre>fwk_tin_prlv_9B in this order, with same priority level</pre>		
	priority level		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A static variable is used to validate	
		triggering order: applet2 is	
	Delete applets instances and packages	triggered before applet1	
10	Trigger 4 applets from 2 packages		
	1-Install packages fwk_tin_prlv_10,		
	fwk_tin_prlv_10A and fwk_tin_prlv_10B.		
	Install (install) 2 applets 1 then 2 from		
	package fwk_tin_prlv_10A, with		
	respectively priority levels 1 and 2.		
	Send an Envelope that triggers the 2		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	2- Install (install) 2 applets 3 then 4		
	<pre>from package fwk_tin_prlv_10B, with respectively priority levels 1 and 2.</pre>		
	respectively priority levels I and 2.		
	Send an Envelope that triggers the 4	1- A static variable is used to	
	applets.	validate triggering order: applet1 is	
		triggered before applet2	
	Delete applets instances and packages		
		2- Applet3 is triggered before	
		applets 1, 4, then 2.	
11	Trigger 4 applets with the same Priority Level		
	then delete them one after another and trigger		
	them each time		
	1		
	1- Install (install) applet1, 2, 3, 4 in this order with same priority level from		
	package fwk_tin_prlv_11.		
	Send an Enveloppe that triggers the 4		
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
		A statistical transfer	
	Delete applet instance 4	1- A static variable is used to	
	Detect applee instance i	validate triggering order: applets	
	2- Send an Enveloppe that triggers the 3	are triggered in order 4, 3, 2, 1.	
	applets with the		
	EVENT_UNFORMATTED_SMS_PP_ENV event.		
	Delete applet instance 3		
	The state of the s		
	3- Send an Enveloppe that triggers the 2	2- Applets are triggered in order 3,	
	applets with the	2. Applets are triggered in order 3, 2, 1.	
	EVENT_UNFORMATTED_SMS_PP_ENV event.	۷, ۱.	

_		T	
	Delete remaining applet instances and packages		
		3- Applets are triggered in order 2, 1.	
12	Trigger 5 applets with different Priority Levels, alternating install and delete		
	1- Install (install) applets 1, 2, 3, 4 in this order with respective priority levels 1, 2, 1, 2		
	Send an Enveloppe that triggers the 4 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	2- Delete applet instance 1 and install (install) applet5 with priority level 2	1- A static variable is used to validate triggering order: applets are triggered in order 3, 1, 4, 2	
	Send an Enveloppe that triggers the 4 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
	3- Re-install (install) applet1 with priority level 1	2- Applets are triggered in order 3, 5, 4, 2	
	Send an Enveloppe that triggers the 5 applets with the EVENT_UNFORMATTED_SMS_PP_ENV event.		
		3- Applets are triggered in order 1, 3, 5, 4, 2	

6.3.8.7.4 Test Coverage

CRR number	Test case number	
CRRN1	1, 2, 3, 4, 6, 8, 10, 12	
CRRN2	5, 7, 9, 11	

6.3.9 File System Context

6.3.9.1 Initial Context

Test Area Reference: FWK_FSC_INIT

6.3.9.1.1 Conformance Requirements

Normal Execution

CRRN1: At the invocation of the processToolkit method of a toolkit applet, the current file is the MF.

6.3.9.1.2 Test Suite Files

Test Script: FWK_FSC_INIT_1.scr

Test Applet: FWK_FSC_INIT_1.java

FWK_FSC_INIT_2.java

Load Script: FWK_FSC_INIT_1.ldr

Cleanup Script: FWK_FSC_INIT_1.clr

FWK_FSC_INIT_2.clr

Parameter File: FWK_FSC_INIT_1.par

6.3.9.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	<pre>MF is the selected DF in processToolkit() An ENVELOPE APDU containing a formatted SMS PP for Applet 1 is issued to the SIM byte[] fci = new byte[10] fciOffset = 0 fciLength = 7 status()</pre>	No exception shall be thrown. Shall return 7. fci shall contain the following part of the FCI structure: < XX XX XX XX 3F 00 01 >	
2	No EF is selected rehabilitate ()	SIMView exception shall be thrown with reason NO_EF_SELECTED	
3	MF is selected even when an applet triggered before selected any other file Applets 1 and 2 register to EVENT_DOWNLOAD_USER_ACTIVITY. Applet 1 has higher priority than Applet 2. An ENVELOPE "EVENT - USER ACTIVITY" is sent to the SIM 1 - Applet 1: - is triggered by event_event_download_user_activity - selects DF_GSM and EF_IMSI 2 - Applet 2: - is triggered by event_event_download_user_activity fciOffset = 0 fciLength = 7 status() 3 - rehabilitate ()	1 - No exception shall be thrown.2 - No exception shall be thrown.Shall return 7.	

6.3.9.1.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1, 2, 3

6.3.9.2 Context Preservation (current file)

Test Area Reference: FWK_FSC_CUFI

6.3.9.2.1 Conformance Requirements

Normal execution

CRRN1: When calling the method select (), the current files (file context) of any other applets shall not be changed (see GSM 03.19 [] - §5.2).

CRRN2: The select() methods select a file without changing the current file of any other applet or of the subscriber session.

CRRN3: After invocation of ProactiveHandler.send() method: the current file context of the toolkit applet is unchanged (see GSM 03.19 [] - §5.2.).

6.3.9.2.2 Test Suite Files

Test Script: FWK_FSC_CUFI_1.scr

Test Applet: FWK_FSC_CUFI_1.java

FWK_FSC_CUFI_2.java

Load Script: FWK_FSC_CUFI_1.ldr

Cleanup Script: FWK_FSC_CUFI_1.clr

 $FWK_FSC_CUFI_2.clr$

Parameter File: FWK_FSC_CUFI_1.par

6.3.9.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	No change to file context by another applet Applet1 registers to EVENT_FORMATTED_SMS_PP_ENV. Applet2 registers to EVENT_CALL_CONTROL_BY_SIM	1 - No exception shall be thrown. 2 - No exception shall be thrown. 3 - No exception shall be thrown. The value of buffer2 is { 0xCA, 0xFE }	A GET INKEY proactive command is fetched from the SIM
	<pre>1 - Applet 1: - is triggered by a formatted SMS - selects DF_SIMTEST and EF_TARU - fileOffset = 0; dataLength = 2; dataOffset = 0; - buffer = {0xCA, 0xFE } - updateBinary (): first 2 bytes of EF_TARU are written as 'CA FE' issues a proactive command "Get Inkey".</pre>	OAI L }	
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL BY SIM - selects DF_TELECOM and EF_ADN.		
	<pre>3 - The terminal response for Get Inkey reactivates Applet 1: - fileOffset = 0; respLength = 2; respOffset = 0; - readBinary () info buffer2</pre>		
2	No change to file context by subscriber session 1 - Applet 1 - issues a proactive command "Get Inkey".	1 - No exception shall be thrown. 3 - No exception shall be thrown. The value of buffer2 is { 0xCA, 0xFE }	1 - A GET INKEY proactive command is fetched from the SIM
	2 - Subscriber session selects DF_TELECOM and EF_ADN.		
	<pre>3 - The terminal response for Get Inkey reactivates Applet 1: - fileOffset = 0; respLength = 2; respOffset = 0; - readBinary () info buffer2</pre>		
3	No change by applet of subscriber session context 1 - Applet 1: - selects DF_SIMTEST and EF_TNU	No exception shall be thrown. No exception shall be thrown.	1 - A GET INKEY proactive command is fetched from the SIM
	- issues a proactive command "Get Inkey". 2 - subscriber session reads record 1 of current file (shall be EF_ADN) 3 - The terminal response for Get Inkey		2 - READ RECORD absolute number 1 shall read "FF FF FF FF FF FF FF FF FF FF FF FF FF FF

ld	Description	API/Framework Expectation	APDU Expectation
	reactivates Applet 1, which terminates execution		FF FF FF FF FF" (from
	CACCUCION		EF _{ADN})

6.3.9.2.4 Test Coverage

CRR Number	Test Case Number	
CRRN1	1	
CRRN2	1,2,3	
CRRN3	1,2	

6.3.9.3 Context Preservation (current record pointer)

Test Area Reference: FWK_FSC_CURE

6.3.9.3.1 Conformance Requirements

Normal execution

CRRN1: When the seek method is called by one applet, the record pointer of any other applet is not changed.

CRRN2: *updateRecord*: the current record pointer of other applets / subscriber shall not be changed in case of linear fixed EF

CRRN3: *updateRecord*: the record pointer of a cyclic EF shall be changed for all other applets / subscriber to the record number 1.

CRRN4: *readRecord*: read data bytes of the linear fixed or cyclic EF currently selected by the applet without changing the current record pointer of any other applet / subscriber.

CRRN5: *increase*: the last updated record of the cyclic EF currently selected becomes record number 1 for every other applet and subscriber session.

6.3.9.3.2 Test Suite Files

Test Script: FWK_FSC_CURE_1.scr

Test Applet: FWK_FSC_CURE_1.java

FWK_FSC_CURE_2.java

Load Script: FWK_FSC_CURE_1.ldr

Cleanup Script: FWK_FSC_CURE_1.clr

FWK_FSC_CURE_2.clr

Parameter File: FWK_FSC_CURE_1.par

6.3.9.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
0	SIM Initialisation	Responses ignored.	
1	Seek without affecting another record pointer	1 - No exception shall be thrown.	1 - A GET INKEY proactive
	Applet1 registers to	2 - No exception shall be thrown.	command is fetched from
	EVENT_FORMATTED_SMS_PP_ENV	3 - No exception shall be thrown.	the SIM
	Applet 2 registers to		
	EVENT_CALL_CONTROL_BY_SIM		
	1 - Applet 1:		
	- is triggered by a formatted SMS event		
	- selects DF_SIMTEST and EF_LARU		
	- reads record 2 using NEXT so that the		

ld	Description	API/Framework Expectation	APDU Expectation
	current record pointer is set to record 2 - issues a proactive command, e.g. Get Inkey.		
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM Applet 2:		
	- is triggered by a CALL CONTROL event - selects DF_SIMTEST and EF_LARU - performs a seek of pattern {0x55} from beginning forward, which finds record 1 returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		
2	updateRecord in linear fixed EF without affecting current pointer of others 1 - Applet 1: - is triggered by a formatted SMS event - selects DF_SIMTEST and EF_LARU - reads record 2 using NEXT so that the current record pointer is set to record 2 - issues a proactive command, e.g. Get	1 - No exception shall be thrown.2 - No exception shall be thrown.3 - No exception shall be thrown.	1 - A GET INKEY proactive command is fetched from the SIM
	Inkey. 2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL BY SIM event - selects DF_SIMTEST and EF_LARU - updates record 1, by using mode "NEXT" returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		
3	readRecord in linear fixed EF without affecting current pointer of others 1 - Applet 1: - is triggered by a formatted SMS event - selects DF_SIMTEST and EF_LARU - reads record 2 using NEXT so that the current record pointer is set to record 2 - issues a proactive command, e.g. Get Inkey.	1 - No exception shall be thrown.2 - No exception shall be thrown.3 - No exception shall be thrown.	1 - A GET INKEY proactive command is fetched from the SIM
	2 - An ENVELOPE APDU containing a CALL CONTROL BY SIM is issued to the SIM		
	Applet 2: - is triggered by a CALL CONTROL BY SIM event - selects DF_SIMTEST and EF_LARU - reads record 1, by using mode "NEXT" returns from processToolkit		
	3 - The terminal response for Get Inkey reactivates Applet 1: - call readRecord() using CURRENT - the record read should still be record 2 of EF_LARU, containing {0xAA, 0xAA, 0xAA, 0xAA}		

6.3.9.3.4 Test Coverage

CRR Number	Test Case Number
CRRN1	1
CRRN2	2
CRRN3	not tested (see Note)
CRRN4	3
CRRN5	not tested (see Note)

Note: These requirements have not been tested because of an inconsistent behavior in 03.19, which is foreseen to be corrected in future releases.

6.3.10 Other parts transferred to framework from API

6.3.10.1 A handler is a temporary JCRE Entry Point object

Test Area Reference: FWK_API_HEPO

6.3.10.1.1 Conformance Requirement:

Normal execution

CRRN1: The EnvelopeHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

CRRN2: The EnvelopeResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

CRRN3: The ProactiveHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

CRRN4: The ProactiveResponseHandler is a Temporary JCRE Entry Point Object (see Javacard 2.1 Runtime Environment (JCRE) Specification [12]).

Parameters error

Context errors

6.3.10.1.2 Test suite files

Test Script: FWK_API_HEPO_1.scr

Test Applet: FWK_API_HEPO_1.java

Load Script: FWK_API_HEPO_1.ldr

Cleanup Script: FWK_API_HEPO_1.clr

Parameter File: FWK_API_HEPO_1.par

6.3.10.1.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	EnvelopeHandler.getTheHandler and store it in	SecurityException is thrown	
	a static field of the toolkit applet	·	
2	EnvelopeHandler.getTheHandler and store it in	SecurityException is thrown	
	a field of the toolkit applet		
3	EnvelopeResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a static field of the toolkit applet		
4	EnvelopeResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a field of the toolkit applet		

5	ProactiveHandler.getTheHandler and store it in a static field of the toolkit applet	SecurityException is thrown	
6	ProactiveHandler.getTheHandler and store it in	SecurityException is thrown	
	a field of the toolkit applet		
7	Build and send a DISPLAY TEXT command to		
	be able to get the reference of the		Proactive command fetched
	ProactiveReponseHandler		and terminal response is
			issued
	ProactiveResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a static field of the toolkit applet	·	
8	ProactiveResponseHandler.getTheHandler and	SecurityException is thrown	
	store it in a field of the toolkit applet		

6.3.10.1.4 Test Coverage

CRR number	Test case number
N1	1, 2
N2	3, 4
N3	5, 6
N4	7, 8

6.3.10.2 Transaction

Test Area Reference: FWK_API_TRAN

6.3.10.2.1 Conformance Requirement:

Normal execution

CRRN1: A pending toolkit applet transaction at the ProactiveHandler.send() method invocation is aborted..

6.3.10.2.2 Test suite files

Test Script: FWK_API_TRAN_1.scr

Test Applet: FWK_API_TRAN_1.java

Load Script: FWK_API_TRAN_1.ldr

Cleanup Script: FWK_API_TRAN_1.clr

Parameter File: FWK_API_TRAN_1.par

6.3.10.2.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	Verify that transaction is aborted when a proactive command is sent Initialise a byte field with 0x05		
	Build a display text proactive command.		
	beginTransaction()		
	Update the byte with 0x02		
	send the proactive command		
			Proactive command fetched and terminal response is issued
	Verify that the byte value is 0x05		
	JCSystem.getTransactionDepth()	Shall return 0	

6.3.10.2.4 Test Coverage

CRR number	Test case number
N1	1

6.3.10.3 Timer Id between Applets

Test Area Reference: FWK_API_TMID

6.3.10.3.1 Conformance Requirement:

Context errors

CRRC1: The method ToolkitRegistry.releaseTimer() shall throw a ToolkitException with INVALID_TIMER_ID reason if the timer is valid but isn't allocated to this applet.

6.3.10.3.2 Test suite files

Test Script: FWK_API_TMID_1.scr

Test Applet: FWK_API_TMID_1.java

Load Script: FWK_API_TMID_1.ldr

Cleanup Script: FWK_API_TMID_1.clr

Parameter File: FWK_API_TMID_1.par

6.3.10.3.3 Test Procedure

ld	Description	API/Framework Expectation	APDU Expectation
1	During installation :		
	First instance allocate a timer and store the		
	returned value in a static field.		
	Second instance allocate a timer.		
	Trig second instance and try to releaseTimer()	releaseTimer() shall throw a	
	with the static field value.	ToolkitException with	
		INVALID_TIMER_ID reason	

6.3.10.3.4 Test Coverage

CRR number	Test case number
N1	1

Annex A (normative): Class and Methods AID numbering and acronyms

A.1 Sim.access

Class Name	Acronyms	Numbering on 5 bits
SIMView	SVW	00001
SIMSystem	SSY	00010
SIMViewException	SVE	00011

A.1.1 SIMView methods

Method Name	Acronyms	Numbering on 6 bits
static final Constants		000001
<pre>short increase(byte[] incr, short incrOffset, byte[] resp, short respOffset)</pre>	INCR_BS_BS	000010
<pre>void invalidate()</pre>	INVL	000011
<pre>void readBinary(short fileOffset, byte[] resp, short respOffset, short respLength)</pre>	REDBS_BSS	000100
<pre>short readRecord(short recNumber, byte mode, short recOffset, byte[] resp, short respOffset, short respLength)</pre>	REDRSBS_BSS	000101
<pre>void rehabilitate()</pre>	REHA	000110
<pre>short seek(byte mode, byte[] patt, short pattOffset, short pattLength)</pre>	SEEKB_BSS	000111
<pre>void select(short fid)</pre>	SLCTS	001000
<pre>short select(short fid, byte[] fci, short fciOffset, short fciLength)</pre>	SLCTS_BSS	001001
<pre>short status(byte[] fci, short fciOffset, short fciLength)</pre>	STAT_BSS	001010
<pre>short updateBinary(short fileOffset, byte[] data, short dataOffset, short dataLength)</pre>	UPDBS_BSS	001011
<pre>void updateRecord(short recNumber, byte mode, short recOffset, byte[] data, short dataOffset, short dataLength)</pre>	UPDRSBS_BSS	001100

A.1.2 SIMSystem methods

Method Name	Acronyms	Numbering on 6 bits
static SIMView getTheSIMView()	GETS	000001

A.1.3 SIMViewException methods

Method Name	Acronyms	Numbering on 6 bits
static void throwIt(short reason)	THITS	000001
SIMViewException(short reason)	COORS	000010
Constants	CONS	000011

A.2 Sim.toolkit

Class Name	Acronyms	Numbering on 5 bits
ToolkitConstants	TKC	00001
ToolkitInterface	TKI	00010
EditHandler	EDH	00011
EnvelopeHandler	ENH	00100
EnvelopeResponseHandler	ERH	00101
MEProfile	MEP	00110
ProactiveHandler	PAH	00111
ProactiveResponseHandler	PRH	01000
ToolkitRegistry	TKR	01001
ViewHandler	VWH	01010
ToolkitException	TKE	01011

A.2.1 ToolkitConstants

Method Name	Acronyms	Numbering on 6 bits
Constants	CONS	000001

A.2.2 ToolkitInterface methods

Method Name	Acronyms	Numbering on 6 bits
<pre>void processToolkit (byte event)</pre>	PRTKB	000001

A.2.3 EditHandler methods

The numbering of the EditHandler methods it will be done in the classes inherit it: EnvelopeResponseHandler, ProactiveHandler, because the methods provided by this class as it is declared 'abstract'.

A.2.4 EnvelopeHandler methods

Method Name	Acronyms	Numbering on 6 bits
<pre>byte getEnvelopeTag()</pre>	GENT	000001
<pre>byte getItemIdentifier()</pre>	GIID	000010
<pre>short getSecuredDataLength()</pre>	GSDL	000011
<pre>short getSecuredDataOffset()</pre>	GSDO	000100
EnvelopeHandler getTheHandler()	GTHD	000101
<pre>short getTPUDLOffset()</pre>	GTPO	000110
Inherited Method Name: ViewHandler		
Byte compareValue(short valueOffset,byte[] compareBuffer, short compareOffset, short compareLength)	CPRVS_BSS	000111
Short copy(byte[] dstBuffer,short dstOffset,short dstLengt h)	COPY_BSS	001000
Short copyValue(short valueOffset, byte[] dstBuffer,short dstOffset,short dstLength)	CPYVS_BSS	001001
Byte findAndCompareValue(byte tag,byte[] compareBuffer,sh ort compareOffset)	FACRB_BS	001010
Byte findAndCompareValue(byte tag,byte occurence, short valueOffset,byte[] compareBuffer,short compare Offset,short compareLength)	FACRBBS_BSS	001011
Short FindAndCopyValue(byte tag,byte occurence,short value	FACYBBS_BSS	001100

Offset, byte[] dstBuffer, short dstOffset,		
short dstLength)		
Short	FACYB BS	001101
<pre>findAndCopyValue(byte tag,byte[] dstBuffer,short dst</pre>	_	
Offset)		
Byte	FINDBB	001110
FindTLV(byte tag,byte occurrence)		
Short	GLEN	001111
GetLength()		
Byte	GVBYS	010000
<pre>GetValueByte(short valueOffset)</pre>		
Short	GVLE	010001
GetValueLength()		

A.2.5 EnvelopeResponseHandler methods

Method Name	Acronym	Numbering on 6 bits 000001	
EnvelopeResponseHandler getTheHandler()	GTHD		
Void post(byte statusType)	POSTB	000010	
Void postAsBERTLV(byte statusType, byte tag)	POSTBB	000011	
Inherited Method Name: EditHandler			
Void appendArray(byte[] buffer, short offset, short	APDA	000100	
length, short dstLength)			
Void appendTLV(byte tag, byte value)	APTLBB	000101	
Void appendTLV(byte tag, byte[] value, short	APTLB_BSS	000110	
valueOffset, short valueLength)	ADTIBBB		
Void appendTLV(byte tag, byte value1, byte value2)	APTLBBB	000111	
Void appendTLV(byte tag, byte value1, byte[] value2, short value2Offset, short value2Length)	APTLBB_BSS	001000	
Void clear()	CLER	001001	
void cital()	OLEK	001001	
Inherited Method Name: ViewHandler			
Byte	CPRVS_BSS	001010	
compareValue(short valueOffset,byte[] compareBuffer,	CPRVS_BSS	001010	
short compareOffset, short compareLength)			
Short	COPY_BSS	001011	
Copy(byte[] dstBuffer,short dstOffset,short dstLengt			
h)			
Short	CPYVS_BSS	001100	
<pre>CopyValue(short valueOffset, byte[] dstBuffer,short dstOffset,short dstLength)</pre>			
byte[] ustburrer, short ustorrset, short ustbength)			
Byte	FACRB_BS	001101	
FindAndCompareValue(byte tag,byte[] compareBuffer,sh			
ort compareOffset)			
Byte findAndCompareValue(byte tag,byte occurence, short valueOffset,byte[] compareBuffer,short compare	FACRBBS_BSS	001110	
Offset, short compareLength)			
Short	FACYBBS_BSS	001111	
FindAndCopyValue(byte tag,byte occurence,short value	17101220_200	001111	
Offset, byte[] dstBuffer, short dstOffset,			
short dstLength)			
Short	FACYB_BS	010000	
<pre>findAndCopyValue(byte tag,byte[] dstBuffer,short dst Offset)</pre>			
Byte	FINDBB	010001	
FindTLV(byte tag,byte occurrence)	TINDDD	010001	
Short	GLEN	010010	
GetLength()			
Byte	GVBYS	010011	
GetValueByte(short valueOffset)	0) " =	0.45 (5.5	
Short CotValueLongth()	GVLE	010100	
GetValueLength()		1	

A.2.6 MEProfile methods

Method Name	Acronym	Numbering on 6 bits
Static boolean check(byte index)	CHECB	000001
Static boolean check(byte[] mask, short offset, short	CHECBSS	000010
length)		

A.2.7 ProactiveHandler methods

Method Name	Acronyms	Numbering on 6 bits
GetTheHandler()	GTHD	000001
<pre>Init(byte type, byte qualifier, byte dstDevice)</pre>	INITBBB	000010
<pre>InitDisplayText(byte qualifier, byte dcs, byte[] buffer, short offset, short length)</pre>	INDTBB_BSS	000011
<pre>InitGetInkey(byte qualifier, byte dcs, byte[] buffer, short offset, short length)</pre>	INGKBB_BSS	000100
<pre>InitGetInput(byte qualifier, byte dcs, byte[] buffer, short offset, short length, short minRespLength, short maxRespLength)</pre>	INGPBB_BSSSS	000101
Byte send()	SEND	000110
Inherited Method Name: EditHandler		
Void appendArray(byte[] buffer, short offset, short length, short dstLength)	APDA	000111
Void appendTLV(byte tag, byte value)	APTLBB	001000
Void appendTLV(byte tag, byte[] value, short valueOffset, short valueLength)	APTLB_BSS	001001
Void appendTLV(byte tag, byte value1, byte value2)	APTLBBB	001010
Void appendTLV(byte tag, byte value1, byte[] value2, short value2Offset, short value2Length)	APTLBB_BSS	001011
Void clear()	CLER	001100
Inherited Method Name: ViewHandler		
Byte CompareValue(short valueOffset,byte[] compareBuffer,s hort compareOffset, short compareLength)	CPRVS_BSS	001101
Short Copy(byte[] dstBuffer,short dstOffset,short dstLength	COPY_BSS	001110
Short CopyValue(short valueOffset, byte[] dstBuffer,short dstOffset,short dstLength)	CPYVS_BSS	001111
Byte FindAndCompareValue(byte tag,byte[] compareBuffer,sho rt compareOffset)	FACRB_BS	010000
Byte findAndCompareValue(byte tag,byte occurence, short valueOffset,byte[] compareBuffer,short compareO ffset,short compareLength)	FACRBBS_BSS	010001
Short FindAndCopyValue(byte tag,byte occurence,short value0 ffset, byte[] dstBuffer, short dstOffset, short dstLength)	FACYBBS_BSS	010010
Short findAndCopyValue(byte tag,byte[] dstBuffer,short dstO ffset)	FACYB_BS	010011
Byte FindTLV(byte tag,byte occurrence)	FINDBB	010100
Short GetLength()	GLEN	010101
Byte GetValueByte(short valueOffset)	GVBYS	010110
Short GetValueLength()	GVLE	010111

A.2.8 ProactiveResponseHandler methods

Method Name	Acronyms	Numbering on 6 bits	
nort <pre>CopyAdditionalInformation(byte[] dstBuffer,</pre>		000001	
short dstOffset, short dstLength)			
Short <pre>copyTextString(byte[] dstBuffer, short</pre>	CPTS_BS	000010	
dstOffset)			
Short <pre>getAdditionalInformationLength()</pre>	GTIL	000011	
Byte getGeneralResult()	GTGR	000100	
Byte getItemIdentifier()	GTII	000101	
Byte getTextStringCodingScheme()	GTCS	000110	
Short getTextStringLength()	GTTL	000111	
GetTheHandler()	GTHD	001000	
· · · · · · · · · · · · · · · · · · ·	OTTE	001000	
Inherited Method Name: ViewHandler			
Byte	CPRVS_BSS	001001	
CompareValue(short valueOffset,byte[] compareBuffer,s	0o_200	00.001	
hort compareOffset, short compareLength)			
Short	COPY_BSS	001010	
<pre>Copy(byte[] dstBuffer,short dstOffset,short dstLength)</pre>			
Short	CPYVS_BSS	001011	
CopyValue(short valueOffset,	0. 1 vo_boo	001011	
<pre>byte[] dstBuffer,short dstOffset,short dstLength)</pre>			
Byte	FACRB_BS	001100	
FindAndCompareValue(byte tag,byte[] compareBuffer,sho			
rt compareOffset)			
Byte findAndCompareValue(byte tag,byte occurence,	EAODDDO DOO	004404	
short valueOffset,byte[] compareBuffer,short compareO	FACRBBS_BSS	001101	
ffset, short compareLength)			
Short	FACYBBS_BSS	001110	
FindAndCopyValue(byte tag,byte occurence,short value0	1 AC1BB0_B00	001110	
ffset, byte[] dstBuffer, short dstOffset,			
short dstLength)			
Short	FACYB_BS	001111	
findAndCopyValue(byte tag,byte[] dstBuffer,short dstO			
ffset)			
Byte	FINDBB	010000	
FindTLV(byte tag,byte occurrence)	21 = 11		
Short Catlongth()	GLEN	010001	
GetLength() Byte	GVBYS	010010	
GetValueByte(short valueOffset)	GVDIS	010010	
Short.	GVLE	010011	
GetValueLength()	GVLE	010011	
		1	

A.2.9 ToolkitRegistry methods

Method Name	Acronyms	Numbering on 6 bits
<u>AllocateTimer</u> ()	ATIM	000001
<pre>changeMenuEntry(byte id, byte[] menuEntry, short</pre>	CMETB_BSSBZBS	000010
offset, short length, byte nextAction, boolean		
helpSupported, byte iconQualifier, short		
iconIdentifier)		
<u>clearEvent</u> (byte event)	CEVTB	000011
<u>disableMenuEntry</u> (byte id)	DMETB	000100
<pre>enableMenuEntry(byte id)</pre>	EMETB	000101
<pre>getEntry()</pre>	GETY	000110
<pre>getPollInterval()</pre>	GPOL	000111
<pre>initMenuEntry(byte[] menuEntry, short offset, short</pre>	IMET_BSSBZBS	001000
length, byte nextAction, boolean helpSupported, byte		
iconQualifier, short iconIdentifier)		
<u>isEventSet</u> (byte event)	IEVSB	001001
<pre>releaseTimer(byte timerIdentifier)</pre>	RTIM	001010
requestPollInterval(short duration)	RPOL	001011
<pre>setEvent(byte event)</pre>	SEVTB	001100
<pre>setEventList(byte[] eventList, short offset, short length)</pre>	SEVL_BSS	001101

A.2.10 ViewHandler methods

The numbering of the ViewHandler methods it will be done in the classes inherit it: EditHandler, EnvelopeHandler, ProactiveResponseHandler, ProactiveHandler, because the methods provided by this class as it is declared 'abstract'.

A.2.11 ToolkitException methods

Method Name	Acronyms	Numbering on 6 bits
Static void throwIt(short reason)	THITS	000001
ToolkitException(short reason)	COORS	000010
Constants	CONS	000011

Annex B (normative): Script file syntax and format description

B.1 Syntax description

Following is a syntax description in BNF.

```
<statement list> ::=
                     [ < statement > \n] +
<statement> ::=
                      <simple> | <switch> | <blank line>
<simple> ::=
                      <reset> | <init> | <command> | <remark>
<reset> ::=
                      RST
<init> ::=
                      INI < hexdata>
<command> ::=
                      CMD < hexdata > [ < response > ] ( < status > )
                      [ < hexdata> ]
<response> ::=
<status> ::=
                      ( < hexdata > )
<remark> ::=
                      REM < text line>
<switch> ::=
                      SWI { [<|abelled list>] + }
<labelled list> ::=
                      <label>: \n <statement list>
```

Description of syntax metalanguage:

\n represents a linebreak

[x] means x can appear optionally

[x] + means 1 or more appearances of x

x | y means x or y

[]{}: (bold) these are characters that appear literally in the script files

<text line> any character until the end of the line <blank line> a line containing no text is acceptable

< hexdata> data written in hexadecimal, each byte separated from the following by a whitespace

Each simple statement beginning with 3 characters different than the ones defined indicates another tool command, and shall be ignored by the parser if not recognised.

```
' ', '\t' : Can be used as separator
```

A long statement can be broken into several lines by using the character '\' at the end of each line which is not the last one in the statement.

For more details refer to the examples in B.3.

B.2 Semantics

Following is the meaning of each of the statements:

Sends an APDU Command to the card, including (optionally) the expected response data and also (optionally) the expected status words SW1, SW2.

RST: Resets and powers on the card

INI: Performs the terminal profile with the following data. Afterwards, it shall perform all the fetch and terminal response commands until there is no proactive session in progress.

REM: Used for comments

sw: Activates a switch condition. Every labelled list represents a list of statements to be executed, if the label matches the SW resulting from the previously executed command.

Evaluation of expected response and status in the case of a CMD:

data within [...] has to be checked, it needs to be present for an outgoing command. Bytes written as XX shall not be checked by the APDU tool.

status contained within (...) has to be checked; when several status are valid they shall be separated by commas. Bytes written as XX shall not be checked by the APDU tool.

B.3 Example

```
REM this is an example
RST
REM Case 1 example
CMD A0 C2 00 00 00 (91 33 , 69 XX)
REM Case 2 example
CMD A0 B6 00 00 07
   [XX XX XX 55 55 XX 55] \
    (91 33 , 67 XX)
CMD A0 B6 00 00 07 \
   (91 33 , 67 XX)
CMD A0 C0 00 00 1F \setminus
    [10 A0 00 00 00 09 00 02 FF FF FF FF 89 28 A4 05 \backslash
   02 0D CC CC CC CC CC CC CC CC CC CC CC CC ] \
   (90 00)
REM Case 3 example
CMD A0 C2 00 00 33 \
   D1 31 82 02 83 81 06 05 80 11 22 33 44 8B 24 40 \
   08 00 24 23 85 18 41 04 51 10 10 00 00 00 00 13 \
   02 70 00 00 0E 0D 00 00 00 00 28 A4 05 00 00 00 \
   00 00 00 \
   (90 00)
REM Case 4 example with switch statement
CMD 00 A4 04 00 10 \
   A0 00 00 00 09 00 02 FF FF FF FF 89 41 04 44 02 \
   (61 XX, 6A 82)
SWI {
CMD 00 C0 00 00 14 \
   [10 A0 00 00 00 09 00 02 FF FF FF FF 89 41 04 44 \
   02 02 CC CC] \
   (90 00)
CMD A0 A4 00 00 02 \
   3F 00
6A 82:
RST
```

B.4 Style and formatting

In order to show a common appearance all the scripts shall follow those format rules:

- start always with a 'RST' followed by an 'INI' command.
- The command, data to be checked and status to be checked shall be presented in the following order:
 - CMD COMMAND [EXPECTED DATA] (EXPECTED STATUS)
- APDU shall be presented with command (CLA INS P1 P2 P3) in one line and data (if present) in next line grouped 16 bytes per line (see example above).
- The expected data (if present) shall be presented in 16 bytes groups per line (see example above).

Annex C (normative): Default Prepersonalisation

C.1 General Default Prepersonalisation

This table shows the default prepersonalisation, the file system and the files' content, that the test SIM cards shall contain unless otherwise stated.

Name	Identifier	Default Value	Special Features
EFICCID	2FE2	OF FF FF FF FF FF FF FF	This value is not compliant with GSM 11.11
EF _{IMSI}	6F07	FF FF FF FF FF FF FF	This value is not compliant with GSM 11.11
EFLP	6F05	01 FF FF FF	
EF _{Kc}	6F20	FF FF FF FF FF FF FF 07	
EF _{PLMNsel}	6F30	FF FF FF FF FF FF FF FF FF FF	
		FF FF FF FF FF FF FF FF FF FF	
EF _{HPLMN}	6F31	05	
EF _{ACMmax}	6F37	00 00 00	Access condition UPDATE: CHV1
EF _{SST}	6F38	FF 3F C3 03 0C 00 FF 0F 00 33	
EF _{ACM}	6F39	00 00 00	Access condition UPDATE: CHV1
EF _{PUCT}	6F41	FF FF FF 00 00	Access condition UPDATE: CHV1
EF _{BCCH}	6F74	FF FF FF FF FF FF FF FF FF FF FF	
FF	0570	FF FF FF FF 00 00	
EF	6F78 6F7B	FF FF FF FF FF FF FF FF FF FF FF	
EF _{FPLMN}	6F7B	FF FF FF FF 00 F0 00 00 00 FF 01	
EF _{LOCI}	6FAD	00 FF FF	
EF _{AD}	6FAE	03	
EF _{Phase}	6F3B	Default value in all the records:	Departe F
EF _{FDN}	0535	FF FF FF FF FF FF FF FF FF FF FF	Records: 5
		FF FF FF FF FF FF FF FF FF FF	
		FF FF FF FF	
EF _{SMSP}	6F42	FF FF FF FF FF FF FF FF FF FF	Records: 1
		FF FF FF FF FF FF FF FF FF FF FF FF	
		FF FF FF FF FF FF	
EF _{LND}	6F44	FF FF FF FF FF FF FF FF FF FF FF	Records: 1
_ LIND	0	FF FF FF FF FF FF FF FF FF FF	Troopids. 1
		FF FF FF FF	
EF _{SMSS}	6F43	FF FF	
EF _{SMS}	6F3C	1 st record: 00 FF FF(length 176)	Records: 3
		2 nd record:00 FF FF(length 176) 3 rd record: 00 FF FF(length 176)	
EF _{ADN}	6F3A	FF FF FF FF FF FF FF FF FF FF FF	Records: 1
L' ADN	0134	FF FF FF FF FF FF FF FF FF FF	Trecords. 1
		FF FF FF FF	
EF _{CCP}	6F3D	FF FF FF FF FF FF FF FF FF FF	
	25.40	FF FF	
EF _{MSISDN}	6F40	FF FF FF FF FF FF FF FF FF FF FF FF	Records: 1
		FF FF FF FF	
EF _{SDN}	6F49	FF FF FF FF FF FF FF FF FF FF FF	Records: 1
ODIV.		FF FF FF FF FF FF FF FF FF FF	
		FF FF FF FF	
EF _{SUME}	6F54	85 0C 54 4F 4F 4C 4B 49 54 20 54 45	
	GE 1E	53 54 FF FF FF FF FF FF	
EF _{CBMI}	6F45	FF FF FF FF FF FF FF FF FF	
EF _{IM}	4F20	ce ce ce th th th th th th th	

The default value for the CHV1 shall be "0x31 0x31 0x31 0x31 0xFF 0xFF 0xFF 0xFF" and its state shall be 'disabled' during test applets execution.

C.2 Sim.Access.SimView test default prepersonalisation

C.2.1 DF_{SIMTEST} (SIM Test)

Identifier: '0319'

C.2.2 EF_{TNR} (Transparent Never Read)

Identifier: '6F01' St		ructure: transparent N		Mandatory	
File size: 3 bytes		Update activity: low			
Access Conditions:					
	READ	NEVER			
	UPDAT	Έ	ALWAYS		
	INVALI	DATE	ALWAYS		
	REHAB	SILITATE	ALWAYS		
Bytes	Description	[Default Value	M/O	Length
1 – 3	Test Data		AA AA AA	М	3 bytes

C.2.3 EF_{TNU} (Transparent Never Update)

Identifier: '6F02'		Str	ucture: transparent	Mandatory	
	File size: 3 bytes		Update activity	y: low	
Access Conditions:					
	READ		ALWAYS		
	UPDA	TE	NEVER		
	INVALI	DATE	ALWAYS		
	REHAE	BILITATE	ALWAYS		
Bytes	Description	I	Default Value	M/O	Length
1 - 3	Test Data		55 55 55	М	3 bytes

C.2.4 EF_{TARU} (Transparent Always Read and Update)

	Identifier: '6F03'		Structure: transparent Mandatory		indatory
	File size: 260 bytes		Update activity: low		
	Access Conditions:				
	READ		ALWAYS		
	UPDATE		ALWAYS		
	INVALIDATE		ALWAYS		
	REHAB	ILITATE	ALWAYS		
				I I	
Bytes	Description	[Default Value	M/O	Length
1 - 260	Test Data		FF FF	M	260
					bytes

C.2.5 EF_{CNR} (Cyclic Never Read)

Iden	tifier: '6F04'		Structure: cyclic Mandatory				
R	Record length: 3 bytes		Update activity: high				
	Access Conditions:						
	READ		NEVER	3			
	UPDA		ALWAY	_			
	INCRE		ALWAYS				
		IDATE	ALWAY	_			
	KEHA	BILITATE	ALWAY	5			
Logical	Description		Default Value	M/O	Length		
Record							
Number							
1	Test Data		00 00 00	M	3 bytes		
2	Test Data		00 00 00	M	3 bytes		

C.2.6 EF_{CNU} (Cyclic Never Update)

Iden	Identifier: '6F05'		Structure: cyclic		Mandatory
R	ecord length: 3 bytes		Update	activity:	high
	READ	Access Cor	nditions:	S	
	UPDA INCR INVAL		NEVEF NEVEF ALWAY ALWAY	₹ ₹ 'S	
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes
2	Test Data		00 00 00	М	3 bytes

C.2.7 EF_{CNIC} (Cyclic Never Increase)

Ider	Identifier: '6F06		Structure: cyclic		Mandatory		
F	Record length: 3 bytes		Update activity: high				
	Access Conditions:						
	READ UPDAT INCRE INVALI REHAE	EASE	ALWAY ALWAY NEVEI ALWAY ALWAY	/S R /S			
Logical Record Number	Description		Default Value	M/O	Length		
1	Test Data		00 00 00	М	3 bytes		
2	Test Data		00 00 00	М	3 bytes		

C.2.8 EF_{CNIV} (Cyclic Never Invalidate)

Iden	Identifier: '6F07		Structure: cyclic		Mandatory			
R	Record length: 3 bytes		Update activity: high					
	Access Conditions:							
		TE	ALWAY ALWAY ALWAY NEVEF ALWAY	S S				
Logical Record Number	Description		Default Value	M/O	Length			
1	Test Data		00 00 00	М	3 bytes			
2	Test Data		00 00 00	М	3 bytes			

C.2.9 EF_{CNRH} (Cyclic Never Rehabilitate)

Iden	tifier: '6F08'	(Structure: cyclic		Mandatory				
R	Record length: 3 bytes		Update activity: high						
	Access Conditions:								
READ ALWAYS UPDATE ALWAYS INCREASE ALWAYS INVALIDATE ALWAYS REHABILITATE NEVER									
Logical Record Number	Description		Default Value	M/O	Length				
1	Test Data		00 00 00	М	3 bytes				
2	Test Data		00 00 00	М	3 bytes				

C.2.10 EF_{CARU} (Cyclic Always Read and Update)

Iden	Identifier: '6F09'		Structure: cyclic Mandatory		
R	Record length: 3 bytes		Update activity: high		
	READ	Access Cor	nditions:	S	
	UPDA' INCRE	TE	ALWAY ALWAY	S	
		IDATE BILITATE	ALWAY ALWAY	_	
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		55 55 55	М	3 bytes
2	Test Data		AA AA AA	М	3 bytes

C.2.11 EF_{LNR} (Linear Fixed Never Read)

	Identifier: '6F0A'		Structure: linear fixed Mandatory		ndatory
	Record length: 4 bytes		Update activity: low		
Access Conditions:					
	READ	_	NEVER		
	UPDATI INVALIF		ALWAYS ALWAYS		
	REHAB	· · · · -	ALWAYS		
Logical	Description		Default Value	M/O	Length
Record Number					
1	Test Data - Record 1		FF FF FF FF	М	4 bytes
2	Test Data - Record 2		FF FF FF FF	М	4 bytes

C.2.12 EF_{LNU} (Linear Fixed Never Update)

Identifier: '6F0B'		Str	Structure: linear fixed Mandatory		ndatory
Record length: 4 bytes			Update activity	: low	
Access Conditions:					
	READ UPDAT INVALIE REHABI	ATE	ALWAYS NEVER ALWAYS ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data - Record 1		FF FF FF FF	М	4 bytes
2	Test Data - Record 2		FF FF FF FF	М	4 bytes

C.2.13 EF_{LARU} (Linear Fixed Always Read and Update)

Identifier: '6F0C'		Str	Structure: linear fixed Mandatory		ndatory
	Record length: 4 bytes		Update activity: low		
	Access Conditions:				
	READ UPDATI INVALIE REHAB	DATE	ALWAYS ALWAYS ALWAYS ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data - Record 1		55 55 55 55	М	4 bytes
2	Test Data - Record 2		AA AA AA AA	М	4 bytes

C.2.14EF_{CINA} (Cyclic Increase Not Allowed)

Identifier: '6F0D'		Structure: cyclic	Structure: cyclic Mandatory		
Rec	Record length: 3 bytes		Update activity: high		
Access Conditions:					
	READ UPDATE		ALWAYS ALWAYS		
	INCREASE		ALWAYS (see note 1)		
	INVALID REHABII		ALWAYS ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes
2	Test Data		00 00 00	М	3 bytes
Note 1: This file will be personalised in a way such that increase is not allowed, as indicated by the FCI byte 8, bit 7 (GSM 11.11: FCI structure of an EF returned by the SELECT command)					

C.2.15EF_{TRAC} (Transparent Read Access Condition CHV2)

Identifier: '6F0E'		Str	ucture: transparent	Man	datory
Record length: 3 bytes		Update activity: low			
Access Conditio					
	READ UPDATE		CHV2 ALWAYS		
	INCREA	_	ALWAYS		
	INVALID REHABII		ALWAYS ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes

C.2.16EF_{TIAC} (Transparent Invalidate Access Condition CHV1)

Identifier: '6F0F' S		Str	ucture: transparent	Man	datory
Record length: 3 bytes		Update activ	ity: low		
	A	ccess Condi	tions:		
	READ		ALWAYS		
	UPDATE		ALWAYS		
	INCREAS	SE	ALWAYS		
	INVALI	DATE	CHV1		
	REHABII	LITATE	ALWAYS		
	.	1	D (10)/ 1	14/0	1 4
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes

C.2.17EF_{CIAC} (Cyclic Increase Access Condition CHV2)

Identific	er: '6F10'	Structure: cyclic	Man	datory	
Rec	ord length: 3 bytes	Update activity	Update activity: low		
Access Conditions:					
	READ UPDATE INCREASI INVALIDATE REHABILITA	ALWAYS ALWAYS E CHV2 E ALWAYS			
Logical Record Number	Description	Default Value	M/O	Length	
1	Test Data	00 00 00	М	3 bytes	
2	Test Data	00 00 00	М	3 bytes	

C.2.18EF_{CIAA} (Cyclic Increase Access Condition ADM)

Identific	er: '6F11'		Structure: cyclic	Mar	datory
Rec	Record length: 3 bytes		Update activity: low		
	AI READ UPDATE INCREA:		litions: ALWAYS ALWAYS ADM		
	INVALIDA REHABIL		ALWAYS ALWAYS		
Logical Record Number	Description		Default Value	M/O	Length
1	Test Data		00 00 00	М	3 bytes
2	Test Data		00 00 00	М	3 bytes

C.2.19EF_{CNRI} (Cyclic Never Rehabilitate Invalidated)

Identific	er: '6F12'		Structure: cyclic		Man	datory
Rec	Record length: 3 bytes		Update activity: low			
	Access Conditions:					
	READ UPDATE INCREAS INVALIDA REHABI	SE ATE	ALWAYS ALWAYS ALWAYS ALWAYS NEVER			
Logical Record	Description		Default Value		M/O	Length
Number	Bosonption		Boladit Valdo		101/10	Longui
1	Test Data		00 00 00		М	3 bytes
2	Test Data		00 00 00		М	3 bytes

The file status shall be invalidated as defined in [3]

Annex D (normative): sim.test.util package and loading, testing and cleaning script examples.

See attached files:

- Annex_D_SimTestUtil.zip
- Annex_D_Examples.zip

Annex E (normative): Test Area files.

See attached file:

- Annex_E_SourceCode.zip

Annex F (Normative): AID numbering and acronyms for Framework tests

F.1 Toolkit Installation Parameters (TIN)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Timer allocation	TMAL	000001
Item identifier	ITID	000010
Item position	ITPO	000011
Access conditions	ACCO	000100
Priority level	PRLV	000101
Maximum length for each menu entry	MLME	000110
Number of menu entries	NBME	000111
Memory space	MESP	001000

F.2 Minimum Handler Availability (MHA)

Test Area within the chapter	Acronyms	Numbering on 6 bits
ProactiveHandler	PAHD	000001
ProactiveResponseHandler	PRHD	000010
EnvelopeHandler	ENHD	000011
EnvelopeResponseHandler	ERHD	000100

F.3 Handler Integrity (HIN)

Test Area within the chapter	Acronyms	Numbering on 6 bits
ProactiveHandler	PAHD	000001
ProactiveResponseHandler	PRHD	000010
EnvelopeHandler	ENHD	000011
RFU (EnvelopeResponseHandler)	(ERHD)	000100

F.4 Applet Triggering (APT)

Test Area within the chapter	Acronyms	Numbering on 6 bits
EVENT_PROFILE_DOWNLOAD	EPDW	000001
EVENT_MENU_SELECTION	EMSE	000010
EVENT_MENU_SELECTION_HELP_REQUEST	EMSH	000011
EVENT_FORMATTED_SMS_PP_ENV	EFSE	000100
EVENT_UNFORMATTED_SMS_PP_ENV	EUSE	000101
EVENT_CALL_CONTROL_BY_SIM	ECCN	000110
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	EMCN	000111
EVENT_TIMER_EXPIRATION	ETEX	001000
EVENT_UNFORMATTED_SMS_CB	EUCB	001001
EVENT_EVENT_DOWNLOAD_MT_CALL	EDMC	001010
EVENT_EVENT_DOWNLOAD_CALL_CONNECTED	EDCC	001011
EVENT_EVENT_DOWNLOAD_CALL_DISCONNECTED	EDCD	001100
EVENT_EVENT_DOWNLOAD_LOCATION_STATUS	EDLS	001101
EVENT_EVENT_DOWNLOAD_USER_ACTIVITY	EDUA	001110
EVENT_EVENT_DOWNLOAD_IDLE_SCREEN_AVAILABLE	EDIS	001111
EVENT_EVENT_DOWNLOAD_CARD_READER_STATUS	EDCR	010000
RFU (EVENT_UNRECOGNIZED_ENVELOPE)	(EUEN)	010001
EVENT_STATUS_COMMAND	ESTC	010010

F.5 Proactive Command Sending (PCS)

Test Area within the chapter	Acronyms	Numbering on 6 bits
System Proactive commands	SPCO	000001
Interaction with GSM commands	IGCO	000010
Errors during proactive command sending	EPCS	000011

F.6 Envelope Response Posting (ERP)

Test Area within the chapter	Acronyms	Numbering on 6 bits
EVENT_CALL_CONTROL_BY_SIM	ECCN	000001
EVENT_MO_SHORT_MESSAGE_CONTROL_BY_SIM	EMCN	000010
EVENT_UNRECOGNIZED_ENVELOPE	EUEN	000011

F.7 Framework Security (FWS)

Test Area within the chapter	Acronyms	Numbering on 6 bits		
Input data	INDA	000001		
Output data	OUDA	000010		

F.8 File System Context (FSC)

Test Area within the chapter	Acronyms	Numbering on 6 bits
Initial Context	INIT	000001
Context Preservation for Current File	CUFI	000010
Context Preservation for Current Record	CURE	000011

F.9 Exception Handling (EXH)

Test Area within the chapter	Acronyms	Numbering on 6 bits		
Hide exception to the mobile	HEME	000001		
Interaction with multi-triggering	IMTG	000010		

F.10 Other parts transferred to framework from API (API)

Test Area within the chapter	Acronyms	Numbering on 6 bits
A handler is a temporary JCRE Entry Point object	HEPO	000001
Transaction	TRAN	000010
Timer Id between Applets	TMID	000011

Annex G (Normative): Configuration Parameters File

This file describes all the mandatory and optional parameters that are used in order to create the loading script(s) for one test area. The configuration parameters file contains the values for the parameters needed in order to generate the loading and cleanup scripts.

The name of the parameters file will be <test area reference>_<n>.par.

The number <n> is associated with the loading/cleanup script number, i.e. API_2_TKR_ SEVL_BSS_1.par is used to generate API_2_TKR_ SEVL_BSS_1.ldr etc.

G.1 Syntax

The general syntax for this file will be:

```
<file> ::= <section>+

<section> ::= <section heading> line break> <section body>
<section heading> ::= `[' <name> `]'
<section body> ::= <parameter assignment>+
<parameter assignment> ::= <name> `=' <value> line break>
```

Where '+' indicates one or more repetitions of the previous syntax element.

Any text included between the symbol ';' and the end of line is considered a comment and ignored by parsing tools.

Empty values are considered valid. They are used to indicate that an optional value is not present.

Names of sections, names of parameters and values are case-sensitive.

Blank spaces and Tabs between tokens are allowed and will be ignored by the parser.

When values represent a sequence of bytes, they are expressed in hexadecimal format, where every 2 digits represent one byte. Blank space between bytes is optional.

Example:

```
; comment

[Section1]

Parameter11 = 00 11 22 33

Parameter12 = 0101  ; another comment

[Section2]

Parameter21 = vvwwxxyyzz
```

G.2 File Contents and Organisation

Parameters in this file are organised in the following sections:

[CONVERT]	Conversion parameters used during conversion (i.e. CAP file generation)
[INSTALL(load)]	Parameters used by the Install for Load command
[LOAD]	Parameters used by the Load command
[INSTALL(install)]	Parameters used by the Install for Install command

All sections may appear only once in the file, except for the "INSTALL(install)" section. If that section appears more than once, it will apply to different applet instances, in sequence.

G.2.1 Default values, order and processing

The ordering of the parameters and the sections is relevant, since parameter names may be repeated and apply to different applets.

When one single parameter is repeated within one section, it refers to different applets. The value of the n^{th} appearance of the parameter applies to applet n.

When one section is repeated (INSTALL(install)), then the n^{th} appearance of the section applies to applet n. Parameter/value pairs which are found in one appearance of the section are valid for the subsequent applets as long as they are not overriden. For example, first INSTALL(install) may contain all values for parameters, whereas the subsequent INSTALL(install) sections may only contain parameters whose values change.

If one required parameter is missing from one section, the last defined value of this parameter in a previous section of the same file will be used.

G.2.2 CONVERT Section

These parameters allow configuration of the conversion process of the Java class file(s) into one CAP file.

Parameter	Description
PackageAID	AID of the package
PackageName	Fully qualified name of the package
PackageVersion	Version of the package
AppletClassAID	AID of the applet
AppletClassName	Name of the applet

G.2.3 INSTALL(load) Section

Here are the parameters to be included in the Install(Load) command (as specified in GSM 03.48 [8]).

Parameter	Description
PackageAID	AID of the package
PackageNonVolatileMemSize	Non Volatile memory space (in bytes) required for package loading
InstallationNonVolatileMemSize	Non volatile memory required for installation, in bytes
InstallationVolatileMemSize	Volatile memory required for installation, in bytes

G.2.4 LOAD Section

Here are the parameters to be included in the Load command (as specified in GSM 03.48 [8]).

Parameter	Description
MaxLoadCommandDataLength	Maximum length of the data provided in the load command (P3
	parameter of the LOAD APDU embedded in the command packet)

G.2.5 INSTALL(install) Section

Here are the parameters to be included in the Install(Install) command (as specified in GSM 03.48 [8]

Parameter	Description
PackageAID	AID of the package
AppletClassAID	AID of the applet
InstanceAID	AID of the instance of the applet
InstallationNonVolatileMemSize	Non volatile memory required for installation, in bytes
InstallationVolatileMemSize	Volatile memory required for installation, in bytes
AccessDomain	Specify the SIM files that may be accessed by the applet and the
	operations allowed on these files. This parameter includes the
	Access Domain Parameter (ADP) and Access Domain Data (ADD)
PriorityLevel	Priority level of the Toolkit applet instance
MaxNumberOfTimers	Maximum number of timers allowed for this applet instance
MaxMenuEntryTextLength	Maximum text length for a menu entry
MaxNumberOfMenuEntries	Maximum number of menu entries allowed for this applet instance
MenuEntriesPositionIdentifier	For each menu entry: Position and identifier of that menu entry
AppletSpecificParameters	Parameters specific to the applet

The applet shall be installed with install(install and make selectable) command.

G.3 Full example

```
[CONVERT]
PackageAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 00
PackageName = sim.test.access.api_1_svw_updrbs
PackageVersion = 1.0
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
AppletClassName = API_1_SVW_UPDRBS_1
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02
AppletClassName = API_1_SVW_UPDRBS_2
[INSTALL(load)]
PackageNonVolatileMemSize = 0D27
;InstallationNonVolatileMemSize = 0400
;InstallationVolatileMemSize = 0000
[LOAD]
MaxLoadCommandDataLength = 6C ; max value
[INSTALL(install)]
AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
InstanceAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 01
InstallationNonVolatileMemSize = 0400
InstallationVolatileMemSize = 0000
AccessDomain = 00
PriorityLevel = FF
MaxNumberOfTimers = 00
MaxMenuEntryTextLength = 10
```

```
MaxNumberOfMenuEntries = 01

MenuEntriesPositionIdentifier = 0001

AppletSpecificParameters =

[INSTALL(install)]

AppletClassAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02

InstanceAID = A0 00 00 00 30 00 02 FF FF FF FF 89 00 00 01 02

InstallationNonVolatileMemSize = 0200

InstallationVolatileMemSize = 0000

MenuEntriesPositionIdentifier = 0002
```

[;] rest of INSTALL(install) parameters are taken from previous INSTALL(install)...

Annex H (informative): Change history

The table below indicates all changes that have been made to the present document since drafting work began.

Change history								
Date	TSG#	TSG Doc	CR	Rev	Cat	Subject/Comment	Old	New
2000-10	-	-				Draft presented at T3 #16		0.2.0
2000-12	TP-10	TP-000208				Presented to TSG-T #10 for information	0.2.0	1.0.0
2001-01	-	-				Input to T3 #17 resulting from T3 ad hoc #24	1.0.0	1.1.0
2001-03	-	-				Document presented for approval at T3 #18	1.1.0	1.2.0
2001-03	TP-11	TP-010041				Doument presented for approval to TSG-T #11 (identical in technical content to v1.2.0)	1.2.0	2.0.0
2001-03						As approved at TSG-T #11 (identical in technical content to v2.0.0)	2.0.0	7.0.0
2001-05						Correction to date on cover page / headers	7.0.0	7.0.1
2001-06	TP-12	TP-010105	A001	-	F	Corrections to the API Test plan, addition of the test area files and modification of the util package	7.0.1	7.1.0
2001-09	TP-13	TP-010206	A002	-	F	Update API Test plan and Test Area Files	7.1.0	7.2.0
2001-11	TP-14	TP-010241	A003	-	F	Specification for framework part	7.2.0	7.3.0
		TP-010241	A004	-	F	API part		
2002-03	TP-15	TP-020073	004	-	F	Testing Framework Update	7.3.0	7.4.0
2002-03						Files for Annexes D and E added, Editorial correction performed in 6.3.2.3.3 colomn 9 (1- Applet triggered)	7.4.0	7.4.1
2002-09	T3#24					Reference [14] changed as TS 101 220 v3.0.0 was withdrawn.	7.4.1	7.4.2

History

Document history				
V7.0.1	May 2001	Publication		
V7.1.0	June 2001	Publication		
V7.2.0	September 2001	Publication		
V7.3.0	December 2001	Publication		
V7.4.1	March 2002	Publication (Withdrawn)		
V7.4.2	September 2002	Publication		