ETSI TS 103 484-1 V9.0.0 (2013-05)

Smart Cards;
Test specification for the Secure Channel interface;
Part 1: Terminal features
(Release 9)

Reference
DTS/SCP-00SC_test_A2A_ISO-1

Keywords
Smart Card, terminal

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

Contents

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://ipr.etsi.org).

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 ETSI Technical Committee Smart Card Platform (SCP).

The contents of the present document are subject to continuing work within TC SCP and may change following formal TC SCP approval. If TC SCP modifies the contents of the present document, it will then be republished by ETSI with an identifying change of release date and an increase in version number as follows:

Version x.y.z

where:

x the first digit:

0 early working draft;

1 presented to TC SCP for information;

2 presented to TC SCP for approval;

3 or greater indicates TC SCP 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.

The present document is part 1 of a multi-part deliverable covering the Test specification for the Secure Channel interface, as identified below:

Part 1: "Terminal features";

Part 2: "UICC features".

Introduction

The present document defines test cases for the terminal relating to the Secure Channel interface, as specified in TS 102 484 [1] and TS 102 221 [2].

The aim of the present document is to ensure interoperability between the terminal and the UICC independently of the respective manufacturer, card issuer or operator.

TS 102 484 [1] details four types of secure channel:

- TLS- Application to Application.
- Secured APDU - Application to Application.

- IPsec - USB Class to USB Class.
- Secured APDU - Platform to Platform.

TS 102 484 [1] also defines 4 types of key agreement mechanism:

- Strong Pre-shared Keys - GBA.
- Strong Pre-shared Keys - Proprietary Pre-Shared Keys.
- Weak Pre-shared Keys - Proprietary Pre-Shared Keys.
- Certificate Exchange.

The present document may be used to test either:

- Terminal Capability - the terminal support for an application that implements the TS 102 484 [1] secure channel specification.
- Terminal Application - a terminal and application that implements the TS 102 484 [1] secure channel specification.

1 Scope

The present document covers the minimum characteristics which are considered necessary for the terminal or terminal and terminal application in order to provide compliance to TS 102 484 [1].

The present document specifies the test cases for the Secured APDU - Application to Application type of secure channel and includes tests for:

- the characteristics of the Secure Channel interface between the UICC and the UICC-enabled terminal;
- the Discovery and Channel Administration;
- Key Agreement for Strong Pre-shared Keys - Proprietary Pre-Shared Keys;
- the Channel Operation between the UICC-enabled terminal and the UICC.

Both tests for Terminal capability and Terminal applications are specified.

The following are out of scope of this document:

- TLS- Application to Application.
- IPsec - USB Class to USB Class.
- Secured APDU - Platform to Platform.
- Strong Pre-shared Keys - GBA key agreement.
- Weak Pre-shared Keys - Proprietary Pre-Shared Keys key agreement.
- Certificate Exchange key agreement.

2 References

References are either specific (identified by date of publication and/or edition number or version number) or non-specific. For specific references, only the cited version applies. For non-specific references, the latest version of the reference document (including any amendments) applies.

- In the case of a reference to a TC SCP document, a non specific reference implicitly refers to the latest version of that document in the same Release as the present document.

Referenced documents which are not found to be publicly available in the expected location might be found at http://docbox.etsi.org/Reference.

NOTE: While any hyperlinks included in this clause were valid at the time of publication ETSI cannot guarantee their long term validity.

2.1 Normative references

The following referenced documents are necessary for the application of the present document.

[1] ETSI TS 102 484: "Smart Cards; Secure channel between a UICC and an end-point terminal".

[2] ETSI TS 102 221: "Smart Cards; UICC-Terminal interface; Physical and logical characteristics".

[3] ETSI TS 133 110: "Universal Mobile Telecommunications System (UMTS); LTE; Key establishment between a Universal Integrated Circuit Card (UICC) and a terminal (3GPP TS 33.110)".

[4] ETSI TS 102 223: "Smart Cards; Card Application Toolkit (CAT)".

[5] ETSI TS 124 008: "Digital cellular telecommunications system (Phase 2+); Universal Mobile Telecommunications System (UMTS); LTE; Mobile radio interface Layer 3 specification; Core network protocols; Stage 3 (3GPP TS 24.008)".

[6] IETF RFC 4634 (2006): "US Secure Hash Algorithms (SHA and HMAC-SHA)".

[7] IETF RFC 2104 (1997): "HMAC: Keyed-Hashing for Message Authentication".

[8] FIPS PUB 180-2: "Secure Hash Standard (SHS)".

[9] ETSI TS 102 225 (V7.3.0): "Smart Cards; Secured packet structure for UICC based applications (Release 7)".

[10] ETSI TS 102 600: "Smart Cards; UICC-Terminal interface; Characteristics of the USB interface".

[11] ISO/IEC 9797-1: "Information technology - Security techniques - Message Authentication Codes (MACs) - Part 1: Mechanisms using a block cipher".

[12] ETSI TS 102 483: "Smart cards; UICC-Terminal interface; Internet Protocol connectivity between UICC and terminal".

[13] ETSI TS 102 613: "Smart Cards; UICC - Contactless Front-end (CLF) Interface; Part 1: Physical and data link layer characteristics".

[14] ISO/IEC 9646-7: "Information technology -- Open Systems Interconnection -- Conformance testing methodology and framework -- Part 7: Implementation Conformance Statements".

[15] ETSI TS 101 220: "Smart Cards; ETSI numbering system for telecommunication application providers".

[16] IETF RFC 4306: "Internet Key Exchange (IKEv2) Protocol".

2.2 Informative references

The following referenced documents are not necessary for the application of the present document but they assist the user with regard to a particular subject area.

Not applicable.

3 Definitions, symbols and abbreviations

3.1 Definitions

For the purposes of the present document, the terms and definitions given in TS 102 484 [1] and TS 102 221 [2] apply.

3.2 Symbols

For the purposes of the present document, the symbols given in TS 102 484 [1] apply.

3.3 Abbreviations

For the purposes of the present document, the abbreviations given in TS 102 484 [1] and the following apply:

DUT	Device Under Test
HEX	HEXadecimal
RQ	Conformance requirement
T	Terminal
TCK	ChecK character

3.4 Formats

3.4.1 Format of the table of optional features

The columns in table 4.1 have the following meaning:

Column	Meaning
Option:	The optional feature supported or not by the implementation.
Status:	See chapter 3.4.3 'Status and Notations'.
Support:	The support columns are to be filled in by the supplier of the implementation. The following common notations, defined in ISO/IEC 9646-7 [14], are used for the support column in table 4.1. Y or y supported by the implementation. N or n not supported by the implementation. N/A, n/a or - no answer required (allowed only if the status is N/A, directly or after evaluation of a conditional status).
Mnemonic:	The mnemonic column contains mnemonic identifiers for each item.

3.4.2 Format of the applicability table

The applicability of every test in table 4.2 a) is formally expressed by the use of Boolean expression defined in the following clause.

The columns in table 4.2 have the following meaning:

Column	Meaning
Test case:	The "Test case" column gives a reference to the test case number(s) detailed in the present document and required to validate the implementation of the corresponding item in the "Description" column.
Description:	In the "Description" column a short non-exhaustive description of the requirement is found.
Release:	The "Release" column gives the Release applicable and onwards, for the item in the "Description" column.
Rel-x Terminal:	For a given Release, the corresponding "Rel-x Terminal" column lists the tests required for a Terminal to be declared compliant to this Release.
Support:	The "Support" column is blank in the proforma, and is to be completed by the manufacturer in respect of each particular requirement to indicate the choices, which have been made in the implementation.

3.4.3 Status and Notations

The "Rel-x Terminal" columns show the status of the entries as follows:

The following notations, defined in ISO/IEC 9646-7 [14], are used for the status column:

M mandatory - the capability is required to be supported.

O optional - the capability may be supported or not.

N/A not applicable - in the given context, it is impossible to use the capability.

X prohibited (excluded) - there is a requirement not to use this capability in the given context.

O.i qualified optional - for mutually exclusive or selectable options from a set. "i" is an integer which identifies an unique group of related optional items and the logic of their selection which is defined immediately following the table.

Ci conditional - the requirement on the capability ("M", "O", "X" or "N/A") depends on the support of other optional or conditional items. "i" is an integer identifying an unique conditional status expression which is defined immediately following the table. For nested conditional expressions, the syntax "IF ... THEN (IF ... THEN ... ELSE...) ELSE ..." is to be used to avoid ambiguities.

References to items

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

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

4 Test environment

4.1 Table of optional features

The supplier of the implementation shall state the support of possible options in table 4.1. See clause 3.4 for the format of table 4.1.

Table 4.1: Options

Item	Option	Status	Support	Mnemonic
1	Terminal Capability - the terminal support for an application that implements the TS 102 484 [1] secure channel specification.	C1/1		C_Term_capability
2	Terminal Application - a terminal and application that implements the TS 102 484 [1] secure channel specification.	C1/2		C_Term_and_App
3	key agreement	O		O_ KeyAgreement
4	The terminal application uses Manage Secure Channel APDU - Retrieve UICC Endpoints command to discover UICC endpoints	O		O_RetrieveEndpoints
5	Terminal Secure Channel Master SA setup using pre-shared keys	O		O_PresharedKeys

4.2 Applicability table

Table 4.2.1 specifies the applicability of each test case to the device under test. See clause 3.4 for the format of table 4.2.1.

Table 4.2.1: Applicability of tests

Test case	Description	Release	Rel-7 Terminal	Rel-8 Terminal	Rel-9 Terminal	Support
	Discovery					
6.1.1.1	ATR		M	M	M	
	Channel Administration					
6.2.1.1	Retrieve UICC Endpoints - No Endpoints	Rel-7	C002	C002	C002	
6.2.1.2	Retrieve UICC Endpoints - Single Endpoint	Rel-7	C002	C002	C002	
6.2.1.3	Retrieve UICC Endpoints - Multiple Endpoints	Rel-7	C002	C002	C002	
6.2.1.4	Retrieve UICC Endpoints - Multiple Endpoints Transferred in Blocks	Rel-7	C002	C002	C002	
	Manage Secure Channel - Establish SA - Master SA					
6.2.2.1	Establish Master SA	Rel-7	M	M	M	
6.2.2.2	Establish Master SA UICC Rejects	Rel-7	M	M	M	
6.2.2.3	Establish Master SA - 4 Master SAs	Rel-7	C001	C001	C001	
	Manage Secure Channel - Establish SA - Connection SA					
6.2.3.1	Establish Connection SA	Rel-7	M	M	M	
6.2.3.2	Establish Connection SA (Incorrect CSAMAC)	Rel-7	M	M	M	
6.2.3.3	Establish Connection SA - 4 Connection SAs	Rel-7	C001	C001	C001	
	Manage Secure Channel - Establish SA - Start Secure Channel					
6.2.4.1	Start Secure Channel	Rel-7	M	M	M	
6.2.4.2	Start Secure Channel - UICC Error	Rel-7	M	M	M	
	Manage Secure Channel - Terminate Secure Channel					
6.2.5.1	Terminate Secure Channel - Master SA	Rel-7	M	M	M	
6.2.5.2	Terminate Secure Channel - Connection SA	Rel-7	M	M	M	
6.2.5.3	Suspend and resume Secure channel - Terminal application expires	Rel-7	C001	C001	C001	
6.2.5.4	Suspend and resume Secure Channel - UICC application expires	Rel-7	M	M	M	
	Transact Data - Command Data					
6.4.1.1	Transact Data - Command Data in 1 secure channel TLV	Rel-7	M	M	M	
6.4.1.2	Transact Data - Command Data in 2 secure channel TLVs	Rel-7	M	M	M	
6.4.1.3	Transact Data - Command Data in 25 secure channel TLVs	Rel-7	M	M	M	
6.4.1.4	Transact Data - Command Data Maximum Size APDU	Rel-7	M	M	M	
	Transact Data - Response Data					
6.4.2.1	Transact Data - Response Data in 1 secure channel TLV	Rel-7	M	M	M	
6.4.2.2	Transact Data - Response Data in 2 secure channel TLVs	Rel-7	M	M	M	
6.4.2.3	Transact Data - Response Data in 25 secure channel TLVs	Rel-7	M	M	M	
	Retransmission					
6.4.3.1	Transact Data using resend mechanism	Rel-7	M	M	M	
	Transact Data - Multiple Secure Channels					

Test case	Description	Release	Rel-7 Terminal	Rel-8 Terminal	Rel-9 Terminal	Support
6.4.4.1	Transact Data - Multiple secure channels in 1 message	Rel-7	M	M	M	
6.4.4.2	Transact Data - Multiple secure channels with different secure channel block sizes	Rel-7	M	M	M	

Table 4.2.2: Conditional items referenced by table 4.2.1

Conditional item	Condition
C001	IF C_Term_and_App THEN M ELSE N/A (see note)
C002	IF O_RetrieveEndpoints THEN M ELSE N/A
NOTE: The clauses are optional as the terminal application is not required to support all of the options that the terminal has to.	

4.3 Information provided by the device supplier

Void.

4.4 Test equipment

The test equipment shall provide a UICC simulator which is connected to the DUT during test procedure execution, unless otherwise specified.

With respect to the Terminal, the UICC simulator shall act as a valid UICC according to TS 102 484 [1] and TS 102 221 [2], unless otherwise specified. In particular, during test procedure execution, the UICC simulator shall respect the electrical and signalling conditions for all UICC contacts within the limits given by TS 102 484 [1] and TS 102 221 [2]. The accuracy of the UICC simulator's settings shall be taken into account when ensuring this.

4.4.1 Measurement / setting uncertainties

As the tests being carried out are measured based on the protocol response there are no measurement tolerances specified.

4.4.2 Default conditions for DUT operation

These tests shall be performed on a DUT that supports the TS 102 221 [2] APDU interface. The terminal shall attach to the UICC simulator using the terminals default voltage and speed parameters.

The USB TS 102 600 [10] and SWP interfaces TS 102 613 [13] shall be disabled for these tests.

These tests shall be carried out in ambient environmental conditions.

4.4.3 UICC Simulator requirements

The following are requirements for a UICC simulator to be used to perform the tests specified.

4.4.3.1 General Requirements

REQ_UICC_TEST_GEN_01	The UICC simulator shall be designed to TS 102 484 [1] Release 9 and TS 102 221 [2] Release 9.
REQ_UICC_TEST_GEN_02	UICC simulator shall be capable of communicating with a terminal.
REQ_UICC_TEST_GEN_03	The UICC simulator shall be capable of performing all of the UICC operations for at least 4 Secured APDU - Application to Application secure channels simultaneously.
REQ_UICC_TEST_GEN_04	The UICC simulator shall inform the tester of all communication between the UICC and the terminal.
REQ_UICC_TEST_GEN_05	The Tester shall be able to cause the UICC simulator to perform specific actions required for the tests as detailed below.
REQ_UICC_TEST_GEN_06	The UICC simulator shall display all terminal commands and any data sent, to the tester. The UICC simulator may highlight to the user: • Any variance in the received result to the expected result. • An interpretation of the received data.
REQ_UICC_TEST_GEN_07	The UICC simulator shall be capable of interleaving the test command responses with other UICC none 'secure channel' command responses.

4.4.3.2 MANAGE SECURE CHANNEL - ATR Requirements

REQ_UICC_TEST_ATR_01	The UICC simulator shall allow the tester to set the contents of the ATR.

4.4.3.3 MANAGE SECURE CHANNEL - Retrieve UICC Endpoints Requirements

REQ_UICC_TEST_RUE_01	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data".
REQ_UICC_TEST_RUE_02	If the terminal requests the "First block of response data", the UICC simulator shall be able to send the response data and SW1 SW2.
REQ_UICC_TEST_RUE_03	If terminal requests the "Next block of response data", the UICC simulator shall be able to send the response data and SW1 SW2.

4.4.3.4 MANAGE SECURE CHANNEL - Key Agreement Requirements

REQ_UICC_TEST_KEY_01	The UICC simulator shall support the entry of a strong proprietary key to be used in the Secure channel setup and operation.
REQ_UICC_TEST_KEY_02	The tester shall be able to set the key lifetime as either a counter value or as a time value.

4.4.3.5 MANAGE SECURE CHANNEL - Establish SA - Master SA Requirements

REQ_UICC_TEST_MSA_01	The UICC simulator shall support the reception of the MANAGE SECURE CHANNEL - Establish SA - Master SA without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints to be issued.
REQ_UICC_TEST_MSA_02	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_MSA_03	If the terminal requests the "First block of response data" then the UICC simulator shall respond with the response data, SW1 and SW2".
REQ_UICC_TEST_MSA_04	The terminal test application shall support the responding to the MANAGE SECURE CHANNEL - Establish SA - Master SA multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.3.6 MANAGE SECURE CHANNEL - Establish SA - Connection SA Requirements

REQ_UICC_TEST_CSA_01	The UICC simulator shall support the reception of the MANAGE SECURE CHANNEL - Establish SA - Connection SA without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints or MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued.
REQ_UICC_TEST_CSA_02	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_CSA_03	If the terminal requests the "First block of response data" then the UICC simulator shall respond with the response data, SW1 and SW2".
REQ_UICC_TEST_CSA_04	The UICC simulator shall support the responding to the MANAGE SECURE CHANNEL - Establish SA - Connection SA multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.3.7 MANAGE SECURE CHANNEL - Establish SA - Start Secure Channel Requirements

REQ_UICC_TEST_SSC_01	The UICC simulator shall support the reception of the MANAGE SECURE CHANNEL - Establish SA - Start secure channel without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints, MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued or MANAGE SECURE CHANNEL - Establish SA - Connection SA.
REQ_UICC_TEST_SSC_02	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_SSC_03	If the terminal requests the "First block of response data" then the UICC simulator shall respond with the response data, SW1 and SW2".
REQ_UICC_TEST_SSC_04	The UICC simulator shall support the responding the MANAGE SECURE CHANNEL - Establish SA - Start secure channel multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.3.8 MANAGE SECURE CHANNEL - Terminate Secure Channel Requirements

REQ_UICC_TEST_TSC_01	The UICC simulator shall support the reception of the MANAGE SECURE CHANNEL - Terminate secure channel without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints, MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued, MANAGE SECURE CHANNEL - Establish SA - Connection SA or MANAGE SECURE CHANNEL - Establish SA - Start secure channel.
REQ_UICC_TEST_TSC_02	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_TSC_03	The UICC simulator shall support the response to the MANAGE SECURE CHANNEL - Establish SA - Terminate secure channel multiple times.

4.4.3.9 TRANSACT DATA Requirements

REQ_UICC_TEST_TRD_01	The UICC simulator shall support the processing of the TRANSACT DATA command regardless as to whether any MANAGE SECURE CHANNEL commands have been issued and regardless of the state of the secure channel being used.
REQ_UICC_TEST_TRD_02	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "First block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_TRD_03	At the request of the tester, the UICC simulator shall be able to interpret and respond to the "Next block of command data". The SW1 and SW2 response shall be set by the tester.
REQ_UICC_TEST_TRD_04	If the terminal requests the "First block of response data" then the UICC simulator shall respond with the response data, SW1 and SW2".
REQ_UICC_TEST_TRD_05	If the terminal requests the "Next block of response data" then the UICC simulator shall respond with the response data, SW1 and SW2".
REQ_UICC_TEST_TRD_06	If the SW1 SW2 response to this command from the UICC is '62 F3'and there is no more data to send and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_TRD_02.

4.4.4 Terminal Application Requirements (for Terminal Capability Testing)

The following are requirements for a test terminal application to be used when the test type is 'terminal capability'.

4.4.4.1 General Requirements

REQ_TERM_TEST_GEN_01	The terminal test application shall be designed to TS 102 484 [1] Release 9 and TS 102 221 [2] Release 9.
REQ_TERM_TEST_GEN_02	The terminal test application shall be capable of communicating with a UICC.
REQ_TERM_TEST_GEN_03	The terminal test application shall be capable of performing all of the terminal operations for at least 4 Secured APDU - Application to Application secure channels simultaneously.
REQ_TERM_TEST_GEN_04	The terminal test application shall inform the tester of all communication between the UICC and the test application.
REQ_TERM_TEST_GEN_05	The Tester shall be able to cause the terminal test application to perform specific actions required for the tests as detailed below.
REQ_TERM_TEST_GEN_06	The tester shall be able to choose the logical channel that the test is to be carried out on. If it is not already open, the terminal test application shall negotiate and open the logical channel requested.
REQ_TERM_TEST_GEN_07	The terminal test application shall display all SW1 and SW2 responses and any data returned, to the tester. The terminal test application may highlight to the user: • Any variance in the received result to the expected result. • An interpretation of the received data.
REQ_TERM_TEST_GEN_08	The terminal test application shall be capable of interleaving the test commands with other UICC none 'secure channel' commands.
REQ_TERM_TEST_GEN_09	The terminal test application shall be capable of powering the UICC off and on at any point in a test.
REQ_TERM_TEST_GEN_10	The terminal test application shall be capable of resetting the UICC at any point in a test.

4.4.4.2 MANAGE SECURE CHANNEL - ATR Requirements

REQ_TERM_TEST_ATR_01	The terminal test application shall be able to retrieve the contents of the ATR from the UICC and display it to the tester unmodified.

4.4.4.3 MANAGE SECURE CHANNEL - Retrieve UICC Endpoints Requirements

REQ_TERM_TEST_RUE_01	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The CLA byte shall be set to the logical channel chosen by the tester for this test.
REQ_TERM_TEST_RUE_02	If the SW1 SW2 response to this command from the UICC is '62 F3' and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_RUE_01.
REQ_TERM_TEST_RUE_03	If the response to this message is "More data available" and at the request of the tester, the terminal test application shall request the "Next block of response data" using the same CLA byte as for REQ_TERM_TEST_RUE_01.
REQ_TERM_TEST_RUE_04	If the response to either REQ_TERM_TEST_RUE_02 or REQ_TERM_TEST_RUE_03 is "normal ending of command" then the terminal test application may interpret the endpoint data so that it can be used in later tests.

4.4.4.4 MANAGE SECURE CHANNEL - Key Agreement Requirements

REQ_TERM_TEST_KEY_01	The terminal test application shall support the entry of a strong proprietary key to be used in the Secure channel setup and operation.
REQ_TERM_TEST_KEY_02	The tester shall be able to set the key lifetime as either a counter value or as a time value.

4.4.4.5 MANAGE SECURE CHANNEL - Establish SA - Master SA Requirements

REQ_TERM_TEST_MSA_01	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Master SA without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints to be issued.
REQ_TERM_TEST_MSA_02	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The Key Agreement Mechanism value, UICC_ID, UICC_AID, IMEI and Terminal_application_ID shall be set by the tester.
REQ_TERM_TEST_MSA_03	If the SW1 SW2 response to this command from the UICC is '62 F3' and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_MSA_02.
REQ_TERM_TEST_MSA_04	If the response to either REQ_TERM_TEST_MSA_03 is "normal ending of command" then the terminal test application may interpret the response so that it can be used in later tests.
REQ_TERM_TEST_MSA_05	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Master SA multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.4.6 MANAGE SECURE CHANNEL - Establish SA - Connection SA Requirements

REQ_TERM_TEST_CSA_01	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Connection SA without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints or MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued.
REQ_TERM_TEST_CSA_02	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The Algorithm and integrity TLV value, MSA_ID value and the Tnonce value shall be set by the tester.
REQ_TERM_TEST_CSA_03	If the SW1 SW2 response to this command from the UICC is '62 F3' and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_CSA_02.
REQ_TERM_TEST_CSA_04	If the response to either REQ_TERM_TEST_CSA_03 is "normal ending of command" then the terminal test application may interpret the response so that it can be used in later tests.
REQ_TERM_TEST_CSA_05	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Connection SA multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.4.7 MANAGE SECURE CHANNEL - Establish SA - Start Secure Channel Requirements

REQ_TERM_TEST_SSC_01	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Start secure channel without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints, MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued or MANAGE SECURE CHANNEL - Establish SA - Connection SA.
REQ_TERM_TEST_SSC_02	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The Algorithm and integrity TLV value, CSA_ID value, the SSCMAC value and the Endpoint data container size value shall be set by the tester. The terminal test application can calculate the SSCMAC for the tester.
REQ_TERM_TEST_SSC_03	If the SW1 SW2 response to this command from the UICC is '62 F3' and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_SSC_02.
REQ_TERM_TEST_SSC_04	If the response to either REQ_TERM_TEST_SSC_03 is "normal ending of command" then the terminal test application may interpret the response so that it can be used in later tests.
REQ_TERM_TEST_SSC_05	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Start secure channel multiple times without limit and without requiring the MANAGE SECURE CHANNEL - Terminate secure channel SA to be issued.

4.4.4.8 MANAGE SECURE CHANNEL - Terminate Secure Channel Requirements

REQ_TERM_TEST_TSC_01	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Terminate secure channel without requiring a MANAGE SECURE CHANNEL - Retrieve UICC Endpoints, MANAGE SECURE CHANNEL - Establish SA - Master SA to be issued, MANAGE SECURE CHANNEL - Establish SA - Connection SA or MANAGE SECURE CHANNEL - Establish SA - Start secure channel.
REQ_TERM_TEST_TSC_02	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The MSA_ID value and/or CSA_ID values along with their MAC values shall be set by the tester. The terminal test application can calculate the MAC values for the tester.
REQ_TERM_TEST_TSC_03	The terminal test application shall support the issuing of the MANAGE SECURE CHANNEL - Establish SA - Terminate secure channel multiple times.

4.4.4.9 TRANSACT DATA Requirements

REQ_TERM_TEST_TRD_01	The terminal test application shall support the issuing of the TRANSACT DATA command regardless as to whether any MANAGE SECURE CHANNEL commands have been issued and regardless of the state of the secure channel being used.
REQ_TERM_TEST_TRD_02	At the request of the tester, the terminal test application shall be able to send the "First block of command data". The data used shall be either as directly supplied by the tester or calculated by the terminal test application based on 'clear' data provided by the tester. Where the data is calculated by the terminal test application, the tester shall provide the necessary data for this calculation.
REQ_TERM_TEST_TRD_03	If there is more data to send and regardless of the SW1 SW2 response from the UICC and at the request of the tester, the terminal test application shall send the "Next block of response data" using the same CLA byte as for REQ_TERM_TEST_TRD_02. This step may be repeated until all of the data to be sent in this block has been sent.
REQ_TERM_TEST_TRD_04	The terminal test application shall support the interleaving of TRANSACT DATA commands for different secure channels. The terminal test application shall support these interleaved TRANSACT DATA being a different size for each secure channel.
REQ_TERM_TEST_TRD_05	If the SW1 SW2 response to this command from the UICC is '62 F3'and there is no more data to send and at the request of the tester, the terminal test application shall request the "First block of response data" using the same CLA byte as for REQ_TERM_TEST_TRD_02.
REQ_TERM_TEST_TRD_06	If the SW1 SW2 response to this command from the UICC is '62 and at the request of the tester, the terminal test application shall request the "Next block of response data" using the same CLA byte as for REQ_TERM_TEST_TRD_02. This step may be repeated as required by the tester.

4.4.5 ATRs to be used by the UICC simulator

For some particular test cases, the ATRs described below need to be specifically used.

4.4.5.1 ATR that indicates that indicates support of Secure Channel

Table 4.4.5.1.1

Character	Value	Description
TS	'3B'	Indicates direct convention
T0	'97'	TA1 and TD1 are present 7 historical bytes
TA1	'96'	Clock rate conversion factor FI=9 (F=512) Baud rate adjustment factor DI=6 (D=32)
TD1	'80'	TD2 only is present Protocol T=0 is supported by UICC
TD2	'3F'	TA3 and TB3 are present Global interface bytes are following (T=15)
TA3	'C6'	Clock stop supported (No preferred state) Accepted voltage class B and C
TB3	'88'	Secure Channel supported as defined in TS 102 484 [1]
T1	'80'	
T2	'31'	Card data services
T3	'A0'	SELECTED by AID supported, EFDIR present
T4	'73'	Card capabilities
T5	'BE'	SFI supported
T6	'21'	Data Coding Byte
T7	'00'	No extended Lc and Le No logical channels supported
TCK	'XX'	Check byte

4.4.6 Defined TLVs to be used by the UICC simulator

4.4.6.1 Endpoints to be used by the UICC simulator

For the following endpoints to be used by the UICC simulator the following definitions shall apply:

YY (1 byte): The maximum data container size as detailed in the test procedure or using the default value 'FF'.

UICC_identifier: The Endpoint identifier value of Tag '82' shall be the AID of the application and allocated by the test equipment manufacturer in accordance with TS 101 220 [15]. This may be that required by the Terminal application.

Z: is the length of the Endpoint identifier within the '82'.

4.4.6.1.1 No Endpoints

Table 4.4.6.1.1.1

Tag	Length	Value
'73'	'0C'	UICC_ID TLV as detailed in table 4.4.6.1.1.2

Table 4.4.6.1.1.2

Tag	Length	Value
81	'0A'	ICCID as defined for EF_{ICCID}

4.4.6.1.2 Single Endpoint

Table 4.4.6.1.2.1

Tag	Length	Value
'73'	'XX'	UICC_ID and Endpoint information TLVs as detailed in table 4.4.6.1.2.2

Table 4.4.6.1.2.2

Tag	Length	Value
81	'0A'	ICCID as defined for EF_{ICCID}
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier

4.4.6.1.3 Multiple Endpoints - 4 Endpoints

Table 4.4.6.1.3.1

Tag	Length	Value
'73'	'XX'	UICC_ID and Endpoint information TLVs as detailed in table 4.4.6.1.3.2

Table 4.4.6.1.3.2

Tag	Length	Value
81	'0A'	ICCID as defined for EF_{ICCID}
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier

4.4.6.1.4 Multiple Endpoints - 20 Endpoints over 2 Blocks

4.4.6.1.4.1 Multiple Endpoints 1st Block

Table 4.4.6.1.4.1.1

Tag	Length	Value
'73'	'FF'	The first 255 bytes of UICC_ID and Endpoint information TLVs as detailed in table 4.4.6.1.4.3.1

4.4.6.1.4.2 Multiple Endpoints 2nd Block

Table 4.4.6.1.4.2.1

Tag	Length	Value
'73'	'XX'	Remaining Endpoint information after first 255 bytes TLVs as detailed in table 4.4.6.1.4.3.1

4.4.6.1.4.3 Multiple Endpoints Block greater than 255

Table 4.4.6.1.4.3.1

Tag	Length	Value
81	'0A'	ICCID as defined for EF_{ICCID}
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier
82	7+Z	02 01 04 02 YY FF FF Endpoint_identifier

4.4.6.2 Master SA Response to be used by UICC Simulator

Table 4.4.6.2.1

Tag	Length	Value
'73'	'15'	Key Agreement Mechanism and MSA_ID as detailed in table 4.4.6.2.2

Table 4.4.6.2.2

Tag	Length	Value
87	'01'	'82'
88	'10'	MSA_ID which is a unique 16 byte HEX number that identifies the specific Master_SA and shall be allocated by the test equipment manufacturer

4.4.6.3 Connection SA Response to be used by UICC Simulator

Table 4.4.6.3.2.1

Tag	Length	Value
'73'	'3A'	Algorithm and integrity, CSA_ID, Unonce and CSAMAC as detailed in table 4.4.6.3.2

Table 4.4.6.3.2

Tag	Length	Value
89	'02'	'07 07'
8B	'10'	CSA_ID which is a unique 16 byte HEX number that identifies the specific Connection_SA and shall be allocated by the test equipment manufacturer
8C	'10'	Unonce a randomly generated 16 byte HEX number by the test equipment manufacturer
8F	'10'	CSAMAC truncated to the first 16 bytes and calculated as detailed below

CSAMAC =HMAC-SHA-256(K_MAC, MSA_ID||Tnonce||TSCA||TSIM||CSA_ID||Unonce||UCA||UIM)

4.4.6.4 Start Secure Channel Session Number response

The session number returned in the Establish SA - Start Secure Channel response shall be in accordance with table 4.4.6.4.1 of value as given in the test procedure being used.

Table 4.4.6.4.1

Tag	Length	Value
'53'	'01'	XX000000 XX=Session number as indicated in the test procedure which may be '0', '1', '2' or '3' coded in 2 bits. Value = '00' or '40 or '80' or 'C0'

4.4.6.5 Transact Data Responses

4.4.6.5.1 Transact Data Response 1

The response data shall consist of no data with SW1 SW2 Normal ending of the command bytes encrypted within the TLV.

Table 4.4.6.5.1.1

Data	SW1	SW2
See table 4.4.6.5.1.2	'92'	'00000XX0' XX=Session Number as indicated in the test procedure which may be '0', '1', '2' or '3' coded in 2 bits. Value = '00' or '20 or '40' or '60'

Table 4.4.6.5.1.2

Tag	Length	Value	Padding
80	'81 FC'	The data in table 4.4.6.5.1.3 encrypted using the encryption method and encryption Key agreed for the current secure channel	'00 .. 00' (218 bytes)

Table 4.4.6.5.1.3

Tag	Length	Value
81	'20'	The data in table 4.4.6.5.1.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 4.4.6.5.1.4

Byte	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 20	APDU Command BER-TLV	4	APDU BER TLV - See Table 4.4.6.5.1.5 below
21 to 24	Padding	4	4 byte random number
25 to 32	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel. Calculated as per clause 10.1.1 TS 102 484 [1]

Table 4.4.6.5.1.5

Byte(s)	Description	Length	Value
1	Tag	1	'83'
2	Length	1	'02'
3 to 4	APDU response - SW1 SW2	2	'90 00'

4.4.6.5.2 Transact Data Response 2

The response data shall consist of 210 bytes of repeated data bytes 'A5' with SW1 SW2 Normal ending of the command bytes encrypted within the TLV.

Table 4.4.6.5.2.1

Type	Tag	Length	Value
Encrypted Data Container	81	'81 F0'	The data in table 4.4.6.5.2.2 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 4.4.6.5.2.2

Byte	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 20	APDU Command BER-TLV	215	APDU BER TLV - See table 4.4.6.5.2.3
21 to 24	Padding	1	3 byte random number
25 to 32	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel Calculated as per clause 10.1.1 TS 102 484 [1]

Table 4.4.6.5.2.3

Byte(s)	Description	Length	Value
1	Tag	1	'83'
2 to 3	Length	2	'81 D4'
4 to 213	APDU response - Data	210	'A5..A5'
214 to 215	APDU response - SW1 SW2	2	'90 00'

4.4.6.5.3 Transact Data Response 3

The response data shall consist of 162 bytes of repeated data bytes '40' with SW1 SW2 Normal ending of the command bytes encrypted within the TLV.

Table 4.4.6.5.3

Type	Tag	Length	Value
Encrypted Data Container	81	'81 C0'	The data in table 4.4.6.5.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 4.4.6.5.4

Byte	Description	Length	Description
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 20	APDU Command BER-TLV	167	APDU BER TLV - See table 4.4.6.5.5
21 to 23	Padding	1	1 byte random number
23 to 31	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel Calculated as per clause 10.1.1 TS 102 484 [1]

Table 4.4.6.5.5

Byte(s)	Description	Length	Value
1	Tag	1	'83'
2 to 3	Length	2	'81 D4'
4 to 213	APDU response - Data	162	'40..40
214 to 215	APDU response - SW1 SW2	2	'90 00'

4.4.7 Terminal Application Data

4.4.7.1 TRANSACT DATA commands

Several types of content are required for the TRANSACT DATA tests. The data provided in this section and the responses are the 'clear' data / responses prior to the Secure Channel encryption process.

When testing a specific Terminal application using secure channel, alternative messages may be used that satisfy the criteria in the test sections.

4.4.7.1.1 Transact Data Command APDU 1

This is a message that is 207 bytes long and that produces only a SW1 SW2 response.

Data to Send:

The Terminal application will be implemented to send the following APDU command encrypted and contained in the TRANSACT DATA command, The INS is chosen to be a non-standard APDU which will trigger the UICC simulator to reply.

Table 4.4.7.1.1.1: Coding of Command APDU 1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X'	'EE'	'00'	'00'	'CF'	00 11 22 33 44 55 66 77 88 99 00 11 22 33 44 55 66

4.4.7.1.2 Transact Data Command APDU 2

This is a message that is 255 bytes long and that produces only a SW1 SW2 response.

Data to Send:

The Terminal application will be implemented to send the following APDU command encrypted and contained in the TRANSACT DATA command, The INS is chosen to be a non-standard APDU which will trigger the UICC simulator to reply.

Table 4.4.7.1.2.1: Coding of Command APDU 2

Code	CLA	INS	P1	P2	Le	DATA
Value	'0X'	'EE'	'00'	'00'	'FF'	00 11 22 33 44 55 66 77 88 99 00 11 22 33 44

4.4.7.1.3 Transact Data Command APDU 3

Table 4.4.7.1.3.1: Coding of Command APDU 3

Code	CLA	INS	P1	P2	Le
Value	'0X'	'EE'	'01'	'A5'	'D2'

4.5 Test execution

The tests may be run as a suite or run as individual tests.

4.6 Pass criterion

A test shall only be considered as successful if the test procedure was carried out successfully under all parameter variations with the DUT respecting all conformance requirements referenced in the test procedure.

5 Conformance Requirements

5.1 Secure Channel Properties

Reference: TS 102 484 [1], clause 5.

5.1.1 Secure Channel Lifecycle and Discovery

RQ number	Clause	Description
RQ01_0101	5.1	The lifecycle of each secure channel will include the following step: Discovery of support for secure channels by the terminal and the UICC as detailed in the present document.
RQ01_0102	5.1	The lifecycle of each secure channel will include the following step: Discover endpoints that can communicate securely on the UICC.
RQ01_0103	5.1	The lifecycle of each secure channel will include the following step: Negotiate secure channel parameters.
RQ01_0104	5.1	The lifecycle of each secure channel will include the following step: Create a secure channel.
RQ01_0105	5.1	The lifecycle of each secure channel will include the following step: Communicate over a secure channel.
RQ01_0106	5.1	The lifecycle of each secure channel will include the following step: Suspend and resume a secure channel.
RQ01_0107	5.1	The lifecycle of each secure channel will include the following step: Terminate a secure channel.
RQ01_0201	5.1.1	Support for the mandatory procedures defined in the present document … shall be indicated in the ATR as defined in TS 102 221 [2].

5.1.2 Secure Channel Administration

RQ number	Clause	Description
RQ02_0101	5.1.3	For a secure channel to be setup, both ends of the secure channel must agree on the parameters to be used for this channel. The present document defines these parameters as a "Security Association".
RQ02_0103	5.1.3.1	A Security Association has the following parameter: Identified and authenticated endpoints for both the terminal and the UICC.
RQ02_0104	5.1.3.1	A Security Association has the following parameter: Cryptographic keys.
RQ02_0105	5.1.3.1	A Security Association has the following parameter: Protection algorithms.
RQ02_0106	5.1.3.1	A Security Association has the following parameter: Any additional parameters to be used for securing data transmissions.
RQ02_0107	5.1.3.1	A Security Association has the following parameter: Mechanisms and parameters for identifying secure connections and managing the secure channel.
RQ02_0108	5.1.3.1	The following Security Association is defined in the present document: Master SA.
RQ02_0109	5.1.3.1	The following Security Association is defined in the present document: Connection SA.
RQ02_0110	5.1.3.1	Each secure channel shall have one Master SA and at least one Connection SA.
RQ02_0111	5.1.3.1	The terminal and the UICC shall be able to securely store all of the parameters for a minimum of 4 Master SAs and 4 Connections SAs.
RQ02_0112	5.1.3.1	Master SA and Connection SA - Security Association parameters shall not be visible or editable by any process outside of the present document.
RQ02_0201	5.1.3.2	The Master SA records:Channel endpoints.
RQ02_0202	5.1.3.2	The Master SA records:Master SA identifier.
RQ02_0203	5.1.3.2	The Master SA records:Master SA cryptographic keys (defined as the Master Secret (MS)).
RQ02_0204	5.1.3.2	The Master SA records:The algorithms used to establish secure connections.
RQ02_0205	5.1.3.2	The Master SA records:Expiration information for the Master SA.
RQ02_0206	5.1.3.2	The definition of the Master SA parameters is specific to the type of channel being opened (e.g. Secured APDU - Application to Application).
RQ02_0207	5.1.3.2	A Master SA is specific to the endpoints being used and the type of channel being used. If two endpoints need to communicate over a different secure channel type or a secure channel is required to a different endpoint (even if it is on the same device), then a new Master SA shall be used.
RQ02_0208	5.1.3.2	Master SA's shall exist until they expire or until they are terminated.
RQ02_0209	5.1.3.2	A Master SA is used to setup one or more Connection SAs. This enables the setup of a new Connection SA for a secure channel to be negotiated before the current Connection SA expires.
RQ02_0210	5.1.3.2	A Master SA shall not be used to directly setup a secure channel.
RQ02_0301	5.1.3.3	Each Connection SA contains the operational security parameters for a specific secure channel, these parameters are specific to each secure channel type.
RQ02_0302	5.1.3.3	Connection SAs derive their parameters from a Master SA and have their own lifetime limit.
RQ02_0303	5.1.3.3	Connection SAs shall be active until: The Connection SA is terminated.
RQ02_0304	5.1.3.3	Connection SAs shall be active until: The Master SA that the Connection SA is derived from is terminated.
RQ02_0305	5.1.3.3	Connection SAs shall be active until: The terminal determines that the lifetime of the SA has expired.
RQ02_0306	5.1.3.3	Connection SAs shall be active until: The UICC determines that the Connection SA usage counter has reached its limit.
RQ02_0307	5.1.3.3	It is possible for a secure channel to have more than one active Connection SA, however for security reasons the amount of time that multiple Connection SAs exist should be minimized.

5.1.3 Key Agreement

RQ number	Clause	Description
RQ03_0101	5.1.4	A mechanism to share key material between the two endpoints ... can be: Strong Pre-shared Keys - GBA.
RQ03_0102	5.1.4	A mechanism to share key material between the two endpoints ... can be: Strong Pre-shared Keys - Proprietary Pre-agreed keys.
RQ03_0103	5.1.4	A mechanism to share key material between the two endpoints ... can be: Weak Pre-shared Keys - Proprietary Pre-agreed keys.
RQ03_0104	5.1.4	A mechanism to share key material between the two endpoints ... can be: Certificate exchange.
RQ03_0105	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Ks_local: This is the secret key used to secure the data transmission between the two endpoints.
RQ03_0106	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Terminal_ID: This is a unique identifier for the terminal or device connected to the terminal where the terminal endpoint is. This may be the IMEI of the terminal as defined in TS 124 008 [5].
RQ03_0107	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Terminal_appli_ID: This is a unique identifier for the application that hosts the terminal endpoint. If Ks_local is intended to be used for 'Secured APDU - Platform to Platform' or 'IPsec - USB class to USB class' secure channel types then Terminal_appli_ID shall be set to the ASCII encoded string "platform".
RQ03_0108	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Weak Key: This indicates the strength of Ks_local. Weak Key shall be set to 1 if the pre-shared key is based on a low entropy key (i.e. a key of less than 128 bits of entropy such as a user entered PIN or password), otherwise it shall be set to 0.
RQ03_0109	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Key Lifetime: This is the date and time that the key is valid until.
RQ03_0110	5.1.4	All pre-shared key agreement mechanisms shall produce values for the following parameter: Key Counter Limit (CL): This is the maximum number of times that the key and any derived keys can be used. This is 16 bytes defined as follows: Bytes 1 - 2: Reserved for future use. Bytes 3 - 4: Number of Master SAs that can be created from this pre-shared key. Bytes 5 - 8: Number of Connection SAs that can be derived from each Master SA using this pre-shared key. Bytes 9 - 16: Number of individual secure transactions that can be made before the Connection SA, derived from a Master SA using this pre-shared key, shall expire.
RQ03_0111	5.1.4	The terminal and UICC shall create a value, Ks_Local_Ref, to reference Ks_local where: Ks_Local_Ref = Terminal_ID || Terminal_appli_ID || UICC_ID || UICC_appli_ID.
RQ03_0112	5.1.4	The terminal shall terminate a Master SA if the values ICCID_ID and UICC_appli_ID do not match the intended UICC (and UICC application) that the terminal wishes to securely communicate with.
RQ03_0201	5.1.4.1	A method for establishing a pre-shared key using GBA is defined in TS 133 110 [3].
RQ03_0202	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A 256 bit shared secret key Ks_local.
RQ03_0203	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A 10 byte UICC identifier UICC_ID encoded as for ICCID as defined in TS 102 221 [2].
RQ03_0204	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A 16 byte UICC application identifier UICC_appli_ID (up to 16 bytes).
RQ03_0205	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A 10 byte terminal identifier Terminal_ID encoded using BCD coding as defined in TS 124 008 [5].
RQ03_0206	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A terminal application Identifier Terminal_appli_ID (up to 32 bytes).
RQ03_0207	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A variable length Ks_local Key Lifetime (for use in the terminal).
RQ03_0208	5.1.4.1	The method for establishing a pre-shared key using GBA defined in TS 133 110 [3] shall agree the following value between the UICC and the terminal or connected device: … A 16 byte Ks_local Counter (for use in the UICC).
RQ03_0209	5.1.4.1	For GBA agreed keys, WeakKey shall be set to 0.

RQ number	Clause	Description
RQ03_0210	5.1.4.1	Only one GBA key shall be allowed per individual Ks_Local_Ref.
RQ03_0211	5.1.4.1	If GBA is run again for the same Ks_Local_Ref then the GBA key for that Ks_Local_Ref shall be overwritten by the new key generated.
RQ03_0212	5.1.4.1	If GBA is run again for the same Ks_Local_Ref any Master SA or Connection SAs that were setup using the old key shall be terminated.
RQ03_0301	5.1.4.2	The terminal and UICC may share strong pre-shared keys (with an entropy of 128 bits or greater) using a proprietary mechanism known to both devices.
RQ03_0302	5.1.4.2	The proprietary mechanism used shall agree values for the parameters defined in clause 5.1.4 of TS 102 484 [1].
RQ03_0401	5.1.4.3	The terminal and UICC may share weak pre-shared keys (with an entropy of less than 128 bits) using a proprietary mechanism known to both devices such as password exchange.
RQ03_0402	5.1.4.3	The proprietary mechanism used shall agree values for the parameters defined in clause 5.1.4 of TS 102 484 [1].
RQ03_0403	5.1.4.3	Both the UICC and the terminal shall be able to restrict the use of secure channels that are based on a weak pre-shared key.
RQ03_0501	5.1.4.5	To maintain the security of keys used by secure channels, all used keys shall have a defined lifetime. The lifetime of a key and a counter limit are set as part of the key agreement and is specific to the strength of the key, how many times it can be used and what it is being used for.
RQ03_0502	5.1.4.5	The terminal or connected device shall have a key lifetime for each key.
RQ03_0503	5.1.4.5	The terminal or connected device … may have a counter limit for each key.
RQ03_0504	5.1.4.5	Once either the key lifetime limit or the counter limit is reached for that key, the terminal or connected device shall delete that key.
RQ03_0505	5.1.4.5	Once either the key lifetime limit or the counter limit is reached for that key, the terminal or connected device shall … refuse all transactions based on that key.
RQ03_0506	5.1.4.5	The terminal or connected device shall terminate all security associations that rely on the expired key (including any derived Security Associations).

5.1.5 Secure Channel Operation

RQ number	Clause	Description
RQ04_0102	5.1.6	A secure channel shall be considered 'suspended' if all of the Connection SAs for that secure channel have been terminated.
RQ04_0103	5.1.6	A suspended secure channel shall be resumed when a Connection SA is created using the Master SA for that secure channel.
RQ04_0201	5.2	A terminal or UICC conforming to the present document shall be able to support multiple application to application secure channels.
RQ04_0302	5.3	Once an application to application secure channel is setup, each application that has an endpoint for this channel shall prevent communication, outside of this secure channel and for this type of secure channel, with the other endpoint. It is the responsibility of each application to implement restrictions on other communication channels it may be using, including communication channels that are out of scope of the present document.
RQ04_0303	5.3	Applications on the Terminal or the UICC shall be able to refuse the communication of information with another application if a secure channel is not active between these applications.

5.2 Secured APDU - Application to Application lifecycle

Reference: TS 102 484 [1], clause 7.

5.2.1 Discovery

RQ number	Clause	Description
RQ05_0101	7.1	The terminal application may use Manage Secure Channel APDU - Retrieve UICC Endpoints command to discover UICC endpoints.

5.2.2 Master SA setup

RQ number	Clause	Description
RQ06_0101	7.2	Application to application secure channel setup shall only be initiated by a terminal application using the Manage Secure Channel APDU - Establish SA - Master SA command.
RQ06_0102	7.2	Using the Manage Secure Channel APDU - Establish SA - Master SA command, the terminal application shall supply the Ks_Local_Ref.
RQ06_0103	7.2	Using the Manage Secure Channel APDU - Establish SA - Master SA command, the terminal application shall … indicate the supported key agreement mechanisms available for that secure channel.
RQ06_0104	7.2	If the UICC application rejects the setup request … then the UICC shall set the SW1 SW2 to 'Execution error - no information given, state of non-volatile memory unchanged' and the Master SA and secure channel procedure shall end.
RQ06_0105	7.2	If the UICC application … there are no available mechanisms for key agreement indicated, then the UICC shall set the SW1 SW2 to 'Execution error - no information given, state of non-volatile memory unchanged' and the Master SA and secure channel procedure shall end.
RQ06_0106	7.2	An Application to application APDU secure channel Master SA may be setup using: a pre-shared key
RQ06_0107	7.2	An Application to application APDU secure channel Master SA may be setup using: certificates.
RQ06_0108	7.2	If a pre-shared key (e.g. from a GBA run) exists and WeakKey=0, then this may be used directly to derive a Master secret for the Master SA.
RQ06_0109	7.2	If a pre-shared key exists but WeakKey=1, then a TLS handshake protocol run is required to generate a strong Master secret for the Master SA.
RQ06_0110	7.2	If no pre-shared key exists but UICC and terminal certificates are available, then the terminal application and UICC application may run a TLS handshake protocol to establish a Master secret for the Master SA.
RQ06_0111	7.2	The terminal application shall instigate the agreed key agreement mechanism.
RQ06_0112	7.2	If a certificate-based key agreement or a weak pre-shared key is to be used for the key agreement then a TLS handshake shall be used to provide key material for the Master SA.
RQ06_0114	7.2	If a certificate-based key agreement or a weak pre-shared key is to be used for the key agreement then a TLS handshake shall be used to provide key material for the Master SA as follows: An IP channel shall be established over… a TCP connection using BIP - UICC Server mode as detailed in TS 102 223 [4] using the TLS port specified in TS 102 483 [12].
RQ06_0115	7.2	The terminal application sends a 'Client Hello' message to the UICC application to initiate a TLS handshake.
RQ06_0116	7.2	If a certificate-based key agreement or a weak pre-shared key is to be used … The same key agreement algorithms shall be supported as for the Application to Application TLS secure channel.
RQ06_0117	7.2	The UICC application and terminal application shall use the 48 byte TLS Master secret (MS_TLS) obtained from the TLS handshake to derive the 256 bit Master Secret (MS) of the Master SA as follows: MS = HMAC-SHA-256(MS_TLS, Ks_Local_Ref, MSA_ID). HMAC-SHA-256 is defined in defined in RFC 4634 [6] and FIPS PUB 180-2 [8].
RQ06_0118	7.2	If a strong pre-shared key agreement is indicated, then the UICC application takes the pre-shared key (PSK) referenced by Ks_Local_Ref and derives the Master Secret as MS = HMAC-SHA-256 (PSK,MSA_ID).
RQ06_0119	7.2	If a strong pre-shared key agreement is indicated. The terminal application uses the string Ks_Local_Ref to identify the key PSK and then derives the key Master Secret by computing MS=HMAC-SHA-256 (PSK,MSA_ID).

5.2.3 Connection SA setup

RQ number	Clause	Description
RQ07_0101	7.3	The terminal application shall setup a Connection SA using the Manage Secure Channel APDU - Establish SA - Connection SA command.
RQ07_0102	7.3	The terminal application shall generate a 16 byte Terminal nonce defined as Tnonce.
RQ07_0103	7.3	The terminal application shall send the UICC a Manage Secure Channel APDU - Establish SA - Connection SA command including the MSA_ID, Tnonce, the supported ciphering algorithms for the terminal (TSCA) and the supported integrity mechanisms for the terminal (TSIM).
RQ07_0104	7.3	The following Ciphering Algorithm shall be supported by the terminal application as a minimum ... 3DES - outer CBC using 2 keys as defined in TS 102 225 [9].
RQ07_0105	7.3	The following Ciphering Algorithm shall be supported by the terminal application as a minimum … 3DES - outer CBC using 3 keys as defined in TS 102 225 [9].
RQ07_0106	7.3	The following Ciphering Algorithm shall be supported by the terminal application as a minimum … AES with 128 bit key length in CBC mode with initial chaining value as defined in TS 102 225 [9].
RQ07_0107	7.3	The following integrity mechanism shall be supported by the terminal application as a minimum … CRC32 as defined in TS 102 225 [9].
RQ07_0108	7.3	The following integrity mechanism shall be supported by the terminal application as a minimum … ANSI Retail MAC (i.e. MAC algorithm 3 using block cipher DES and padding method 1 as defined in ISO/IEC 9797-1 [11]) without MAC truncation, i.e producing a checksum of 8 bytes length.
RQ07_0109	7.3	The following integrity mechanism shall be supported by the terminal application as a minimum … AES with 128 bit key length in CMAC mode as defined in TS 102 225 [9] with a checksum length truncated to the first 64 bits (8 bytes) as output.
RQ07_0110	7.3	Upon receipt of the Establish SA - Connection SA response from the UICC application, the terminal application retrieves the nonce Unonce.
RQ07_0111	7.3	The terminal application shall derive 464 bits of key material (KMaterial) from the key MS, and the nonces Unonce and Tnonce as follows: KMaterial = Kexp(MS, Unonce || Tnonce), using the key expansion algorithm KExp as defined in clause 10.
RQ07_0112	7.3	The first 128 bits of this key material shall be used as the MAC key K_MAC leaving a remaining 336 bits of key material for ciphering and integrity keys.
RQ07_0113	7.3	The terminal application uses K_MAC to verify the Establish SA - Connection SA response from the UICC application as follows. terminal application CSAMAC' = HMAC-SHA-256 (K_MAC, MSA_ID || Tnonce || TSCA || TSIM || CSA_ID || Unonce || UCA || UIM) truncated to the first 16 bytes as defined in RFC 2104 [7].
RQ07_0114	7.3	If CSAMAC' does not equal the value CSAMAC received, then the terminal shall terminate the Connection SA.
RQ07_0115	7.3	If CSAMAC'=CSAMAC, then the terminal application shall send a Manage Secure Channel APDU - Start Secure Channel command to the UICC application
RQ07_0116	7.3	Manage Secure Channel APDU - Start Secure Channel command to the UICC application… contains • the secure connection identifier CSA_ID; • confirmation of the ciphering algorithm to be used (UCA); and • confirmation of the integrity mechanism (UIM). The data in this command is protected by the value SSCMAC where SSCMAC = HMAC-SHA-256(K_MAC, CSA_ID || Unonce || UCA || UIM || CSAMAC) truncated to the first 16 bytes as defined in RFC 2104 [7].
RQ07_0117	7.3	If SSCMAC' does not equal the value SSCMAC sent, then the UICC application shall terminate the Connection SA establishment and set SW1 SW2 to "Authentication error, application specific".
RQ07_0118	7.3	If SSCMAC'=SSCMAC then the UICC application returns the unique secure channel session number to be used for secure data transfer using this Connection SA. This session number is used in the session control within the TRANSACT DATA APDU.
RQ07_0119	7.3	The terminal application and UICC application then derive the ciphering and integrity keys from the remaining 336 bits of KMaterial.

RQ number	Clause	Description
RQ07_0120	7.3	The ciphering key indicated by KIC shall be taken from the start of the remaining 336 bits of KMaterial. The ciphering key can be at most 168 bits (a 3 key 3DES key), leaving at least 168 remaining bits for the integrity key.
RQ07_0121	7.3	The integrity key indicated by KID is then taken from the start of the remaining bits left after both the K_MAC and ciphering keys have been taken.

5.2.4 Secure Connection Initiation and Data Transmission

RQ number	Clause	Description
RQ08_0101	7.4	Once a Manage Secure Channel APDU - Start SecureChannel command has been received by the UICC application and acknowledged, the UICC application and terminal application can initiate their security policy and start to secure transmitted data.
RQ08_0102	7.4	To send and receive APDUs securely through the APDU secure channel, the terminal application shall use the Transact Data APDU command defined in TS 102 221 [2] with the P1 parameter set to the value returned in the response to the Manage Secure Channel APDU - Start Secure Channel command. The coding of encrypted APDUs exchanged using the Transact Data command is described in clause 10 "Encrypted Data coding".
RQ08_0103	7.4	The terminal application and UICC application shall handle the encryption / decryption of APDUs, and their responses, with up to 255 bytes of data using the secure channel segmentation detailed in TS 102 221 [2].
RQ08_0104	7.4	Each encrypted message, in either direction, shall have its own 8 bytes transaction counter value that shall be the last successful message counter value + 1.
RQ08_0105	7.4	This transaction counter is incremented regardless of execution errors or aborted transactions.
RQ08_0106	7.4	The same transaction counter shall be used for both directions of communication.
RQ08_0107	7.4	The transaction counter is reset when a new Connection SA is established.
RQ08_0108	7.4	On receipt of encrypted blobs, the terminal application receiving the blob shall … Re-assemble the encrypted blobs.
RQ08_0109	7.4	On receipt of encrypted blobs, the terminal application receiving the blob shall … Decrypt the combined encrypted blob using the keys and mechanisms agreed for that secure channel.
RQ08_0110	7.4	On receipt of encrypted blobs, the terminal application receiving the blob shall … Verify that the message is valid by checking the integrity protection.
RQ08_0111	7.4	On receipt of encrypted blobs, the terminal application receiving the blob shall … Check that the counter is valid.
RQ08_0112	7.4	If the message is valid then the terminal application or UICC application that has decoded the message shall action the APDU or APDU response.
RQ08_0113	7.4	If the message is invalid then the terminal application or UICC application that has decoded the message shall not action the APDU or APDU response.

5.2.5 SA Termination and Resumption

RQ number	Clause	Description
RQ09_0101	7.5	To terminate an existing APDU secure channel Master SA ... the terminal application shall use the Manage Secure Channel APDU - Terminate secure channel SA command defined in TS 102 221 [2].
RQ09_0102	7.5	To terminate an existing APDU secure channel … Connection SA, the terminal application shall use the Manage Secure Channel APDU - Terminate secure channel SA command defined in TS 102 221 [2].
RQ09_0103	7.5	If a Connection SA is indicated then the MAC value shall be set to the first 16 bytes of HMAC SHA 256(K_MAC, CSA_ID).
RQ09_0104	7.5	If a Master SA is indicated then the MAC value shall be set to the first 16 bytes of HMAC SHA 256 (MS, MSA_ID).
RQ09_0105	7.5	The UICC application shall acknowledge the Manage Secure Channel APDU - Terminate secure channel SA command with a status word indicating success or failure.
RQ09_0106	7.5	The UICC application can indicate that a key or a security association has expired using the SW1 SW2 response "Security session or association expired".
RQ09_0107	7.5	In order to resume a secure channel, the terminal application shall start a completely new Connection SA using the Manage Secure Channel APDU - Establish SA - Connection SA command.

5.3 Encrypted data coding

Reference: TS 102 484 [1], clause 10.

RQ number	Clause	Description
RQ10_0101	10	Data to be sent and its response is encrypted together with a nonce, a counter, padding and a checksum.
RQ10_0102	10	The padding length shall be chosen so that the data to be encrypted is a multiple of the block size for the algorithm used.
RQ10_0103	10	Encrypted data is sent using the TRANSACT DATA command as described in TS 102 221 [2]. The encrypted data is sent in encrypted data TLV objects.
RQ10_0104	10	For each secure channel, TRANSACT DATA APDUs with encrypted data TLV objects shall always contain fixed number of bytes of data.
RQ10_0105	10	If the data is sent using several APDUs, each of the APDUs, including the last one, shall contain the same fixed number of bytes of data.
RQ10_0106	10	This data size is indicated in the endpoint discovery mechanism for each secure channel.
RQ10_0107	10	If necessary, the TRANSACT DATA command data shall be padded to create the correct length of message as described below.

5.4 Key Expansion Function Definition

Reference: TS 102 484 [1], clause 11.

RQ number	Clause	description
RQ11_0101	11	"The key expansion function Kexp is based on the Key Expansion function defined in IKE v2 (RFC 4306 [16]) and is designed to produce any required amount of key material from a single cryptographic key. In order to do this, the HMAC-SHA-256 algorithm, which produces output of 256 bits is used iteratively until enough key material is available. For input a key K and an arbitrary length string str, the function Kexp produces a stream of 256 bit output strings T1, T2, T3, etc using HMAC-SHA-256 as follows: Kexp(K,str) = T1 || T2 || T3 || … Where: • T1 = HMAC-SHA-256(K, str || 0x01); • T2 = HMAC-SHA-256(K, T1 || str || 0x02); • T3 = HMAC-SHA-256(K, T2 || str || 0x03). And so on until enough key material has been produced. Key material of the desired length (e.g. 464 bits are required for Kmaterial in clause 7.3) is taken from the output key stream of Kexp."

5.5 ATR

Reference: TS 102 221 [2], clause 6.

RQ number	Clause	Description
RQ11_0201	6.3.3	Table 6.7: Coding of the first TBi (i > 2) after T = 15 of the ATR

5.6 MANAGE SECURE CHANNEL Command

Reference: TS 102 221 [2], clause 11.

RQ number	Clause	Test clauses
RQ12_0101	11.1.20.1	P1 determines which sub procedure is required, the P2 parameter value meaning is specific to each P1 value.
RQ12_0102	11.1.20.1	The command and response data are encapsulated in BER-TLV objects structured as defined in clause 11.3 using tag '73' for BER-TLV structured data and tag '53' otherwise.
RQ12_0103	11.1.20.1	This command can chain successive blocks of command data, if present, with a maximum size of 255 bytes each, required for one operation using P2 to indicate the first/next block.
RQ12_0104	11.1.20.1	The terminal performs the segmentation of the data, and the UICC the concatenation of the data.
RQ12_0105	11.1.20.1	The first MANAGE SECURE CHANNEL APDU is sent with P2 indicating "First block of command data".
RQ12_0106	11.1.20.1	Following MANAGE SECURE CHANNEL APDUs are sent with P2 indicating "Next block of command data".
RQ12_0107	11.1.20.1	The command response data is retrieved from the UICC using one or more separate MANAGE SECURE CHANNEL APDUs with the same chaining mechanism as for the command data.
RQ12_0108	11.1.20.1	The UICC performs the segmentation of the data, and the terminal the concatenation of the response data.
RQ12_0109	11.1.20.1	The first MANAGE SECURE CHANNEL APDU is sent with P2 indicating "First block of response data".
RQ12_0110	11.1.20.1	Following MANAGE SECURE CHANNEL APDUs are sent with P2 indicating "Next block of response data".
RQ12_0201	11.1.20.2.1	MANAGE SECURE CHANNEL 'Retrieve UICC Endpoints'. This command allows the terminal to retrieve a list of secure channel endpoints from the UICC as defined in TS 102 484 [1] and the maximum data container size available for the TRANSACT DATA command.
RQ12_0202	11.1.20.2.1	In order to retrieve the end point information P2 is set to "First block of response data".
RQ12_0203	11.1.20.2.1	In order to retrieve the end point information P2 is set to ... or in case of the response data longer than 255 bytes following blocks are retrieved be setting P2 to "Next block of response data".
RQ12_0301	11.1.20.3.1	Establish SA - Master SA. This command allows the terminal to establish a Master SA with the UICC as defined in TS 102 484 [1].
RQ12_0302	11.1.20.3.1	Establish SA - Master SA. The command data is sent to the UICC using P2='80'
RQ12_0303	11.1.20.3.1	Establish SA - Master SA. The response data is retrieved using P2='A0'.
RQ12_0304	11.1.20.3.1	The command and response data is encapsulated using tag '73'.
RQ12_0305	11.1.20.3.1	Key Agreement Mechanism Tag. Coding of Byte 1- Supported key agreement methods: • In the present document only the first byte is defined. Supported key agreement methods: • Strong Preshared Keys.
RQ12_0306	11.1.20.3.1	Coding of Terminal_ID: This shall be a unique value that identifies that terminal. This may be the IMEI as defined in TS 124 008 [5].
RQ12_0307	11.1.20.3.1	Coding of Terminal_appli_ID: This shall be a value that identifies the application in that terminal that hosts the terminal endpoint. This value shall uniquely identify an application within the terminal.
RQ12_0308	11.1.20.3.1	Coding of UICC_ID: This shall be a unique value that identifies that UICC. This shall be the ICCID as defined for EFICCID.
RQ12_0309	11.1.20.3.1	Coding of UICC_appli_ID: This shall be the AID of the application in that UICC that hosts the UICC endpoint.
RQ12_0401	11.1.20.4.2	This clause defines the MANAGE SECURE CHANNEL function and coding when P1 = 'Establish SA - Connection_SA'. This command allows the terminal to establish a Connection SA with the UICC as defined in TS 102 484 [1].

RQ number	Clause	Test clauses
RQ12_0402	11.1.20.4.2	Establish SA - Connection_SA. The command data is sent to the UICC using P2='80'.
RQ12_0403	11.1.20.4.2	Establish SA - Connection_SA. The response data is retrieved using P2='A0'.
RQ12_0404	11.1.20.4.2	The command and response data is encapsulated using tag '73'.
RQ12_0405	11.1.20.4.2	Coding of Algorithm and Integrity BER-TLV, tag '89': Coding of Byte 1 - Ciphering Algorithm (UCA).
RQ12_0406	11.1.20.4.2	Coding of Algorithm and Integrity BER-TLV, tag '89': Coding of Byte 2 - Integrity mechanism (UIM).
RQ12_0408	11.1.20.4.2	Coding of MSA_ID BER-TLV, tag '88': Unique 16 byte HEX number that identifies a specific Master_SA.
RQ12_0409	11.1.20.4.2	Coding of Tnonce BER-TLV, tag '8A': Randomly generated 16 byte Tnonce in HEX.
RQ12_0501	11.1.20.5.1	This clause defines the MANAGE SECURE CHANNEL function and coding when P1 = 'Establish SA - Start Secure Channel'. This command allows the terminal to secure a logical channel with the UICC as defined in TS 102 484 [1]. It contains the final part of the authenticated handshake for the MANAGE SECURE CHANNEL - 'Establish SA - Connection_SA' command.
RQ12_0601	11.1.20.5.2	Establish SA - Start Secure Channel. The command data is sent to the UICC using P2='80'.
RQ12_0602	11.1.20.5.2	Establish SA - Start Secure Channel. The response data is retrieved using P2='A0'.
RQ12_0603	11.1.20.5.2	Establish SA - Start Secure Channel. The command data is encapsulated using tag '73'.
RQ12_0604	11.1.20.5.2	Establish SA - Start Secure Channel. The response data is encapsulated using tag '53'.
RQ12_0605	11.1.20.5.2	Coding of Algorithm and Integrity BER-TLV, tag '89': Coding of Byte 1 - Ciphering Algorithm (UCA): Only one bit shall be indicated.
RQ12_0606	11.1.20.5.2	Coding of Algorithm and Integrity BER-TLV, tag '89': Coding of Byte 2 - Integrity mechanism (UIM): Only one bit shall be indicated.
RQ12_0607	11.1.20.5.2	Coding of CSA_ID BER-TLV, tag '8B': • Unique 16 byte HEX number that identifies a specific Connection_SA. See TS 102 484 [1].
RQ12_0608	11.1.20.5.2	Coding of SSCMAC BER-TLV, tag '8D': • 16 Byte HEX value. See TS 102 484 [1].
RQ12_0609	11.1.20.5.2	Coding of the Endpoint data container size BER-TLV, tag '8E': • This is the length of the value part of the secure channel data TLV specified for the TRANSACT DATA command. The data container size set by the terminal shall be less or equal to the value indicated in the BER-TLV object returned with Tag '82' returned by the Retrieve UICC Endpoints command.
RQ12_0701	11.1.20.6.1	This clause defines the MANAGE SECURE CHANNEL function and coding when P1 = "Terminate secure channel SA". This command allows the terminal to terminate one or several secure channel Security Association(s) with the UICC as defined in TS 102 484 [1].
RQ12_0801	11.1.20.6.2	Terminate secure channel SA. The command data is sent to the UICC using P2='80'.
RQ12_0802	11.1.20.6.2	Terminate secure channel SA. The response data is retrieved using P2='A0'.
RQ12_0803	11.1.20.6.2	Terminate secure channel SA. The command and response data are encapsulated using tag '73'.
RQ12_0804	11.1.20.6.2	The command data shall contain either a Master_SA TLV only or a list of Connection_SA TLVs associated to the same MSA.
RQ12_0805	11.1.20.6.2	Coding of MSA_ID: Master Security Association Identity MSA_ID as defined in TS 102 484 [1].
RQ12_0806	11.1.20.6.2	Coding of MAC: MAC as defined in TS 102 484 [1].
RQ12_0807	11.1.20.6.2	Coding of CSA_ID: Connection Security Association Identity CSA_ID as defined in TS 102 484 [1].
RQ12_0808	11.1.20.6.2	Coding of MAC: MAC as defined in TS 102 484 [1].

5.7 TRANSACT DATA Command

Reference: TS 102 221 [2], clause 11.

RQ number	Clause	description
RQ13_0101	11.1.21.1	The Transact Data command transports large amounts of data on APDU based communication with different data formats.
RQ13_0102	11.1.21.1	The Transact Data command is either a case 2 or case 3 command depending on P1 b3.
RQ13_0103	11.1.21.1	The Transact Data becomes a case 1 command when P1 b2 is set (session abort).
RQ13_0104	11.1.21.1	The Transact Data P1 defines the data transfer session number.
RQ13_0105	11.1.21.1	The Transact Data P1 is also used for requesting retransmission.
RQ13_0106	11.1.21.1	The Transact Data P1 is also used for session abort.
RQ13_0107	11.1.21.1	The session number allows up to four transfer sessions to be interleaved.
RQ13_0108	11.1.21.1	The P2 parameter contains the number of remaining data blocks going from terminal to the UICC in this transaction.
RQ13_0109	11.1.21.1	Both the terminal and the UICC can abort the data transfer session.
RQ13_0110	11.1.21.1	A data transfer session is ongoing until it is aborted by the UICC or terminal or completed in normal circumstances. Upon session abort by the terminal, the Connection SA remains open and all data related to the current transaction are lost.
RQ13_0201	11.1.21.2	If P2 is different from 0 then the APDU shall contain data.
RQ13_0202	11.1.21.2	Once P2 has reached zero the terminal shall not start sending more data in the same session as long as the UICC is producing response data.
RQ13_0203	11.1.21.2	The data transmitted is encapsulated in a BER-TLV data object structure.
RQ13_0204	11.1.21.2	The length of the TLV objects shall be coded one or two bytes.
RQ13_0205	11.1.21.2	The same tag value shall be used within one transfer session. All data within subsequent TRANSACT DATA commands within the same session shall use the same tag as the first TRANSACT DATA command in the session.
RQ13_0206	11.1.21.2	The normal response to the TRANSACT DATA APDU is '92 XX': Data transaction ongoing.

6 Test cases

6.1 Test group 1: Discovery

6.1.1 Sub Test group 1.1: Discovery of secure channel support

6.1.1.1 Test Case 1: ATR

6.1.1.1.1 Test execution

The test procedure shall be executed for each of the following parameters:

- There are no test case-specific parameters for this test case.

6.1.1.1.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel, but none of the contacts are activated.

6.1.1.1.3 Test procedure

Step	Direction	Description	RQ
1	User → T	Trigger the terminal to activate Secure Channel Interface.	
2	UICC → T	Send ATR given in clause 4.4.5.1, that indicates: • support of Secure Channel.	
3	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints; or Manage Secure Channel - Establish SA - Master SA. First block of command data as detailed in table 6.1.1.1.3.1.	RQ01_0101 RQ01_0201 RQ05_0101 RQ11_0201

Table 6.1.1.1.3.1

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00' or '01'	'80'	'00'

6.2 Test group 2: Channel Administration

6.2.1 Sub Test group 2.1 Retrieve UICC Endpoints

6.2.1.1 Test case 1: Retrieve UICC Endpoints - No Endpoints

6.2.1.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.2.1.1.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- Terminal is connected to UICC simulator supporting Secure Channel with no Endpoints defined, as detailed in clause 4.4.6.1.1.1.

6.2.1.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of command data as detailed in table 6.2.1.1.3.1.	RQ01_0102 RQ01_0103
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'	
3	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of response data as detailed in table 6.2.1.1.3.2.	
4	UICC → T	Return data for Retrieve UICC Endpoints as detailed in clause 4.4.6.1.1 and Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.1.1.3.1

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'80'	'00'

Table 6.2.1.1.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'A0'	'00'

6.2.1.2 Test case 2: Test case 2: Manage Secure Channel - Retrieve UICC Endpoints - Single Endpoint

6.2.1.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- Maximum container size indicated in Endpoint = 255.

6.2.1.2.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- Terminal is connected to UICC simulator supporting Secure Channel Endpoints as detailed in clause 4.4.6.1.2.

6.2.1.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of command data as detailed in table 6.2.2.1.3.1.	RQ01_0102 RQ02_0103 RQ05_0101
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'	
3	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of response data as detailed in table 6.2.1.1.3.2.	
4	UICC → T	Return data for Retrieve UICC Endpoints as detailed in clause 4.4.6.1.2. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.1.2.3.1

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'80'	'00'

Table 6.2.1.2.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'A0'	'00'

6.2.1.3 Test case 3: Retrieve UICC Endpoints - Multiple Endpoints

6.2.1.3.1 Test execution

The test procedure shall only be executed for the following considerations:

- Maximum container size indicated in Endpoint = 255.

6.2.1.3.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- Terminal is connected to UICC simulator supporting Secure Channel Endpoints as detailed in clause 4.4.6.1.3.

6.2.1.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of command data as detailed in table 6.2.1.3.3.1.	RQ01_0102 RQ05_0101
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of response data as detailed in table 6.2.1.3.3.2.	
4	UICC → T	Return data for Retrieve UICC Endpoints as detailed in clause 4.4.6.1.3. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.1.3.3.1

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'80'	'00'

Table 6.2.1.3.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'A0'	'00'

6.2.1.4 Test case 4: Retrieve UICC Endpoints - Multiple Endpoints Transferred in Blocks

6.2.1.4.1 Test execution

The test procedure shall only be executed for the following considerations:

- Maximum container size indicated in Endpoint = 255.

6.2.1.4.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- Terminal is connected to UICC simulator supporting Secure Channel Endpoints as detailed in clause 4.4.6.1.4.

6.2.1.4.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of command data as detailed in table 6.2.1.4.3.1.	RQ01_0102 RQ05_0101
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints. First block of response data as detailed in table 6.2.1.4.3.2.	
4	UICC → T	Return data for Retrieve UICC Endpoints as detailed in clause 4.4.6.1.4.1 Endpoint information 1st Block. Status Words SW1 SW2 set to "More data available '62 F1'.	
5	T → UICC	Send Manage Secure Channel - Retrieve UICC Endpoints 2nd Block. Next block of response data as detailed in table 6.2.1.4.3.3.	
6	UICC → T	Return data for Retrieve UICC Endpoints as detailed in clause 4.4.6.1.4.2 Endpoint information 2nd Block. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.1.4.3.1

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'80'	'00'

Table 6.2.1.4.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'A0'	'00'

Table 6.2.1.4.3.3

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'00'	'20'	'00'

6.2.2 Sub Test group 2.2 Manage Secure Channel - Establish SA - Master SA

6.2.2.1 Test case 1: Establish Master SA

6.2.2.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.
- Maximum container size indicated in Endpoint = 255.

6.2.2.1.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal is connected to a UICC simulator supporting a Master SA as defined in clause 4.4.6.2.

6.2.2.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Establish SA - Master SA command. First block of command data as detailed in table 6.2.2.1.3.1.	RQ01_0103 RQ02_0101 RQ02_0107 RQ02_0108 RQ02_0110 RQ02_0201 RQ02_0202 RQ02_0203 RQ02_0204 RQ02_0206 RQ06_0101 RQ06_0102 RQ06_0103
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Establish SA - Master SA. First block of response data as detailed in table 6.2.2.1.3.4.	
4	UICC → T	Return Master SA as defined in clause 4.4.6.2 Master SA response. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.2.1.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'01'	'80'	'XX'	Data as detailed in table 6.2.2.1.3.2
NOTE: XX = Total length of Tag 73 with contained data below.						

Table 6.2.2.1.3.2

Tag	Length	Value
'73'	'XX'	Master SA Command Data TLVs as detailed in table 6.2.2.1.3.3
NOTE: XX = Total length of Tags 83-87 with contained data below.		

Table 6.2.2.1.3.3

Type	Tag	Length	Value	Description
Key Agreement mechanism	87	'01'	'02'	
Terminal ID	83	XX	XX	Length and value not evaluated, may be IMEI i.e. 358701044528124
Terminal_appli_ID	84	XX	XX	Length and value not evaluated, IDs application in the terminal
UICC_Identifier	85	'0A'		ICCID as detailed in clause .4.6.1.2 Tag 81
UICC_appli_ID	86	12-17		AID as detailed in clause 4.4.6.1.2 Tag 82

Table 6.2.2.1.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'01'	'A0'	'00'

6.2.2.2 Test case 2: Establish Master SA UICC Rejects

6.2.2.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.2.2.2.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal is connected to a UICC simulator supporting a Master SA as defined in clause 4.4.6.2.

6.2.2.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Establish SA - Master SA command. First block of command data as detailed in table 6.2.2.2.3.1.	
2	UICC → T	Status words SW1 SW2 set to 'Execution error - no information given, state of non-volatile memory unchanged" '62 00'.	
3	T → UICC	Secure channel procedure ends.	RQ06_0104 RQ06_0105

Table 6.2.2.2.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'01'	'80'	'XX'	Length and value is not evaluated

6.2.2.3 Test case 4: Establish Master SA - 4 Master SAs

6.2.2.3.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.2.2.3.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the 4 endpoints defined in clause 4.4.6.1.3 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.3.
- The Terminal is connected to a UICC simulator supporting 4 Master SAs as defined in clause 4.4.6.2 each with a unique MSA_ID.

6.2.2.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Establish SA - Master SA command. First block of command data as detailed in table 6.2.2.3.3.1.	RQ02_0111 RQ02_0201 RQ02_0202 RQ02_0203 RQ02_0204 RQ02_0206 RQ02_0207 RQ04_0201
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Establish SA - Master SA. First block of response data as detailed in table 6.2.2.3.3.4.	
4	UICC → T	Return Master SA as defined in clause 4.4.6.2. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	
5 - 16	T → UICC → T	Step 1 to 4 shall be repeated 3 more times. Each AID for the endpoints as detailed in clause 4.4.6.1.3 shall be sent in Tag 86 as detailed in table 6.2.2.3.3.3, once only and in any order.	

Table 6.2.2.3.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'01'	'80'	'XX'	Data as detailed in table 6.2.2.3.3.2

Table 6.2.2.3.3.2

Tag	Length	Value
'73'	'XX'	Master SA Command Data TLVs as detailed in table 6.2.2.3.3.3

Table 6.2.2.3.3.3

Type	Tag	Length	Value	Description
Key Agreement mechanism	87	'01'	'02'	
Terminal ID	83	XX	XX	Length and value not evaluated, may be IMEI i.e. 358701044528124
Terminal_appli_ID	84	XX	XX	Length and value not evaluated, IDs application in the terminal
UICC_Identifier	85	'0A'		ICCID as detailed in 4.4.6.1.3 Tag 81
UICC_appli_ID	86	12-17		AID as detailed in 4.4.6.1.3 Tag 82 once only for each endpoint in any order.

Table 6.2.2.3.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'01'	'A0'	'00'

6.2.3 Sub Test group 2.3 Manage Secure Channel - Establish SA - Connection SA

6.2.3.1 Test case 1: Establish Connection SA

6.2.3.1.1 Test execution

The test procedure shall be executed for each of the following parameters:

- There are no test case-specific parameters for this test case.

6.2.3.1.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal has established a Master SA using data as defined in clause 4.4.6.2 in accordance with the procedure outlined in clause 6.2.2.1.

6.2.3.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel - Establish SA - Connection SA. First block of command data as detailed in table 6.2.3.1.3.1.	RQ01_0103 RQ02_0101 RQ03_0105 RQ03_0106 RQ03_0107 RQ07_0101 RQ07_0102 RQ07_0103 RQ07_0104 RQ07_0105 RQ07_0106 RQ07_0107 RQ07_0108 RQ07_0109
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Establish SA - Connection SA. First block of response data as detailed in table 6.2.3.1.3.4.	RQ02_0104 RQ02_0105 RQ02_0106 RQ02_0107 RQ02_0109 RQ02_0110 RQ02_0209 RQ02_0301 RQ02_0302 RQ07_0110
4	UICC → T	Return Connection SA as defined in clause 4.4.6.3 Connection SA response. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.3.1.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2A'	Data as detailed in table 6.2.3.1.3.2

Table 6.2.3.1.3.2

Tag	Length	Value
'73'	'28'	Connection SA Command Data TLVs as detailed in table 6.2.3.1.3.3

Table 6.2.3.1.3.3

Type	Tag	Length	Value	Description
Algorithm and integrity	'89'	'02'	'07 07'	Supported ciphering algorithms and supported integrity mechanism all must be supported by the terminal.
MSA_ID	'88'	'10'	16 byte number	16 byte HEX number returned in the Master SA command in the Initial conditions. This shall be the MSA_ID value of Tag '88' defined in table 4.4.6.2.2. of the Master SA response given in clause 4.4.6.2.
Tnonce	'8A'	'10'	16 byte number	16 byte Tnonce in HEX randomly generated by the terminal application.

Table 6.2.3.1.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'02'	'A0'	'00'

6.2.3.2 Test case 2: Establish Connection SA (Incorrect CSAMAC)

6.2.3.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.2.3.2.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal has established Master SA using data as defined in clause 4.4.6.2 in accordance with the procedure outlined in clause 6.2.2.1.

6.2.3.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel - Establish SA - Connection SA. First block of command data as detailed in table 6.2.3.2.3.1.	
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'	
3	T → UICC	Send Manage Secure Channel - Establish SA - Connection SA. First block of response data as detailed in table 6.2.3.2.3.2.	
4	UICC → T	Return Connection SA (Incorrect APDU Response). Data as detailed in clause 4.4.6.3 Connection SA response with the CSAMAC = calculated value +1.	
5	T → UICC	No further secure channel communication takes place.	RQ07_0114

Table 6.2.3.2.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2A'	Data not evaluated

Table 6.2.3.2.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'02'	'A0'	'00'

6.2.3.3 Test case 3: Establish Connection SA - 4 Connection SAs

6.2.3.3.1 Test execution

The test procedure shall be executed for each of the following parameters:

- There are no test case-specific parameters for this test case.

6.2.3.3.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoints defined in clause 4.4.6.1.3 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.3.
- The Terminal is connected to a UICC simulator supporting Master SAs as defined in clause 4.4.6.2.

- The Terminal has established 4 Master SAs as defined in 4.4.6.2 in accordance with the procedure outlined in clause 6.2.2.3.

6.2.3.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel - Establish SA - Connection SA. First block of command data as detailed in table 6.2.3.3.3.1.	RQ02_0301 RQ02_0302 RQ04_0201 RQ07_0101 RQ07_0102 RQ07_0103 RQ07_0104 RQ07_0105 RQ07_0106 RQ07_0107 RQ07_0108 RQ07_0109 RQ07_0110 RQ08_0107
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Establish SA - Connection SA. First block of response data as detailed in table 6.2.3.3.3.4.	
4	UICC → T	Return Connection SA as defined in clause 4.4.6.3 Connection SA response. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	
5 - 16	T → UICC → T	Step 1 to 4 shall be repeated 3 more times. Each MSA_ID for the Master SA as detailed in clause 4.4.6.2 shall sent in Tag '88' as detailed in table 6.2.3.3.3.3 once, only and in any order.	RQ02_0111 RQ02_0209 RQ02_0307

Table 6.2.3.3.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2A'	Data as detailed in table 6.2.3.3.3.2

Table 6.2.3.3.3.2

Tag	Length	Value
'73'	'28'	Master SA Command Data TLVs as detailed in table 6.2.3.3.3.3

Table 6.2.3.3.3.3

Type	Tag	Length	Value	Description
Algorithm and integrity	'89'	'02'	'07 07'	Supported ciphering algorithms and supported integrity mechanism all must be supported by the terminal.
MSA_ID	'88'	'10'	16 byte number	16 byte HEX number returned in the Master SA command in the Initial conditions. This shall be the MSA_ID value of ' 88' defined in table 4.4.6.2.2. of the Master SA response given in clause 4.4.6.2.
Tnonce	'8A'	'10'	16 byte number	16 byte Tnonce in HEX randomly generated by the terminal application.

Table 6.2.3.3.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'02'	'A0'	'XX'

6.2.4 Sub Test group 2.4 Manage Secure Channel - Establish SA - Start Secure Channel

6.2.4.1 Test case 1: Start Secure Channel

6.2.4.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- The test shall be executed for each of the Ciphering and Integrity key combinations that the Terminal supports and is selectable.

6.2.4.1.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal has established a Master SA using data as defined in clause 4.4.6.2 in accordance with the procedure outlined in clause 6.2.2.1.
- The Terminal has established Connection SA using data as defined in clause 4.4.6.3 in accordance with the procedure outlined in clause 6.2.3.1.

6.2.4.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel - Start Secure Channel command. First block of command data as detailed in table 6.2.4.1.3.1.	RQ01_0104 RQ02_0210 RQ07_0111 RQ07_0112 RQ07_0113 RQ07_0115 RQ07_0116
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel. First block of response data as detailed in table 6.2.4.1.3.4.	RQ02_0101 RQ02_0104 RQ02_0105 RQ02_0106
4	UICC → T	Return Session Number = 3. Data as detailed in clause 4.4.6.4. Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.4.1.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2D'	Data as detailed in table 6.2.4.1.3.2

Table 6.2.4.1.3.2

Tag	Length	Value
'73'	'2B'	Start Secure Channel Command Data TLVs as detailed in table 6.2.4.1.3.3

Table 6.2.4.1.3.3

Type	Tag	Length	Value	Description
Algorithm and integrity	'89'	'02'	'ZZ YY'	Supported ciphering algorithms and supported integrity mechanism: • ZZ = '01, 02, 04 or 80'. • YY = '01, 02, 04 or 80'.
CSA_ID	'8B'	'10'	16 byte HEX value	16 byte HEX number returned in the Connection SA command in the initial conditions. This shall be the CSA_ID value of Tag '8B' defined in table 4.4.6.3.2 of the Connection SA response given in clause 4.4.6.3.
SSCMAC	'8D'	'10'	16 byte HEX value	SSCMAC truncated to the first 16 bytes and calculated as detailed below.
Endpoint data container size	'8E'	'01'	'XX'	data container size less than or equal to the value returned in endpoint discovery i.e 255.

SSCMAC =HMAC-SHA-256(K_MAC, CSA_ID||Unonce||UCA||UIM||CSAMAC)

Table 6.2.4.1.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'03'	'A0'	'00'

Table 6.2.4.1.3.5

Tag	Length	Value
'53'	'01'	XX000000 XX=Session number as indicated in the test procedure which may be '0', '1', '2' or '3' coded in 2 bits. Value = '00' or '40 or '80' or 'C0'.

6.2.4.2 Test case 2: Start Secure Channel - UICC Error

6.2.4.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.2.4.2.2 Initial conditions

- Terminal is connected to UICC simulator supporting Secure Channel and successfully reset.
- The Terminal has retrieved the endpoint defined in clause 4.4.6.1.2 with a Maximum data container size = 255 in accordance with the procedure outlined in clause 6.2.1.2.
- The Terminal has established a Master SA using data as defined in clause 4.4.6.2 in accordance with the procedure outlined in clause 6.2.2.1.
- The Terminal has established Connection SA using data as defined in clause 4.4.6.3 in accordance with the procedure outlined in clause 6.2.3.1.

6.2.4.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel - Start Secure Channel command. First block of command data as detailed in table 6.2.4.2.3.1.	
2	UICC → T	Status Words SW1 SW2 'Authentication error, application specific.	
3	T → UICC	No further commands for this CSA_ID.	RQ07_0117

Table 6.2.4.2.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2D'	Data as detailed in table 6.2.4.2.3.2

Table 6.2.4.2.3.2

Tag	Length	Value
'73'	'2B'	Data not evaluated

6.2.5 Sub Test group 2.5 Manage Secure Channel - Terminate Secure Channel SA

6.2.5.1 Test case 1: Terminate Secure Channel - Master SA

6.2.5.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.
- Terminal application key lifetime.

6.2.5.1.2 Initial conditions

- Master SA successfully established as defined in clause 6.2.2.1.
- Connection SA successfully established as defined in clause 6.2.3.1.
- Secure Channel is successfully started as defined in clause 6.2.4.1.

6.2.5.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Terminate Secure Channel command as detailed in table 6.2.5.1.3.1.	RQ01_0107 RQ02_0208 RQ02_0304 RQ09_0101 RQ09_0104 RQ09_0105
2	UICC → T	Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.5.1.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'04'	'80'	'XX'	As detailed in table 6.2.5.1.3.2

Table 6.2.5.1.3.2

Type	Tag	Length	Value	Description
MSA_ID	'88'	'20'	X	Master SA ID as received in the response to Manage Secure Channel Establish SA - Master SA

6.2.5.2 Test case 2: Terminate Secure Channel - Connection SA

6.2.5.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.
- Terminal application key lifetime.

6.2.5.1.2 Initial conditions

- Master SA successfully established as defined in clause 6.2.2.1.
- Connection SA successfully established as defined in clause 6.2.3.1.
- Secure Channel is successfully started as defined in clause 6.2.4.1.

6.2.5.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Terminate Secure Channel command as detailed in table 6.2.5.1.3.1.	RQ01_0107 RQ02_0302 RQ02_0303 RQ02_0305 RQ04_0102 RQ09_0102 RQ09_0103 RQ09_0105
2	UICC → T	Status Words SW1 SW2 set to "Normal ending of command" '90 00'.	

Table 6.2.5.2.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'04'	'80'	'XX'	As detailed in table 6.2.5.2.3.2

Table 6.2.5.2.3.2

Type	Tag	Length	Value	Description
CSA_ID	'8B'	'20'	X	Connection SA ID

6.2.5.3 Test case 3: Suspend and resume Secure Channel - Terminal application expires

6.2.5.3.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.
- Terminal application key lifetime.

6.2.5.3.2 Initial conditions

- Master SA successfully established as defined in clause 6.2.2.1.
- Connection SA successfully established as defined in clause 6.2.3.1.
- Secure Channel is successfully started as defined in clause 6.2.4.1.

6.2.5.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Terminate Secure Channel command as detailed in table 6.2.5.3.3.1.	RQ01_0106
2	UICC → T	Normal ending of command.	
3	T → UICC	Manage Secure Channel - Establish SA - Connection SA. First block of command data as detailed in table 6.2.5.3.3.3.	
4	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
5	T → UICC	Send Manage Secure Channel - Establish SA - Connection SA. First block of response data as detailed in table 6.2.5.3.3.4.	RQ04_0103
6	UICC → T	Return Connection SA. Data as detailed in clause 4.4.6.3 Connection SA response.	

Table 6.2.5.3.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'04'	'80'	'XX'	Table 6.2.5.3.3.2

Table 6.2.5.3.3.2

Type	Tag	Length	Value	Description
CSA_ID	'8B'	'20'	CSA_ID	Connection CSA ID

Table 6.2.5.3.3.3

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2A'	Data not evauluated

Table 6.2.5.3.3.4

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'02'	'A0'	'00'

6.2.5.4 Test case 4: Suspend and resume Secure Channel - UICC application expires

6.2.5.4.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.
- Terminal application key lifetime.

6.2.5.4.2 Initial conditions

- Master SA successfully established as defined in clause 6.2.2.1.
- Connection SA successfully established as defined in clause 6.2.3.1.
- Secure Channel is successfully started as defined in clause 6.2.4.1.

6.2.5.4.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Manage Secure Channel command or Transact Data command.	
2	UICC → T	SW1 SW2 "Security session or association expired".	RQ02_0306 RQ09_0106
3	T → UICC	Manage Secure Channel - Establish SA - Connection SA. First block of command data as detailed table 6.2.5.4.3.1.	RQ01_0106 RQ04_1013
4	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'	
5	T → UICC	Send Manage Secure Channel - Establish SA - Connection SA. First block of response data as detailed in table 6.2.3.1.3.3.	
6	UICC → T	Return Connection SA. Data as determined in clause 4.4.6.3 Connection SA response.	

Table 6.2.5.4.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'73'	'02'	'80'	'2A'	Data not evauluated

Table 6.2.5.4.3.2

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'73'	'02'	'A0'	'00'

6.3 Test group 3: Key Agreement

6.3.1 Sub Test group 3.1 GBA

Out of scope for this specification.

6.3.2 Sub Test group 3.2 Strong

As the mechanism is, by definition, not standardised, there are no tests for this feature.

6.3.3 Sub Test group 3.3 Weak

Out of scope for this specification

6.3.4 Sub Test group 3.4 Certificate Exchange

Out of scope for this specification.

6.4 Test group 4: Secure Channel Operation

6.4.1 Sub Test group 4.1 Transact Data - Command Data

6.4.1.1 Test case 1: Transact Data - Command Data in 1 secure channel TLV

6.4.1.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- The Terminal shall support at least a minimal application as defined in clause 4.4.7.1.1 which shall send a APDU command defined in table 4.4.7.1.1.1.
- The Terminal shall support the endpoint container size of 255 as indicated in the retrieve endpoint provided by the UICC simulator.

- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.1.1.2 Initial conditions

- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.

6.4.1.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command.	
2	UICC → T	Status Words SW1 SW2 set to "Response data available" '62 F3'.	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response.	
4	UICC → T	Return Session Number = 3. Data as detailed in clause 4.4.6.4.	
5	T → UICC	Send TRANSACT DATA command. First block of command data as detailed in table 6.4.1.1.3.1.	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0105 RQ10_0106 RQ10_0107
6	UICC → T	Status words SW1 SW2 set to "More data blocks pending" '92 07'. Session Number =3.	
7	T → UICC	Send TRANSACT DATA command. First block of response data as detailed in table 6.4.1.1.3.6.	RQ08_0104 RQ08_0106
8	UICC → T	Return Data and Status Words as detailed in clause 4.4.6.5.1 Transact Data Response 1.	

Table 6.4.1.1.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'FF'	Encrypted Blob TLV - See table 6.4.1.1.3.2

Table 6.4.1.1.3.2: Secure Channel TLV for Transact Data Command

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 FC'	The data in table 6.4.1.1.3.3 encrypted using the encryption method and encryption Key agreed for the current secure channel	'000000000000000000'

Table 6.4.1.1.3.3: Encrypted Blob TLV

Type	Tag	Length	Value
Encrypted Data Container	'81'	'81 F0'	The data in table 6.4.1.1.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.1.1.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 231	APDU Command BER-TLV	215	APDU BER TLV - See table 6.4.1.1.3.5
232	Padding	1	1 byte random number
233 to 240	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel

Table 6.4.1.1.3.5

Byte(s)	Description	Length	Value
1	Tag	1	8 byte random value
2 to 3	Length	2	'81 D4'
4 to 215	APDU	212	Value contains APDU as defined in clause 4.4.7.1.1 Transact Data Command APDU 1

Table 6.4.1.1.3.6

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'75'	11000000	'00'	'FF'

6.4.1.2 Test case 2: Transact Data - Command Data in 2 secure channel TLVs

6.4.1.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- The Terminal shall support at least a minimal application as defined in clause 4.4.7.1.1 which shall send a APDU command defined in table 4.4.7.1.1.1.
- The Terminal shall support the endpoint container size of 127 as indicated in the retrieve endpoint provided by the UICC simulator.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.1.2.2 Initial conditions

- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.

6.4.1.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as detailed in table 6.4.1.2.3.1	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0103 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0106 RQ10_0107
6	UICC → T	Status words SW1 SW2 set to "Data Transaction ongoing - Send next block" '92 06' Session Number = 3	
7	T → UICC	Send TRANSACT DATA command Second block of command data as detailed in table 6.4.1.2.3.6	RQ10_0105
8	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
9	T → UICC	Send TRANSACT DATA command First block of response data as detailed in table 6.4.1.2.3.8	RQ08_0104 RQ08_0106
10	UICC → T	Status Words SW1 SW2 set to "Normal ending of command" '90 00' As detailed in clause 4.4.6.5.1 Transact Data Response 1	

Table 6.4.1.2.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'7F'	Encrypted Blob TLV - See table 6.4.1.2.3.2

Table 6.4.1.2.3.2

Tag	Length	Value	Padding
'80'	'7D'	First 125 bytes of the Encrypted Blob TLV as detailed in table 6.4.1.2.3.3	None

Table 6.4.1.2.3.3: Encrypted Blob TLV

Tag	Length	Value
'81	'81 F0'	The data in table 6.4.1.2.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.1.2.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 231	APDU Command BER-TLV	215	APDU BER TLV - See table 6.4.1.2.3.5
232	Padding	1	1 byte random number
233 to 240	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel

Table 6.4.1.2.3.5

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2 to 3	Length	2	'81 D4'
4 to 215	APDU	212	Value contains APDU as defined in clause 4.4.7.1.1 Transact Data Command APDU 1

Table 6.4.1.2.3.6

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'7F'	Secure Channel TLV - See table 6.4.1.2.3.7

Table 6.4.1.2.3.7

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'7D'	Last 116 bytes of the Encrypted Blob TLV see clause 4.4.5.1	'000000000000000000'

Table 6.4.1.2.3.8

Code	CLA	INS	P1	P2	Le
Value	'0X', '4X' or '6X'	'75'	11000000	'00'	'7F'

6.4.1.3 Test case 3: Transact Data - Command Data in 25 secure channel TLVs

6.4.1.3.1 Test execution

The test procedure shall only be executed for the following considerations:

- The Terminal shall support at least a minimal application as defined in clause 4.4.7.1.1 which shall send a APDU command defined in table 4.4.7.1.1.1.
- The Terminal shall support the endpoint container size of 10 as indicated in the retrieve endpoint provided by the UICC simulator.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.1.3.2 Initial conditions

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel with the application on the UICC Simulator.
- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.
- The UICC has Endpoints that can handle an endpoint data container size of 255 bytes.

6.4.1.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command. First block of command data as detailed in table 6.4.1.3.3.1	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0103 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0106 RQ10_0107
6	UICC → T	Status words SW1 SW2 set to "Data Transaction ongoing - Send next block" '92 06' Session Number = 3	
7	T → UICC	Send TRANSACT DATA command Next block of command data as detailed in table 6.4.1.3.3.6	RQ10_0105
8	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - Send next block" '92 06'	
	T → UICC → T	Repeat steps 7 and 8 until 24 TRANSACT DATA commands have been sent and responded to.	
9	UICC → T	Status words SW1 SW2 set to "Data transaction ongoing - Send next block" '92 06' Session Number = 3	
10	T → UICC	Send TRANSACT DATA command Last block of command data as detailed in table 6.4.1.3.3.8	
11	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
12	T → UICC	Send TRANSACT DATA command Request first block of response data from UICC	RQ08_0104 RQ08_0106
13	UICC → T	Return SW1 SW2 = 'normal ending of command' as detailed in clause 4.4.6.5.1 Transact Data Response 1	

Table 6.4.1.3.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'0A'	Secure channel TLV - See table 6.4.1.3.3.2

Table 6.4.1.3.3.2

Tag	Length	Value	Padding
'80'	'08'	First 8 bytes of the Encrypted Blob TLV see table 6.4.1.3.3.3	None

Table 6.4.1.3.3.3: Encrypted Blob TLV

Tag	Length	Value
'81'	'81 F0'	The data in table 6.4.1.3.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel This TLV is calculated once for each test

Table 6.4.1.3.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 231	APDU Command BER-TLV	215	APDU BER TLV - See table 6.4.1.3.3.5
232	Padding	1	1 byte random number
233 to 240	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel

Table 6.4.1.3.3.5

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2 to 3	Length	2	'81 D4'
4 to 215	APDU	212	Value contains APDU as defined in clause 4.4.7.1.1 Transact Data Command APDU 1

Table 6.4.1.3.3.6: Second to twenty fourth command to send

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'17' for the second command reducing by 1 for each subsequent command.	'0A'	Secure channel TLV - See table 6.4.1.3.3.7

Table 6.4.1.3.3.7

Tag	Length	Value	Padding
'80'	'08'	next 8 bytes of the Encrypted Blob TLV see Table 6.4.1.3.3.3	None

Table 6.4.1.3.3.8: Twenty fifth command to send

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'00'	'0A'	Secure channel TLV - See table 6.4.1.3.3.9

Table 6.4.1.3.3.9

Tag	Length	Value	Padding
'80'	'08'	Last 3 bytes of the Encrypted Blob TLV see table 6.4.1.3.3.3	'00 00 00 00 00'

6.4.1.4 Test case 4: Transact Data - Command Data Maximum Size APDU

6.4.1.4.1 Test execution

The test procedure shall only be executed for the following considerations:

- The Terminal shall support at least a minimal application as defined in clause 4.4.7.1.1 which shall send a APDU command defined in table 4.4.7.1.1.1.
- The Terminal shall support the endpoint container size of 160 as indicated in the retrieve endpoint provided by the UICC simulator.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.1.4.2 Initial conditions

- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.

6.4.1.4.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as in table 6.4.1.4.3.1	RQ01_0105 RQ02_0107 RQ04_0101 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0103
6	UICC → T	Status words SW1 SW2 set to "Data Transaction ongoing - Send next block" '92 06' Session Number = 3	
7	T → UICC	Send TRANSACT DATA command Second block of command data as detailed in table 6.4.1.4.3.6	RQ08_0104
8		Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
9	T → UICC	Send TRANSACT DATA command First block of response data as detailed in table 6.4.1.4.3.8	
10	UICC → T	Return SW1 SW2 = 'normal ending of command' as detailed in clause 4.4.6.5.1 Transact Data Response 1	

Table 6.4.1.4.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'A0'	Secure channel data TLV - See table 6.4.1.4.3.2

Table 6.4.1.4.3.2

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'819D'	First 157 bytes of the Encrypted Blob TLV see table 6.4.1.4.3.3	No padding used

Table 6.4.1.4.3.3

Encrypted Blob Tag	Length	Value
'81'	'820120'	The data in table 6.4.1.4.3.4 encrypted using the encryption method and encryption Key agreed on for the current secure channel. This TLV is calculated once for each test.

Table 6.4.1.4.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	Random 8 byte number
9 to 16	Counter	8	The next valid counter value for the current secure channel
17 to 280	APDU Command BER-TLV	264	APDU BER TLV - see table 6.4.1.4.3.5
-	Padding	0	None
281 to 288	Checksum	8	Calculated as per clause 10.1.1 TS 102 484 [1]

Table 6.4.1.4.3.5: Coding of the APDU BER-TLV object

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2 to 4	Length	3	'820104'
5 to 264	APDU	260	APDU command to be encapsulated - see clause 4.4.7.1.2 Transact Data Command APDU 2

Table 6.4.1.4.3.6: Second Command

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'00'	'A0'	Secure channel data TLV - See table 6.4.1.4.3.7

Table 6.4.1.4.3.7

Tag	Length	Value	Padding
'80'	'81 9D'	Last 135 bytes of the Encrypted Blob TLV see table 6.4.1.4.3.3	No padding used

Table 6.4.1.4.3.8

Code	CLA	INS	P1	P2	Le	Data
Value	'0X', '4X' or '6X'	75	11000000	'00'	'A0'	none

6.4.2 Sub Test group 4.2 Transact Data - Response Data

6.4.2.1 Test case 1: Transact Data - Response Data in 1 secure channel TLV

6.4.2.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.4.2.1.2 Initial conditions

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel.
- The secure channel MSA and CSA have been that has been negotiated and the secure channel has not been started.
- The UICC has an Endpoint that can handle an endpoint data container size of 255 bytes.

6.4.2.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as in table 6.4.4.1.3.1	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0108 RQ08_0109 RQ08_0110 RQ08_0111 RQ08_0112 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0105 RQ10_0106 RQ10_0107
6	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
7	T → UICC	Send TRANSACT DATA command First block of response data as detailed in table 6.4.4.1.3.6	RQ08_0104 RQ08_0106
8	UICC → T	Return data and status words encrypted as detailed in clause 4.4.6.5.2 Transact Data Response 2	

Table 6.4.2.1.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'00'	'FF'	Encrypted Blob TLV - See table 6.4.2.1.3.2

Table 6.4.2.1.3.2

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 FC'	The data in table 6.4.2.1.3.3 encrypted using the encryption method and encryption Key agreed for the current secure channel	'00 ... 00' (218 bytes)

Table 6.4.2.1.3.3

Type	Tag	Length	Value
Encrypted Data Container	'81'	'20'	The data in table 6.4.2.1.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.2.1.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	Random 8 byte number
9 to 16	Counter	8	The next valid counter value for the current secure channel.
17 to 23	APDU Command BER-TLV	7	APDU BER TLV - see table 6.4.2.1.3.5
24	Padding	1	1 byte random number
25 to 32	Checksum	8	Calculated as per clause 10.1.1 TS 102 484 [1]

Table 6.4.2.1.3.5: Coding of the APDU BER-TLV object

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2	Length	2	'05'
3 to 7	APDU	5	Value contains APDU as defined in clause 4.4.7.1.3 Transact Data Command APDU 3

Table 6.4.2.1.3.6

Code	CLA	INS	P1	P2	Le	DATA
Value	'0X', '4X' or '6X'	'75'	11000000	'00'	'FF'	None

6.4.2.2 Test case 2: Transact Data - Response Data in 2 secure channel TLVs

6.4.2.2.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.4.2.2.2 Initial conditions

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel with the application on the UICC Simulator.
- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.
- The UICC has Endpoints that can handle an endpoint data container size of 150 bytes.

6.4.2.2.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as in table 6.4.2.2.3.1	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0103 RQ08_0108 RQ08_0109 RQ08_0110 RQ08_0111 RQ08_0112 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0105 RQ10_0106 RQ10_0107
6	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
7	T → UICC	Send TRANSACT DATA command First block of response data as detailed in table 6.4.2.2.3.6	RQ08_0104 RQ08_0106
8	UICC → T	Return data and status words encrypted as detailed in table 6.4.2.2.3.7	
9	T → UICC	Send TRANSACT DATA command Second block of response data as detailed in table 6.4.2.2.3.9	
10	UICC → T	Return data and status words encrypted as detailed in table 6.4.2.2.3.10	

Table 6.4.2.2.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'00'	'96'	Secure channel data TLV - See table 6.4.2.2.3.2

Table 6.4.2.2.3.2

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 93'	Encrypted Blob TLV - see table 6.4.2.2.3.3	'00 .. 00' (113 bytes)

Table 6.4.2.2.3.3

Type	Tag	Length	Value
Encrypted Data Container	'81'	'20'	The data in table 6.4.2.2.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.2.2.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	Random 8 byte number
9 to 16	Counter	8	The next valid counter value for the current secure channel.
17 to 23	APDU Command BER-TLV	7	APDU BER TLV - see table 6.4.2.2.3.5
24	Padding	1	1 byte random number
25 to 32	Checksum	8	Calculated as per clause 10.1.1 TS 102 484 [1]

Table 6.4.2.2.3.5: Coding of the APDU BER-TLV object

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2	Length	2	'05'
3 to 7	APDU	5	Value contains APDU as defined in clause 4.4.7.1.3 Transact Data Command APDU 3

Table 6.4.2.2.3.6

Code	CLA	INS	P1	P2	Le	Data
Value	'0X', '4X' or '6X'	'75'	1100000	'00'	'96'	None

Table 6.4.2.2.3.7

Data	SW1	SW2
See table 6.4.2.2.3.8	'92'	'06'

Table 6.4.2.2.3.8

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 93'	First 147 bytes of Encrypted Blob TLV as detailed in clause 4.4.6.5.2 Transact Data Response 2	None

Table 6.4.2.2.3.9

Code	CLA	INS	P1	P2	Le	DATA
Value	'0X', '4X' or '6X'	'75'	1100000	'00'	'96'	None

Table 6.4.2.2.3.10

Data	SW1	SW2
See table 6.4.2.2.3.11	'92'	'06'

Table 6.4.2.2.3.11

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 93'	Last 141 bytes of Encrypted Blob TLV as detailed in clause 4.4.6.5.2 Transact Data Response 2	'00..00' (6 bytes)

6.4.2.3 Test case 3: Transact Data - Response Data in 25 secure channel TLVs

6.4.2.3.1 Test execution

The test procedure shall only be executed for the following considerations:

- There are no test case-specific parameters for this test case.

6.4.2.3.2 Initial conditions

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel with the application on the UICC Simulator.
- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.
- The UICC has Endpoints that can handle an endpoint data container size of 10 bytes.

6.4.2.3.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as in table 6.4.2.3.3.1	RQ01_0105 RQ07_0118 RQ07_0119 RQ07_0120 RQ07_0121 RQ08_0101 RQ08_0102 RQ08_0103 RQ08_0108 RQ08_0109 RQ08_0110 RQ08_0111 RQ08_0112 RQ10_0101 RQ10_0102 RQ10_0103 RQ10_0104 RQ10_0105 RQ10_0106 RQ10_0107
6	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
7	T → UICC	Send TRANSACT DATA command. First block of response data as detailed in table 6.4.2.3.3.6	RQ08_0104 RQ08_0106
8	UICC → T	Return data and status words encrypted as detailed in table 6.4.2.2.3.7	
15	T → UICC → T	Repeat steps 9 to 10 until 25 TRANSACT DATA commands as defined in clause have been sent and responded to	

Table 6.4.2.3.3.1

Code	CLA	INS	P1	P2	Lc	Data
Value	'0X', '4X' or '6X'	'75'	11000100	Defined in Test Procedure	'0A'	Secure channel data TLV - See table 6.4.2.3.3.2

Table 6.4.2.3.3.2

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'08'	First 8 bytes of Encrypted Blob TLV - see table 6.4.2.3.3.3	None

Table 6.4.2.3.3.3

Type	Tag	Length	Value
Encrypted Data Container	'81'	'20'	The data in table 6.4.2.3.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.2.3.3.4: Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	Random 8 byte number
9 to 16	Counter	8	The next valid counter value for the current secure channel.
17 to 23	APDU Command BER-TLV	7	APDU BER TLV - see table 6.4.2.2.3.5
24	Padding	1	1 byte random number
25 to 32	Checksum	8	Calculated as per clause 10.1.1 TS 102 484 [1]

Table 6.4.2.3.3.5: Coding of the APDU BER-TLV object

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2	Length	2	'05'
3 to 7	APDU	5	Value contains APDU as defined in clause 4.4.7.1.3 Transact Data Command APDU 3

Table 6.4.2.3.3.6

Code	CLA	INS	P1	P2	Le	DATA
Value	'0X', '4X' or '6X'	'75'	1100000	'00'	'96'	None

Table 6.4.2.3.3.7

Data	SW1	SW2
See table 6.4.2.2.3.8	'92'	'06'

Table 6.4.2.3.3.8

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 93'	First 147 bytes of Encrypted Blob TLV as detailed in clause 4.4.6.5.2 Transact Data Response 2	No padding used

Table 6.4.2.3.3.9

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	XX00000	'00'	'96'	Encrypted Blob TLV - See table 6.4.2.3.3.5

Table 6.4.2.3.3.10

Data	SW1	SW2
See table 6.4.2.3.3.11	'92'	'06'

Table 6.4.2.3.3.11

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'81 93'	Last 141 bytes of Encrypted Blob TLV The data in encrypted using the encryption method and encryption Key agreed for the current secure channel	No padding used

6.4.3 Sub Test group 4.3 Retransmission

6.4.3.1 Test case 1: Transact Data using resend mechanism

6.4.3.1.1 Test execution

The test procedure shall only be executed for the following considerations:

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel with the application on the UICC Simulator.
- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.
- The UICC has Endpoints that can handle an endpoint data container size of 248 bytes.
- Applicable The Terminal shall support the endpoint container size as indicated in the retrieve endpoint provided by the UICC simulator.

The test procedure shall be performed with variation in the following values and combinations:

- The test procedure shall be performed with the Ciphering and Integrity key as chosen by the Terminal.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.3.1.2 Initial conditions

- The Terminal has an application that is able to manage and communicate using an APDU application - application secure channel with the application on the UICC Simulator.
- The secure channel Master SA and Connection SA have been negotiated and the secure channel has not been started.
- The UICC has Endpoints that can handle an endpoint data container size of 127 bytes.

6.4.3.1.3 Test procedure

Step	Direction	Description	RQ
1	T → UICC	Send Manage Secure Channel - Start Secure Channel command	
2	UICC → T	Normal ending of command	
3	T → UICC	Send Manage Secure Channel - Start Secure Channel response	
4	UICC → T	Return Session Number = 3 Data as detailed in clause 4.4.6.4	
5	T → UICC	Send TRANSACT DATA command First block of command data as in table 6.4.3.1.3.1	RQ08_0103 RQ08_0105
6	UICC → T	Status Words SW1 SW2 set to = '92 16'	
7	T → UICC	Send TRANSACT DATA command. Command data repeated in step 5	
8		Status words SW1 SW2 set to "Data Transaction ongoing - Send next block" '92 06' Session Number = 3	
9	T → UICC	Send TRANSACT DATA command Second block of command data as detailed in table 6.4.3.1.3.7	
10	UICC → T	Status Words SW1 SW2 set to "Data transaction ongoing - More data blocks pending" '92 07'	
11	T → UICC	Send TRANSACT DATA command First block of response data as detailed in table 6.4.3.1.3.8	
12	UICC → T	Status Words SW1 SW2 set to "Normal ending of command" '90 00' as detailed in clause 4.4.6.5.1 Transact Data Response 1	

Table 6.4.3.1.3.1

Code	CLA	INS	P1	P2	Lc	DATA
Value	'0X', '4X' or '6X'	'75'	11000100	'01'	'7F'	Encrypted Blob TLV - See table 6. 6.4.3.1.3.2

Table 6.4.3.1.3.2

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'7D'	First 125 bytes of the Encrypted Blob TLV as detailed in table 6.4.3.1.3.3	No padding used

Table 6.4.3.1.3.3 Encrypted Blob TLV

Tag	Length	Value
'81'	'81 F0'	The data in table 6.4.3.1.3.4 encrypted using the encryption method and encryption Key agreed for the current secure channel

Table 6.4.3.1.3.4 Unencrypted data for the Encrypted Blob TLV

Byte(s)	Description	Length	Value
1 to 8	Nonce	8	8 byte random value
9 to 16	Counter	8	The next valid counter value for the current secure channel.
17 to 231	APDU Command BER-TLV	215	APDU BER TLV - See table 6.4.3.1.3.5
232	Padding	1	1 byte random number
233 to 240	Checksum	8	Checksum algorithm used shall be the agreed integrity mechanism for this secure channel

Table 6.4.3.1.3.5

Byte(s)	Description	Length	Value
1	Tag	1	'82'
2 to 3	Length	2	'81 D4'
4 to 215	APDU	212	Value contains APDU as defined in clause 4.4.7.1.1 Transact Data Command APDU 1

Table 6.4.3.1.3.6

Code	CLA	INS	P1	P2	Lc	DATA
Value	XX	'75'	11000100	01	7F	Last 116 bytes of Encrypted Blob TLV - See table 6.4.3.1.3.3

Table 6.4.3.1.3.7

Type	Tag	Length	Value	Padding
Encrypted Data Container	'80'	'7D'	Last 116 bytes of the Encrypted Blob TLV see table in clause 4.4.5.1	No padding used

Table 6.4.3.1.3.8

Code	CLA	INS	P1	P2	Le	Data
Value	'0X', '4X' or '6X'	'75'	11000000	'00'	'7F'	none

6.4.4 Sub Test group 4.4 Transact Data - - Multiple secure channels

6.4.4.1 Test case 1: Transact Data - Multiple secure channels in 1 message

6.4.4.1.1 Test execution

This test verifies that the DUT is able to receive process and respond to two TRANSACT DATA sessions interleaved.

The test procedure shall only be executed for the following considerations:

- Applicable The Terminal shall support the endpoint container size as indicated in the retrieve endpoint provided by the UICC simulator.

The test procedure shall be performed with variation in the parameters Ciphering and Integrity keys, in following values and combinations:

- The test procedure shall be performed with the Ciphering and Integrity key as chosen by the Terminal.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.4.1.2 Initial conditions

- The Terminal has 2 applications that are able to manage and communicate using an APDU application - application secure channel.
- The secure channel Master SAs and Connection SAs have been negotiated and neither secure channel has been started.
- One Terminal Endpoint can handle an endpoint data container size of 127 bytes and the other can handle a endpoint container size of 255 bytes.

6.4.4.1.3 Test procedure

The test procedures as detailed in clauses 6.4.1.1.3 and 6.4.1.2.3 the steps may be interleaved with each other or carried out one after the other.

Step	Direction	Description	RQ
1	T → UICC → T	The test procedures as detailed in clauses 6.4.1.1.3 and 6.4.1.2.3 the steps may be interleaved with each other or carried out one after the other.	RQ01_0105 RQ04_0201

6.4.4.2 Test case 2: Transact Data - Multiple secure channels with different secure channel block sizes

6.4.4.2.1 Test execution

This test verifies that the DUT is able to receive process and respond to two TRANSACT DATA sessions interleaved.

The test procedure shall only be executed for the following considerations:

- Applicable The Terminal shall support the endpoint container size as indicated in the retrieve endpoint provided by the UICC simulator.

The test procedure shall be performed with variation in the parameters Ciphering and Integrity keys, in following values and combinations:

- The test procedure shall be performed with the Ciphering and Integrity key as chosen by the Terminal.
- The test shall be repeated for each of the supported Ciphering and Integrity key where this is selectable by the Terminal Application.

6.4.4.2.2 Initial conditions

- The Terminal has 2 applications that are able to manage and communicate using an APDU application - application secure channel.
- The secure channel Master SAs and Connection SAs have been negotiated and neither secure channel has been started.
- One Terminal Endpoint can handle an endpoint data container size of 127 bytes and the other can handle a endpoint container size of 255 bytes.

6.4.4.2.3 Test procedure

The test procedures as detailed in clauses 6.4.1.1.3 and 6.4.1.2.3 the steps may be interleaved with each other or carried out one after the other.

Step	Direction	Description	RQ
1	T → UICC → T	The test procedures as detailed in clauses 6.4.1.1.3 and 6.4.1.2.3 the steps may be interleaved with each other or carried out one after the other.	RQ04_0201

Annex A (informative): List of test cases for each conformance requirement

A.1 Secure Channel Properties

Reference: TS 102 484 [1], clause 5.

A.1.1 Secure Channel Lifecycle and Discovery

RQ number	Test clauses
RQ01_0101	6.1.1.1
RQ01_0102	6.2.1.1, 6.2.1.2, 6.2.1.3, 6.2.1.4
RQ01_0103	6.2.1.1, 6.2.2.1, 6.2.3.1
RQ01_0104	6.2.4.1
RQ01_0105	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ01_0106	6.2.5.3, 6.2.5.4
RQ01_0107	6.2.5.1, 6.2.5.2
RQ01_0201	6.1.1.1

A.1.2 Secure Channel Administration

RQ number	Test clauses
RQ02_0101	6.2.2.1, 6.2.3.1, 6.2.4.1
RQ02_0103	6.2.1.2
RQ02_0104	6.2.3.1, 6.2.4.1
RQ02_0105	6.2.3.1, 6.2.4.1
RQ02_0106	6.2.3.1, 6.2.4.1
RQ02_0107	6.2.2.1, 6.2.3.1
RQ02_0108	6.2.2.1
RQ02_0109	6.2.3.1
RQ02_0110	6.2.2.1, 6.2.3.1
RQ02_0111	6.2.2.4, 6.2.3.3
RQ02_0112	Not Testable
RQ02_0201	6.2.2.1, 6.2.2.4
RQ02_0202	6.2.2.1, 6.2.2.4
RQ02_0203	6.2.2.1, 6.2.2.4
RQ02_0204	6.2.2.1, 6.2.2.4
RQ02_0205	No mechanism detailed within core specification
RQ02_0206	6.2.2.1, 6.2.2.4
RQ02_0207	6.2.2.4
RQ02_0208	6.2.5.1
RQ02_0209	6.2.3.1, 6.2.3.3
RQ02_0210	6.2.4.1
RQ02_0301	6.2.3.1, 6.2.3.3
RQ02_0302	6.2.3.1, 6.2.3.3, 6.2.5.2
RQ02_0303	6.2.5.2
RQ02_0304	6.2.5.1
RQ02_0305	6.2.5.2
RQ02_0306	6.2.5.4
RQ02_0307	6.2.3.3

A.1.3 Key Agreement

RQ number	Test clauses
RQ03_0101	Not Testable
RQ03_0102	Not Testable
RQ03_0103	Not Testable
RQ03_0104	Not Testable
RQ03_0105	Not Testable
RQ03_0106	Not Testable
RQ03_0107	Not Testable
RQ03_0108	Not Testable
RQ03_0109	Not Testable
RQ03_0110	Not Testable
RQ03_0111	Not Testable
RQ03_0112	Not Testable
RQ03_0201	Not Testable
RQ03_0202	Not Testable
RQ03_0203	Not Testable
RQ03_0204	Not Testable
RQ03_0205	Not Testable
RQ03_0206	Not Testable
RQ03_0207	Not Testable
RQ03_0208	Not Testable
RQ03_0209	Not Testable
RQ03_0210	Not Testable
RQ03_0211	Not Testable
RQ03_0212	Not Testable
RQ03_0301	Not Testable
RQ03_0302	Not Testable
RQ03_0401	Not Testable
RQ03_0402	Not Testable
RQ03_0403	Not Testable
RQ03_0501	Not Testable
RQ03_0502	Not Testable
RQ03_0503	Not Testable
RQ03_0504	Not Testable
RQ03_0505	Not Testable
RQ03_0506	Not Testable

A.1.4 Secure Channel Operation

RQ number	Test clauses
RQ04_0102	6.2.5.2
RQ04_0103	6.2.5.3
RQ04_0201	6.2.2.4, 6.2.3.3
RQ04_0302	Not testable
RQ04_0303	Not testable

A.2 Secured APDU - Application to Application lifecycle

Reference: TS 102 484 [1], clause 7.

A.2.1 Discovery

RQ number	Test clauses
RQ05_0101	6.2.1.1,6.2.1.2, 6.2.1.3, 6.2.1.4

A.2.2 Master SA setup

RQ number	Test clauses
RQ06_0101	6.2.2.1
RQ06_0102	6.2.2.1
RQ06_0103	6.2.2.1
RQ06_0104	6.2.2.2
RQ06_0105	6.2.2.2
RQ06_0106	Not Testable
RQ06_0107	Not Testable
RQ06_0108	Not Testable
RQ06_0109	Not Testable
RQ06_0110	Not Testable
RQ06_0111	Not Testable
RQ06_0112	Not Testable
RQ06_0114	Not Testable
RQ06_0115	Not Testable
RQ06_0116	Not Testable
RQ06_0117	Not Testable
RQ06_0118	Not Testable
RQ06_0119	Not Testable

A.2.3 Connection SA setup

RQ number	Test clauses
RQ07_0101	6.2.3.1, 6.2.3.3
RQ07_0102	6.2.3.1, 6.2.3.3
RQ07_0103	6.2.3.1, 6.2.3.3
RQ07_0104	6.2.3.1, 6.2.3.3
RQ07_0105	6.2.3.1, 6.2.3.3
RQ07_0106	6.2.3.1, 6.2.3.3
RQ07_0107	6.2.3.1, 6.2.3.3
RQ07_0108	6.2.3.1, 6.2.3.3
RQ07_0109	6.2.3.1, 6.2.3.3
RQ07_0110	6.2.3.1, 6.2.3.3, 6.2.4.1
RQ07_0111	6.2.4.1
RQ07_0112	6.2.4.1
RQ07_0113	6.2.4.1
RQ07_0114	6.2.3.2
RQ07_0115	6.2.4.1
RQ07_0116	6.2.4.2
RQ07_0117	6.2.5.2
RQ07_0118	6.2.5.2
RQ07_0119	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ07_0120	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ07_0121	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3

A.2.4 Secure Connection Initiation and Data Transmission

RQ number	Test clauses
RQ08_0101	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0102	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0103	6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.3.1, 6.4.2.2, 6.4.2.3
RQ08_0104	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0105	6.4.3.1
RQ08_0106	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.1.4, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0107	6.2.3.1, 6.2.3.3
RQ08_0108	6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0109	6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0110	6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0111	6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0112	6.4.2.1, 6.4.2.2, 6.4.2.3
RQ08_0113	Not Testable

A.2.5 SA Termination and Resumption

RQ number	Test clauses
RQ09_0101	6.2.5.1
RQ09_0102	6.2.5.2
RQ09_0103	6.2.5.2
RQ09_0104	6.2.5.1
RQ09_0105	6.2.5.4.3, 6.2.5.1, 6.2.5.2
RQ09_0106	6.2.5.4
RQ09_0107	Not Tested

A.3 Encrypted data coding

Reference: TS 102 484 [1], clause 10.

RQ number	Test clauses
RQ10_0101	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0102	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0103	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0104	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0105	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0106	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3
RQ10_0107	6.4.1.1, 6.4.1.2, 6.4.1.3, 6.4.2.1, 6.4.2.2, 6.4.2.3

A.4 Key Expansion Function Definition

Reference: TS 102 484 [1], clause 11.

RQ number	Test clauses
RQ11_0101	Not Testable

A.5 ATR

Reference: TS 102 221 [2], clause 6.

RQ number	Test clauses
RQ11_0201	6.1.1.1

A.6 MANAGE SECURE CHANNEL Command

Reference: TS 102 221 [2], clause 11.

RQ number	Test clauses
RQ12_0101	6.2.1.2, 6.2.2.1, 6.2.3.1, 6.2.4.1, 6.2.5.1
RQ12_0102	6.2.1.2, 6.2.2.1, 6.2.3.1, 6.2.4.1, 6.2.5.1
RQ12_0103	6.2.1.4
RQ12_0104	6.2.1.4
RQ12_0105	6.2.1.2, 6.2.2.1, 6.2.3.1, 6.2.4.1, 6.2.5.1
RQ12_0106	Not Testable
RQ12_0107	6.2.1.4
RQ12_0108	6.2.1.4
RQ12_0109	6.2.1.2, 6.2.2.1, 6.2.3.1, 6.2.4.1, 6.2.5.1
RQ12_0110	6.2.1.4
RQ12_0201	6.2.1.2
RQ12_0202	6.2.1.2
RQ12_0203	6.2.1.4
RQ12_0301	6.2.2.1
RQ12_0302	6.2.2.1
RQ12_0303	6.2.2.1
RQ12_0304	6.2.2.1
RQ12_0305	6.2.2.1
RQ12_0306	6.2.2.1
RQ12_0307	6.2.2.1
RQ12_0308	6.2.2.1
RQ12_0309	6.2.2.1
RQ12_0401	6.2.3.1
RQ12_0402	6.2.3.1
RQ12_0403	6.2.3.1
RQ12_0404	6.2.3.1
RQ12_0405	6.2.3.1
RQ12_0406	6.2.3.1
RQ12_0408	6.2.3.1
RQ12_0409	6.2.3.1
RQ12_0501	6.2.4.1
RQ12_0601	6.2.4.1
RQ12_0602	6.2.4.1
RQ12_0603	6.2.4.1
RQ12_0604	6.2.4.1
RQ12_0605	6.2.4.1
RQ12_0606	6.2.4.1
RQ12_0607	6.2.4.1
RQ12_0608	6.2.4.1
RQ12_0609	6.2.4.1
RQ12_0701	6.2.5.1
RQ12_0801	6.2.5.1
RQ12_0802	6.2.5.1
RQ12_0803	6.2.5.1
RQ12_0804	6.2.5.1
RQ12_0805	6.2.5.1
RQ12_0806	6.2.5.1
RQ12_0807	6.2.5.1
RQ12_0808	6.2.5.1

A.7 TRANSACT DATA Command

Reference: TS 102 221 [2], clause 11.

RQ number	Test clauses
RQ13_0101	6.4.1.4
RQ13_0102	6.4.1.1, 6.4.2.1
RQ13_0103	6.4.1.1
RQ13_0104	6.4.1.1
RQ13_0105	6.4.3.1
RQ13_0106	Not testable
RQ13_0107	6.4.4.1, 6.4.4.2
RQ13_0108	6.4.1.3
RQ13_0109	Not Testable
RQ13_0110	Not Testable
RQ13_0201	6.4.1.1
RQ13_0202	6.4.2.3
RQ13_0203	6.4.1.1
RQ13_0204	6.4.1.1
RQ13_0205	6.4.1.3
RQ13_0206	6.4.1.1

Annex B (informative): Core specification version information

Unless otherwise specified, the versions of TS 102 484 [1] from which conformance requirements have been extracted are as follows.

Release	Latest version from which conformance requirements have been extracted
7	V7.8.0
8	V8.2.0
9	V9.2.0

Unless otherwise specified, the versions of TS 102 221 [2] from which conformance requirements have been extracted are as follows.

Release	Latest version from which conformance requirements have been extracted
7	V7.18.0
8	V8.5.0
9	V9.2.0

History

Document history		
V9.0.0	May 2013	Publication