BS ISO/IEC 24727-5:2011

BSI Standards Publication

Identification cards — Intergrated circuit card programming interface

Part 5: Testing procedures

bsi.

...making excellence a habit.™

Document Sharing Hub

Full document details: BS ISO/IEC 24727-5:2011

National foreword

This British Standard is the UK implementation of ISO/IEC 24727-5:2011.

The UK participation in its preparation was entrusted to Technical Committee IST/17, Cards and personal identification.

A list of organizations represented on this committee can be obtained on request to its secretary.

This publication does not purport to include all the necessary provisions of a contract. Users are responsible for its correct application.

© BSI 2011

ISBN 978 0 580 62872 6

ICS 35.240.15

Compliance with a British Standard cannot confer immunity from legal obligations.

This British Standard was published under the authority of the Standards Policy and Strategy Committee on 30 April 2011.

Amendments issued since publication

Date	Text affected

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 24727-5:2011

INTERNATIONAL STANDARD

ISO/IEC 24727-5

First edition
2011-03-15

Identification cards — Integrated circuit card programming interfaces —

Part 5: Testing procedures

Cartes d'identification — Interfaces programmables de cartes à puce —

Partie 5: Essais

Reference number
ISO/IEC 24727-5:2011(E)

Documents
Document Sharing Hub

© ISO/IEC 2011

Full document details: BS ISO/IEC 24727-5:2011

PDF disclaimer

This PDF file may contain embedded typefaces. In accordance with Adobe's licensing policy, this file may be printed or viewed but shall not be edited unless the typefaces which are embedded are licensed to and installed on the computer performing the editing. In downloading this file, parties accept therein the responsibility of not infringing Adobe's licensing policy. The ISO Central Secretariat accepts no liability in this area.

Adobe is a trademark of Adobe Systems Incorporated.

Details of the software products used to create this PDF file can be found in the General Info relative to the file; the PDF-creation parameters were optimized for printing. Every care has been taken to ensure that the file is suitable for use by ISO member bodies. In the unlikely event that a problem relating to it is found, please inform the Central Secretariat at the address given below.

COPYRIGHT PROTECTED DOCUMENT

© ISO/IEC 2011

All rights reserved. Unless otherwise specified, no part of this publication may be reproduced or utilized in any form or by any means, electronic or mechanical, including photocopying and microfilm, without permission in writing from either ISO at the address below or ISO's member body in the country of the requester.

ISO copyright office
Case postale 56 • CH-1211 Geneva 20
Tel. + 41 22 749 01 11
Fax + 41 22 749 09 47
E-mail copyright@iso.org
Web www.iso.org

Published in Switzerland

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 24727-5:2011

Contents

Page

Foreword vi

Introduction vii

1 Scope 1

2 Normative references 1

3 Terms and definitions 2

4 Symbols and abbreviated terms 3

5 Testing methodology 4
5.1 Terms of testing 4
5.1.1 Purpose of testing 4
5.1.2 Testing objective 4
5.1.3 Testing Principles 4
5.1.4 GCI under test: 6
5.1.5 SAL under test: 6
5.1.6 Conformance attainment 7
5.2 Conformance vector 8
5.3 Structure of tests 10
5.4 Test environment 12
5.4.1 Stack configurations 12
5.4.2 Card-application emulators 12
5.4.3 Verification and logging capability of components 12
5.4.4 Procedural element 12

6 Components 12
6.1 Service access layer API 12
6.1.1 Basic tests 12
6.1.2 Discoverability tests 12
6.2 Generic card interface 15
6.2.1 Basic test 15
6.2.2 Processing tests 15
6.2.3 Discoverability tests 17
6.2.4 Generic card interface acted on ISO/IEC 24727-2 implementation (i.e. CLA = "FF") 18
6.3 Interface device API 19
6.4 Trusted channel API 19
6.4.1 TC_API_Open 19
6.4.2 TC_API_Close 19
6.4.3 TC_API Write 19
6.4.4 TC_API Read 19
6.5 SAL on-card implementation component testing 20

7 Authentication protocols 20
7.1 General 20
7.2 SAL security test sequences 21
7.2.1 Cryptographic operations 22
7.2.2 Simple assertion 26
7.2.3 Asymmetric internal authenticate 27
7.2.4 Asymmetric external authenticate 28
7.2.5 Symmetric internal authenticate 30
7.2.6 Symmetric external authenticate 31
7.2.7 Compare 33
7.2.8 PIN compare 35

Documents
Document Sharing Hub

© ISO/IEC 2011 – All rights reserved

Full document details: BS ISO/IEC 24727-5:2011

7.2.9 Biometric compare 36
7.2.10 Mutual authentication with key establishment 37
7.2.11 Client-application mutual authentication with key establishment 39
7.2.12 Client-application asymmetric external authenticate 41
7.2.13 Modular extended access control protocol (M-EAC) 43
7.2.14 Key transport with mutual authentication based on RSA 45
7.2.15 Age attainment 47
7.2.16 Asymmetric session key establishment 48
7.2.17 Secure PIN compare 49
7.2.18 EC key agreement with card-application authentication 51
7.2.19 EC key agreement with mutual authentication 52
7.2.20 Simple EC-DH key agreement 54
7.2.21 GP asymmetric authentication 55
7.2.22 GP symmetric authentication (explicit mode) 56
7.2.23 GP symmetric authentication (implicit mode) 58

8 Secure messaging 60

9 Marshalling 61
9.1 ASN.1 representation 61
9.2 Web-services representation 61

10 Stack configuration testing 61
10.1 Testable interface definitions 61
10.1.1 Full-network-stack 63
10.1.2 Loyal-stack 64
10.1.3 Opaque-ICC-stack 65
10.1.4 Remote-loyal-stack 66
10.1.5 ICC-resident-stack 67
10.1.6 Remote-ICC-stack 68

11 Operational testing 68
11.1 SAL test sequences 69

12 Operational test reporting 69

13 ISO/IEC 24727-6 authentication protocol testing 69
13.1 SAL test sequences 70
13.1.1 ISO/IEC 24727-6 defined authentication protocol 70
13.2 Reference model implementations 70
13.2.1 Off card-application 70
13.2.2 Card-application emulator or test card use 70

Annex A (normative) **SAL operational test sequence descriptions** 71
A.1 Application management – alpha card-application data structure construction 71
A.2 Application management – first application data structure construction 92
A.3 Application management – application data structure construction error conditions 148
A.4 Application management – second application data structure construction 163
A.5 Data manipulation - card application path 207
A.6 Data manipulation – general 215
A.7 Data manipulation - global authentication 255
A.8 Application management - data structure destruction 282

Annex B (informative) **Envelope APDU implementation ICC-Resident stack expected component test inputs and outputs** 326
B.1 Application management - alpha card-application data structure construction 326
B.2 Application management - first application data structure construction 434
B.3 Application management - application data structure construction error conditions 755
B.4 Application management - second application data structure construction 907
B.5 Data manipulation - card application path 1156
B.6 Data manipulation – general 1271
B.7 Data manipulation - global authentication 1633
B.8 Application management - data structure destruction 1913

Documents
Document Sharing Hub

© ISO/IEC 2011 – All rights reserved

Full document details: BS ISO/IEC 24727-5:2011

Annex C (informative) **Non ICC-Resident stack expected component test inputs and outputs** 2346
C.1 Application management - alpha card-application data structure construction 2346
C.2 Application management - first application data structure construction 2443
C.3 Application management - application data structure construction error conditions 2734
C.4 Application management - second application data structure construction 2884
C.5 Data manipulation - card application path 3112
C.6 Data manipulation – general 3228
C.7 Data manipulation - global authentication 3579
C.8 Application management - data structure destruction 3855

Annex D (informative) **TLS implementation ICC-Resident stack expected component test inputs and outputs** 4277
D.1 Application management - alpha card-application data structure construction 4277
D.2 Application management - first application data structure construction 4329
D.3 Application management - application data structure construction error conditions 4501
D.4 Application management - second application data structure construction 4578
D.5 Data manipulation – general 4711
D.6 Data manipulation - global authentication 4908
D.7 Application management - data structure destruction 5060

Annex E (informative) **WSDL encoded IFD data structures** 5307
E.1 Establish Context 5307
E.2 ReleaseContext 5308
E.3 ListIFDs 5308
E.4 GetIFDCapabilities 5309
E.5 GetStatus 5310
E.6 Wait 5312
E.7 Cancel 5314
E.8 ControlIFD 5315
E.9 Connect 5315
E.10 Disconnect 5316
E.11 BeginTransaction 5317
E.12 EndTransaction 5317
E.13 Transmit 5318
E.14 VerifyUser 5319
E.15 ModifyVerificationData 5320
E.16 Output 5321
E.17 SignalEvent 5322

Annex F (informative) **ISO/IEC 24727-3 C language binding for common definitions** 5323

Annex G (informative) **ISO/IEC 24727-4 C language binding for algorithm definitions** 5326

Annex H (informative) **ISO/IEC 24727-4 C language binding for API and authentication protocol data** 5329

Annex I (informative) **ISO/IEC 24727-4 C language binding for TC-API** 5374

Annex J (informative) **ISO/IEC 24727-4 C language binding for IFD-API** 5379

Annex K (informative) **ISO/IEC 24727 Java language binding for common definitions** 5396

Annex L (informative) **ISO/IEC 24727-3 Java language binding for API and authentication protocol data** 5398

Annex M (informative) **ISO/IEC 24727-3 Java language binding for algorithms** 5590

Annex N (informative) **ISO/IEC 24727-4 Java language binding for TC-API** 5592

Annex O (informative) **ISO/IEC 24727-4 Java language binding for IFD-API** 5614

Bibliography 5656

Documents
Document Sharing Hub

© ISO/IEC 2011 – All rights reserved

Full document details: BS ISO/IEC 24727-5:2011

Foreword

ISO (the International Organization for Standardization) and IEC (the International Electrotechnical Commission) form the specialized system for worldwide standardization. National bodies that are members of ISO or IEC participate in the development of International Standards through technical committees established by the respective organization to deal with particular fields of technical activity. ISO and IEC technical committees collaborate in fields of mutual interest. Other international organizations, governmental and non-governmental, in liaison with ISO and IEC, also take part in the work. In the field of information technology, ISO and IEC have established a joint technical committee, ISO/IEC JTC 1.

International Standards are drafted in accordance with the rules given in the ISO/IEC Directives, Part 2.

The main task of the joint technical committee is to prepare International Standards. Draft International Standards adopted by the joint technical committee are circulated to national bodies for voting. Publication as an International Standard requires approval by at least 75 % of the national bodies casting a vote.

Attention is drawn to the possibility that some of the elements of this document may be the subject of patent rights. ISO and IEC shall not be held responsible for identifying any or all such patent rights.

ISO/IEC 24727-5 was prepared by Joint Technical Committee ISO/IEC JTC 1, *Information technology*, Subcommittee SC 17, *Cards and personal identification*.

ISO/IEC 24727 consists of the following parts, under the general title *Identification cards — Integrated circuit card programming interfaces*:

— *Part 1: Architecture*

— *Part 2: Generic card interface*

— *Part 3: Application interface*

— *Part 4: Application programming interface (API) administration*

— *Part 5: Testing procedures*

— *Part 6: Registration authority procedures for the authentication protocols for interoperability*

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Introduction

ISO/IEC 24727 is a set of programming interfaces for interactions between integrated circuit cards (ICCs) and external applications to include generic services for multi-sector use. The organization and the operation of the ICCs conform to ISO/IEC 7816-4.

ISO/IEC 24727 is relevant to ICC applications desiring interoperability among diverse application domains.

ISO/IEC 7498-1:1994, *Information technology — Open Systems Interconnection — Basic Reference Model: The Basic Model*, is used as the layered architecture of the client-application to card-application connectivity. That is, the client-application, through the Application Interface, assumes that there is a protocol stack through which it will exchange information and transactions among card-applications using commands conveyed through the message structures defined in ISO/IEC 7816. The semantics of action requests through the interface defined in ISO/IEC 24727-3 refers to application protocol data units (APDUs) as characterized through the interface defined in ISO/IEC 24727-2 and in the following International Standards:

— ISO/IEC 7816-4:2005, *Identification cards — Integrated circuit cards — Part 4: Organization, security and commands for interchange*

— ISO/IEC 7816-8:2004, *Identification cards — Integrated circuit cards — Part 8: Commands for security operations*

— ISO/IEC 7816-9:2004, *Identification cards — Integrated circuit cards — Part 9: Commands for card management*

The goal of ISO/IEC 24727 is to maximize the applicability and solution space of software tools that provide application interface support to card-aware client-applications. This effort includes supporting the evolution of card systems as the cards become more powerful, peer-level partners with existing and future applications while minimizing the impact to existing solutions conforming to this part of ISO/IEC 24727.

This part of ISO/IEC 24727 specifies an application-independent and implementation-independent testing regimen through which conformance of specific implementations to the relevant part of ISO/IEC 24727 can be confirmed. It is assumed that such testing will be performed through test environments and procedures developed in accordance with this part of ISO/IEC 24727.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 24727-5:2011

Identification cards — Integrated circuit card programming interfaces —

Part 5:
Testing procedures

1 Scope

ISO/IEC 24727 is a set of programming interfaces for interactions between integrated circuit cards and external applications to include generic services for multi-sector use.

This part of ISO/IEC 24727 specifies conformance testing procedures designed to determine if interfaces developed with the ISO/IEC 24727 series meet the requirement of ISO/IEC 24727. By conforming to this part of ISO/IEC 24727, interoperable implementations of ISO/IEC 24727 can be realized.

Test procedures for ISO/IEC 24727-2, ISO/IEC 24727-3 and ISO/IEC 24727-4 are described with sufficient detailing in support of ISO/IEC 24727 interoperability requirements, i.e. the connectivity, ISO/IEC 24727 security mechanisms and discovery mechanisms between the client-application and the card-application. This part of ISO/IEC 24727 defines calls on ISO/IEC 24727-3 in an ordered sequence. It also defines the confirmation of integrity of transmitted data by an implementation under test, as well as the syntax of that data received from the implementation under test for the marshalling procedures defined in ISO/IEC 24727-3 and ISO/IEC 24727-4.

For each test procedure, the conditions required for its execution are defined, along with the conditions under which it has to be executed and the expected results. Structures and entities used for the tests, as well as a common set of recurring sequences used for the various procedures, are identified and documented in this part of ISO/IEC 24727.

2 Normative references

The following referenced documents are indispensable for the application of this document. For dated references, only the edition cited applies. For undated references, the latest edition of the referenced document (including any amendments) applies.

ISO/IEC 24727-2:2008, *Identification cards — Integrated circuit card programming interfaces — Part 2: Generic card interface*

ISO/IEC 24727-3:2008, *Identification cards — Integrated circuit card programming interfaces — Part 3: Application interface*

ISO/IEC 24727-4:2008, *Identification cards — Integrated circuit card programming interfaces — Part 4: Application programming interface (API) administration*

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub

Full document details: BS ISO/IEC 24727-5:2011

3 Terms and definitions

For the purposes of this document, the following terms and definitions apply.

3.1
certification
formal confirmation of successful conformance testing by a certified party

3.2
component
executable code comprising a processing layer accessed with ISO/IEC 24727 defined application programming interfaces

[ISO/IEC 24727-4:2008]

3.3
component-under-test
ISO/IEC 24727 component being tested

3.4
conformance-testing
formal demonstration that a component complies with specified critera established by a standard

3.5
conformance-vector
concatenation of the tuple element values

3.6
dimension-of-conformance
tuple element of a conformance-vector

3.7
native-implementation
on-card implementation connected to ISO/IEC 24727-2 layer either directly or via a procedural element

3.8
card-application emulator
processing component accessed through the interface device application programming interface (IFD-API) that simulates the behaviour of a smart card-application

3.9
card emulator
processing component accessed through the interface device application programming interface (IFD-API) that simulates the behaviour of a card

3.10
stack-configuration
series of processing components connected by communication channels that connect a client-application to a card-application as specified by the ISO/IEC 24727-4 protocol stacks

3.11
test-component
ISO/IEC 24727 trusted component that is provided and used by the testee

3.12
test-environment
systems, personnel and standard procedures to perform conformance-testing

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

3.13
test-implementation
ISO/IEC 24727 stack including the component(s) under test and testing components

3.14
testing-body
organization providing a test-environment, test-implementation and means to perform conformance-testing

4 Symbols and abbreviated terms

ACA	alpha card application
ACD	application capability description
ACL	access control list
AK	asymmetric key
APDU	application protocol data unit
API	application programming interface
AR	access rule
CA	card application
CCD	card capability description
CIA	cryptographic information application
CUT	component under test
DER-TLV	distinguished encoding rules – tag length value
DF	dedicated file
DID	differential identity
DS	data set
DSI	data structure for interoperability
EF	elementary file
GCI	generic card interface
IFD	interface device
MF	master file
PC/SC	personal computer / smart card consortium
RMI	reference model implementation
RFU	reserved for future use
SAL	service access layer

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

SE security environment

TC test component

TLS transport layer security

5 Testing methodology

ISO/IEC 24727 establishes the facilities for interoperability through the specification of a number of interfaces. Independent implementations of each interface shall be interchangeable and thus provide for interoperability at each interface. Consequently, ISO/IEC 24727 enables a wide variety of interoperability. This part of ISO/IEC 24727 shall provide a methodology for conformance-testing such that distinct interface implementations can be affirmed as being interchangeable with other independent implementations.

The testing methodology assumes a test-environment that presents standard characteristics that include:

1) Equipment configuration interfaces

2) Test-implementations

3) Test-specifications

Given these standard characteristics a testing-body can make use of this test-environment to confirm the behaviour of the various interfaces through which interoperability is enabled.

This part of ISO/IEC 24727 does not test the following:

1) Tests for quality, performance, robustness

2) Testing of standards and protocols that are referenced within

3) Testing of proprietary card application services, their creation and execution

4) Error conditions that cannot be produced by a correctly functioning implementation

5.1 Terms of testing

5.1.1 Purpose of testing

The purpose of ISO/IEC 24727-5 is to provide testing specifications at a level that allow independent testing bodies to conduct testing with the assurance of independent equivalent results.

5.1.2 Testing objective

The objective of ISO/IEC 24727-5 is to specify a testing regimen that enables independent, equivalent implementations. Independent testing bodies can thereby evaluate conformance to ISO/IEC 24727 with equivalent results. Levels of certification can then indicate which component or set of components is ISO/IEC 24727 conformant.

5.1.3 Testing Principles

The testing principles are derived from the ISO/IEC 24727 objective of interoperability. Interoperability is achieved by specifying interfaces and behaviour through which a component is accessed. Since each component is tested independently there will be varying levels of conformance and a distinct set of certification levels will be defined that address single components as well as a concatenation of components that form ISO/IEC 24727 component-configurations.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

ISO/IEC 24727-5 defines test configurations and unique parameters to be applied uniformly to each component for different implementations to assure conformance.

It is assumed that automated testing tools can be used for:

- the generation of test data that is used as input and output during testing to ensure that all user data is differs for each test execution;
- consistent definition;
- performance of tests; and
- result reporting.

ISO/IEC 24727 certification is enabled through ISO/IEC 24727-5 tests that can be administered by testing bodies.

Table 1 —Testing Principles

	Principle
1.	To attain conformance, all aspects of the ISO/IEC 24727 interfaces for the Component-under-test shall comply with ISO/IEC 24727.
2.	ISO/IEC 24727-5 testing shall only include behavioural testing at the component interfaces.
3.	Testing shall encompass one or more of the following ISO/IEC 24727 components: i. ASN.1 DER-TLV defined APIs; ii. communication interfaces; iii. C language bindings for API marshalling and binding of the APIs; iv. stack configurations; v. authentication protocols.
4.	A test facility shall provide a reference test implementation that permits a component under test to be tested in a complete stack test configuration.
5.	The use of automated testing tools is assumed for generating tests and to aid the certification process.
6.	ISO/IEC 24727-5 should enable the development of conformance and certification testing abilities to be derived in a consistent manner.
7.	Every component under test shall be subject to tests defined at the ISO/IEC 24727-3 API.
8.	Only services and actions as defined by ISO/IEC 24727-3 shall be tested. The testing of proprietary services, actions and the associated creation and loading of these is out of scope.
9.	Test inputs and results shall vary each time a test is performed in order to ascertain that tests are not predictable.
10.	ISO/IEC 24727-3 testing verifies the request and response of the interface and the APDUs it generates.
11.	ISO/IEC 24727-3 proxy testing verifies the request and response of the interface plus the DER-TLV that it generates.
12.	ISO/IEC 24727-3 (including proxy) tests shall be run against blank and populated (partially provisioned) smart cards.
13.	Interoperability is only fully tested in the C language implementation (until other language binding is provided)
14.	Any other languages shall use an agent to translate the messages (e.g. DER-TLV) into language

Documents
Document Sharing Hub

© ISO/IEC 2011 – All rights reserved

Full document details: BS ISO/IEC 24727-5:2011

	Principle
	specific bindings.
15.	Data transfer over communication channels is intercepted by a "man in the middle" and compared to the test data (protocol analysers).
16.	The only testable interface at the card application is the ISO/IEC 24727-3 on-card implementation.
17.	Only IFD API syntax (i.e. the structure and not the containing data) shall be tested. The testing of data contained within the IFD API syntax is out of scope.
18.	Only TC API syntax shall be tested. The testing of data contained within the TC API syntax is out of scope.
19.	Testing of procedural element low-level coding and functions is out of scope
20.	Testing of other standards that are referenced within ISO/IEC 24727 is out of scope
21.	Testing of other protocols (i.e. not ISO/IEC 24727 authentication protocols) that are referenced within ISO/IEC 24727 is out of scope
22.	Testing of error conditions that cannot be produced by a correctly functioning implementation is out of scope
23.	Testing of proprietary services, actions and the associated creation and loading of these is out of scope.
24.	The standard shall not test application specific software module.
25.	Checking of CIA updates is out of scope.

5.1.4 GCI under test:

The GCI under test shall be submitted to test that shall confirm the following minimum set features:

- Whenever procedural elements are present, all the incoming APDUs that are part of the GCI shall be submitted to these procedural elements. Therefore, the GCI under test capability to handle the request and response APDUs shall be tested. The GCI under test shall pass on the request APDU from SAL to Procedural Elements, and the response APDU from the ICC to procedural elements. The procedural elements testing component shall require a “probe” to prove such APDU processing.

- GCI under test shall be capable to parse CCD and to ensure availability of interoperability data to the SAL. The CCD content shall be retrieved according to the bootstrap mechanism. Some CCD data object values may be loaded from remote location specified in the CCD.

- GCI under test capability to deliver upon request the CCD content to the SAL.

- ISO/IEC 24727-2 interface (request and response) (only for C language binding).

- GCI under test capability to handle request APDUs with CLA=’FF’ (packageA) intended for GCI internal/resources management

- GCI under test capability to forward any APDU that is neither part of ISO/IEC 24727-2 Table 2 nor Table 3 directly to ICC without any processing.

- GCI under test capability to call IFD-API functions.

5.1.5 SAL under test:

The SAL under test shall be submitted to test that shall confirm the following minimum set features:

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

- SAL shall be able to recover the whole data mapping between off-card data representation an on-card data representation using the CIA containing within the ACD.
- SAL shall be enabled to parse the CIA values and to generate all the discovery data and to encode it according to ISO/IEC 24727-3, Annex C ASN.1 normative definitions.
- Upon or prior to the call to CardApplicationEndSession action, any change impacting the CIA entries (as per ISO/IEC 24727-2) that occurred during the transaction shall be updated in the CIA that is either stored or referenced in the ACD.
- The retrieved (by ISO/IEC 24727-3) CIA must reflect the card implementation and any changes that have been applied.
- ISO/IEC 24727-3 interface (request and response) (only for C language binding), generated APDUs and ASN.1 DER-TLV (for proxies)
- The off-card SAL shall be enabled to act on client-application requests according to the related ACL (e.g denied actions shall not be translated into APDUs)
- SAL claiming compatibility with a given Authentication Protocol shall prove capability to translate generic requests into specific set of request APDUs according to the Protocol OID description and the rest of information provided in the DID bearing this Protocol.
- Authentication protocols that are internal to ISO/IEC 24727-3 are tested using a trusted test implementation below the ISO/IEC 24727-3 layer.
- Authentication protocols that are internal to ISO/IEC 24727-3 are tested using a trusted test ISO/IEC 24727-3 implementation.
- Specified authentication protocol algorithms and their use are verified by the trusted test components.
- DIDCreate and DIDUpdate requests shall uniquely be executed by the SAL according to existing specification of these requests parameters in ISO/IEC 24727-3, Annex A.
- SAL shall be enabled to translate ExecuteAction requests into IFD-API calls without resorting to GCI under test layer. The SAL shall be able to extract the parameter “request” from within ExecuteAction call and to deliver it unchanged as InputAPDU parameter to the Transmit command from IFD-API.
- SAL shall be enabled to process post-issuance personalization data as a separate package of requests. This package is comprised of the following SAL-API calls:

    i) DIDCreate

    ii) DIDUpdate

    iii) DataSetCreate

    iv) DSICreate

5.1.6 Conformance attainment

For a component-under-test to attain conformance all its mandatory features, as defined in all other parts of ISO/IEC 24727, shall be implemented.

All tests contained within ISO/IEC 24727-5 shall be mandatory for each relevant component-under-test.

Distinct tests shall identify the component-under-test interface behaviour.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

A test report shall be generated stating what aspects of the component-under-test interface do and do not conform to ISO/IEC 24727.

Testing shall be performed on a single component or on component-configurations comprising one or more of the stack models specified in ISO/IEC 24727-4.

The following clause identifies a conformance-vector comprised of a tuple of dimension-of-conformance. The dimensions of this conformance-vector are defined by ISO/IEC 24727 component, ISO/IEC 24727-3 authentication protocol, ISO/IEC 24727-4 stack, ISO/IEC 24727-4 marshalling and cryptographic algorithms.

5.2 Conformance vector

Four distinct criteria can be identified for each testable implementation. These criteria are entitled dimension-of-conformance.

1) ISO/IEC 24727 component

2) ISO/IEC 24727-3 authentication protocol

3) ISO/IEC 24727-4 stack configuration

4) ISO/IEC 24727-4 marshalling

5) ISO/IEC 24727 cryptographic algorithms

A bit-mask is applied to the possible values of these dimensions-of-conformance and by concatenating these bit-masks to form a five-tuple, a conformance-vector is established that will provide a quantitative indicator of the level of conformance of a particular testable-interface. Table 2 provides values for the dimension-of-conformance.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Table 2 — Conformance vector

	Dimension-of-conformance				
Conformance indicator (bit-set i.e. least significant bit)	**ISO/IEC 24727 component (tuple 1)**	**ISO/IEC 24727-3 authentication protocol (tuple 2)**	**ISO/IEC 24727-4 stack (tuple 3)**	**ISO/IEC 24727-3 marshalling (tuple 4)**	**ISO/IEC 24727-3 cryptographic algorithms (tuple 5)**
1	Native implementation	Simple Assertion	Loyal stack	DER-TLV	ISO/IEC 18033-3
2	ISO/IEC 24727-2 Package A	Asymmetric Internal Authenticate	Opaque ICC stack	SOAP	AES (128-bit key)
3	ISO/IEC 24727-2 Package B	Asymmetric External Authenticate	Remote loyal stack	APDU	AES (192-bit key)
4	ISO/IEC 24727-3	Symmetric Internal Authenticate	ICC resident stack	TLS forward protocol	AES (256-bit key)
5	Trusted channel interface	Symmetric External Authenticate	Remote ICC stack	TLS backward protocol	DES
6	Interface device API	Compare	RFU	RFU	3DES
7	RFU	PIN Compare	RFU	RFU	RC2
8	RFU	Biometric Compare	RFU	RFU	RC3
9	RFU	Mutual Authentication with Key Establishment	RFU	RFU	RSA
10	RFU	Client-Application Mutual Authentication with Key Establishment	RFU	RFU	DSA
11	RFU	Client-Application Asymmetric External Authenticate	RFU	RFU	EC
12	RFU	Modular Extended Access Control Protocol (M-EAC)	RFU	RFU	RSA-OAEP
13	RFU	Key Transport with mutual authentication based on RSA	RFU	RFU	md2

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

	Dimension-of-conformance				
Conformance indicator (bit-set i.e. least significant bit)	**ISO/IEC 24727 component (tuple 1)**	**ISO/IEC 24727-3 authentication protocol (tuple 2)**	**ISO/IEC 24727-4 stack (tuple 3)**	**ISO/IEC 24727-3 marshalling (tuple 4)**	**ISO/IEC 24727-3 cryptographic algorithms (tuple 5)**
14	RFU	Age Attainment	RFU	RFU	md5
15	RFU	Asymmetric Session Key Establishment	RFU	RFU	sha1
16	RFU	Secure PIN Compare	RFU	RFU	sha224
17	RFU	EC Key Agreement with Card-Application Authentication	RFU	RFU	sha256
18	RFU	EC Key Agreement with Mutual Authentication	RFU	RFU	sha384
19	RFU	Simple EC-DH Key Agreement	RFU	RFU	sha512
20	RFU	GP Asymmetric Authentication	RFU	RFU	ID-HMAC
21	RFU	GP Symmetric Authentication (Explicit Mode)	RFU	RFU	HMAC-MD5
22	RFU	GP Symmetric Authentication (Implicit Mode)	RFU	RFU	DH
23	RFU	RFU	RFU	RFU	X9-63-scheme
24	RFU	RFU	RFU	RFU	SECG-scheme
25	RFU	RFU	RFU	RFU	RFU

Conformance indicator bit set from 26 to 32 shall be Reserved for Future Use (RFU).

5.3 Structure of tests

Specification of conformance tests shall address three distinct testing domains:

1) Basic (atomic) tests across the allowed range of status responses

2) Processing tests

3) Discoverability tests

4) Operational tests

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Considering that an application programming interface is comprised of distinct requests or functions. For each request or function, a series of tests shall be defined to exercise the request and elicit the complete range of status responses that a specific request can support. These are the basic or "atomic" tests.

Figure 1 — Testable data in transfer between CUTs depicts the CUT and the data transered between each component. When testing components in the ICC-Resident stack the protocol structure as well as the data encapsulated within that protocol shall be tested. Data transferred between non ICC-Resident stack model components can vary to a large degree, therefore only the protocol structure shall be tested and not the data contained within.

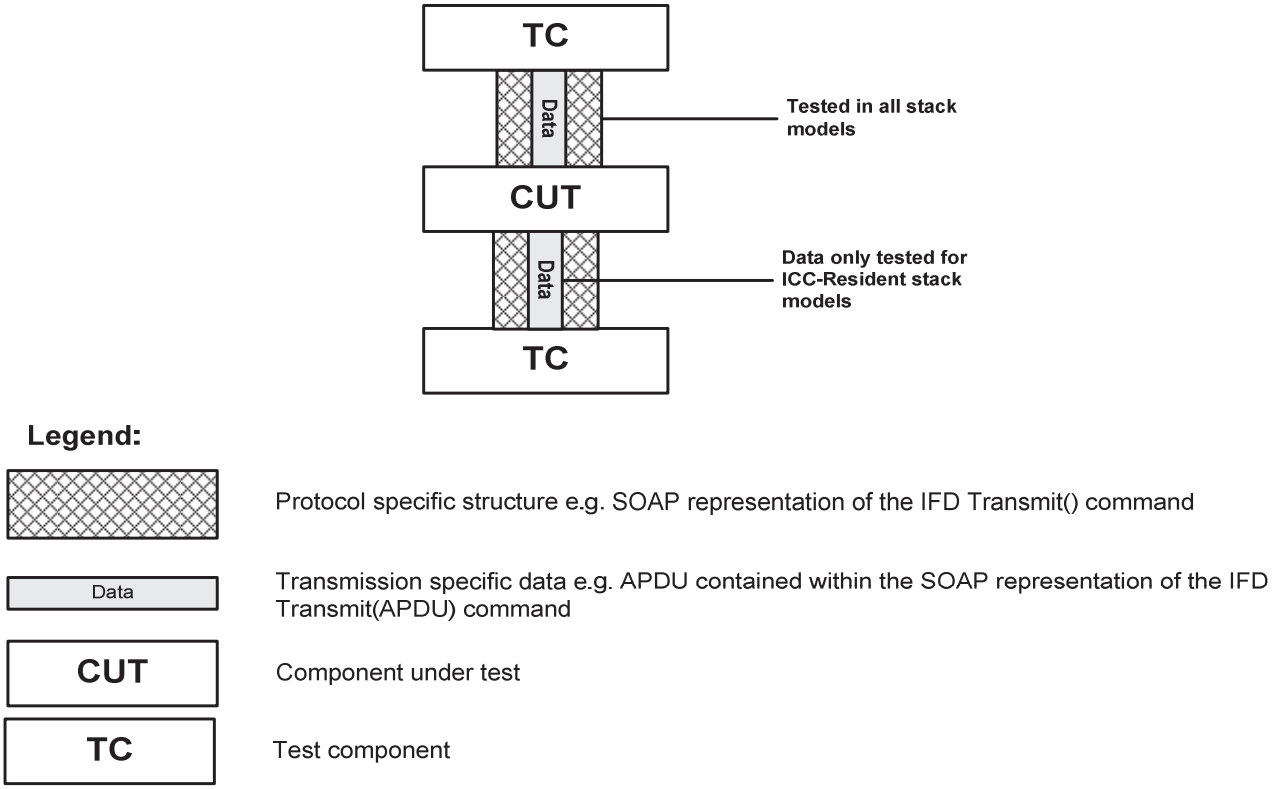

Figure 1 — Testable data in transfer between CUTs

A more comprehensive set of tests shall be defined to completely exercise all of the requests that the specific interface encompasses. Such tests shall comprise inter-related tests; that is, testing sequences of requests that may be related through their individual impact on the target(s) of the interface. These sets of tests are refered to as pocessing tests.

An alternative yet still comprehensive set of tests shall be defined to fully exercise the complete life-cycle of the targets of interfaces. These sets of tests are aimed at exercising application level utilization of an interface. These sets of tests are termed operational tests.

In the following clauses, sets of tests in these three domains shall be specified.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

5.4 Test environment

5.4.1 Stack configurations

The components that will facilitate the testing shall be implemented as a full network stack, that both provide a programming language agnostic API as well as the program language specific binding interface. Each test component must be interchangeable with a given component-under-test. No ISO/IEC 24727 implementation shall be tested remotely i.e. all elements to be tested shall be tested in the given test environment. Two possible constraints apply to the components under test.

If the component under test is intended to reside on the same machine as the stack used for testing, it shall provide a program specific language binding for the interfaces through which it communicates.

If the component under test is intended to be accessed through a network, it shall provide network connectivity by supporting at least one standard marshalling and at least one secure transmission protocol for the interfaces it supports.

5.4.2 Card-application emulators

Within the stack used for testing, a card emulator or card-application emulator (depending on what is to be tested) may be used instead of an actual smart card.

5.4.3 Verification and logging capability of components

Each test component shall have the capability to verify that all action performed upon it and commands received by it are valid. A log shall be kept of all actions and input and output parameters for a given component. This log may be generated through an internal debugging system or a man in the middle interception between components. The generated log shall contribute to the final conformance test report.

5.4.4 Procedural element

A test procedural element shall exist and be used during ISO/IEC 24727-2 conformance testing. The test procedural element shall take the test APDU as an input argument and forward that APDU to the GCI this process is described in detail in clause 6.2. The purpose of the test procecdural element is not to test the functionality of a given procedural element, but rather the ISO/IEC 24727-2 implementation under test conformance to integrate procedural elements.

6 Components

6.1 Service access layer API

6.1.1 Basic tests

All service actions defined in ISO/IEC 24727-3 shall be tested according to clauses 6 through to and including clause 12 of ISO/IEC 24727-3 and the returned values shall be checked to be compliant to ISO/IEC 24727-3, clauses 6 through to and including clause 12.

6.1.2 Discoverability tests

Figure 2 — Discoverability testing ISO/IEC 24727-3 implementation shows the structure of the component under test for an ISO/IEC 24727-3 implementation.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

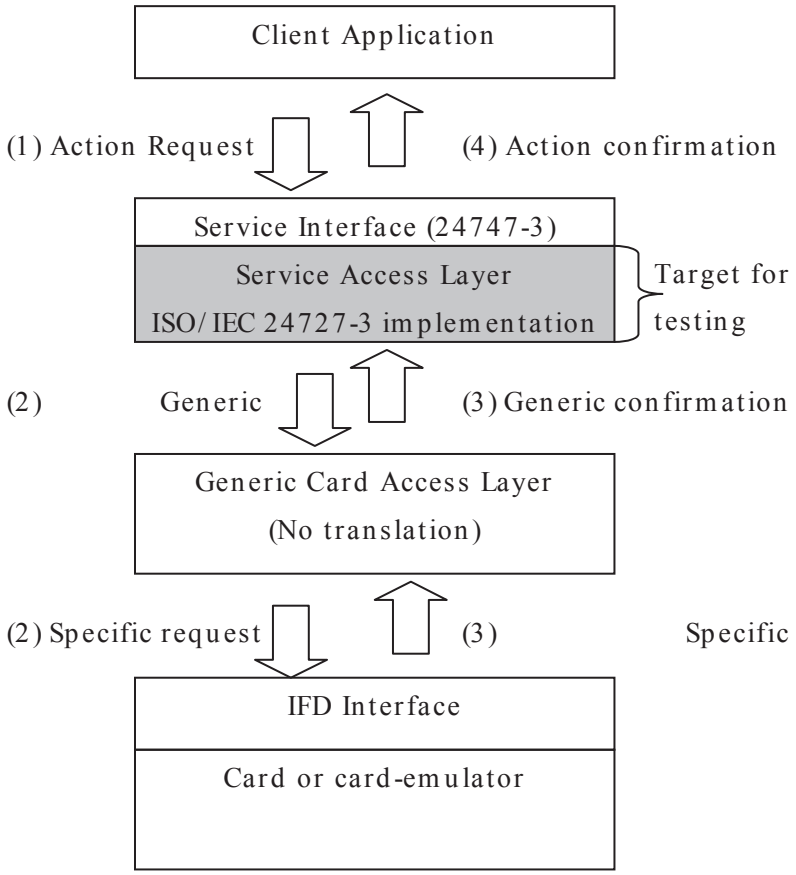

Figure 2 — Discoverability testing ISO/IEC 24727-3 implementation

The purpose of the test is to evaluate that the CUT i.e. the SAL shall be capable to bootstrap the CIA component retrieved from the CCD or from a location as defined within the CCD.

Figure 3 — Discoverability test for ISO/IEC 24727-3 with TC ICC depicts how a test-component (TC) shall be used to test ISO/IEC 24727-3 discoverability via consumption of a trusted procedural element that has been discovered from the TC components.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

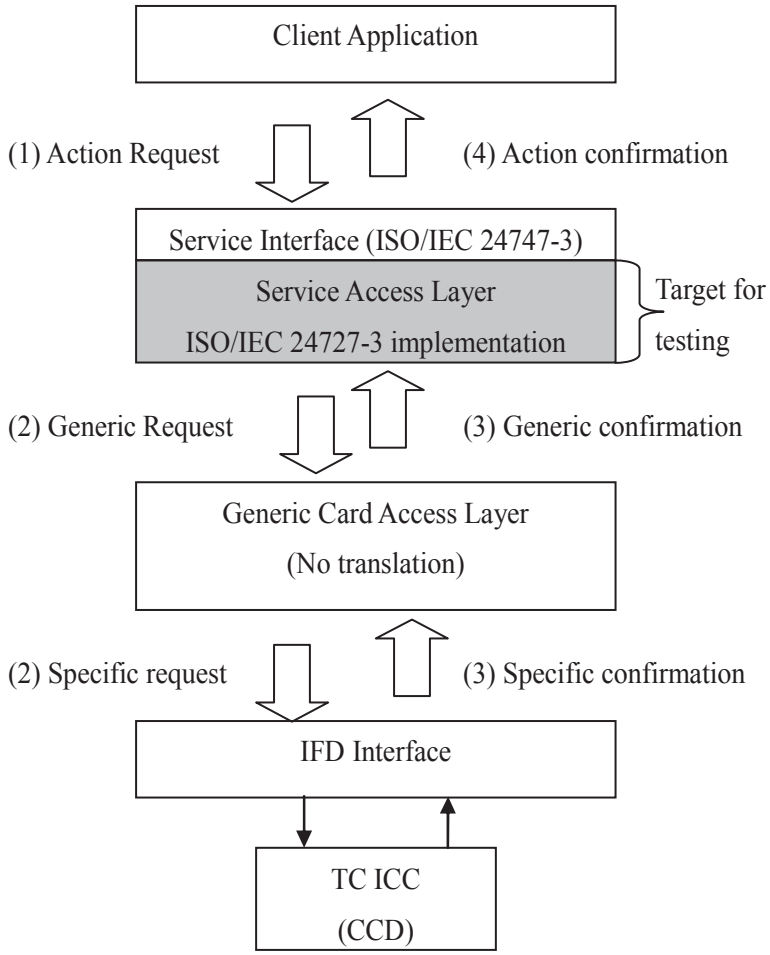

Figure 3 — Discoverability test for ISO/IEC 24727-3 with TC ICC

These tests shall use a trusted TC ICC or equivalent software module (e.g. card-application-emulator) that contains a CCD that will perform specific mapping between the off-card data representation and the on-card data representation as defined below.

The test shall use a TC ICC or card-application-emulator to bootstrap a Cryptographic Information Application (CIA) component that shall perform:

— An action that takes a DSIName as a parameter. The mapping between the DSIName (i.e. off-card data representation) and the file identifier (i.e. on-card data representation) is contained with in the CIA of the trusted TC ICC. The test shall confirm that the SAL has correctly performed the mapping between the off-card DSIName and the on-card file identifier. If a non ICC-Residential stack model is under test that does not cater for Package "A" commands as specified in ISO/IEC 24727-2 clause 5.1.3 Table 2 then these tests shall not be performed.

— The required actions in order to create a DSI (e.g. DSIName = DSI1). The session with the TC ICC shall be ended and the connection terminated. A new connection to the TC ICC shall be established and the CIA shall be bootstrapped during the Initialize action. An action shall be performed that takes a DSIName (i.e. DSI1) as a parameter. The mapping between the DSIName (i.e. off-card data representation) and the file identifier (i.e. on-card data representation) has been created during the previous sequence and shall be contained with in the CIA of the trusted TC ICC. The test shall confirm that the SAL has correctly performed the mapping between the off-card DSIName (i.e. DSI1) and the on-card file identifier that was created during the previous sequence. If a non ICC-Residential stack model is under test that does not cater for Package "B" commands as specified in ISO/IEC 24727-2 clause 5.1.3 Table 2 then these tests shall not be performed.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

6.2 Generic card interface

6.2.1 Basic test

All commands defined in ISO/IEC clause 5.1.3, package A shall be tested according to the limitations given in Table 2 of ISO/IEC 24727-2 and the status returned shall be checked to be compliant to ISO/IEC 24727-2, clause 5.3. Commands defined in ISO/IEC 24727-2, clause 5.3 package B, may be subject to the same procedure

6.2.2 Processing tests

All GCI APDUs defined in ISO/IEC 24727-2 clause 5.1.3 Table 2 shall trigger the *"Transmit"* function of the IFD API, with the input of that *"Transmit"* function being the APDUs generated by the procedural element.

The ENVELOPE command defined in ISO/IEC 24727-4 clause 5.5 shall be the input and trigger the *"Transmit"* function of the IFD API without being passed into the procedural element.

The following GCI APDUs defined in ISO/IEC 24727-2 clause 5.1.3 Table 3 shall not trigger a request of the IFD API but shall be processed within the GCI itself i.e. these are not passed into the procedural element:

- SELECT PROCEDURAL ELEMENT (CLA = 'FF')
- GET DATA (CLA = 'FF')
- PUT DATA (CLA = 'FF')

Table 3 — GCI commands to IFD API mapping shall map the GCI APDUs defined in ISO/IEC 24727-2 clause 5.1.3 Table 3 to the relevant IFD API functions.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Table 3 — GCI commands to IFD API mapping

	COLD RESET (CLA = 'FF')	*WARM RESET (CLA = 'FF')*	*DEACTIVATE CONTACTS (CLA = 'FF')*	*DEACTIVATE CONTACTS AND EJECT (CLA = 'FF')*	*LIST READERS (CLA = 'FF')*
Establish Context	X	X			
ReleaseContext	X	X			
ListIFDs					X
GetIFDCapabilities					
GetStatus					
Wait					
Cancel					
ControlIFD	X	X		X	
Connect	X	X			
Disconnect	X	X	X	X	
BeginTransaction					
EndTransaction					
Transmit					
VerifyUser					
ModifyVerificationData					
Output					
SignalEvent					

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

6.2.3 Discoverability tests

Figure 4 — Discoverability testing ISO/IEC 24727-2 implementation shows the structure of the component under test for an ISO/IEC 24727-2 implementation.

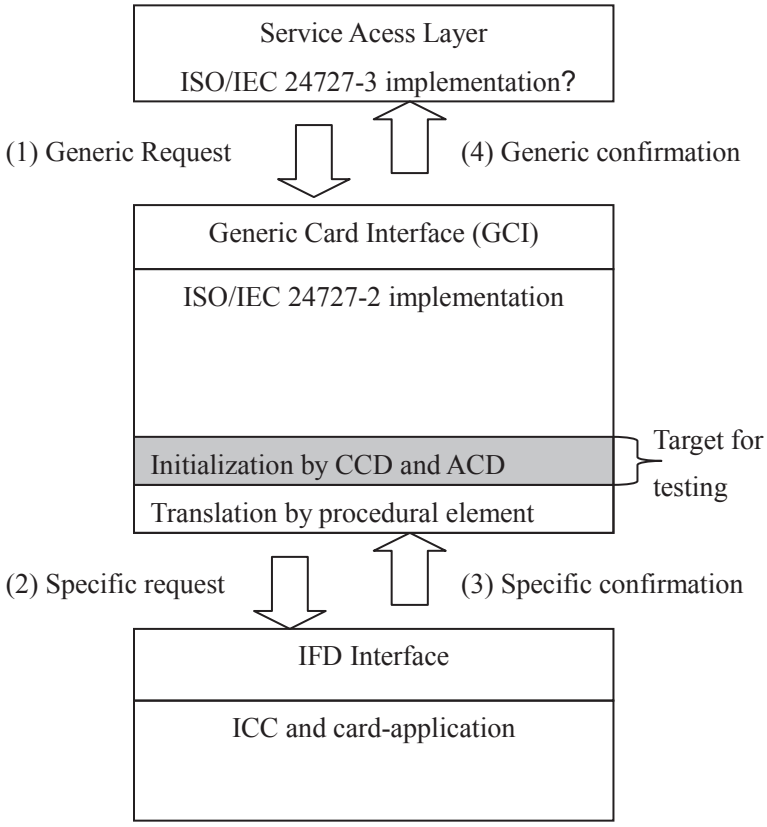

Figure 4 — Discoverability testing ISO/IEC 24727-2 implementation

The purpose of the test is to evaluate that the software module under test has the interoperable functions, especially following three functions specific to ISO/IEC 24727-2.

APDU request/response defined in clause 5.1.3 Table 2 of ISO/IEC 24727-2 shall not be independently tested.

The translation of ISO/IEC 24727-2 defined APDUs to any commands by the procedural element shall not be tested.

6.2.3.1 Discovery Mechanism

Testing shall evaluate that the software module under evaluation supports all CCD and ACD retrieval procedures defined in ISO/IEC 24727-2 and returns the expected return codes. The return code may differ within the content of retrieved CCD and ACD information, especially with and without signature on the procedural element.

6.2.3.2 Testing Mechanism

Figure 5 — Discoverability test for ISO/IEC 24727-2 with TC ICC depicts how a test-component (TC) shall be used to test ISO/IEC 24727-2 discoverability via consumption of a trusted procedural element that has been discovered from the TC components.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

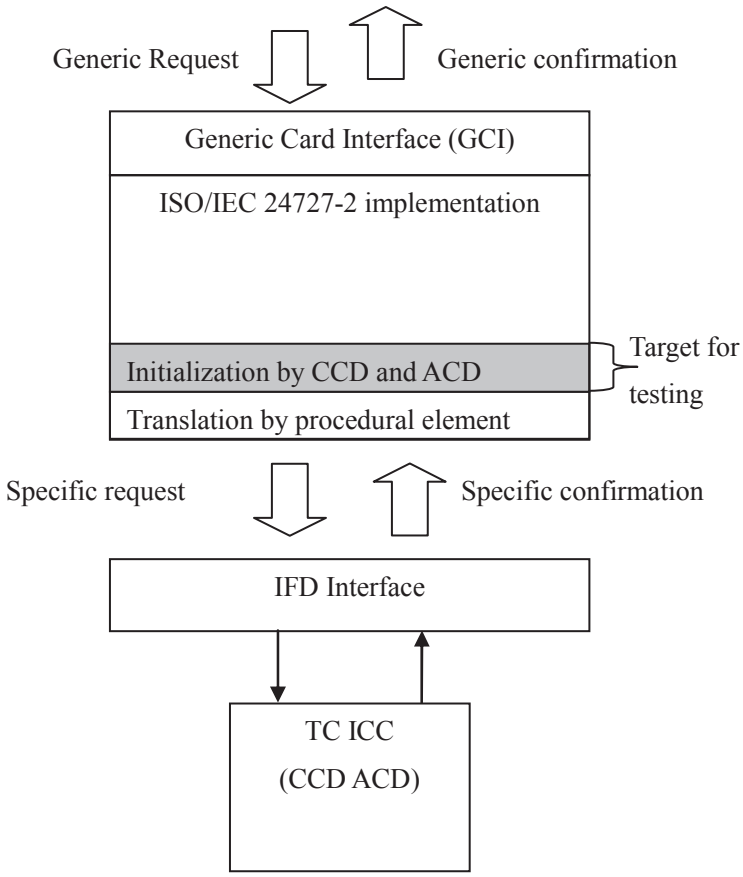

Figure 5 — Discoverability test for ISO/IEC 24727-2 with TC ICC

These tests shall use a trusted TC ICC or equivalent software component (e.g. card-application-emulator) that contains a CCD and ACD that will perform specific APDU translationas defined below.

The test shall use a TC ICC or card-application-emulator to bootstrap a procedural element that shall:

— Receive a request APDU as defined in ISO/IEC 24727-2 clause 5.1.3 Table 2 and transform the class byte from '00' to '01' i.e. the request APDU passed into the procedural element shall contain a CLA = '00' and the output (i.e. the request APDU) from the procedural element request APDU shall contain a CLA '01'.

— Receive a response APDU with a status word of '9000' and transform the status word to '9001' i.e. the response APDU passed into the procedural element shall have a SW1 = '90' and SW2 = '00' the output from the procedural element shall contain a response APDU that shall have a SW1 = '90' and SW2 = '01'.

6.2.4 Generic card interface acted on ISO/IEC 24727-2 implementation (i.e. CLA = "FF")

6.2.4.1 Basic test

The support of commands with CLA=FF defined in ISO/IEC 24727-2 clause 5.1.3, Table 3 are not necessarily supported at the GCI.

If the implementation is expected to support them, they shall be tested according to the limitations given in Table 3 of ISO/IEC 24727-2 and the status returned shall be checked to be compliant to ISO/IEC 24727-2, clause 5.3

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

6.2.4.2 Application oriented tests

The purpose of those tests is to verify that the GCAL implementation translates correctly the APDUs received at the GCI into the proper command at the IFD API

6.3 Interface device API

The IFD API shall be tested by intercepting the data communication flow between the CUT during operational testing as defined in clause 10. The format of the IFD data encapsulation mechanism shall be verified; the actual transmitted data shall only be verified for data sent between ICC-Resident stack components (i.e. Appendix A) as this is the only stack model for which the transfer data shall not vary.

Refere to Annex E for example data structures.

6.4 Trusted channel API

The trusted channel API is implementation specific and therefore cannot be tested in a generic way. In order for an ISO/IEC 24727 implementation to claim trusted channel compliance the API shall be tested using a approved ISO/IEC 24727 binding.

The trusted channel consists of two remote components and each transmission of data shall be tested by comparing the input into the TC_API_Write function with the corresponding output of the TC_API_Read function.

6.4.1 TC_API_Open

This function shall be invoked during the ISO/IEC 24727-3 Initialize action if the TC channel is not already open.

Tests shall confirm that a channelHandle is returned by this function upon processing valid input parameters as defined by ISO/IEC 24727-4 clause 7.3.1.

6.4.2 TC_API_Close

This function shall be invoked during the ISO/IEC 24727-3 Terminate action if the TC channel was opened using the ISO/IEC 24727-3 Initialize action.

Tests shall confirm that the tursted channel is closed by this function upon processing the channelHandle input parameter of a valid established trusted channel as defined by ISO/IEC 24727-4 clause 7.3.2.

6.4.3 TC_API Write

This function shall be invoked during all ISO/IEC 24727-3 actions.

Tests shall confirm that a message can be written to the trusted channel as defined by ISO/IEC 24727-4 clause 7.3.4.

6.4.4 TC_API Read

This function shall be invoked during all ISO/IEC 24727-3 actions.

Tests shall confirm that a message can be read from the trusted channel as defined by ISO/IEC 24727-4 clause 7.3.3.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

6.5 SAL on-card implementation component testing

The SAL on-card implementation is the only on-card implementation that shall be tested. The test cases for this CUT shall be identical to those of the SAL off-card component, that is the actions and their associated parameters shall be DER/TLV encoded and wrapped in either the APDU or TCP/IP web server based interface as defined in ISO/IEC 24727-4.

7 Authentication protocols

7.1 General

An authentication protocol consists of two components, the on-card implementation and the off-card implementation. The off-card implementation of the authentication protocol may either be implemented in the SAL layer or by the client application.

The whole authentication protocol details are handled by the SAL unless the DIDAuthenticate() or CardApplicationStartSession() request conveys authenticationProtocolData that can be processed by both the client-application and the SAL.

For an authentication protocol to achieve conformance, two types of tests shall be performed:

1) The off-card CUT shall be tested independently by using a card-application emulator that shall provide the required on-card functionality of the authentication protocol under test.

2) The on-card CUT shall be tested in conjunction with the off-card CUT.

All authentication protocols shall be actioned as described in ISO/IEC 24727-3 and all necessary authenticationProtocolData parameters shall be constructed as defined in ISO/IEC 24727-2, Annex A.

Authentication protocols shall be tested by the implementor providing the on-card and off-card CUT. The testing facility shall provide the equivelant trusted on-card and off-card TC. The CUT shall be verified by passing the test scenarios defined in clauses below that shall be executed using the following combinations of CUT and TC:

- Full authentication protocol components under test system testing
- Authentication protocol components under test off-card testing, where the on-card CUT is replaced with the on-card TC.
- Authentication protocol components under test on-card testing, where the off-card CUT is replaced with the off-card TC.

This concept is described in Figure 6 — Authentication protocol testing strategy.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

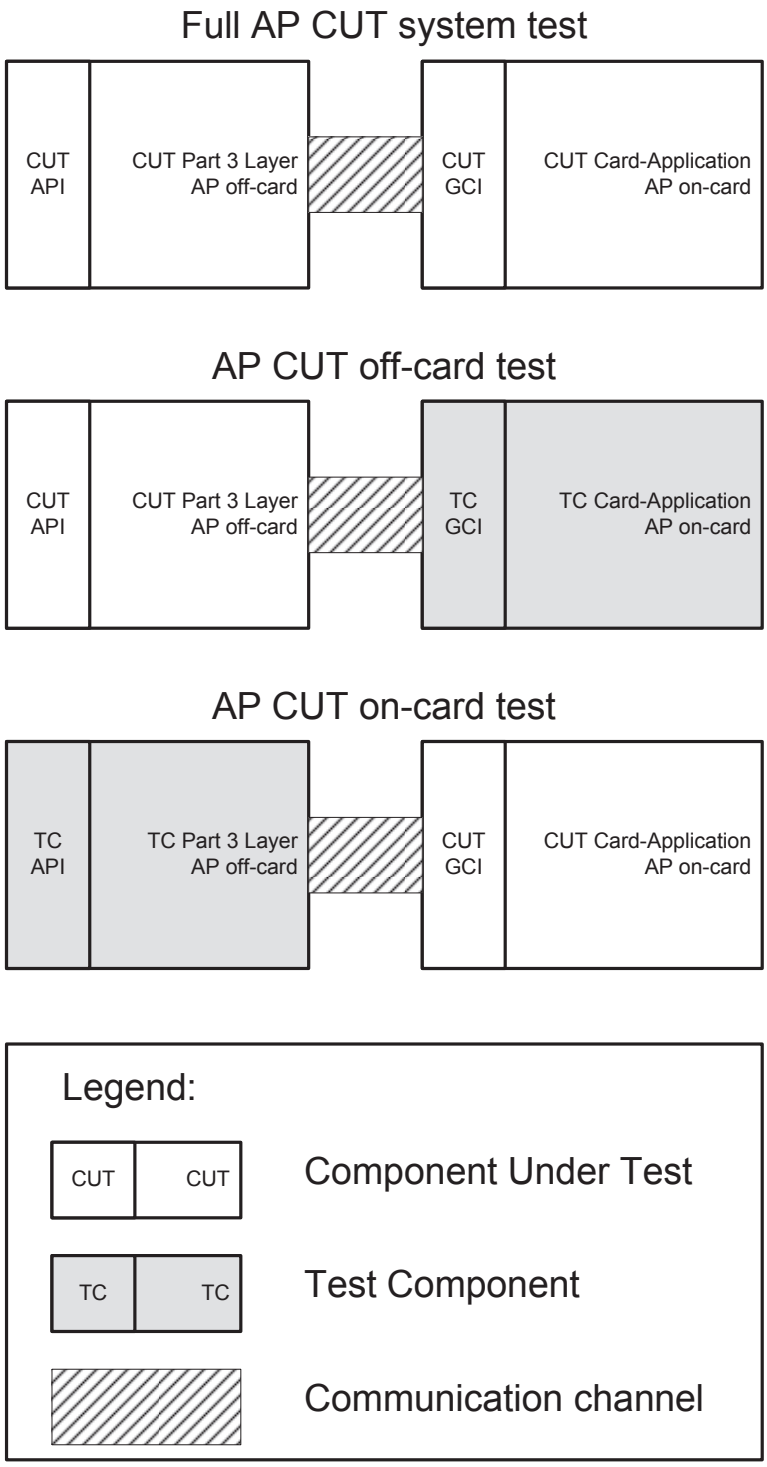

Figure 6 — Authentication protocol testing strategy

7.2 SAL security test sequences

The SAL test sequences for authentication protocols shall not test any functionality of the qualifier that is contained within the DID for a specific authentication protocol.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.1 Cryptographic operations

All authentication protocols shall be tested for the cryptographic actions that they support as well as for the correct error condition i.e. return codes, when a specific cryptographic action is not supported by the authentication protocol under test.

7.2.1.1 Test sequences for supported cryptographic functionality

Table 4 through Table 10 define the tests for cryptographic actions if these are supported by the authentication protocol under test.

7.2.1.1.1 Encipher

Table 4 — Encipher – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Encipher	As defined in ISO/IEC 24727-3 clause 10.2	return code = API_OK cipherText = the cipher of the plain text passed into this action	Test for correct execution of the Encipher action

7.2.1.1.2 Decipher

Table 5 — Decipher – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Decipher	As defined in ISO/IEC 24727-3 clause 10.3	return code = API_OK plainText = plain text that is the deciphered cipher text passed into this action	Test for correct execution of the Decipher action

7.2.1.1.3 GetRandom

Table 6 — GetRandom – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	GetRandom	As defined in ISO/IEC 24727-3 clause 10.4	return code = API_OK random = random value	Test for correct execution of the GetRandom action

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.1.1.4 Hash

Table 7 — Hash – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Hash	As defined in ISO/IEC 24727-3 clause 10.5	return code = API_OK hash = hash of the message passed in	Test for correct execution of the Hash action

7.2.1.1.5 Sign

Table 8 — Sign – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Sign	As defined in ISO/IEC 24727-3 clause 10.6	return code = API_OK signature= signature of the message passed in	Test for correct execution of the Sign action

7.2.1.1.6 VerifySignature

Table 9 — VerifySignature – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	VerifySignature	As defined in ISO/IEC 24727-3 clause 10.7	return code = API_OK	Test for correct execution of the VerifySignature action
2	VerifySignature	As defined in ISO/IEC 24727-3 clause 10.7 An invalid signature shall be passed into this action.	return code = API_INVALID_SIGNATURE	Test for an error condition when an invalid signature is passed into this action.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.1.1.7 VerifyCertificate

Table 10 — VerifyCertificate – Action supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	VerifyCertificate	As defined in ISO/IEC 24727-3 clause 10.8	return code = API_OK	Test for correct execution of the VerifyCertificate action
2	VerifyCertificate	As defined in ISO/IEC 24727-3 clause 10.8 A certificate with an invalid signature shall be passed into this action.	return code = API_INVALID_SIGNATURE	Test for an error condition when a certificate with an invalid signature is passed into this action.

7.2.1.2 Test sequences for unsupported cryptographic functionality

Table 11 through Table 17 define the tests for cryptographic actions if these are not supported by the authentication protocol under test.

7.2.1.2.1 Encipher

Table 11 — Encipher – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Encipher	As defined in ISO/IEC 24727-3 clause 10.2	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

7.2.1.2.2 Decipher

Table 12 — Decipher – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Decipher	As defined in ISO/IEC 24727-3 clause 10.3	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.1.2.3 GetRandom

Table 13 — GetRandom – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	GetRandom	As defined in ISO/IEC 24727-3 clause 10.4	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

7.2.1.2.4 Hash

Table 14 — Hash – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Hash	As defined in ISO/IEC 24727-3 clause 10.5	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

7.2.1.2.5 Sign

Table 15 — Sign – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	Sign	As defined in ISO/IEC 24727-3 clause 10.6	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

7.2.1.2.6 VerifySignature

Table 16 — Verify Signature – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	VerifySignature	As defined in ISO/IEC 24727-3 clause 10.7	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.1.2.7 VerifyCertificate

Table 17 — VerifyCertificate – Action not supported

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	VerifyCertificate	As defined in ISO/IEC 24727-3 clause 10.8	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported

7.2.2 Simple assertion

Table 18 defines the test for Simple Assertion defined in clause A.3 of ISO/IEC 24727-3 if supported.

Table 18 — Authentication protocol sequence – Simple assertion

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = empty qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = empty	Test for correct execution of the DIDGet action
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = empty qualifier = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION	Test for error condition as this action shall not be supported
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action

7.2.3 Asymmetric internal authenticate

Table 19 defines the test for Asymmetric Internal Authenticate defined in clause A.4 of ISO/IEC 24727-3 if supported.

Table 19 — Authentication protocol sequence – Asymmetric internal authenticate

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.4.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.4.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.4.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty	return code = API_OK authenticationProtocolData = result(true)	Test for correct execution of the DIDAuthenticate action
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty	return code = API_OK authenticationProtocolData = result(false)	Test for correct execution of the DIDAuthenticate action, when the asymmetric internal authentication fails. In order to test this scenario an incorrect private key shall be used to verify the signature that is returned from the smartcard.

7.2.4 Asymmetric external authenticate

Table 20 defines the test for Asymmetric External Authenticate defined in clause A.5 of ISO/IEC 24727-3 if supported.

Table 20 — Authentication protocol sequence – Asymmetric external authenticate

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.5.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.5.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.5.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty that is, the samConnectionHandle option was used in the request to DIDAuthenticate	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = privateKey	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
7	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = privateKey an invalid private key shall be passed into this action in order to generate the required return data	return code = API_INVALID_PARAMETER authenticationProtocolData = empty	Test for an error condition of the DIDAuthenticate action that is, an incorrect private key shall be passed into this action.

7.2.5 Symmetric internal authenticate

Table 21 defines the test for Symmetric Internal Authenticate defined in clause A.6 of ISO/IEC 24727-3 if supported.

Table 21 — Authentication protocol sequence – Symmetric internal authenticate

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.6.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.6.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC	return code = API_OK	Test for correct execution of the DIDUpdate action

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
		24727-3 Annex A.6.2 qualifier = empty		
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty that is, the samConnectionHandle option was used in the request to DIDAuthenticate	return code = API_OK authenticationProtocolData = Boolean(true)	Test for correct execution of the DIDAuthenticate action
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = secretKey	return code = API_OK authenticationProtocolData = Boolean(true)	Test for correct execution of the DIDAuthenticate action
7	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = secretKey that is, an invalid secretKey shall be passed in to test the expected return parameters	return code = API_OK authenticationProtocolData = Boolean(false)	Test for correct execution of the DIDAuthenticate action, when the symmetric internal authentication fails. In order to test this scenario an incorrect secret key shall be used to verify the signature that is returned from the smartcard.

7.2.6 Symmetric external authenticate

Table 22 defines the test for Symmetric External Authenticate defined in clause A.7 of ISO/IEC 24727-3 if supported.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Table 22 —Authentication protocol sequence – Symmetric external authenticate

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.7.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.7.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.7.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = empty	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty that is, the samConnectionHandle option was used in the request to DIDAuthenticate	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = secretKey	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action
7	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = secretKey that is, an invalid secretKey shall be passed in to test the expected return parameters	return code = API_INVALID_PARAMETER authenticationProtocolData = empty	Test for error condition of the DIDAuthenticate action, when the symmetric internal authentication fails. In order to test this scenario an incorrect secret key shall be used to verify the signature that is returned from the smartcard.

7.2.7 Compare

Table 23 defines the test for Compare defined in clause A.8 of ISO/IEC 24727-3 if supported.

Table 23 — Authentication protocol sequence – Compare

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.8.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.8.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.8.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = markerTemplate	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = markerTemplate	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = markerTemplate that is, a marker template that is longer than the maxDataLength shall be passed in to test the expected return parameters	return code = API_INVALID_PARAMETER authenticationProtocolData = empty	Test for error condition of the DIDAuthenticate action, when the symmetric internal authentication fails. In order to test this scenario a markerTemplate that is longer than the maxDataLength shall be used to verify the signature that is returned from the smartcard.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

7.2.8 PIN compare

Table 24 defines the test for PIN Compare defined in clause A.9 of ISO/IEC 24727-3 if supported.

Table 24 — Authentication protocol sequence – PIN compare

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.9.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.9.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.9.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = markerTemplate	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = markerTemplate	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = markerTemplate that is, a marker template that is longer than the maxDataLength shall be passed in to test the expected return parameters	return code = API_INVALID_PARAMETER authenticationProtocolData = empty	Test for error condition of the DIDAuthenticate action, when the symmetric internal authentication fails. In order to test this scenario a markerTemplate that is longer than the maxDataLength shall be used to verify the signature that is returned from the smartcard.

7.2.9 Biometric compare

Table 25 defines the test for Biometric Compare defined in clause A.10 of ISO/IEC 24727-3 if supported.

Table 25 — Authentication protocol sequence – Biometric compare

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.10.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.10.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.10.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = MarkerAP010	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = MarkerAP010	return code = API_OK authenticationProtocolData = empty	Test for correct execution of the DIDAuthenticate action
6	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = MarkerAP010 that is, an invalid MarkerAP010 shall be passed in to test the expected return parameters	return code = API_INVALID_PARAMETER authenticationProtocolData = empty	Test for error condition of the DIDAuthenticate action, when the symmetric internal authentication fails. In order to test this scenario an invalid MarkerAP010 shall be used to verify the signature that is returned from the smartcard.

7.2.10 Mutual authentication with key establishment

Table 26 defines the test for Mutual Authenticate with Key Establishment defined in clause A.11 of ISO/IEC 24727-3 if supported.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Table 26 — Authentication protocol sequence – Mutual authentication with key establishment

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.11.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.11.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.11.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = K_seed	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = K_seed	return code = API_OK authenticationProtocolData = Boolean(true)	Test for correct execution of the CardApplicationStartSession action

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
6	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = K_seed	return code = API_INVALID_PARAMETER authenticationProtocolData = Boolean(false)	Test for correct execution of the CardApplicationStartSession action, when the symmetric internal authentication fails. In order to test this scenario an incorrect key, in the off-card application, shall be used during the mutual authentication process.

7.2.11 Client-application mutual authentication with key establishment

Table 27 defines the test for Client-Application Mutual Authentication with Key Establishment defined in clause A.12 of ISO/IEC 24727-3 if supported.

Table 27 — Authentication protocol sequence – Client-application mutual authentication with key establishment

Row number	*Action*	*Parameters*	*Return data*	*Description*
1	DIDCreate	As defined in ISO/IEC 24727-3 clause 11.3 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.12.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDCreate action
2	DIDGet	As defined in ISO/IEC 24727-3 clause 11.4	return code = API_OK DIDStructure as defined in ISO/IEC 24727-3 Annex A.2.2 where marker = as defined in ISO/IEC 24727-3 Annex A.12.2	Test for correct execution of the DIDGet action Not all marker data needs to be returned, some marker data may be left empty to protect the integrity of this authentication protocol.

© ISO/IEC 2011 – All rights reserved

Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011

Row number	*Action*	*Parameters*	*Return data*	*Description*
3	DIDUpdate	As defined in ISO/IEC 24727-3 clause 11.5 with didUpdateData where marker = as defined in ISO/IEC 24727-3 Annex A.12.2 qualifier = empty	return code = API_OK	Test for correct execution of the DIDUpdate action
4	DIDAuthenticate	As defined in ISO/IEC 24727-3 clause 7.4 with authenticationProtocolData = K_seed	return code = API_INAPPROPRIATE_PROTOCOL_FOR_ACTION authenticationProtocolData = empty	Test for error condition as this action shall not be supported
5	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty	return code = API_NEXT_REQUEST authenticationProtocolData = RND.ICC where RND is a challenge from the smartcard.	Test for correct execution of the first CardApplicationStartSession action in a sequence of two.
6	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = Data and MAC that is for mutual authentication purposes	return code = API_OK authenticationProtocolData = Data and MAC	Test for correct execution of the second CardApplicationStartSession action in a sequence of two.
7	CardApplicationStartSession	As defined in ISO/IEC 24727-3 clause 11.7 with authenticationProtocolData = empty	return code = API_NEXT_REQUEST authenticationProtocolData = RND.ICC where RND is a challenge from the smartcard.	Test for correct execution of the first CardApplicationStartSession action in a sequence of two.

© ISO/IEC 2011 – All rights reserved
Documents
Document Sharing Hub
Full document details: BS ISO/IEC 24727-5:2011