Open Mobile API Test Specification for Transport API

Errata for Version 2.2

Published by simalliance now Trusted Connectivity Alliance

July 2016

Table of Contents

1. Introduction 3
2. List of modifications 3
2.1 Chapter 4.2 – Table of DUT options 3
2.2 Chapter 4.3 - Applicability table 3
2.3 Chapter 5.2.2 – UICC, eSE and mSD 4
2.4 Chapter 5.7 - Secure element test applets 4
2.5 Test case 6.1.6 Int getVersion() 6
2.6 Test case 6.4.7 openLogicalChannel() ID18 7
2.7 Test case 6.4.7 openLogicalChannel() ID19 7
2.8 Test cases 6.5.4 getSelectResponse() 8
2.9 Test cases 6.5.6 transmit() 10
2.10 Test cases 6.5.7 selectNext() 11
2.11 Annex B 12
Access Control Applet (ARA) 12
Access Control File System (ARF) 12
2.12 Annex F 14

1. Introduction

This document contains errata notes for Open Mobile API Test Specification for Transport API v2.2. This document uses revision marks to show the new applicable content. The document contains only those parts of the subchapters where the errata are made. All the other parts of the Open Mobile API Test Specification for Transport API v2.2 remain unchanged and applicable.

2. List of modifications

2.1 Chapter 4.2 – Table of DUT options

Rationale of the errata: OP-015 and OP-016 DUT options are deleted, because these are not used in any test cases. New DUT options are added.

Item	Option	Status	Optional item
15	~~DUT relies on the ATR to know if the SE supports partial DF selection~~VOID	~~OP~~	~~OP-015~~
16	~~DUT does not rely on the ATR to know if the SE supports partial DF selection~~VOID	~~OP~~	~~OP-016~~
17	DUT has a reader from which the SE (UICC, or mSD) is removable also when the DUT is in power on state (eg: DUT with UICC/mSD tray)	OP	OP-017
18	DUT has a reader from which the SE (UICC, or mSD) is removable only when the DUT is in power off state (eg: DUT with UICC/mSD under the battery)	OP	OP-018

2.2 Chapter 4.3 - Applicability table

Rationale of the errata: correct the applicability of some test cases.

Clause	Test case number and description	Core version	SUE	RSE		
				UICC	eSE	mSD
6.1.5	Method: String getVersion()	2.05 ~~N/A from 3.2~~	M	M	M	M
6.1.6	~~Method: Int getVersion()~~VOID	~~3.2~~	~~M~~	~~M~~	~~M~~	~~M~~
6.3.7	Method:void registerReaderEventCallback(Reader.EventCallBack cb) ID4 – ID6~~8~~	3.2	M	M	N/A	M
6.3.7	Method:void registerReaderEventCallback(Reader.EventCallBack cb) ID7 – ID8	3.2	OP-017	OP-017	N/A	OP-017
6.3.7	Method:void registerReaderEventCallback(Reader.EventCallBack cb) ID10 – ID11	3.2	OP-003	N/A	N/A	N/A
6.3.7	Method:void registerReaderEventCallback(Reader.EventCallBack cb) ID12~~10~~ – ID15	3.2	M	N/A	N/A	N/A

Clause	Test case number and description	Core version	SUE	RSE		
				UICC	eSE	mSD
6.3.8	Method:boolean unregisterReaderEventCallback(Reader.EventCallBack cb) ID4, ID5	3.2	OP-017~~M~~	OP-017~~M~~	N/A	OP-017~~M~~
6.3.8	Method:boolean unregisterReaderEventCallback(Reader.EventCallBack cb) ID7 – ID8	3.2	OP-003	N/A	N/A	N/A
6.3.8	Method:boolean unregisterReaderEventCallback(Reader.EventCallBack cb) ID~~9~~7 – ID12	3.2	M	N/A	N/A	N/A
6.5.1	Method: void close() ID2	2.05 NA from 3.2	OP-003	OP-003	OP-003	OP-003
~~6.5.7~~	~~Method: Boolean[] selectNext() ID6a~~	~~2.05~~	~~OP-016~~	~~NA~~	~~NA~~	~~NA~~
~~6.5.7~~	~~Method: Boolean[] selectNext() ID6b~~	~~2.05~~	~~OP-015~~	~~NA~~	~~NA~~	~~NA~~

2.3 Chapter 5.2.2 – UICC, eSE and mSD

Rationale of the errata: modify test tool requirement for test tools using real UICC. For T=0 transmission protocol when the response of the case 4 APDU command contains data and warning status word, ISO and ETSI transport layers provides two different recommendations. For testing there is no need to use two different types of SE, the ISO/ETSI recommended behaviours can be implemented by the test applets. Unless otherwise specified, OMAPI test applets implement the ISO recommended behaviour. For those test cases where ETSI behaviour is required specific applets implementing the ETSI recommended behaviour are referenced. These test cases are also listed in Annex F.

- ~~two types of SEs are required:~~
  - ~~SE implementing transport layer according to ISO recommendation (send first a "61 xx" and then - after receiving GET RESPONSE command from the device - the data with the warning status word)~~
  - ~~SE implementing transport layer according to ETSI recommendation: send first SW warning instead of 61 XX and follow the procedure as described in Annex C of [12]~~

~~Note: Unless otherwise specified, the SE to be used is the one implementing ISO behaviour. A list of test cases that shall use the SE implementing ETSI behaviour can be found in Annex F:.~~

2.4 Chapter 5.7 - Secure element test applets

Rationale of the errata: delete and modify some AIDs and add some new test applets.

The following AIDs are used in the present document:

AID_TestApp	A0 00 00 06 00 01 00 01 EE 05 01
AID_TestApp_SW6999	A0 00 00 06 00 01 00 01 EE 05 02
AID_TestApp_SW6280	A0 00 00 06 00 01 00 01 EE 05 03
AID_TestApp_SW6283	A0 00 00 06 00 01 00 01 EE 05 04
AID_TestApp_SW6310	A0 00 00 06 00 01 00 01 EE 05 05

AID_TestApp_SW63C1	A0 00 00 06 00 01 00 01 EE 05 06
AID_TestApp_selectresponse	A0 00 00 06 00 01 00 01 EE 05 07
AID_TestApp_SW6280_selectresponse	A0 00 00 06 00 01 00 01 EE 05 08
AID_TestApp_SW6283_selectresponse	A0 00 00 06 00 01 00 01 EE 05 09
AID_TestApp_SW6310_selectresponse	A0 00 00 06 00 01 00 01 EE 05 0A
AID_TestApp_SW63C1_selectresponse	A0 00 00 06 00 01 00 01 EE 05 0B
AID_TestApp_p1p2	A0 00 00 06 00 01 00 01 EE 05 0C
AID_TestApp_clains	A0 00 00 06 00 01 00 01 EE 05 0D
AID_Partial_1	A0 00 00 06 00 01 00 01 EE 05 0E
AID_Partial_1_instance_1	<AID_Partial_1> 01
AID_Partial_1_instance_2	<AID_Partial_1> 02
AID_Partial_2	<AID_Partial_1_instance_1>
AID_Partial_2_instance_1	<AID_Partial_2>
~~AID_Length_5~~	~~A0 00 00 06 00~~
AID_Length_6	A0 00 00 06 00 02~~1~~
AID_Length_7	A0 00 00 06 00 02~~1~~ 00
AID_Length_8	A0 00 00 06 00 02~~1~~ 00 01
AID_Length_9	A0 00 00 06 00 02~~1~~ 00 01 EE
AID_Length_10	A0 00 00 06 00 02~~1~~ 00 01 EE 05
AID_Length_11	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15
AID_Length_12	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15 01
AID_Length_13	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15 01 01
AID_Length_14	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15 01 01 01
AID_Length_15	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15 01 01 01 01
AID_Length_16	A0 00 00 06 00 02~~1~~ 00 01 EE 05 15 01 01 01 01 01
AID_Partial_SW6280	A0 00 00 06 00 01 00 01 EE 05 0F
AID_Partial_SW6280_instance_1	<AID_Partial_SW6280> 01
AID_Partial_SW6280_instance_2	<AID_Partial_SW6280> 02
AID_Partial_SW6283	A0 00 00 06 00 01 00 01 EE 05 10
AID_TestApp_SW61xx	A0 00 00 06 00 01 00 01 EE 05 11
AID_Partial_SW6283_instance_1	<AID_Partial_SW6283> 01
AID_Partial_SW6283_instance_2	<AID_Partial_SW6283> 02
AID_TestApp_multiselecteable	A0 00 00 06 00 01 00 01 EE 55 01
AID_accessdenied	A0 00 00 06 00 01 00 01 EE 05 FE
AID_nonexisting	A0 00 00 06 00 01 00 01 EE 05 FF
AID_illegal_1	A0 00 00 06
AID_illegal_2	A0 00 00 06 00 01 00 01 EE 10 00 10 00 60 00 00 0A
AID_TestApp_Multi_SW61xx	A0 00 00 06 00 01 00 01 EE 05 12
AID_TestApp_Get_Response	A0 00 00 06 00 01 00 01 EE 05 13
AID_TestApp_Case4_SWwarning	A0 00 00 06 00 01 00 01 EE 05 14
AID_TestApp_Case4_SWwarning_nodata	A0 00 00 06 00 01 00 01 EE 56 01
AID_TestApp_SW6280_selectresponse_etsi	A0 00 00 06 00 01 00 01 EE 56 03
AID_TestApp_SW6283_selectresponse_etsi	A0 00 00 06 00 01 00 01 EE 56 04
AID_TestApp_SW6310_selectresponse_etsi	A0 00 00 06 00 01 00 01 EE 56 05

AID_TestApp_SW63C1_selectresponse_etsi	A0 00 00 06 00 01 00 01 EE 56 06
AID_TestApp_p1p2_etsi	A0 00 00 06 00 01 00 01 EE 56 02

Table 1: Used AIDs

2.5 Test case 6.1.6 Int getVersion()

Rationale of the errata: the Int getVersion() method was removed from the core specification through an Errata. The related test case is deleted.

6.1.6 Method Int getVersion()

VOID

~~***(a) Conformance Requirements***~~

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

~~`Int getVersion()`~~

~~Normal execution~~

~~CRN1: Returns the version of the Open Mobile API Specification this implementation is based on.~~

~~Parameter errors~~

~~None~~

~~Context errors~~

~~None~~

~~***(b) Initial Conditions***~~

~~SEService Object has been created and the isConnected() method has been called and has returned true.~~

~~***(c) Mapping to procedural interface***~~

~~No specific mapping information~~

~~***(d) Test Procedure***~~

~~Test case~~					
~~ID~~	~~API Description~~	~~ISO Command Expectation DUT → UICC Simulator / SE~~	~~ISO Response UICC Simulator / SE → DUT~~	~~API Expectation~~	~~CRR~~
~~1~~	~~getVersion returns version string~~				
	~~1. seService.getVersion( )~~	~~None~~	~~None~~	~~1. returns an Integer that contains the Open Mobile API version the DUT implementation is based on (e.g. 3002)~~	~~CRN1~~

2.6 Test case 6.4.7 openLogicalChannel() ID18

Rationale of the errata: change the title of the test case and the length of the first AID used in the test case. It applies to 6.4.10 ID21 also.

Test case					
ID	**API Description**	**ISO Command Expectation DUT -> UICC Simulator / SE**	**ISO Response UICC Simulator / SE -> DUT**	**API Expectation**	**CRR**
18	**Open a logical channel with different AID lengths from ~~5~~6 bytes till 16 bytes and check, if the selected SE applet answers**				
	From AID_Length_~~6~~5 to AID_Length_16 perform the following steps:				CRN1
	1. session.openLogicalChannel (AID_Length_X);	CMD 1-1: APDU_MANAGE_CH_OPEN CMD 1-2: APDU_SELECT_BY_DF – CLA contains the Channel Number returned by the card in RESP 1-1; Data = 'AID_Length_X'	RESP 1-1: R-APDU - Data: Channel Number; SW '90 00' RESP 1-2: R-APDU - SW '90 00'	1. Returned Channel object is not null. No exception is expected.	
	2. channel.transmit(Test_APDU1)	CMD 2: C-APDU ('XX 10 01 00 04 01 02 03 04 00')	RESP 2: R-APDU - Data = '01 02 03 04'; SW '90 00'	2. Returned Response equals to 'R-APDU' - Data = '01 02 03 04'; SW '90 00'. No exception is expected.	
	3. channel.close()	CMD 3: MANAGE CHANNEL (P1='80')	RESP 3: R-APDU - SW '90 00' '	3. No exception is expected.	

2.7 Test case 6.4.7 openLogicalChannel() ID19

Rationale of the errata: change the applet used in the test case. This change applies to 6.4.10 ID22 also.

Test case					
ID	**API Description**	**ISO Command Expectation DUT -> UICC Simulator / SE**	**ISO Response UICC Simulator / SE -> DUT**	**API Expectation**	**CRR**
19	**Open a logical channel and check – expectDataWithWarningSW is set to "false"**				
	1. session.openLogicalChannel (AID_TestApp_Case4_SWwarning);	CMD 1-1: APDU_MANAGE_CH_OPEN CMD 1-2: APDU_SELECT_BY_DF – CLA contains the Channel Number returned by the card in RESP 1-1;; Data = 'AID_TestApp_Case4_SWwarning'	RESP 1-1: R-APDU - Data: Channel Number; SW '90 00' RESP 1-2: R-APDU - SW '90 00'	1. Returned Channel object is not null. No exception is expected.	CRN9

	2. Channel.transmit(APDU_case4_SW warning); P1 = 0x03	CMD 2: C-APDU ('XX 11 03 00 FF' <Data field of 255 bytes> FF) No GET RESPONSE is sent	RESP 2: R-APDU – SW '62 80'	2. Returned Response equals to 'R-APDU' - SW '62 80'. No exception is expected.	

2.8 Test cases 6.5.4 getSelectResponse()

Rationale of the errata: change the applet used for some of the test cases; clarify in the initial conditions which method shall be used to open the channel to the specific applet and correct some expected results.

(a) Conformance Requirements

Normal execution
CRN1: Returns the data as received from the application select command inclusively the status word.
CRN2: The returned byte array contains the data bytes in the following order:
[<first data byte>, …, <last data byte>, <sw1>, <sw2>].

(b) Initial Conditions

Test case ID1: A logical channel with "AID_TestApp_selectresponse" is already open with openLogicalChannel() method.
Test cases ID2, ID6: A logical channel with "AID_TestApp" is already open with openLogicalChannel() method.
Test case ID3: A logical channel with "null" AID is already open with openLogicalChannel() method.
Test case ID4 ~~and ID17~~: A logical channel with "AID_TestApp_SW6283_selectresponse" is already open with openLogicalChannel() method.
Test case ID5 ~~and ID18~~: A logical channel with "AID_TestApp_SW6280_selectresponse" is already open with openLogicalChannel() method.
Test case ID7 ~~and ID19~~: A logical channel with "AID_TestApp_SW6310_selectresponse" is already open with openLogicalChannel() method.
Test case ID8 ~~and ID20~~: A logical channel with "AID_TestApp_SW63C1_selectresponse" is already open with openLogicalChannel() method.
Test case ID9: A logical channel with "AID_TestApp_selectresponse" is opened with openLogicalChannel(P2=00) method ~~using APDU_SELECT_BY_DF_P2 with P2 set to 00~~.
Test case ID10: A logical channel with "AID_TestApp_selectresponse" is opened with openLogicalChannel(P2=04) method ~~using APDU_SELECT_BY_DF_P2 with P2 set to 04~~.
Test case ID11: A logical channel with "AID_TestApp_selectresponse" is opened with openLogicalChannel(P2=08) method ~~using APDU_SELECT_BY_DF_P2 with P2 set to 08~~.
Test case ID12: A logical channel with "AID_TestApp_selectresponse" is opened with openLogicalChannel(P2=0C) method ~~using APDU_SELECT_BY_DF_P2 with P2 set to 0C~~.
Test case ID13: A logical channel with "AID_TestApp_SW6283" is already open with openLogicalChannel() method.
Test case ID14: A logical channel with "AID_TestApp_SW6280" is already open with openLogicalChannel() method.
Test case ID15: A logical channel with "AID_TestApp_SW6310" is already open with openLogicalChannel() method.
Test case ID16: A logical channel with "AID_TestApp_SW63C1" is already open with openLogicalChannel() method.

Test case ID17: A logical channel with "AID_TestApp_SW6283_selectresponse_etsi" is already open with openLogicalChannel() method.
Test case ID18: A logical channel with "AID_TestApp_SW6280_selectresponse_etsi" is already open with openLogicalChannel() method.
Test case ID19: A logical channel with "AID_TestApp_SW6310_selectresponse_etsi" is already open with openLogicalChannel() method.
Test case ID20: A logical channel with "AID_TestApp_SW63C1_selectresponse_etsi" is already open with openLogicalChannel() method.
Test case ID21: A logical channel with "AID_TestApp_SW6283_selectresponse_etsi" is opened with openLogicalChannel(P2=00) method.
Test case ID22: A logical channel with "AID_TestApp_SW6280_selectresponse_etsi" is opened with openLogicalChannel(P2=00) method.
Test case ID23: A logical channel with "AID_TestApp_SW6310_selectresponse_etsi" is opened with openLogicalChannel(P2=00) method.
Test case ID24: A logical channel with "AID_TestApp_SW63C1_selectresponse_etsi is opened with openLogicalChannel(P2=00) method.
Test case ID25: A logical channel with "AID_TestApp_SW6283" is opened with openLogicalChannel(P2=00) method~~using APDU_SELECT_BY_DF_P2 with P2 set to 00~~.
Test case ID26: A logical channel with "AID_TestApp_SW6280" is opened with openLogicalChannel(P2=00) method~~using APDU_SELECT_BY_DF_P2 with P2 set to 00~~.
Test case ID27: A logical channel with "AID_TestApp_SW6310" is opened with openLogicalChannel(P2=00) method~~using APDU_SELECT_BY_DF_P2 with P2 set to 00~~.
Test case ID28: A logical channel with "AID_TestApp_SW63C1" is opened with openLogicalChannel(P2=00) method~~using APDU_SELECT_BY_DF_P2 with P2 set to 00~~.
Test case ~~ID21 and~~ ID29: A logical channel with "AID_TestApp_SW6283_selectresponse" is opened using APDU_SELECT_BY_DF_P2 with P2 set to 00.
Test case ~~ID22 and~~ ID30: A logical channel with "AID_TestApp_SW6280_selectresponse" is opened using APDU_SELECT_BY_DF_P2 with P2 set to 00.
Test case ~~ID23 and~~ ID31: A logical channel with "AID_TestApp_SW6310_selectresponse" is opened using APDU_SELECT_BY_DF_P2 with P2 set to 00.
Test case ~~ID24 and~~ ID32: A logical channel with "AID_TestApp_SW63C1_selectresponse" is opened using APDU_SELECT_BY_DF_P2 with P2 set to 00.

Test case					
ID	**API Description**	**ISO Command Expectation DUT → UICC Simulator/SE**	**UICC Simulator - ISO Response UICC Simulator/SE → DUT**	**API Expectation**	**CRR**
22	**Check the handset correctly handles the select application command when the status word is 6280 and P2 is set to 00 – ETSI behavior**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~04,~~62, 80}	CRN1, CRN2
23	**Check the handset correctly handles the select application command when the status word is 6310 and P2 is set to 00 – ETSI behavior**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~08,~~63, 10}	CRN1, CRN2

24	**Check the handset correctly handles the select application command when the status word is 63C1 and P2 is set to 00 – ETSI behavior**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~0C,~~ 63, C1}	CRN1, CRN2
30	**Check the handset correctly handles the select application command when the status word is 6280 and P2 is set to 00**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~04,~~ 62, 80}	CRN1, CRN2
31	**Check the handset correctly handles the select application command when the status word is 6310 and P2 is set to 00**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~08,~~ 63, 10}	CRN1, CRN2
32	**Check the handset correctly handles the select application command when the status word is 63C1 and P2 is set to 00**				
	1. Channel.getSelectResponse()	CMD 1-1: None	RESP 1-1: None	1. byte[ ]= { DE, AD, C0, DE, ~~0C,~~ 63, C1}	CRN1, CRN2

2.9 Test cases 6.5.6 transmit()

Rationale of the errata: change the applet used for some of the test cases and change some test case title.

(a) Initial Conditions

Test case ID30-ID33~~ and ID36-ID39~~: A logical channel with “AID_ TestApp_Case4_SWwarning” is already open.

Test case ID34: A logical channel with “AID_TestApp_p1p2_etsi” is already open. The value of the expectDataWithWarningSW attribute of this channel object is set to “true”.

Test case ~~ID34,~~ ID35: A logical channel with “AID_TestApp_p1p2” is already open. The value of the expectDataWithWarningSW attribute of this channel object is set to “true”.

Test case ID36, ID37, ID38, ID39: A logical channel with “AID_~~–~~TestApp_Case4_SWwarning_nodata” is already open. The value of the~~-~~ expectDataWithWarningSW attribute of this channel object is set to “true”.

Test case					
ID	**API Description**	**ISO Command Expectation DUT → UICC Simulator/SE**	**UICC Simulator - ISO Response UICC Simulator/SE → DUT**	**API Expectation**	**CRR**
30	**Handling the response to case4 command with SW warning "6280" ~~and no data~~**				
	1. Channel.transmit(APDU_case4_SWwarning); P1 = 0x03	CMD 1-1: C-APDU ('XX 11 03 00 FF' <Data field of 255 bytes> FF) No get response command is sent by the modem	RESP 1-1: R-APDU – SW '62 80'	`1. byte[ ]= {62, 80}`	CRN7
31	**Handling the response to case4 command with SW warning "6283" ~~and no data~~**				
	1. Channel.transmit(APDU_case4_SWwarning); P1 = 0x06	CMD 1-1: C-APDU ('XX 11 06 00 FF' <Data field of 255 bytes> FF) No get response command is sent by the modem	RESP 1-1: R-APDU – SW '62 83'	`1. byte[ ]= {62, 83}`	CRN7
32	**Handling the response to case4 command with SW warning "6310" ~~and no data~~**				
	1. Channel.transmit(APDU_case4_SWwarning); P1 = 0x0E	CMD 1-1: C-APDU ('XX 11 0E 00 FF' <Data field of 255 bytes> FF) No get response command is sent by the modem	RESP 1-1: R-APDU – SW '63 10'	`1. byte[ ]= {63, 10}`	CRN7
33	**Handling the response to case4 command with SW warning "63C2" ~~and no data~~**				
	1. Channel.transmit(APDU_case4_SWwarning); P1 = 0x0F	CMD 1-1: C-APDU ('XX 11 0F 00 FF' <Data field of 255 bytes> FF) No get response command is sent by the modem	RESP 1-1: R-APDU – SW '63 C2'	`1. byte[ ]= {63, C2}`	CRN7

2.10 Test cases 6.5.7 selectNext()

Rationale of the errata: 6.5.7 ID6a and 6b were removed from the test specification, because there is no real use case for them on the field for multi-application UICC cards.

(a) Conformance Requirements

Context errors
CRC1: IOError - if there is a communication problem to the reader or the SE.
~~CRC2: OperationNotSupportedError - if this operation is not supported by the card.~~

(b) Initial Conditions

~~Test case ID6a, ID6b: the SE indicates in the historical byte T3 of the ATR that the partial DF selection is not supported.~~

(d) Test Procedure

Test case					
ID	**API Description**	**ISO Command Expectation DUT → UICC Simulator/SE**	**UICC Simulator - ISO Response UICC Simulator/SE → DUT**	**API Expectation**	**CRR**
6a	~~Operation not supported by the Secure Element, DUT does not rely on ATR~~VOID				
	~~1. Channel.selectNext( );~~	~~CMD 1-1: APDU_SELECT_BY_DF – CLA with Channel Number =1 ; P2='02' (Next occurrence); Data = 'AID_Partial_1'~~	~~RESP 1-1: R-APDU – SW '6A 81'~~	~~1. OperationNotSupportedError~~	~~CRC2~~
6b	~~Operation not supported by the Secure Element, DUT relies on ATR~~VOID				
	~~1. Channel.selectNext( );~~	~~None~~	~~None~~	~~1. OperationNotSupportedError~~	~~CRC2~~

2.11 Annex B

Rationale of the errata: ARA applet will not be provided for the test specification on the SIMalliance website. The PKCS15 description is updated with Applet AIDs and EF DIR.

Access Control Applet (ARA)

A simple ARA applet provides the access rules to the Enforcer application in the mobile. ~~This will be provided on the SIMalliance website.~~ According to these access rules, the Enforcer will decide whether to allow access to any applet instance or not (see GP SEAC specification).

Access Control File System (ARF)

~~Additionally a~~A PKCS#15 file structure is provided with the access rules. Here it is described following PKCS#15 examples in GP SEAC specification (see also PKCS#15 v1.1 spec):

PKCS#15 file system

```
MF (3F00)
|- EF DIR (2F00) --> shall reference PKCS-15
|
|- DF PKCS-15 (7F50)
   |
   |- ODF (5031) --> shall reference DODF
   |- DODF (5207) --> shall reference EF ACMain
   |- EF ACMain (4200) --> shall reference EF ACRules
   |- EF ACRules (4300) --> shall reference EF ACConditions files
   |- EF ACConditions1 (4310)
   |- EF ACConditions2 (4311)
   |- EF ACConditions3 (4312)
```

The following file identifiers are decided by the application issuer: PKCS-15, DODF, ACMain, ACConditions,..

EF DIR:
61 14 4F 0C A0 00 00 00 63 50 4B 43 53 2D 31 35 51 04 3F 00 7F 50

ODF:
A7 06 30 04 04 02 52 07

DODF:
A1 29 30 00 30 0F 0C 0D 47 50 20 53 45 20 41 63 63 20 43 74 6C A1 14 30 12 06 0A 2A 86 48 86 FC 6B 81 48 01 01 30 04 04 02 42 00

ACMain:
30 10 04 08 01 02 03 04 05 06 07 08 30 04 04 02 43 00

ACRules:
30 15 A0 0D 04 XX XX XX XX .. 0B A0 00 00 06 00 01 00 01 EE 05 FE 30 04 04 02 43 10
30 15 A0 0D 04 XX XX XX XX .. 0B A0 00 00 06 00 01 00 01 EE 05 01 30 04 04 02 43 11
30 08 82 00 30 04 04 02 43 12

ACConditions1:
FF FF

ACConditions2:
30 67
04 00
A0 63
A0 5C
A1 5A
04 08 00 10 01 00 F0 FF FF FF
04 08 00 10 02 00 F0 FF FF FF
04 08 00 30 00 00 F0 FF FF FF
04 08 00 40 00 00 F0 EF FF FF
04 08 00 55 00 00 F0 FF FF FF
04 08 00 A4 00 00 F0 FF FB FF
04 08 00 70 00 00 F0 FF 7F E0
04 08 00 50 00 00 F0 FF FF FF
04 08 00 10 00 00 F0 FF FF FF
A1 03
80 01 00

ACConditions3:
30 00

2.12 Annex F

Rationale of the errata: correct the ID number of 6.4.10.

Test cases where ETSI behavior is expected for case 4 commands with SW warning
6.4.7. Channel openLogicalChannel(byte[] aid) ID19
6.4.10. Channel openLogicalChannel(byte[] aid, Byte P2) ID22~~19~~
6.5.4. byte[] getSelectResponse() ID17 – ID24
6.5.6. byte[] transmit(byte[] command) ID34

Table 2: Test cases where ETSI behavior is expected for case 4 commands with SW warning