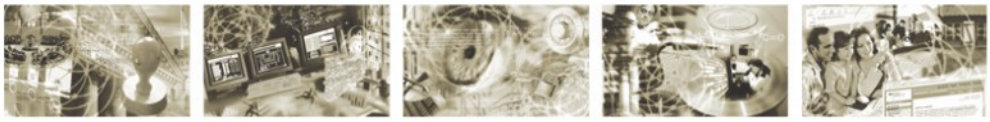
Bundesamt
für Sicherheit in der
Informationstechnik



Technical Guideline

BSI TR-03105 Part 5.2

Test plan for eID and eSign compliant eCard reader systems with EAC 2

Version: 1.2

Date: 2013-11-19

Bundesamt für Sicherheit in der Informationstechnik
Postfach 20 03 63
53133 Bonn
Internet: https://www.bsi.bund.de

Contents

List of Figures

List of Tables

1 Introduction

This Technical Guideline describes conformity criteria for eID and eSign compliant eCard reader systems with EAC 2, in the following also denoted as eCard readers or briefly readers.

The Technical Guideline contains the parts validation rules and test specification.

The validation rules comprise the description of test objects, their test interfaces and their functions as well as the verification requirements. The test specification contains the general structure of test cases and the test cases themselves together with the necessary steps for test preparation, test execution and evaluation of the test results.

The conformity tests defined in this Technical Guideline are functional tests that shall guarantee the interoperability between eCard and reader. They do not address security issues. These topics are treated elsewhere.

The conformity criteria of this Technical Guideline hold for OSI layers 6 and 7 and are based on the EAC 2.0 specification [TR-03110] and on the specification of the contactless interface for signature generation [TR-03117].

Specific properties of the readers as the PC/SC functionality for the PACE protocol follow the Technical Guideline to requirements on eCard readers that support ePA applications [TR-03119].

2 Abbreviations

AID	Application Identification
API	Application Programming Interface
AT	Authentication Template
CA	Chip Authentication
CAN	Card Access Number
CHAT	Card Holder Authorization Template
CIA	Cryptographic Information Application
CLI	Contactless Interface
CT-API	Card Terminal API
CVCA	Country Verifying Certification Authority
DF	Dedicated File
DG	Data Group
DST	Digital Signature Template
DUT	Device Under Test
DV	Document Verifier
EAC	Extended Access Control
eID	Electronic Identity
ID	Identifier
LT	Lower Tester
MRZ	Machine Readable Zone
MSE	Manage Security Environment
OSI	Open Systems Interconnection Reference Model
PACE	Password Authenticated Connection Establishment
PC/SC	Personal Computer / Smartcard
PIN	Personal Identification Number
PSO	Perform Security Operation
PUK	PIN Unblocking Key
RI	Restricted Identification
QCA	Certification Authority issuing qualified certificates
QES	Qualified Electronic Signature
SICCT	Secure Interoperable ChipCard Terminal
SigG	Digital Signature Act (German)

SigV	Digital Signature Ordinance (German)
SM	Secure Messaging
SSCD	Secure Signature Creation Device
TA	Terminal Authentication
UT	Upper Tester

3 Validation Rules

3.1 General Definitions

3.1.1 Verification Task and Scope

The verification task consists in proving interoperability between readers and host applications that access reader functions. Here conformity to the EAC 2.0 specification [TR-03110] and to the specification of the contactless interface for signature generation [TR-03117] must be proven.

The scope for the verification task can be outlined as follows:

- prove conformity of interfaces exclusively for OSI layers 6 and 7,
- verify exclusively reader functions, that are needed for access to eID and biometric data as well as for QES over the contactless interface,
- restrict focus to PC/SC for the interface between host PC and reader, other interfaces as CT-API and SICCT are not relevant in this framework,
- do not examine the behavior of the reader at other interfaces, especially at the display and at the contact interface.

If a manufacturer does not use a PC/SC interface, he has to provide an appropriate adapter to perform the tests.

3.1.2 Test Objects

Due to different hardware and software architectures for the readers, parts of the terminal functions may be implemented outside the reader. These parts of terminal functions are implemented in the host PC and called terminal software in the following. A possible implementation is the eCard API framework as specified in [TR-03112-1].

Since implementation of the reader and of the terminal software in general lead to different products, possibly designed by different manufacturers, and since these products must be interoperable, they are considered as two separate test objects in the document at hand.

Although reader and terminal software are separate test objects, it may be the case in implemented products that this separation cannot be seen from the outside. For evaluation purposes it is admitted to introduce such additional interfaces which are not contained in the product delivered to the customer.

The decision, whether a function of a test object must be included into a test or not, depends on the relevance of this function for the contactless reader interface. Thus, functions for signature generation are incorporated into the tests, but functions for signature verification are not.

3.1.2.1 Test Object Reader

The conformity tests for readers must ensure the correct behavior at interfaces for the PACE, terminal authentication and chip authentication protocols, for the access to eID and biometric data as well as for signature generation. The following sections will describe in detail which functionality has to be verified.

3.1.2.2 Test Object Terminal Software

Depending on the reader architecture, the PACE, terminal authentication and chip authentication protocols as well as the access to the eID application, biometric data and the eSign application have to be performed as a whole or in parts by the terminal software and are therefore subject to the conformity tests.

3.1.3 Test Tracks

The test objects reader and terminal software need separate test tracks for conformity testing because of the differences in their technical interfaces.

3.1.3.1 Test Track for the Reader

In order to access eID and biometric data as well as to use QES, the reader supports two technical interfaces:

1. the PC/SC interface for access by the host PC,
2. the contactless interface (ISO 14443) for communication with an eCard.

To test the functionality of one interface, an appropriate behavior at the other interface is necessary. Therefore a test system is needed, where the test object is situated between an upper tester (UT) and a lower tester (LT). The resulting test track for testing the test object reader is shown in Figure 1

UT sends commands to the test object reader by a call of the PC/SC interface. The reader now generates an expected command sequence that it sends to the eCard. Instead of a physical eCard, the contactless interface at LT uses an eCard simulation. LT analyzes a command received by the reader and generates an answer message for the reader, with or without incorrect data and/or error code, according to the simulator configuration. The reader transmits the answer received from LT back to UT which checks it against the expected result.

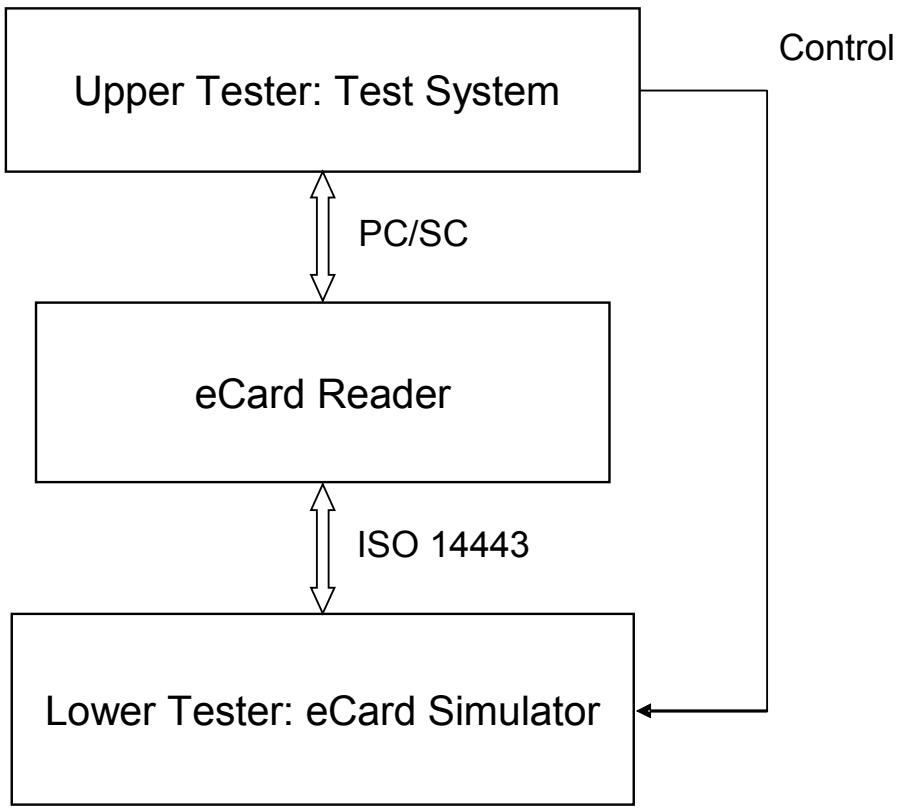


Figure 1: Test track for the reader

Thus, an appropriate configuration of the LT is necessary for test execution. This configuration may be manually or automatically triggered by UT.

3.1.3.2 Test Track for the Terminal Software

In order to access eID and biometric data as well as to use QES, the terminal software supports two technical interfaces:

1. the interface to the calling host PC application,
2. the PC/SC interface for access by the host PC.

Thus, the test system consists of a UT at the interface of the calling host PC application and a LT that simulates the reader with the eCard and receives the PC/SC calls from the terminal software. The resulting test track for testing the data object terminal software is shown in Figure 2

Test system 1 (UT) calls the terminal software. The terminal software generates an expected command sequence of PC/SC calls for a reader. Reader and eCard are simulated by a test system 2 as LT.

LT analyzes a PC/SC call received by the terminal software and generates an answer message for the terminal software, with or without incorrect data and/or error code, according to LT configuration. The terminal software sends this answer message back to UT.

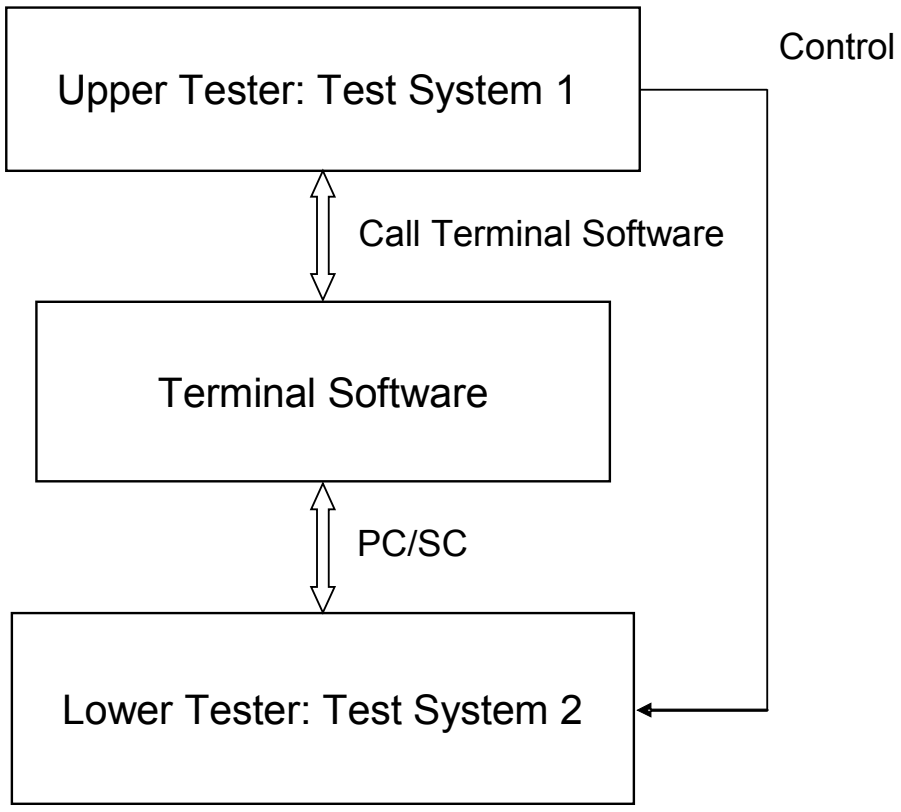


Figure 2: Test track for the terminal software

Thus a triggering mechanism between UT and LT is needed, to appropriately configure LT for test execution. Since the implementations of UT and LT are based on software solutions, it is recommended to implement this mechanism in software.

Note that UT is operated by a human tester. The tester must get an instruction how to activate the necessary functions in the software.

3.1.4 Functions, Options and Profiles

According to the overall hardware and software architecture for reader and host PC parts of the functionality are provided by the reader and/or the terminal software.

Such a distinguished functionality will be called function in the following.

The functions of the whole system, whose conformity has to be proved, are listed in Table 1.

Function	***Task***
Transparent	Transmission of card commands over the PC/SC reader interface to the eCard
PACE	Execution of PACE protocol according to [TR-03110], 4.2
TA	Execution of terminal authentication protocol according to [TR-03110], 4.4
CA	Execution of chip authentication protocol (version 2) according to [TR-03110], 4.3.1.2
eID	Access to eID application
Biometric data	Access to biometric data in the eCard
QES	Generation of qualified electronic signatures according to [TR-03117]

Table 1: Functions of the whole system

The functions of the whole system are taken as a basis to structure the verification requirements (see chapter 3.1.5).

A functionality which may be supported optionally by the whole system will be called an option. The options allowed are listed in Table 2.

Option	***Task***
Change_PIN	Password management function to change the PIN
Change_CAN	Password management function to change the CAN
Change_PIN_PUK	Password management function to change the PIN after using PUK
PIN_MGT_AT	PIN management functions for authentication terminals
PIN_MGT_uT	PIN management functions for unauthenticated terminals after PACE
Write_eID	Writing Access to eID data with EAC

Table 2: Options of the whole system

For the build-up of the tests it is essential whether a function or an option is implemented in the test object reader or in the test object terminal software. Therefore, a profile defines the assignment of a function or option to test object. The profiles allowed are listed in Table 3.

Profile	***Test Object***	***Function***	***Option***
R_Tra	Reader	Transparent	
R_PACE	Reader	PACE	
R_TA	Reader	TA	
R_CA	Reader	CA	
R_eID	Reader	eID	
R_bio	Reader	Biometric data	
R_Sig	Reader	QES	
R_Chg_PIN	Reader		Change PIN
R_Chg_CAN	Reader		Change CAN
R_Chg_PIN_PUK	Reader		Change_PIN after PUK
R_PIN_MGT_AT	Reader		PIN_MGT_AT
R_PIN_MGT_uT	Reader		PIN_MGT_uT
TS_PACE	Terminal Software	PACE	
TS_TA	Terminal Software	TA	
TS_CA	Terminal Software	CA	
TS_eID	Terminal Software	eID	
TS_bio	Terminal Software	Biometric data	
TS_Sig	Terminal Software	QES	
TS_Chg_PIN	Terminal Software		Change PIN
TS_Chg_CAN	Terminal Software		Change CAN
TS_Chg_PIN_PUK	Terminal Software		Change_PIN after PUK
TS_PIN_MGT_AT	Terminal Software		PIN_MGT_AT
TS_PIN_MGT_uT	Terminal Software		PIN_MGT_uT

TS_Write_eID	Terminal Software		Write_eID

Table 3: Profiles defined for the test objects

3.1.5 Description of Verification Requirements

Special verification requirements are defined for each test object that describe which aspects must be validated with respect to the behavior of the test object at the relevant interfaces. The verification requirements are structured along the functions in separate sections.

A verification requirement handles a special aspect of the specification (e. g. the special part age verification in the eID application) and assigns to this aspect a test parameter or a defined combination of test parameters resp. preconditions as verification focus.

Moreover a verification requirement determines whether the test is a positive or a negative one and according to which principles the test parameters have to be chosen. Table 4 lists the test classes that define different selection modes for test parameters:

Test class	***Description***	***Selection mode of a positive test***	***Selection mode of a negative test***
CT	Complete test	Check all admitted values of a test parameter	Check all forbidden values of a test parameter
LM	Limit test	Check limits of the admitted value range of a test parameter. Apart from the lower and upper limits LL and UL, the values UL-1 and LL+1 are tested as well.	Check limits of the value range of a test parameter., where UL+1 and LL-1 are tested.
SA	Sample Test	Check appropriate values within the range of a test parameter	Check appropriate values outside the range of a test parameter

Table 4: Test classes for parameters and their selection modes

The test classes LM and SA may be combined if necessary.

The verification requirements of the same section are described in one table. Table 5 shows the structure of such a table:

Name in table column	*Contents*
RQ_no	Identifier for verification requirement built up as follows: <code test object>_<code function>_<No. VR>.<No. para> <code test object>: R (for reader) TS (for terminal software) <code function>: Tra (for transparent mode) PACE (for PACE protocol) TA (for terminal authentication protocol) CA (for chip authentication protocol) eID (for eID application) bio (for biometric data) Sig (for eSign application) <No. VR> (verification requirement number): 1, 2, 3 ... <No. para> (distinguish parameters within a verification requirement): 1, 2, 3 ...
Description	Short description of the verification requirement
Parameter	Parameter in focus of examination; this also includes combinations of parameters
Reference	Chapters in the specification which define the verification requirement
PR	Test classes for positive tests („positive requirement“); if this field is empty, there are no positive tests for the considered parameters
NR	Test classes for negative tests („negative requirement“); if this field is empty, there are no negative tests for the considered parameters
IF	Interface of test objects that are in focus of examination: UT – Upper Tester LT – Lower Tester

Table 5: Structure of a table to define verification requirements

3.2 Verification Requirements for Test Object Reader

3.2.1 Transparent Mode

3.2.1.1 List of Verification Requirements

RQ_no R_Tra_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct transmission of card commands to reader in transparent mode	UT sends card command to reader via PC/SC interface to read data from eCard which are freely accessible. It is checked that the reader transmits command message to LT and transmits to UT response message received from LT, both without modifications.	[TR-03110], A.1.1.6	SA		UT LT
1.2		UT sends card command to reader via PC/SC interface to read data from eCard for which access is not allowed. It is checked that the reader transmits command message to LT and transmits to UT response message received from LT, both without modifications.	[TR-03110], A.1.1.6		SA	UT LT

Table 6: Verification requirements for transparent mode, execution in reader

3.2.2 PACE

3.2.2.1 General Preliminary Remarks

The PACE protocol is the first part of the EAC protocol that is used to access eID or signature function in the eCard. In order to perform PACE protocol, the PC/SC function GET_FEATURE_REQUEST ([PCSC10], 2.2) extended by the feature FEATURE_EXECUTE_PACE is called in the reader. The structure to this feature is named EstablishPACEChannel ([TR-03119], A.11.1.1).

InputData of EstablishPACEChannel consist of ([TR-03119], A.11.2.2)

- PIN-ID for user authentication,
- CHAT (Card Holder Authorization Template),
- PIN (if transmitted by host PC) and
- complete description of terminal certificate.

3.2.2.2 List of Verification Requirements

RQ_no R_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of PACE protocol in the reader	Use certificate roles (inspection system, authentication terminal, signature terminal) with specified access rights. Use appropriate certificates for Document Verifier Use all possible combinations of certificate role: inspection system: password-ID = CAN and MRZ password authentication terminal: password-ID = PIN, CAN signature terminal: password-ID = CAN, PIN and PUK	[TR-03110], 3.4, 4.2, C.4.1, C.4.2, C.4.3 [TR-03119], D.3	SA		UT
1.2		Use unauthenticated terminals with CAN, PIN and PUK	[TR-03110], 3.5.1, 4.2 [TR-03119], D.3	SA		UT
1.3		Use of different algorithms	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1	SA		LT
2.1	Check that reader aborts PACE protocol when detecting internal error or error communicated by LT	LT derives cryptographic key from password (PIN, CAN, MRZ-Password, PUK) incorrectly	[TR-03110], 4.2 [TR-03119], D.3.1		SA	LT
2.2		LT returns error code to command MSE: Set AT	[TR-03119], D.3.1		SA	LT
2.3		LT transmits incorrect data for mapping function Map in answer to card command General Authenticate (Step 2).	[TR-03110], 4.2, B.1 [TR-03119], D.3.1		SA	LT
2.4		LT generates incorrect internal data	[TR-03110],		SA	LT

RQ_no R_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		with function Map.	4.2			
2.5		LT generates incorrect ephemeral key pair	[TR-03110], 4.2		SA	LT
2.6		LT computes key data for secure messaging (SM) incorrectly.	[TR-03110], 4.2 [TR-03119], D.3.1		SA	LT
3.1	Check that reader aborts PACE protocol when reader receives incorrect data from LT	Transmission of incorrect PACE parameters from LT	[TR-03110], 4.2 [TR-03119], D.3.1		SA	LT
3.2		Incorrect cryptogram in response message to card command General Authenticate (Step 1)	[TR-03110], 4.2, B.1 [TR-03119], D.3.1		SA	LT
3.3		Ephemeral public key received from LT is identical to ephemeral public key generated by reader	[TR-03110], 4.2		SA	LT
3.4		LT transmits incorrect cryptogram that has been generated with derived SM key data	[TR-03110], 4.2		SA	LT
3.5		LT transmits authentication token that has been generated with an algorithm that differs from the algorithm used by reader	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1		SA	LT
4.1	Check that reader aborts PACE protocol when receiving incorrect input data from UT	Incorrect password-ID in input of PC/SC function	[TR-03119], A.11.2.2		LM SA	UT
4.2		Incorrect password (PIN, CAN, MRZ password, PUK) in input of PC/SC function or in user input	[TR-03110], 4.2 [TR-03119], A.11.2.2		SA	UT

RQ_no R_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
4.3		Incorrect combination of password-ID and CHAT in input of PC/SC function (e. g. password-ID addresses CAN; authentication terminal not authorized to use CAN)	[TR-03110], 4.2 [TR-03119], A.11.2.2		SA	UT
5.1	Check that reader supports Secure Messaging in a second attempt	eID-PIN suspended PACE with unauthenticated terminal using CAN	[TR-03110], 4.2.2, 3.5.1 [TR-03119], D.3	SA		UT

Table 7: Verification requirements for PACE, execution in reader

3.2.3 Terminal Authentication

3.2.3.1 General Preliminary Remarks

The terminal authentication protocol is the second part of the EAC protocol. It is executed only in signature terminals. In authentication terminals command messages and answer messages for the eCard used in the protocol are transmitted between UT and LT. Here the reader merely performs SM using the session keys as derived in PACE protocol.

3.2.3.2 List of Verification Requirements

RQ_no R_TA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of terminal authentication protocol in the reader	Use certificate roles inspection system, authentication terminal and signature terminal with specified access rights	[TR-03110], 3.4, 4.4, B.3 C.4.1, C.4.2, C.4.3 [TR-03119], D.3	SA		UT
1.2	deleted in version 1.2	Use of different algorithms	[TR-03110], 4.4, A.1.1.3, A.6, B.3	SA		LT
2.1	Check that reader aborts terminal authentication when discovering inconsistent internal data	Terminal certificate CHAT used in terminal authentication different from CHAT used in PACE protocol	[TR-03110], 4.4, B.3 [TR-03119], A.10.2		SA	UT
3.1	Check that reader	LT returns error code to command	[TR-03110],		SA	LT

RQ_no R_TA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
	aborts terminal authentication when detecting internal error or error communicated by LT	MSE: Set DST (setting public key for certificate verification)	4.4, B.3			
3.2		LT returns error code to command PSO: Verify Certificate when verifying terminal certificates	[TR-03110], 4.4, B.3		SA	LT
3.3		LT returns error code to command MSE: Set AT (transmitting parameters for terminal authentication to LT)	[TR-03110], 4.4, B.3		SA	LT
3.4		LT returns error code to command Get Challenge (random number for terminal authentication)	[TR-03110], 4.4, B.3		SA	LT
3.5		LT returns error code to command External Authenticate when checking signed data from UT in terminal authentication protocol.	[TR-03110], 4.4, B.3		SA	LT
4.1	Check that reader aborts terminal authentication on Secure Messaging error in LT	Reader does not receive SM data objects from LT in answer messages of LT	[TR-03110], 4.4, B.3		SA	LT
4.2		Reader receives incorrect SM data objects from LT in answer messages of LT	[TR-03110], B.3, 4.4		SA	LT
4.3		The reader aborts terminal authentication, if it receives that wrong SM data from LT in response APDU to command MSE: Set DST	[TR-03110]		SA	LT

Table 8: Verification requirements for terminal authentication, execution in reader

3.2.4 Chip Authentication

3.2.4.1 General Preliminary Remarks

The chip authentication protocol is the third part of the EAC protocol. It is executed completely only in signature terminals. In authentication terminals command messages and answer messages for the eCard used in the protocol are transmitted between UT and LT. Here the reader merely performs SM using the session keys as derived in PACE protocol.

3.2.4.2 List of Verification Requirements

RQ_no R_CA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of chip authentication protocol in the reader	Use certificate roles inspection system, authentication terminal and signature terminal with specified access rights	[TR-03110], 3.4, 4.3, B.2 C.4.1, C.4.2, C.4.3 [TR-03119], D.3	SA		UT
1.2		Use of different algorithms and multiple key pairs	[TR-03110], 4.3, A.1.1.2, A.4, B.2	SA		LT
2.1	Check that reader aborts chip authen-tication when detecting internal error or error communicated by LT	LT returns error code to command MSE: Set AT (transmitting parameters for chip authentication to LT)	[TR-03110], 4.3, B.2		SA	LT
2.2		LT returns error code to command General Authenticate when verifying the ephemeral public key of the reader	[TR-03110], 4.3, B.2		SA	LT
2.3		LT computes incorrect key data for SM	[TR-03110], 4.3, B.2		SA	LT
3.1	Check that reader aborts chip authen-tication when receiving incorrect data from LT	LT returns incorrect random number in answer message to command General Authenticate	[TR-03110], 4.3, B.2		SA	LT
3.2		LT returns incorrect cryptogram in answer message to command	[TR-03110], 4.3, B.2		SA	LT

RQ_no R_CA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		General Authenticate				
3.3		LT transmits authentication token that has been generated with an algorithm that differs from the algorithm used by reader	[TR-03110], 4.3, A.1.1.2, A.4, B.2		SA	LT
3.4		LT transmits authentication token that has been generated with a wrong static key pair.	[TR-03110], 4.3, A.1.1.2, A.4, B.2		SA	LT

Table 9: Verification requirements for chip authentication, execution in reader

3.2.5 Access to the eID Application

3.2.5.1 General Preliminary Remarks

The access to the eID application is only admitted after successful execution of the EAC protocol (General Authentication Procedure). Here using the reader, UT must have been authenticated to LT with certificate role inspection system or authentication terminal. After performing EAC the SM channel is established between UT and LT, i. e. the commands for the eID application are transmitted by the reader in transparent mode. Therefore the verification requirements for the eID application exclusively handle positive tests.

3.2.5.2 List of Verification Requirements

RQ_no R_eID_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct reading access to eID data with EAC by the reader	Use certificate role inspection system with specified access rights	[TR-03110], C.4.1, E.1	CT		UT
1.2		Use certificate role authentication terminal with specified access rights	[TR-03110], C.4.2, E.1	CT		UT
1.3		Use of different algorithms for Secure Messaging	[TR-03110], A.4, C.4.2, E.1, F.2	SA		LT
2.1	Check correct writing access to eID data with EAC by the reader	Use certificate role authentication terminal with specified access rights	[TR-03110], C.4.2, E.1	CT		UT

RQ_no R_eID_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
3.1	Check correct execution of internal eID functions that have been called in LT by the reader	Certificate role authentication terminal with access right restricted identification: Execute restricted identification protocol	[TR-03110], 4.5, C.4.2	SA		UT
3.2		Certificate role authentication terminal with access right age verification: Execute age verification for card holder	[TR-03110], C.4.2	SA LM		UT
3.3		Certificate role authentication terminal with access right community ID verification: Execute verification of card holder's community ID	[TR-03110], C.4.2	SA		UT
4.1	Password management functions for authenticated terminals	Certificate role authentication terminal with access right password management: Changing PIN in LT optional (if supported by terminal)	[TR-03110], 3.5.2, C.4.2	SA		UT
4.2		Certificate role authentication terminal with access right password management: Changing CAN in LT optional (if supported by terminal)	[TR-03110], 3.5.2, C.4.2	SA		UT
4.3		Certificate role authentication terminal with access right password management: Unblock PIN in LT optional (if supported by terminal)	[TR-03110], 3.5.2, C.4.2	SA		UT
4.4		Certificate role authentication terminal with access right password management: Activate PIN in LT	[TR-03110], 3.5.2, C.4.2	SA		UT

RQ_no R_eID_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
4.5		Certificate role authentication terminal with access right password management: Deactivate PIN in LT	[TR-03110], 3.5.2, C.4.2	SA		UT
5.1	Password management functions for unauthenticated terminals after PACE	Setting a new PIN using the currently valid PIN Consider the special case that the current PIN is a transport PIN	[TR-03110], 3.5.1	SA		UT
5.2		Resetting retry counter for PIN using PUK	[TR-03110], 3.5.1	SA		UT
5.3		Setting a new PIN using PUK optional (if supported by reader)	[TR-03110], 3.5.1	SA		UT
5.4		Resume temporarily a PIN using CAN	[TR-03110], 3.5.1	SA		UT
5.5		Resume a temporarily resumed PIN by using it in PACE protocol	[TR-03110], 3.5.1	SA		UT

Table 10: Verification requirements for eID application, execution in reader

3.2.6 Access to Biometric Data

3.2.6.1 General Preliminary Remarks

The access to the data groups of the ePassport application containing biometric data is admitted for inspection systems with appropriate access rights after successful execution of the EAC protocol (General Authentication Procedure).

3.2.6.2 List of Verification Requirements

RQ_no R_bio_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct reading access to biometric data with EAC by the reader	Use certificate role inspection system with access rights for DG 3 (Fingerprint) and DG 4 (Iris) of the ePassport application	[TR-03110], C.4.1	CT		UT

Table 11: Verification requirements for access to biometric data, execution in reader

3.2.7 Use of the Digital Signature Application

3.2.7.1 General Preliminary Remarks

In this section verification requirements will be defined, that focus the correct behaviour of readers during the processing of functions related to qualified electronic signatures according to [TR-03117]. In this application the reader is part of a so called “signature application component” and the eCard is a secure signature creation device (SSCD) according to the German digital signature act [SigG] and the signature ordinance [SigV].

According to the Technical Guideline [TR-03117], that defines requirements for the reader and the eCard, the following functions and procedures have to be considered:

- Activation of the eCard as SSCD including
  - Setting of the operational user authentication data (Setting of the eSign-PIN, see also management of user authentication data)
  - Generation of the signature key pair by the certification authority (QCA) issuing qualified certificates
- Operational use of the eCard as SSCD
  - Options for entering the CAN
  - Generation of qualified electronic signatures
  - Management of the user authentication data
    - Setting a new eSign-PIN
    - Changing the eSign-PIN
    - Unblocking the eSign-PIN
  - Termination of the signature function
    - Terminating the eSign-PIN
    - Terminating the signature key

For the processing of these procedures it is necessary to use the roles authentication reader and signature reader. According to [TR-03117], Annex A2, Table 1 Inspection systems have no access to the eSign application.

Authentication readers must have the access right „Install Qualified Certificate“ and signature readers the access right „Generate qualified electronic signature“. With the exception of access rights requirements for advanced signatures are not specified yet in [TR-03117], so this document at hand does not define any test requirement for advanced signatures.

3.2.7.2 List of Verification Requirements

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Verification of the	Use of the role authentication	[TR-03117],	CT		UT

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
	correct execution of the key pair generation initiated by the QCA	terminal (of the QCA) with the access rights to read personal data of the eCard-User as well as for issuing and installation of the qualified certificate („Install Qualified Certificate“) Use of the PIN for the PACE protocol between the signature terminal and the LT.	4.3.2			
1.2		Access with the authentication terminal of the QCA to the necessary identification data of the eCard-User (stored in the eCard) according to §5 clause 1 SigG and §3 clause 1 SigV (where necessary DG1 to DG21). Select the eID application Read personal data	[TR-03117], 4.3.2	CT		UT
1.3		Access with the authentication terminal to the eSign application (Select the eSign application) Transmission of the command Generate Asymmetric Key Pair for the generation of the key pair from the UT via authentication and signature terminal to the LT. The authentication terminal reads the generated signature verification key from the LT.	[TR-03117], 4.3.2, A.2.5	CT		UT
2.1	Verification that the reader aborts the key pair generation when detecting internal error or error communicated by LT	Authentication terminal does not have the access right „Install Qualified Certificate“ Use of a wrong password for the PACE protocol	[TR-03117], 4.3.2		SA	UT
2.2		The LT returns that the eSign application could not be selected The LT returns an error code to the command Generate Asymmetric Key Pair:	[TR-03117], 4.3.2, A.2.5		SA	LT

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		The private key or the related eSign-PIN is not terminated				
3.1	deleted in Version 1.1					
3.2	deleted in Version 1.1					
4.1	Verification of the correct execution of a signature generation	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Use of the CAN for the PACE protocol between signature terminal and LT	[TR-03117], 4.4.2	CT		UT
4.2		The UT selects the eSign application and requires via the signature terminal the eCard-User to enter the eSign-PIN at the signature terminal Entering the eSign-PIN at the signature terminal Transmission of the command Verify for the successful verification of the eSign-PIN to the LT Response from the LT to the signature terminal or the UT	[TR-03117], 4.4.2, A.2.1	CT		UT
4.3		The UT transfers the command PSO:Compute Digital Signature for the generation of a digital signature to the signature terminal Transfer of the command by the signature terminal to the LT Return of the generated signature from the LT to the signature terminal Return of the received signature by the signature terminal to the UT	[TR-03117], 4.4.2, A.2.4	SA		UT
5.1	Verification that the	Signature terminal does not have	[TR-03117],		SA	UT

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
	reader aborts the signature generation when detecting internal error or error communicated by LT	the access right „Generate qualified electronic signature“ Use of password for the PACE protocol which doesn't have the required permits.	4.4.2			
5.2		The LT returns that the eSign application could not be selected The LT returns an error code to the command Verify: Failed user authentication, X further attempts The error counter of the eSign-PIN is expired (no further attempts for user authentication possible) The eSign-PIN is terminated The referenced password is not available The LT returns an error code to the command PSO:Compute Digital Signature: The related eSign-PIN was not verified successfully The signature key is terminated Incorrect command data	[TR-03117], 4.4.2, A.2.1, A.2.4		CT	LT
6.1	Password management functions for signature terminals	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Setting eSign-PIN in LT using PIN	[TR-03117], 4.4.3, A.2.2	SA		UT
6.2		Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Changing eSign-PIN in LT using	[TR-03117], 4.4.3, A.2.2	SA		UT

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		CAN				
6.3		Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Resetting retry counter for eSign-PIN using PUK	[TR-03117], 4.4.3, A.2.3	SA		UT
7.1	Verification that the signature terminal aborts the password management function when detecting internal error or error communicated by LT	Setting the eSign-PIN Signature terminal does not have the access right „Generate qualified electronic signature“ Use of password for the PACE protocol which doesn't have the required permits.	[TR-03117], 4.4.3		SA	UT
7.2		Setting the eSign-PIN The LT returns that the eSign application could not be selected The LT returns an error code to the command Change Reference Data	[TR-03117], 4.4.3, A.2.2		SA	LT
7.3		Changing the eSign-PIN Signature terminal does not have the access right „Generate qualified electronic signature“ Use of password for the PACE protocol which doesn't have the required permits.	[TR-03117], 4.4.3		SA	UT
7.4		Changing the eSign-PIN The LT returns an error code to the command Change Reference Data	[TR-03117], 4.4.3, A.2.2		SA	LT
7.5		Resetting Retry Counter of eSign-PIN Signature terminal does not have the access right „Generate	[TR-03117], 4.4.3, A.2.3		SA	UT

RQ_no R_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		qualified electronic signature“ Use of password for the PACE protocol which doesn't have the required permits.				
7.6		Resetting Retry Counter of eSign-PIN The LT returns an error code to the command Reset Retry Counter	[TR-03117], 4.4.3		SA	LT
8.1	Verification of the correct execution of the termination of the signature function	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Termination of the eSign-PIN or the signature key with PIN	[TR-03117], 4.4.4, A.2.6	CT		UT
9.1	Verification that the signature terminal aborts a termination function when detecting internal error or error communicated by LT	Termination of the eSign-PIN or the signature key Signature terminal does not have the access right „Generate qualified electronic signature“ Use of password for the PACE protocol which doesn't have the required permits.	[TR-03117], 4.4.4		SA	UT
9.2		The LT returns an error code to the command Terminate	[TR-03117], 4.4.4, A.2.6		SA	LT

Table 12: Verification Requirements use of the signature application, execution in the reader

3.3 Verification Requirements for Test Object Terminal Software

3.3.1 PACE

3.3.1.1 List of Verification Requirements

RQ_no TS_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of PACE protocol in the terminal software	Use certificate roles (inspection system, authentication terminal, signature terminal) with specified access rights. Use appropriate certificates for Document Verifier Use all possible combinations of certificate role: inspection system: password-ID = CAN and MRZ-Password authentication terminal: password-ID = PIN, CAN signature terminal: password-ID = CAN, PIN and PUK	[TR-03110], 3.4, 4.2, C.4.1, C.4.2, C.4.3	SA		UT
1.2		Use unauthenticated terminals with CAN, PIN and PUK	[TR-03110], 3.5.1, 4.2	SA		UT
1.3		Use of different algorithm	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1	SA		LT
2.1	Check that reader aborts PACE protocol when detecting internal error or error communicated by LT	LT derives cryptographic key from password (PIN, CAN) incorrectly	[TR-03110], 4.2		SA	LT
2.2		LT returns error code to command MSE: Set AT	[TR-03110], 4.2		SA	LT
2.3		LT transmits incorrect data for mapping function Map in answer	[TR-03110], 4.2, B.1		SA	LT

RQ_no TS_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		to card command General Authenticate (Step 2).				
2.4	deleted in Version 1.1					
2.5		LT generates incorrect internal data with function Map.	[TR-03110], 4.2		SA	LT
2.6		LT computes key data for secure messaging (SM) incorrectly.	[TR-03110], 4.2		SA	LT
3.1	Check that terminal software aborts PACE protocol when reader receives incorrect data from LT	Transmission of incorrect PACE parameters from LT	[TR-03110], 4.2		SA	LT
3.2		Incorrect cryptogram in response message to card command General Authenticate (Step 1)	[TR-03110], 4.2, B.1		SA	LT
3.3		Ephemeral public key received from LT is identical to ephemeral public key generated by terminal software	[TR-03110], 4.2		SA	LT
3.4		LT transmits incorrect cryptogram that has been generated with derived SM key data	[TR-03110], 4.2		SA	LT
3.5		LT transmits authentication token that has been generated with a wrong algorithm	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1		SA	LT
4.1	Check that terminal software aborts PACE protocol when receiving incorrect input data from UT	Incorrect password (PIN, CAN, MRZ)	[TR-03110], 4.2		SA	UT
4.2		Incorrect combination of password-ID and terminal certificate (e. g. password-ID	[TR-03110], 4.2		SA	UT

RQ_no TS_PACE_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		addresses CAN; authentication terminal not authorized to use CAN)				

Table 13: Verification requirements for PACE, execution in terminal software

3.3.2 Terminal Authentication

3.3.2.1 List of Verification Requirements

RQ_no TS_TA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of terminal authentication protocol in the terminal software	Use certificate roles inspection system, authentication terminal and signature terminal with specified access rights	[TR-03110], 3.4, 4.4, B.3 C.4.1, C.4.2, C.4.3	SA		UT
1.2		Use of different algorithms	[TR-03110], 4.4, A.1.1.3, A.6, B.3	SA		LT
2.1	Check that terminal software aborts terminal authentication when discovering inconsistent internal data	CHAT of terminal certificate used in terminal authentication different from CHAT used in PACE protocol	[TR-03110], 4.2, B.3		SA	UT
3.1	Check that terminal software aborts terminal authentication when detecting internal error or error communicated by LT	LT returns error code to command MSE: Set DST (setting public key for certificate verification)	[TR-03110], 4.4, B.3		SA	LT
3.2		LT returns error code to command PSO: Verify Certificate when verifying terminal certificates	[TR-03110], 4.4, B.3		SA	LT
3.3		LT returns error code to command	[TR-03110],		SA	LT

RQ_no TS_TA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		MSE: Set AT (transmitting parameters for terminal authentication to LT)	4.4, B.3			
3.4		LT returns error code to command Get Challenge (random number for terminal authentication)	[TR-03110], 4.4, B.3		SA	LT
3.5		LT returns error code to command External Authenticate when checking signed data from UT in terminal authentication protocol.	[TR-03110], 4.4, B.3		SA	LT
4.1	Check that terminal software aborts terminal authentication on Secure Messaging error in LT	Terminal software does not receive SM data objects from LT in answer messages of LT	[TR-03110], 4.4, B.3		SA	LT
4.2		Terminal software receives incorrect SM data objects from LT in answer messages of LT	[TR-03110], 4.4, B.3		SA	LT

Table 14: Verification requirements for terminal authentication, execution in terminal software

3.3.3 Chip Authentication

3.3.3.1 List of Verification Requirements

RQ_no TS_CA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct execution of chip authentication protocol in the terminal software	Use certificate roles inspection system, authentication terminal and signature terminal with specified access rights	[TR-03110], 3.4, 4.3, B.2 C.4.1, C.4.2, C.4.3	SA		UT
1.2		Use of different algorithms and multiple key pairs	[TR-03110], 4.3, A.1.1.2, A.4, B.2	SA		LT
2.1	Check that terminal software aborts chip authen-tication when detecting	LT returns error code to command MSE: Set AT (transmitting parameters for chip authentication to LT)	[TR-03110], 4.3, B.2		SA	LT

RQ_no TS_CA_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
	internal error or error communicated by LT					
2.2		LT returns error code to command General Authenticate when verifying the ephemeral public key of the reader	[TR-03110], 4.3, B.2		SA	LT
2.3	deleted in version 1.1					
3.1	Check that terminal software aborts chip authen-tication when receiving incorrect data from LT	LT returns incorrect random number in answer message to command General Authenticate	[TR-03110], 4.3, B.2		SA	LT
3.2	deleted in version 1.1					
3.3		LT transmits authentication token that has been generated with a wrong algorithm	[TR-03110], 4.3, A.1.1.2, A.4, B.2		SA	LT
3.4		LT transmits authentication token that has been generated with a wrong static key pair	[TR-03110], 4.3, A.1.1.2, A.4, B.2		SA	LT

Table 15: Verification requirements for chip authentication, execution in terminal software

3.3.4 Access to the eID Application

3.3.4.1 General Preliminary Remarks

The access to the eID application is only admitted after successful execution of the EAC protocol (General Authentication Procedure). Here using the terminal software, UT must have been authenticated to LT with certificate role inspection system or authentication terminal. After performing EAC the SM channel is established between UT and LT, i. e. the commands for the eID application are transmitted by the terminal software in transparent mode. Therefore the verification requirements for the eID application exclusively handle positive tests.

3.3.4.2 List of Verification Requirements

RQ_no TS_eID_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct reading access to eID data with EAC by the terminal software	Use certificate role inspection system with specified access rights	[TR-03110], C.4.1, E.1	CT		UT
1.2		Use certificate role authentication terminal with specified access rights	[TR-03110], C.4.2, E.1	CT		UT
1.3		Use of different algorithms for Secure Messaging	[TR-03110], A.4, C.4.2, E.1, F.2	SA		LT
2.1	Check correct writing access to eID data with EAC by the terminal software	Use certificate role authentication terminal with specified access rights	[TR-03110], C.4.2, E.1	CT		UT
3.1	Check correct execution of internal eID functions that have been called in LT by the terminal software	Certificate role authentication terminal with access right restricted identification: Execute restricted identification protocol	[TR-03110], 4.5, C.4.2	SA		UT
3.2		Certificate role authentication terminal with access right age verification: Execute age verification for card holder	[TR-03110], C.4.2	SA LM		UT
3.3		Certificate role authentication terminal with access right community ID verification: Execute verification of card holder's community ID	[TR-03110], C.4.2	SA		UT
4.1	Password management functions for authenticated terminals	Certificate role authentication terminal with access right password management: Changing PIN in LT optional (if supported by terminal)	[TR-03110], 3.5.2, C.4.2	SA		UT
4.2		Certificate role authentication	[TR-03110],	SA		UT

RQ_no TS_eID_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
		terminal with access right password management: Changing CAN in LT optional (if supported by terminal)	3.5.2, C.4.2			
4.3		Certificate role authentication terminal with access right password management: Unblock PIN in LT	[TR-03110], 3.5.2, C.4.2	SA		UT
4.4		Certificate role authentication terminal with access right password management: Activate PIN in LT	[TR-03110], 3.5.2, C.4.2	SA		UT
4.5		Certificate role authentication terminal with access right password management: Deactivate PIN in LT	[TR-03110], 3.5.2, C.4.2	SA		UT
5.1	Password management functions for unauthenticated terminals after PACE	Setting a new PIN using the currently valid PIN Consider the special case that the current PIN is a transport PIN	[TR-03110], 3.5.1	SA		UT
5.2		Resetting retry counter for PIN using PUK	[TR-03110], 3.5.1	SA		UT
5.3		Setting a new PIN using PUK optional (if supported by reader)	[TR-03110], 3.5.1	SA		UT
5.4		Resume temporarily a PIN using CAN	[TR-03110], 3.5.1	SA		UT
5.5		Resume a temporarily resumed PIN by using it in PACE protocol	[TR-03110], 3.5.1	SA		UT

Table 16: Verification requirements for eID application, execution in terminal software

3.3.5 Access to Biometric Data

3.3.5.1 General Preliminary Remarks

The access to the data groups of the ePassport application containing biometric data is admitted for inspection systems with appropriate access rights after successful execution of the EAC protocol (General Authentication Procedure).

3.3.5.2 List of Verification Requirements

RQ_no TS_bio_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Check correct reading access to biometric data with EAC by the terminal software	Use certificate role inspection system with access rights for DG 3 (Fingerprint) and DG 4 (Iris) of the ePassport application	[TR-03110], C.4.1	CT		UT

Table 17: Verification requirements for access to biometric data, execution in terminal software

3.3.6 Use of the Signature Application

3.3.6.1 General Preliminary Remarks

The access to the eSign application is admitted for authentication and signature terminals with appropriate access rights “Install Qualified Certificate” for authentication terminals and “Generate qualified electronic signature” for signature terminals. The successful authentications PACE, terminal authentication and chip authentication are preconditions for the execution of functions concerning the qualified electronic signature.

3.3.6.2 List of Verification Requirements

RQ_no TS_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.1	Verification of the correct execution of the key pair generation initiated by the QCA	Use of the role authentication terminal (of the QCA) with the access rights to read personal data of the eCard-User as well as for issuing and installation of the qualified certificate („Install Qualified Certificate“) Use of the PIN for the PACE protocol	[TR-03117], 4.3.2	CT		UT

RQ_no TS_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
1.2		Access with the authentication terminal of the QCA to the necessary identification data of the eCard-User (stored in the eCard) according to §5 clause 1 SigG and §3 clause 1 SigV (where necessary DG1 to DG21) Retrieval of personal data	[TR-03117], 4.3.2	CT		UT
1.3		Access with the authentication terminal to the eSign application Key pair generation and retrieval of public key	[TR-03117], 4.3.2	CT		UT
2.1	Verification that the terminal software handles the key pair generation properly when detecting internal error or error communicated by LT	Authentication terminal does not have the access right „Install Qualified Certificate“ Use of a wrong password for the PACE protocol	[TR-03117], 4.3.2		SA	UT
2.2		The LT returns that the eSign application could not be selected The LT returns an error code to the key pair generation request	[TR-03117], 4.3.2		SA	LT
3.1	deleted in version 1.1					
3.2	deleted in version 1.1					
4.1	Verification of the correct execution of a signature generation	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“	[TR-03117], 4.4.2	CT		UT
4.2		The UT selects the eSign application and requires a password verification via the signature terminal Response from the LT to the software and transfer the UT	[TR-03117], 4.4.2	CT		UT

RQ_no TS_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
4.3		The UT requires the generation of a digital signature Return of the generated signature from the LT to the software and transfer to the UT	[TR-03117], 4.4.2	SA		UT
5.1	Verification that the terminal software handles the signature generation properly when detecting internal error or error communicated by LT	Signature terminal does not have the access right „Generate qualified electronic signature“	[TR-03117], 4.4.2		SA	UT
5.2		The LT returns that the eSign application could not be selected The LT returns an error code to the request for password verification The LT returns an error code to the request for signature generation	[TR-03117], 4.4.2		CT	LT
6.1	Verification of the correct execution of eSign-PIN management functions	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Setting eSign-PIN in LT with PIN	[TR-03117], 4.4.3	SA		UT
6.2		Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Changing eSign-PIN in LT with CAN	[TR-03117], 4.4.3	SA		UT
6.3		Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Resetting retry counter for eSign-PIN in LT with PUK	[TR-03117], 4.4.3	SA		UT

RQ_no TS_Sig_	***Description***	***Parameter***	***Reference***	***PR***	***NR***	***IF***
7.1	Verification that the terminal software handles the password management function properly when detecting internal error or error communicated by LT	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ The LT returns an error code to the request for setting, changing or resetting the eSign-PIN	[TR-03117], 4.4.3		SA	LT
8.1	Verification of the correct execution of the termination function	Use of a signature terminal (of the eCard-User) with the access right „Generate qualified electronic signature“ Termination of the eSign-PIN or the signature key with the PIN	[TR-03117], 4.4.4	CT		UT
9.1	Verification that the terminal software handles the termination function properly when detecting internal error or error communicated by LT	Signature terminal does not have the access right „Generate qualified electronic signature“ The LT returns an error code to the request for termination of the eSign-PIN or the signature key	[TR-03117], 4.4.4		SA	LT

Table 18: Verification requirements use of the signature application, execution in terminal software

4 Implementation Conformance Statement

The purpose of the Implementation Conformance Statement is the declaration of optional functionality of the product to be approved by the applicant. The declarations of the applicant are used for the determination of the set of test cases appropriated to the functionality of the product.

The Implementation Conformance Statement must be filled completely by the applicant. The information of the filled ICS must be documented in the test report.

The test result will only cover the function declared in this statement.

4.1 Supported profiles and functions

All test cases of a profile which is declared with “Yes” by the applicant, have to be performed completely. The test coverage can be limited by declarations in chapter 4.3 and 4.4.

4.1.1 Profiles for Reader

Profile	***Task***	***Applicant declaration (Yes / No)***
R_Tra	Reader supports transmission of card commands over the PC/SC interface to the eCard	
R_PACE	Reader supports execution of PACE protocol according to [TR-03110], 4.2	
R_TA	Reader supports execution of terminal authentication protocol according to [TR-03110], 4.4	
R_CA	Reader supports execution of chip authentication protocol (version 2) according to [TR-03110], 4.3.1.2	
R_eID	Reader supports access to eID application	
R_bio	Reader supports access to biometric data in the eCard	
R_Sig	Reader supports generation of qualified electronic signatures according to [TR-03117]	

Table 19: Profiles for reader

4.1.2 Functions for Reader

Profile	*Task*	*Applicant declaration (Yes / No)*
R_Chg_PIN	Password management function to change the PIN is supported by the reader.	
R_Chg_CAN	Password management function to change the CAN is supported by the reader.	
R_Chg_PIN_PUK	Password management function to change the PIN after using PUK is supported by the reader.	
R_PIN_MGT_AT	PIN management functions for authentication terminals is supported by the reader.	
R_PIN_MGT_uT	PIN management functions for unauthenticated terminals after PACE is supported by the reader.	

Table 20: Functions for reader

4.1.3 Profiles for Terminal Software

Profile	*Task*	*Applicant declaration (Yes / No)*
TS_PACE	Terminal software supports execution of PACE protocol according to [TR-03110], 4.2	
TS_TA	Terminal software supports execution of terminal authentication protocol according to [TR-03110], 4.4	
TS_CA	Terminal software supports execution of chip authentication protocol (version 2) according to [TR-03110], 4.3.1.2	
TS_eID	Terminal software supports access to eID application	
TS_bio	Terminal software supports access to biometric data in the eCard	
TS_Sig	Terminal software supports generation of qualified electronic signatures according to [TR-03117]	

Table 21: Profiles for terminal software

Profile	*Task*	*Applicant declaration (Yes / No)*
TS_Chg_PIN	Password management function to change the PIN is supported by the terminal software.	
TS_Chg_CAN	Password management function to change the CAN is supported by the terminal software.	
TS_Chg_PIN_PUK	Password management function to change the PIN after using PUK is supported by the terminal software.	
TS_PIN_MGT_AT	PIN management functions for authentication terminals is supported by the terminal software	
TS_PIN_MGT_uT	PIN management functions for unauthenticated terminals after PACE is supported by the terminal software	
TS_Write_eID	Writing Access to eID data with EAC is supported by the terminal software	

Table 22: Functions for terminal software

4.2 Cryptographic algorithms

The applicant of the DUT SHALL declare the supported algorithms used to perform the PACE and Chip- and Terminal-Authentication and eSign(QES) if applicable. The algorithm identifiers as defined in [TR-03110] have to be used (e.g. PACE-ECDH-GM-AES-CBC-CMAC-128, ...).

Protocol	*supported algorithms*
PACE	
TA	
CA	
eSign	

Table 23: Supported algorithms

4.3 Terminal type

The applicant of the DUT SHALL declare the supported terminal roles.

Terminal Type	*Applicant declaration (Yes / No)*
Inspection System (IS)	
Authentication Terminal (AT)	
Signature Terminal (ST)	

Table 24: Supported terminal roles

4.4 Passwords

4.4.1 Reader

The reader can accept different types of passwords for the PACE protocol. Some readers may accept the password via a PC/SC command, while others block passwords from the host. The allowed input channel may also be dependent on the terminal type.

		IS	*AT*	*ST*
allows input from host	MRZ			
	CAN			
	PIN			
	PUK			
allows input from pinpad	MRZ			
	CAN			
	PIN			
	PUK			

Table 25: Matrix for the supported passwords dependent on the terminal role

4.4.2 Terminal Software

For the profiles TS_PACE, TS_TA, TS_CA, TS_eID, t TS_Chg_PIN, TS_Chg_CAN, TS_Chg_PIN_PUK and TS_PIN_MGT_AT the tests MUST be applied using the supported passwords as stated in the following table.

Password	***IS***	***AT***	***ST***
MRZ			
CAN			
PIN			
PUK			

Table 26: Matrix for the passwords dependent on the terminal role supported by terminal software

However, for the other profiles the tests MUST be applied with all passwords as stated in the corresponding test cases.

5 Definition of Configuration Data for the Tests

5.1 Terminal Certificates

Terminal certificates used in the tests are built up as follows:

'7F 21'	var.					Certificate template (tag, length)
	'7F 4E'	'XX'				Certificate body (tag, length)
		'5F 29'	'01'		'XX'	Certificate profile identifier
		'42'	var.		'XX .. XX'	Certificate authority reference
		'7F 49'	var.		'XX .. XX'	Public key
		'5F 20'	var.		'XX .. XX'	Certificate holder reference
		'7F 4C'	var.			Certificate Holder Authorization Template (CHAT) (tag, length)
			'06'	'09'	'04 00 7F 00 07 03 01 02 RR'	Object identifier for role RR: RR = 01: inspection system RR = 02: authentication terminal RR = 03: signature terminal
			'53'	'LZ'	'XX .. XX'	Discretionary data (access rights) Value see Tables 28 to 33 = 01, if RR = 01 LZ = 05, if RR = 02 LZ = 01, if RR = 03
		'5F 25'	'06'		'XX .. XX'	Certificate effective date
		'5F 24'	'06'		'XX .. XX'	Certificate expiration date
		'65'	var.			Certificate extensions (Tag, Length)
			'73'	var.		Discretionary Data Template (Tag, Length)
				'06' '0A'	'04 00 7F 00 07 03 01 03 01 01'	Object identifier for certificate description (plain text format)
				'80' var.	'XX .. XX'	Hash value over certificate description

	'73'	var.		Discretionary data template (tag, length)
		'06' '09'	'04 00 7F 00 07 03 01 03 02'	Object identifier for terminal sector
		'80' var.	'XX .. XX'	Hash value over public key DO 1st sector
		'81' var.	'XX .. XX'	Hash value over public key DO 2nd sector
'5F 37'	var.		'XX .. XX'	Certificate signature

Table 27: Structure of a certificate

No.	***Value***	***Description***
1	'03'	*Universal rights:* Reading access to biometric data of ePassport
2	'00'	No access to eID- and ePassport functions

Table 28: Choice of access rights for inspection systems

No.	*Value*	*Description*
1	'3E 1F FF FF F7'	*Universal rights:* Write access to DG 17 – DG 21 Read access to DG 1 – DG 21 Right to install qualified certificates Right to install certificates Right to execute password management functions Right to use CAN Right to perform restricted identification Right to perform community ID verification Right to perform age identification
2	'30 00 00 00 02'	Write access to DG 17 and DG 18 Right to perform community ID verification
3	'3E 1F FF FF 17'	Write access to DG 17 – DG 21 Read access to DG 1 – DG 21 Right to use CAN Right to perform restricted identification Right to perform community ID verification Right to perform age identification
4	'00 01 13 FB 07'	Read access to DG 1, DG 2, DG 4 – DG 10, DG 13 and DG 17 Right to perform restricted identification Right to perform community ID verification Right to perform age identification

Table 29: Choice of access rights for authentication terminals

No.	*Value*	*Description*
1	'03'	*Universal rights*: Right to generate ES + QES
2	'00'	No signature generation

Table 30: Choice of access rights for signature terminals

No.	*Value*	*Description*
1	C3	CVCA
2	'83'	Document Verifier (official domestic)

Table 31: Choice of access rights for CA certificates (inspection systems)

No.	*Value*	*Description*
1	'FE 1F FF FF F7'	CVCA
2	'7E 1F FF FF F7'	Document Verifier (non official / foreign)

Table 32: Choice of access rights for CA certificates (authentication terminals)

No.	*Value*	*Description*
1	'C3'	CVCA
2	'43'	Document Verifier (certification service provider)

Table 33: Choice of access rights for CA certificates (signature terminals)

5.2 Extension of PC/SC Interface

According to [TR-03119], A.11 the call of the PC/SC function SCardControl from [PCSC10] is extended by GetReadersPACECapabilities and EstablishPACEChannel. InBuffer und OutBuffer of SCardControl are specified as follows:

5.2.1 InBuffer (for GetReadersPACECapabilities)

According to [TR-03119], A.11.1.1, A.11.2.1. the value for InBuffer in GetReadersPACECapabilities is:

```
01 00 00
```

5.2.2 OutBuffer (for GetReadersPACECapabilities)

According to [TR-03119], A.11.1.1, A.11.2.1. the value for OutBuffer in GetReadersPACECapabilities is:

```
<Result_Code> 00 02 01 <Bit_Map>
```

Result_Code: Result code according to 37

Bit_Map: Bit map according to 36

5.2.3 InBuffer (for EstablishPACEChannel)

According to [TR-03119], A.11.1.1, A.11.2.2. the value for InBuffer in EstablishPACEChannel is:

```
02 <L_inputData> <PIN-ID> <L_CHAT> <CHAT> <L_PIN> <PIN>
<L_CERT_DESC> <CERT_DESC>
```

PIN-ID: '01' (MRZ-Password), '02' (CAN), '03' (PIN) or '04' (PUK)

CHAT: Restricted CHAT for terminal certificate (coding see 35) or empty

PIN: if provided by host; e. g. CAN

CERT_DESC: complete description of certificate as described in [TR-03119] A.11.2.2 and [TR-03110] C.3.1 See 34 for an example.

T	***L***	***Value***			
30	82 01 02	T	L	V	Comment
		06	0A	04 00 7F 00 07 03 01 03 01 01	descriptionType
		A1	11	0C 0F 54 65 73 74 69 73 73 75 65 72 20 47 6D 62 48	issuerName [1]
		A2	1A	13 18 68 74 74 70 3A 2F 2F 77 77 77 2E 74 65 73 74 69 73 73 75 65 72 2E 64 65	issuerURL [2]
		A3	0F	0C 0D 54 65 73 74 68 61 75 73 20 47 6D 62 48	subjectName [3]
		A4	18	13 16 68 74 74 70 3A 2F 2F 77 77 77 2E 74 65 73 74 68 61 75 73 2E 64 65	subjectURL [4]
		A5	8199	0C 81 96 41 6E 73 63 68 72 69 66 74 3A 20 0D 0A 54 65 73 74 68 61 75 73 20 47 6D 62 48 0D 0A 51 75 61 6C 69 74 C3 A4 74 73 73 74 72 2E 20 31 0D 0A 33 33 31 30 30 20 50 61 64 65 72 62 6F 72 6E 0D 0A 0D 0A 45 2D 4D 61 69 6C 2D 41 64 72 65 73 73 65 3A 20 0D 0A 6E 70 61 40 74 65 73 74 68 61 75 73 2E 64 65 0D 0A 0D 0A 5A 77 65 63 6B 20 64 65 73 20 41 75 73 6C 65 73 65 76 6F 72 67 61 6E 67 73 3A 20 0D 0A 54 65 73 74 20 64 65 72 20 54 65 72 6D 69 6E 61 6C 73 2E	termsOfUsage [5]

Table 34: Example for CERT_DESC

Pos.	*Length (in Bytes)*	*Value*	*Description*
1	1	'06'	Tag for Object Identifier (Role)
2	1	'09'	Length for Object Identifier (Role)
3	9	'04 00 7F 00 07 03 01 02 RR'	Object identifier for role RR: RR = 01: inspection system RR = 02: authentication terminal RR = 03: signature terminal
4	1	'53'	Tag for discretionary data (access rights)
5	1	'XX'	Length for discretionary data (access rights): XX = 01, if RR = 01 XX = 05, if RR = 02 XX = 01, if RR = 03
6	var.	'XX .. XX'	Discretionary data (access rights) Value see Tables 28 to 30

Table 35: Structure of the CHAT data object

5.2.4 OutBuffer (for EstablishPACEChannel)

According to [TR-03119], A.11.1.1, A.11.2.2. the value for OutBuffer in EstablishPACEChannel is:

`<Result_Code> <L_outputData> <status_mse> <L_dca> <data_card_acc> <L_CAR1> <CAR1> <L_CAR2> <CAR2> <L_IDPICC> <IDPICC>`

Result_Code: Result code according to 37

status_mse: Status bytes in response to MSE: Set AT

data_card_acc: Data for card access

CAR1: Current certificate authority reference (CAR)

CAR2: Previous certificate authority reference (CAR)

IDPICC: ID_PICC, necessary for TA

Remark: If the reader uses a certificate with role signature terminal, the data objects `<CAR1>` , `<CAR2>` and `<IDPICC>` are omitted according to [TR-03119], A.11, since the secure channel between eCard and reader will be established automatically.

b7	b6	b5	b4	b3	b2	b1	b0	
0								RFU
	1							PACE supported
		1						eID-function supported
			1					Signature function supported
				0	0	0	0	RFU

Table 36: Bitmap for functions supported by the reader

Result code	*Description*
'00 00 00 00'	No error
'D0 00 00 01'	Inconsistent lengths in input buffer
'D0 00 00 02'	Unexpected data in input buffer
'D0 00 00 03'	Unexpected combination of data in input buffer
'E0 00 00 01'	Card does not support PACE (unexpected structure in response message of eCard)
'E0 00 00 02'	Reader does not support required resp. calculated algorithm
'E0 00 00 03'	Reader does not know password-ID
'0xE0000006'	incorrect for token PACE
'0xE0000007'	certificate chain for Terminal Authentication can't be build
'0xE0000008'	unexpected data structure in response to Chip Authentication
'0xE0000009'	Passive Authentication failed
'0xE000000A'	incorrect token for Chip Authentication
card returns error code for PACE	
'F0 00' | SW1SW2	Negative response of eCard to Select command to access card data
'F0 01' | SW1SW2	Negative response of eCard to Read Binary
'F0 02' | SW1SW2	Negative response of eCard to MSE: Set AT
'F0 03' | SW1SW2 'F0 04' | SW1SW2 'F0 05' | SW1SW2 'F0 06' | SW1SW2	Negative response of eCard to General Authenticate: Step 1 Step 2 Step 3 Step 4
card returns error code for TA/CA	
0xF800SW1SW2	MSE: Set DST (first certificate)
0xF801SW1SW2	PSO: Verify Certificate (first certificate)
0xF802SW1SW2	MSE: Set DST (second certificate)
0xF803SW1SW2	PSO: Verify Certificate (second certificate)

0xF804SW1SW2	MSE: Set DST (third certificate)
0xF805SW1SW2	PSO: Verify Certificate (third certificate)
0xF806SW1SW2	MSE: Set AT for Terminal Authentication
0xF807SW1SW2	Get Challenge
0xF808SW1SW2	External Authenticate
0xF809SW1SW2	Select EF.CardSecurity
0xF80ASW1SW2	Read Binary EF.CardSecurity
0xF80BSW1SW2	MSE: Set AT for Chipauthentisierung
0xF80CSW1SW2	General Authenticate
other errors	
'F0 10 00 01'	Aborting communication with eCard
'F0 10 00 02'	No eCard in the field
'F0 20 00 01'	User abort
'F0 20 00 02'	User timeout

Table 37: Result Codes

5.3 Communication Steps at the Card Interface

The following protocol descriptions have to be executed at the card interface. Some of them can be called directly from the UT while other must not be called from the UT directly but from the reader. If not described in an other way, all passwords have to be entered directly on the readers pinpad (5.3.1 PACE and 5.3.9 PIN Management).

5.3.1 PACE

The PACE protocol can be executed without or with Secure Messaging. If executed with Secure Messaging the SM keys have been derived by a former PACE protocol. If not explicitly mentioned otherwise, the PACE protocol is performed without SM in the test cases.

In the following either all command APDUs and response APDUs are executed without SM (PACE without SM) or with SM (PACE with SM).

Step 1: DUT gets content of EF.CardAccess.

Step 2: LT receives command MSE: Set AT (mutual authentication in PACE)

- Without SM:
  `00 22 C1 A4 <Lc> 80 <L_80> <OID> 83 <L_83> <PIN-ID> 7F 4C <L_7F4C> 06 <L_06> <OID-Role> 53 <L_53> <access rights>`
  OID: PACE OID
  PIN-ID: 01: MRZ-Password
  02: CAN
  03: PIN
  04: PUK
  OID-Role: `04 00 7F 00 07 03 01 02 01`: IS
  `02`: AT
  `03`: ST
  access rights: see Tables 28 to 30
  The data object with tag 7F4C is missing if no terminal authentication is performed after PACE.
- With SM:
  `0C 22 C1 A4 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`
  PI: Padding Indicator '01'
  cryptogram: cryptogram over command data of unsecured command MSE: Set AT

Step 3: LT sends response to MSE: Set AT

- Without SM:
  `<SW1SW2>`
- With SM:
  `99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

Step 4: LT receives command General Authenticate (Step 1)

- Without SM:
  `10 86 00 00 <Le>`
- With SM:
  `1C 86 00 00 <Lc> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>`
  Le_data: Length of data in response APDU

Step 5: LT derives encryption key K_pi from eCard password, generates nonce s and computes encrypted nonce encNonce = E(K_pi, s).

Step 6: LT sends response to General Authenticate (Step 1)

- Without SM:
  `7C <L_7C> 80 <L_80> <encNonce> <SW1SW2>`
- With SM:
  `87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`
  cryptogram: cryptogram of response data to unsecured command General Authenticate (Step 1)

Step 7: LT receives command General Authenticate (Step 2)

- Without SM:
  `10 86 00 00 <Lc> 7C <L_7C> 81 <L_81> <mapping_data> <Le>`
  mapping_data: mapping data of the terminal (specific to crypto algorithm, see [TR-03110], B.1.2)
- With SM:
  `1C 86 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>`
  PI: Padding Indicator '01'
  cryptogram: cryptogram over command data of unsecured command General Authenticate (Step 2)
  Le_data: Length of data in response APDU

Step 8: LT sends response to General Authenticate (Step 2)

- Without SM:
  `7C <L_7C> 82 <L_82> <mapping_data> <SW1SW2>`
  mapping_data: mapping data of the card D_PICC (specific to crypto algorithm, see [TR-03110], B.1.2)
- With SM:
  `87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`
  cryptogram: cryptogram of response data to unsecured command General Authenticate (Step 2)

Step 9: LT receives command General Authenticate (Step 3)

- Without SM:
  `10 86 00 00 <Lc> 7C <L_7C> 83 <L_83> <ephem_pk> <Le>`
  ephem_pk: ephemeral public key of the terminal (PKeph_PCD)
- With SM:
  `1C 86 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'
cryptogram: cryptogram over command data of unsecured command General Authenticate (Step 3)
Le_data: Length of data in response APDU

Step 10: LT generates data Deph = Map(D_PICC, s) and, using Deph, the ephemeral key pair (SKeph_PICC, PKeph_PICC). LT checks that PKeph_ICC is different from PKeph_PCD.

Step 11: LT sends response to General Authenticate (Step 3)

- Without SM:
  `7C <L_7C> 84 <L_84> <ephem_pk> <SW1SW2>`
  ephem_pk: ephemeral public key of the eCard (PKeph_PICC)
- With SM:
  `87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`
  cryptogram: cryptogram of response data to unsecured command General Authenticate (Step 3)

Step 12: LT receives command General Authenticate (Step 4)

- Without SM:
  `00 86 00 00 <Lc> 7C <L_7C> 85 <L_85> <auth_token> <Le>`
  auth_token: authentication token of the terminal (T_PCD)
- With SM:
  `0C 86 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>`
  PI: Padding Indicator '01'
  cryptogram: cryptogram over command data of unsecured command General Authenticate (Step 3)
  Le_data: Length of data in response APDU

Step 13: LT computes key material KA(SKeph_PICC, PKeph_PCD, Deph), extracts Kmac from key material and checks that T_PCD is identical to MAC(Kmac, PKeph_PICC). LT computes authentication token T_PICC = MAC(Kmac, PKeph_PCD).

Step 14: LT sends response to General Authenticate (Step 4)

- Without SM:
  `7C <L_7C> 86 <L_86> <auth_token> 87 <L_87> <CAR1> 88 <L_88> <CAR2> <SW1SW2>`

auth_token: authentication token of the eCard (T_PICC)
CAR1: CAR (exists only if DO with Tag '7F 4C' is provided in command MSE:Set AT, s. step 3)
CAR2: previous CAR (optional if DO with Tag '7F 4C' is provided in command MSE:Set AT, s. step 3)

- With SM:
  `87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`
  cryptogram: cryptogram of response data to unsecured command General Authenticate (Step 4)

5.3.2 Terminal Authentication

Step 1: UT sends command MSE: Set DST for certificate verification

`00 22 81 B6 <Lc> 83 <L_83> <pk_ref>`

pk_ref: reference to public key (CVCA or terminal certificate)

Step 2: LT receives command MSE: Set DST for certificate verification

`0C 22 81 B6 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command MSE: Set DST

Step 3: LT sends response to MSE: Set DST

`99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

Step 4: UT receives response to MSE: Set DST

`<SW1SW2>`

Step 5: UT sends command PSO: Verify Certificate

`00 2A 00 BE <Lc> 7F 4E <L_7F4E> <body> 5F 37 <L_5F37> <sig>`

body: body of certificate to be verified

sig: signature of certificate to be verified

Step 6: LT receives command PSO: Verify Certificate

`0C 2A 00 BE <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command PSO: Verify Certificate

Step 7: LT sends response to PSO: Verify Certificate

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 8: UT receives response to PSO: Verify Certificate

<SW1SW2>

The steps 1-8 are performed for all certificates in the certificate chain, i. e. CA certificates and terminal certificate.

Step 9: UT generates new ephemeral key pair (SKeph_PCD, PKeph_PCD) and computes Comp(PKeph_PCD). Moreover UT provides auxiliary data for key exchange A_PCD.

Step 10: UT sends command MSE: Set AT for external authentication

00 22 81 A4 <Lc> 80 <L_80> <OID> 83 <L_83> <pk_ref> 67 <L_67> <aux_data> 91 <L_91> <ephem_pk>

OID: OID for TA

pk_ref: reference to public terminal key

aux_data: auxiliary authenticated data A_PCD

pk_ephem: compressed ephemeral public key Comp(PKeph_PCD)

Step 11: LT receives command MSE: Set AT for external authentication

0C 22 81 A4 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command MSE: Set AT

Step 12: LT sends response to MSE: Set AT

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 13: UT receives response to MSE: Set AT

<SW1SW2>

Step 14: UT sends command Get Challenge

00 84 00 00 08

Step 15: LT receives command Get Challenge

`0C 84 00 00 <Lc> 97 <L_97> <Le_chall> 8E <L_8E> <MAC> <Le>`

Le_chall: Length of challenge

Step 16: LT sends response to Get Challenge

`87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

cryptogram: cryptogram of response data to unsecured command Get Challenge

Step 17: UT receives response to Get Challenge

`<challenge> <SW1SW2>`

challenge: Challenge r1_PICC from eCard

Step 18: UT computes s_PCD = Sign(SKeph_PCD, ID_PICC | r1_PICC | Comp(PKeph_PCD) | A_PCD). (Hint: ID_PICC has been provided to UT at the end of the PACE protocol).

Step 19: UT sends command External Authenticate

`00 82 00 00 <Lc> <signature>`

signature: Signature s_PCD

Step 20: LT receives command External Authenticate

`0C 82 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command External Authenticate

Step 21: LT sends response to External Authenticate

`99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

Step 22: UT receives response to External Authenticate

`<SW1SW2>`

5.3.3 Chip Authentication

Before performing Chip Authentication the EF.CardSecurity shall be read and Passive Authentication with the Security Object shall be performed.

Step 1: UT sends command MSE: Set AT for internal authentication

```
00 22 41 A4 <Lc> 80 <L_80> <OID>
```

OID: OID for CA

Step 2: LT receives command MSE: Set AT for internal authentication

```
0C 22 41 A4 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>
```

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command MSE: Set AT

Step 3: LT sends response to MSE: Set AT

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 4: UT receives response to MSE: Set AT

```
<SW1SW2>
```

Step 5: UT sends command General Authenticate

```
00 86 00 00 <Lc> 7C <L_7C> 80 <L_80> <ephem_pk> <Le>
```

ephem_pk: ephemeral public key of the terminal (PKeph_PCD)

Step 6: LT receives command General Authenticate

```
0C 86 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 97 <L_97> <Le_data>
8E <L_8E> <MAC> <Le>
```

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command General Authenticate

Le_data: Length of data in response APDU

Step 7: LT generates random number r2_PICC and computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), K_MAC = KDF_MAC(K, r2_PICC), K_ENC = KDF_ENC(K, r2_PICC) and T_PICC = MAC(K_MAC, PKeph_PCD).

Step 8: LT sends response to General Authenticate

```
87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

cryptogram: cryptogram of response data to unsecured command General Authenticate

Step 9: UT receives response to General Authenticate

```
7C <L_7C> 81 <L_81> <nonce> 82 <L_82> <auth_token> <SW1SW2>
```

nonce: Challenge r2_PICC from eCard

auth_token: authentication token T_PICC

Step 10: UT computes K = KA(SKeph_PCD, PK_PICC, D_PICC), K_MAC = KDF_MAC(K, r2_PICC), K_ENC = KDF_ENC(K, r2_PICC) and checks that T_PICC is identical to MAC(K_MAC, PKeph_PCD). K_MAC and K_ENC are the session keys for secure messaging when generating digital signatures or for access to the eID application that ask for authentication with PACE, TA and CA.

5.3.4 Select the eSign Application

Step 1: UT sends command Select

`00 A4 04 0C <Lc> <AID>`

AID: 'A0 00 00 01 67 45 53 49 47 4E' AID of the eSign application

Step 2: LT receives command Select

`0C A4 04 0C <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Select

Step 3: LT sends response to Select

`99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

Step 4: UT receives response to Select

`<SW1SW2>`

5.3.5 Reading Data from the eID Application

Step 1: UT sends command Select

`0C A4 04 0C <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>`

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Select, consisting of the AID of eID application: 'E8 07 04 00 7F 00 07 03 02'

Step 2: LT receives this command Select

Step 3: LT sends response to Select

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 4: UT receives this response to Select

Step 5: UT sends command Read Binary

0C B0 <P1> 00 <Lc> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>

P1:

'81'	(SFI '01' for DG1)
'82'	(SFI '02' for DG2)
'83'	(SFI '03' for DG3)
...	...
'89'	(SFI '09' for DG9)
'8A'	(SFI '0A' for DG10)
...	...
'8F'	(SFI '0F' for DG15)
'90'	(SFI '10' for DG16)
...	...
'95'	(SFI '15' for DG21)

Le_data: Length of data in response APDU

Step 6: LT receives this command Read Binary

Step 7: LT sends response to Read Binary

87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

cryptogram: cryptogram of response data to unsecured command Read Binary

Step 8: UT receives this response to Read Binary

5.3.6 Writing Data into the eID Application

Step 1: UT sends command Select

0C A4 04 0C <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Select, consisting of the AID of eID application: 'E8 07 04 00 7F 00 07 03 02'

Step 2: LT receives this command Select

Step 3: LT sends response to Select

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 4: UT receives this response to Select

Step 5: UT sends command Update Binary

```
0C D6 <P1> 00 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC>
<Le>
```

P1:

'81'	(SFI '01' for DG1)
'82'	(SFI '02' for DG2)
'83'	(SFI '03' for DG3)
...	...
'89'	(SFI '09' for DG9)
'8A'	(SFI '0A' for DG10)
...	...
'8F'	(SFI '0F' for DG15)
'90'	(SFI '10' for DG16)
...	...
'95'	(SFI '15' for DG21)

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Update Binary, consisting of a byte list that has to be specified in the test cases

Step 6: LT receives this command Update Binary

Step 7: LT sends response to Update Binary

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 8: UT receives this response to Update Binary

5.3.7 Restricted Identification

Step 1: UT sends command MSE: Set AT for internal authentication

```
0C 22 41 A4 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>
```

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command MSE: Set AT

The command data of the unsecured command MSE: Set AT are built up as follows:

```
80 <L_80> <OID> 84 <L_84> <sk_id_ref>
```

OID: OID for RI

sk_id_ref: Reference to private key SK_ID in the eCard, used for RI

Step 2: LT receives the secured command MSE: Set AT

Step 3: LT sends response to MSE: Set AT

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 4: UT receives this response to MSE: Set AT

Step 5: UT sends command General Authenticate

```
0C 86 00 00 <Lc> 87 <L_87> <PI> <cryptogram> 97 <L_97> <Le_data>
8E <L_8E> <MAC> <Le>
```

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command General Authenticate

Le_data: Length of data in response APDU

The command data of the unsecured command General Authenticate are built up as follows:

```
7C <L_7C> A0 <L_A0> <pk_sec> <Le>
```

pk_sec: public key of the sector (PK_Sector)

Step 6: LT receives the secured command General Authenticate

Step 7: LT generates its sector specific identifier I_sector_ID using SK_ID and PK_Sector.

Step 8: LT sends response to General Authenticate

```
87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

cryptogram: cryptogram of response data to unsecured command General Authenticate

The response data of the unsecured command General Authenticate are built up as follows:

```
7C <L_7C> 81 <L_81> <sec_id>
```

sec_id: sector specific identifier I_sector_ID

Step 9: UT receives the secured response to command General Authenticate

5.3.8 Auxiliary Data Verification

Step 1: UT sends command Verify

```
8C 20 80 00 <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>
```

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Verify

The command data of the unsecured command Verify are built up as follows:

```
<OID>
```

OID: OID of the auxiliary data to be verified (age verification, document validity verification or community ID verification)

Step 2: LT receives secured command Verify

Step 3: LT sends response to Verify

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 4: UT receives this response to Verify

5.3.9 PIN Management

5.3.9.1 Changing password

Step 1: UT sends command Reset Retry Counter

```
0C 2C 02 <P2> <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC>
<Le>
```

P2: '02' (for changing CAN), '03' (for changing PIN)

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Reset Retry Counter

The command data of the unsecured command Reset Retry Counter are built up as follows:

```
<PIN>
```

PIN: the new password

Step 2: LT receives secured command Reset Retry Counter

Step 3: LT sends response to Reset Retry Counter

```
99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>
```

Step 4: UT receives this response to Reset Retry Counter

5.3.9.2 Unblocking password

Step 1: UT sends command Reset Retry Counter

0C 2C 03 03 <Lc> 8E <L_8E> <MAC> <Le>

Step 2: LT receives secured command Reset Retry Counter

Step 3: LT sends response to Reset Retry Counter

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 4: UT receives this response to Reset Retry Counter

5.3.9.3 Activating / Deactivating password

Step 1: UT sends command Activate / Deactivate

0C <INS> 10 03 <Lc> 8E <L_8E> <MAC> <Le>

INS: '44' (for Activate), '04' (for Deactivate)

Step 2: LT receives secured command Activate / Deactivate

Step 3: LT sends response to Activate / Deactivate

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 4: UT receives this response to Activate / Deactivate

5.3.10 Reading Data from the ePassport Application

Step 1: UT sends command Select

0C A4 04 0C <Lc> 87 <L_87> <PI> <cryptogram> 8E <L_8E> <MAC> <Le>

PI: Padding Indicator '01'

cryptogram: cryptogram over command data of unsecured command Select, consisting of the AID of ePassport application

Step 2: LT receives this command Select

Step 3: LT sends response to Select

99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>

Step 4: UT receives this response to Select

Step 5: UT sends command Read Binary

`0C B0 <P1> 00 <Lc> 97 <L_97> <Le_data> 8E <L_8E> <MAC> <Le>`

P1: `80 + <SFI__DF_to_be_read>`

Le_data: Length of data in response APDU

Step 6: LT receives this command Read Binary

Step 7: LT sends response to Read Binary

`87 <L_87> <cryptogram> 99 <L_99> <SW1SW2> 8E <L_8E> <MAC> <SW1SW2>`

cryptogram: cryptogram of response data to unsecured command Read Binary

Step 8: UT receives this response to Read Binary

6 Test Specification

6.1 General Definitions

The test cases are derived from verification requirements. They refine these requirements by defining a test goal, all conditions, test steps and verifications that are necessary for the implementation and execution of a test.

The structure of this chapter follows the structure of the verification requirements whereas for each requirement number (RQ_no) an additional structuring level is defined that enumerates the test cases which are assigned to the verification requirement. Each test case is described by a table. The structure of such a table is shown in Table 38.

Name in table column	*Contents*
Test ID	Identifier for the test case in the form <code test object>_<code function>_<No. VR>.<No. para>.<No. test case> <No. test case> (distinguishes test cases within a verification requirement): 1, 2, 3 ... The meaning of <code test object>, <code function>, <No. VR> and <No. para> is defined in 5.
Purpose	What shall be tested, which aspect of the verification requirement?
Reference	Reference to the specification requirement
Profiles	Set of profiles required for test execution. The profiles allowed are declared in the ICS (see Table 22). Profiles may be logical connected by AND, OR and NOT.
Preconditions	Roles and access rights of the terminal certificates needed for test execution and other preparatory steps, if necessary
Test scenario	List of all necessary steps to reach the Purpose
Expected results	Verification steps and expected results
Post processing	If necessary, steps after test execution to ensure execution of further test cases

Table 38: Structure of a test case description

For all test cases where no terminal role and password type is defined, these parameters can be chosen from these which are supported by the reader (see chapter 4.3 Terminal type).

If no terminal type and/or password is defined, the priority of the terminal type to use in the test cases are. AT, IS and ST. The priority of the password to use in the in the test cases are CAN, PIN and MRZ. That does mean, first select the first supported terminal type then select the first supported password type which is supported in combination with the terminal type.
The used terminal type and password must be documented in the test report.

6.2 Test Cases for Test Object Reader

6.2.1 Transparent Mode

6.2.1.1 R_Tra_1 – Correct Reading of eCard Data in Transparent Mode

Test ID	R_Tra_1.1.1
Purpose	Check reader for correct reading access to eCard data which are freely accessible.
References	[TR-03110]
Profiles	R_Tra
Preconditions	-
Test scenario	1. UT sends command Read Binary SFI '1C' (EF.CardAccess) to reader: `00 B0 9C 00 <Le>` (Le: Number of Bytes in answer message) 2. Reader transmits command received from UT to LT. 3. LT sends response to Read Binary to reader: `<Data> 90 00` (Data: Data from EF.CardAccess) 4. Reader transmits response received from LT to UT.
Expected results	1. - 2. The command APDU received at LT is correctly coded and coincides with the command APDU sent by UT. 3. - 4. The response APDU received at UT is correctly coded and coincides with the response APDU sent by LT.
Post processing	Reset of eCard and reader

Test ID	R_Tra_1.2.1
Purpose	Check for an abort when trying to read eCard data, which are not freely accessible, without authentication.
References	[TR-03110]

Profiles	R_Tra
Preconditions	-
Test scenario	1. UT sends command Read Binary SFI '1D' (EF.CardSecurity) to reader: `00 B0 9D 00 <Le>` (Le: Number of Bytes in answer message) 2. Reader transmits command received from UT to LT. 3. LT sends response to Read Binary to reader: `<SW1SW2>` (SW1SW2: negative Return Code) 4. Reader transmits response received from LT to UT.
Expected results	1. - 2. The command APDU received at LT is correctly coded and coincides with the command APDU sent by UT. 3. - 4. The response APDU received at UT is correctly coded and coincides with the response APDU sent by LT.
Post processing	Reset of eCard and reader

6.2.2 PACE

6.2.2.1 R_PACE_1 – Correct Execution of PACE Protocol

Test ID	R_PACE_1.1.1_template
Purpose	Check correct execution of PACE protocol in the reader. Use certificate with role inspection system. The test is executed with the passwords CAN and MRZ. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates and the password (CAN resp. MRZ-Password) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • `<PIN-ID>`: as defined in table 39

	• <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 39. • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (inspection system). • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1. • All command APDUs received by the LT are correctly coded and secured by SM from the test object reader. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Testcase ID	Passwort	<PIN-ID>
R_PACE_1.1.1a	Use PACE with MRZ	'01'
R_PACE_1.1.1b	Use PACE with CAN	'02'

Table 39: Test case R_PACE_1.1.1

Test ID	R_PACE_1.1.2_template
Purpose	Check correct execution of PACE protocol in the reader. Use certificate with role authentication terminal. The test is executed with the passwords PIN and CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 4.2, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password (PIN and CAN) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: as defined in table 40 • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 40 • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. • All command APDUs received by the LT are correctly coded and secured by SM from the test object reader. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds:

	• <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Testcase ID	Passwort	<PIN-ID>
R_PACE_1.1.2a	Use PACE with CAN	'02'
R_PACE_1.1.2b	Use PACE with PIN	'03'

Table 40: Test case R_PACE_1.1.2

Test ID	R_PACE_1.1.3_template
Purpose	Check correct execution of PACE protocol in the reader. Use certificate with role signature terminal. The test is executed with the passwords CAN and PIN. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader. All supported possibilities to provide the password have to be tested.
References	[TR-03110], 4.2, C.4.3; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates and the password (CAN and. PIN) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: as defined in table 41 • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, password is entered via PINPad of reader

	• <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 41 • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. • All command APDUs received by the LT are correctly coded and secured by SM from the test object reader. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Testcase ID	Passwort	<PIN-ID>
R_PACE_1.1.3a	Use PACE with CAN	'02'
R_PACE_1.1.3b	Use PACE with PIN	'03'

Table 41: Test case R_PACE_1.1.3

Test ID	R_PACE_1.2.1_template
Purpose	Check correct execution of PACE protocol in the reader. Use an unauthenticated terminal. The test is executed with the passwords CAN, PIN and PUK. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 3.5.1, 4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	-
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>:as defined in table 42 • <CHAT>: empty • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: empty 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 42 • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty). • All command APDUs received by the LT are correctly coded and secured by SM from the test object reader. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT.

	• <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Testcase ID	Passwort	<PIN-ID>
R_PACE_1,2,1a	Use PACE with CAN	'02'
R_PACE_1.2.1b	Use PACE with PIN	'03'
R_PACE_1.2.1c	Use PACE with PUK	'04'

Table 42: Test case R_PACE_1.2.1

Test ID	R_PACE_1.3.1
Purpose	Check correct execution of PACE protocol in the reader. Use of several algorithms for the PACE protocol. The values for PACEInfo and PACEDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. The test has to be executed for each PACE algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password CAN available in UT Make valid values for PACEInfo and PACEDomainParameterInfo within EF.CardAccess available in LT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used)

	• <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Test ID	R_PACE_1.3.2
Purpose	Check correct execution of PACE protocol in the reader, if LT supports several algorithms for PACE. Use three different PACEInfo in EF.CardAccess with different algorithms and standardized domain parameters. Use algorithms and domain parameters which are supported by the reader (chapter 4.2 Cryptographic algorithms). Don't use a PACEDomainParameterInfo within EF.CardAccess. The test is executed with the password CAN. It calls an extension of the

	PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password CAN available in UT Make valid values for PACEInfo and PACEDomainParameterInfo within EF.CardAccess available in LT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access

	coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

6.2.2.2 R_PACE_2 – Abort of PACE Protocol because of Internal LT Error

Test ID	R_PACE_2.1.1_template
Purpose	Check that reader aborts PACE protocol when LT derives cryptographic key from CAN-Password incorrectly. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 43. Make certificates and the CAN available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>:'02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT derives an *incorrect encryption key* K_pi from the password stored in the eCard. The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as

	described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN is used). • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 43. • The <access rights> in command APDU to MSE: Set AT as described in table 43. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, 'F0 06 63 XX', if LT sends return code '63 XX' in response to General Authenticate (Step 4) • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_2.1.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_2.1.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 43: Test case R_PACE_2.1.1

Test ID	R_PACE_2.1.2
Purpose	Check that reader aborts PACE protocol when LT derives cryptographic key from PIN incorrectly. The test is executed with the passwords PIN. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 4.2, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password (PIN) available in UT

Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT derives an *incorrect encryption key* K_pi from the password stored in the eCard. The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03'. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' • The <access rights> in command APDU to MSE: Set AT as defined in 29, No. 1. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 06 63 XX', if LT sends return code '63 XX' in response to General Authenticate (Step 4), or 'F0 10 00 01', if reader aborts communication with LT. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Test ID	R_PACE_2.2.1_template
Purpose	Check that reader aborts PACE protocol when LT returns error code to command MSE: Set AT. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the

	InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 44. Make certificates and the password (CAN) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT sends the *negative* return code '6A 88' in answer message to MSE: Set AT back to the reader. It is expected that the reader aborts protocol execution after receiving this return code. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 44. • The <access rights> in command APDU to MSE: Set AT as described in table 44. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the reader must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 02 6A 88'. • <status_mse>: The status code for command MSE: Set AT is '6A 88'.

Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_2.2.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_2.2.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 44: Test case R_PACE_2.2.1

Test ID	R_PACE_2.3.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits incorrect data for mapping function Map in answer to card command General Authenticate (Step 2). The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, B.1, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 45. Make certificates and the CAN available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT receives card command General Authenticate (Step 2) with Mapping Data of the terminal from the reader via CLI. LT sends *incorrect* Mapping data of the eCard (D_PICC) in the response APDU to card command General Authenticate (Step 2) back to the reader. The return codes in the response

	APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 45. • The <access rights> in command APDU to MSE: Set AT as described in table 45. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 06 63 XX', if LT sends return code '63 XX' in response to General Authenticate (Step 4), or 'F0 10 00 01', if reader aborts communication with LT. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_2.3.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_2.3.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 45: Test case R_PACE_2.3.1

Test ID	R_PACE_2.4.1 deleted in version 1.1

Test ID	R_PACE_2.5.1 deleted in version 1.1

Test ID	R_PACE_2.6.1 deleted in version 1.1

6.2.2.3 R_PACE_3 – Abort of PACE Protocol because of Incorrect LT Data

Test ID	R_PACE_3.1.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits incorrect PACE parameters (inconsistent data in these parameters). The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 46. Make certificates and the CAN available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT sends *inconsistent data* as data from EF.CardAccess in response APDU to command Read Binary with the following change: In SecurityInfo *PACEInfo* change length byte of tag "version" from '01' to '02': 30 0F 06 0A 04 00 7F 00 07 02 02 04 02 02 02 **02** 01. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The command APDUs for MSE: Set AT, General Authenticate (Step 1, 2, 3, 4) are missing, since the reader must abort communication to LT after receiving the inconsistent data in response APDU to Read Binary.

	3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'E0 00 00 01'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_3.1.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_3.1.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 46: Test case R_PACE_3.1.1

Test ID	R_PACE_3.1.2_template
Purpose	Check that reader aborts PACE protocol when LT transmits incorrect PACE parameters (algorithm ID contained in these parameters which is not supported by the reader). The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 47. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends

	response APDUs back to the reader as described in chapter 5.3.1. LT sends data from EF.CardAccess with *an algorithm-ID in the PACE parameters, which is not supported by the reader,* in response APDU to command Read Binary. It is expected that the reader aborts protocol execution after receiving these data. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The command APDUs for MSE: Set AT, General Authenticate (Step 1, 2, 3, 4) are missing, since the reader must abort communication to LT after receiving the response APDU to Read Binary. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • `<Result_Code>`: The result code is negative, i. e. 'E0 00 00 02'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	**`<OID-Role>`**
R_PACE_3.1.2a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	`'01'`
R_PACE_3.1.2b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	`'02'`

Table 47: Test case R_PACE_3.1.2

Test ID	R_PACE_3.2.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits an incorrect cryptogram in response message to card command General Authenticate (Step 1). The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, B.1, C.4.1; [TR-03119], D.3
Profiles	R_PACE

Preconditions	Certificates as described in table 48. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT computes the encrypted nonce encNonce *incorrectly,* i. e. *different from* E(K_pi, s).The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 48. • The <access rights> in command APDU to MSE: Set AT as described in table 48. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 06 63 XX', if LT sends return code '63 XX' in response to General Authenticate (Step 4), or 'F0 10 00 01', if reader aborts communication with LT. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_3.2.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_3.2.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 48: Test case R_PACE_3.2.1

Test ID	R_PACE_3.3.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits an ephemeral public key in response message to card command General Authenticate (Step 3) that coincides with the ephemeral public key transmitted by the reader in the command APDU to this command. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, B.1, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificates as described in table 49. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT generates the ephemeral key pair correctly but transmits in response message to card command General Authenticate (Step 3) the *ephemeral public key received from the reader*. It is expected that the reader aborts protocol execution after receiving the ephemeral public key in response APDU to General Authenticate (Step 3). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).

Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 49. • The <access rights> in command APDU to MSE: Set AT as described in table 49. • The command APDU for General Authenticate (Step 4) must be missing. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 10 00 01', since reader aborts communication with LT. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_3.3.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_3.3.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 49: Test case R_PACE_3.3.1

Test ID	R_PACE_3.4.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits an incorrect cryptogram, that it has generated with derived SM key data, in response message to card command General Authenticate (Step 4). The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3

Profiles	R_PACE
Preconditions	Certificates as described in table 50. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT computes authentication token T_PICC *incorrectly*, i. e. *different from* MAC(Kmac, PKeph_PCD). The return codes in the response APDUs of all commands are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 50. • The <access rights> in command APDU to MSE: Set AT as described in table 50. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative ('0xE0 00 00 06'). • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_3.4.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_3.4.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 50: Test case R_PACE_3.4.1

Test ID	R_PACE_3.5.1_template
Purpose	Check that reader aborts PACE protocol when LT transmits an authentication token (T_PICC), that it has generated with a wrong algorithm, in response message to card command General Authenticate (Step 4). The values for PACEInfo and PACEDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. But for the generation of T_PICC LT uses another algorithm. The test has to be executed for each algorithm specified in the manufacturer's conformance statement. The following substitutions have to be used (algorithm indicated in EF.CardAccess, algorithm used by LT): - id-PACE-DH-GM 1, id-PACE-DH-GM 2 - id-PACE-DH-GM 2, id-PACE-DH-GM 1 - id-PACE-DH-GM 3, id-PACE-DH-GM 1 - id-PACE-DH-GM 4, id-PACE-DH-GM 1 - id-PACE-ECDH-GM 1, id-PACE-ECDH-GM 2 - id-PACE-ECDH-GM 2, id-PACE-ECDH-GM 1 - id-PACE-ECDH-GM 3, id-PACE-ECDH-GM 1 - id-PACE-ECDH-GM 4, id-PACE-ECDH-GM 1 - id-PACE-DH-IM 1, id-PACE-DH-IM 2 - id-PACE-DH-IM 2, id-PACE-DH-IM 1 - id-PACE-DH-IM 3, id-PACE-DH-IM 1 - id-PACE-DH-IM 4, id-PACE-DH-IM 1 - id-PACE-ECDH-IM 1, id-PACE-ECDH-IM 2 - id-PACE-ECDH-IM 2, id-PACE-ECDH-IM 1 - id-PACE-ECDH-IM 3, id-PACE-ECDH-IM 1 - id-PACE-ECDH-IM 4, id-PACE-ECDH-IM 1

	The test is executed with the CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	R_PACE
Preconditions	Certificates as described in table 51. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT computes authentication token T_PICC *incorrectly*, *with a wrong algorithm*. The return codes in the response APDUs of all commands are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 51. • The <access rights> in command APDU to MSE: Set AT as described in table 51. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative ('0xE0 00 00 06'). • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_3.5.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_3.5.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 51: Test case R_PACE_3.5.1

6.2.2.4 R_PACE_4 – Abort of PACE Protocol because of Incorrect UT Data

Test ID	R_PACE_4.1.1_template
Purpose	Check that reader aborts PACE protocol when UT transmits an incorrect password-ID in the input of the PC/SC function. The test is performed subsequently with the incorrect password-IDs '00', '05' and '11'.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: see table 52 • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty (all executions) • <CERT_DESC>: List of certificates as specified in Profiles (all executions) 2. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, '0xE00000003' or '0xD00000002' or '0xD00000003'.
Post processing	Reset of eCard and reader

Testcase ID	<PIN-ID>
R_PACE_4.1.1a	'00'
R_PACE_4.1.1b	'05'
R_PACE_4.1.1c	'11'

Table 52: Test case R_PACE_4.1.1

Test ID	R_PACE_4.1.2_template
Purpose	Check that reader aborts PACE protocol when UT transmits an incorrect password-ID in the input of the PC/SC function. The test is performed subsequently with the incorrect password-IDs '00', '05' and '11'.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: see table 53 • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty (all executions) • <CERT_DESC>: List of certificates as specified in Profiles (all executions) 2. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, '0xE00000003' or '0xD00000002' or '0xD00000003'.
Post processing	Reset of eCard and reader

Testcase ID	<PIN-ID>
R_PACE_4.1.2a	'00'
R_PACE_4.1.2b	'05'
R_PACE_4.1.2c	'11'

Table 53: Test case R_PACE_4.1.2

Test ID	R_PACE_4.2.1_template
Purpose	Check that reader aborts PACE protocol when an incorrect password is transmitted. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate as described in table 54. Make certificates and the CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>:incorrect value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the wrong password in the reader. The return codes in the response APDUs of all commands apart from General Authenticate (Step 4)are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. -

	2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN is used). • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 54. • The <access rights> in command APDU to MSE: Set AT as described in table 54. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 06 63 XX'. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Testcase ID	Terminal type	<OID-Role>
R_PACE_4.2.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
R_PACE_4.2.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 54: Test case R_PACE_4.2.1

Test ID	R_PACE_4.2.2
Purpose	Check that reader aborts PACE protocol when an incorrect password is transmitted. The test is executed with the passwords PIN. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 4.2, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password PIN available in UT

Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, an incorrect password is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the wrong password in the reader. The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03'. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'F0 06 63 XX'. • <status_mse>: The status code for command MSE: Set AT is '90 00'.
Post processing	Reset of eCard and reader

Test ID	R_PACE_4.2.3
Purpose	Check that reader aborts PACE protocol when an incorrect password (MRZ) is transmitted. The test is executed with password MRZ. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.

References	[TR-03110], 4.2, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates and the password (MRZ) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '01' (MRZ is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: wrong value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the transport PIN in the reader. The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions (only if these commands have been transmitted by the reader): • The <PIN-ID> in command APDU to MSE: Set AT is '01' (MRZ is used). • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (inspection system). • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'D0 00 00 02'. • <status_mse>: The status code for command MSE: Set AT is '90 00', if this command has been transmitted, and not specified, otherwise.

Post processing	Reset of eCard and reader

Test ID	R_PACE_4.3.1
Purpose	Check that reader aborts PACE protocol when an incorrect combination of password-ID and CHAT is transmitted in input of PC/SC function. The password-ID addresses CAN. Use certificate with role authentication terminal, that is *not authorized to use the CAN*. The test is executed with the password CAN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.2; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 2 (*not authorized to use CAN*); CA certificates according to 32, No. 1, 2 Make certificates and the CAN available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password • <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT sends the *negative* return code '6A 80', indicating that the terminal type referenced by the CHAT is not authorized to use the referenced password (CAN), in answer message to MSE: Set AT back to the reader. It is expected that the reader aborts protocol execution after receiving this return code. Alternatively, the reader may detect the inconsistence between CHAT and password and thus does not send MSE: Set AT. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used, if this command is sent. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal), if this command is sent. • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 2, if this command is sent. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the reader must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative in all cases. It is 'F0 02 6A 80', if MSE: Set AT is sent, and 'D0 00 00 03', otherwise. • <status_mse>: The status code for command MSE: Set AT is '6A 80', if this command is sent, and not specified, otherwise.
Post processing	Reset of eCard and reader

Test ID	R_PACE_4.3.2
Purpose	Check that reader aborts PACE protocol when an incorrect combination of password-ID and CHAT is transmitted in input of PC/SC function. The password-ID addresses PIN. Use certificate with role inspection system (IS are not authorized to use PIN) The test is executed with the PIN. It calls an extension of the PC/SC function SCardControl where the password is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 4.2, C.4.1; [TR-03119], D.3
Profiles	R_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates and the PIN available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: empty, password is entered via PINPad of reader

	• <CERT_DESC>: List of certificates as specified in Profiles 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. LT sends the *negative* return code '6A 80', indicating that the terminal type referenced by the CHAT is not authorized to use the referenced password, in answer message to MSE: Set AT back to the reader. It is expected that the reader aborts protocol execution after receiving this return code. Alternatively, the reader may detect the inconsistence between CHAT and password and thus does not send MSE: Set AT. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' since PIN is used, if this command is sent. • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (inspection system), if this command is sent. • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1, if this command is sent. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the reader must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative in all cases. It is 'F0 02 6A 80', if MSE: Set AT is sent, and 'D0 00 00 03', otherwise. • <status_mse>: The status code for command MSE: Set AT is '6A 80', if this command is sent, and not specified, otherwise. •
Post processing	Reset of eCard and reader

Test ID	R_PACE_4.4.1
Purpose	Check that reader aborts PACE protocol when UT transmit the PIN via PC/SC command. The test is executed with the eID-PIN. It calls an extension of the PC/SC function SCardControl where the password is transmitted in the InBuffer for EstablishPACEChannel.
References	[TR-03110], 4.2, C.4.2; [TR-03119]

Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password (eID-PIN) available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password (eID-PIN) • <CERT_DESC>: List of certificates as specified in Profiles 2. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative, i. e. 'D0 00 00 03'. • <status_mse>: The status code for command MSE: Set AT is '90 00'
Post processing	Reset of eCard and reader

6.2.3 Terminal Authentication

For all test cases for terminal authentication the precondition is to successful establish a PACE channel. Where no terminal role and password type is defined, these parameters can be chosen from these which are supported by the reader (see chapter 4.3 Terminal type).

If no terminal type and/or password is defined, the priority of the terminal type to use in the test cases are. AT, IS and ST. The priority of the password to use in the in the test cases are CAN, PIN and MRZ. That does mean, first select the first supported terminal type then select the first supported password type which is supported in combination with the terminal type.
The used terminal type and password must be documented in the test report.

6.2.3.1 R_TA_1 – Correct Execution of Terminal Authentication Protocol

Test ID	R_TA_1.1.1
Purpose	Check reader for correct execution of terminal authentication protocol. Use certificate with role inspection system.
References	[TR-03110], 4.4, B.3, C.4.1

Profiles	R_PACE AND R_TA
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE protocol has been executed successfully with CAN in the reader. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM).LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_TA_1.1.2
Purpose	Check reader for correct execution of terminal authentication protocol. Use certificate with role authentication terminal.
References	[TR-03110], 4.4, B.3, C.4.2
Profiles	R_PACE AND R_TA
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE protocol has been executed successfully with PIN in the reader. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard)

	- ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_TA_1.1.3
Purpose	Check reader for correct execution of terminal authentication protocol. Use certificate with role signature terminal.
References	[TR-03110], 4.4, B.3, C.4.3
Profiles	R_PACE AND R_TA
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 The PACE protocol has been executed successfully with CAN in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2 and 5.3.3: All command steps for Terminal Authentication have to be executed directly from the reader without sending commands from the UT. .

Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_TA_1.2.1 deleted in version 1.2

6.2.3.2 R_TA_2 – Abort because of Inconsistent Reader Data

Test ID	R_TA_2.1.1 deleted in version 1.2

Test ID	R_TA_2.1.2
Purpose	Check that reader aborts terminal authentication if the terminal certificate description used in the terminal authentication protocol is different from the terminal certificate description used in the PACE protocol.
References	[TR-03110], 4.4, B.3, C.4.1; [TR-03119],
Profiles	R_PACE
Preconditions	Certificate with role authentication terminal and access rights according to Table 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE protocol has been executed successfully with CAN in the reader. EstablishPACEChannel must contain a complete certificate description (CERT_DESC) as described in chapter 5.2.3. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard)

	- ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). The hash value of the terminal certificate description submitted by UT in the command APDU to PSO: Verify Certificate is different from the hash value which the reader had to calculate from the terminal certificate description (CERT_DESC) used in the PACE protocol.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. UT receives response APDUs to the commands MSE: Set DST and PSO: Verify Certificate for all certificates of the certificate chain which are correctly coded without SM. The response APDUs to MSE: Set DST and PSO: Verify Certificate (for the CV certificates) coincide apart from SM with the response APDU sent by LT directly before. The response APDU to PSO: Verify for the terminal certificate received by UT consists of the negative return code '69 85'. It is expected that the reader aborts protocol execution after sending this return code.
Post processing	Reset of eCard and reader

6.2.3.3 R_TA_3 – Abort because of Internal LT Error

Test ID	R_TA_3.1.1
Purpose	Check that reader aborts terminal authentication when LT returns error code to command MSE: Set DST (setting public key for certificate verification).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard)

	- ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. When LT receive the first command APDU MSE: Set DST the return code in response APDU send from the LT shall be negative ('6A 80').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to MSE: Set DST, the Terminal Authentication must be aborted. The expected result code is '0xF8 00 6A 80' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_TA_3.2.1
Purpose	Check that reader aborts terminal authentication when LT returns error code to command PSO: Verify Certificate when verifying the terminal certificate.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. The LT sends negative return code ('6A 80') to the first PSO: Verify Certificate command received from the reader. The return codes to all other commands sent by the reader are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to PSO: Verify Certificate, the Terminal Authentication must be aborted. The expected result code is '0xF8 01 6A 80' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_TA_3.3.1
Purpose	Check that reader aborts terminal authentication when LT returns error code to command MSE: Set AT (transmitting parameters for terminal authentication to LT).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights.

	The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. The LT sends negative return code ('6A 88') in response APDU to the first MSE: Set AT. The return codes to all other commands sent by the reader are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to MSE: Set AT, the Terminal Authentication must be aborted. The expected result code is '0xF8 06 6A 88' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_TA_3.4.1
Purpose	Check that reader aborts terminal authentication when LT returns error code to command Get Challenge (random number for terminal authentication).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2:

	All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. The LT sends negative return code ('ZZ ZZ', the actual value depends on the eCard operating system) in response APDU to Get Challenge. The return codes to all other commands sent by the reader are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to Get Challenge command, the Terminal Authentication must be aborted. The expected result code is '0xF8 07 ZZ ZZ' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_TA_3.5.1
Purpose	Check that reader aborts terminal authentication when LT returns error code to command External Authenticate when checking signed data from reader in terminal authentication protocol.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. The LT sends negative return code ('69 85') in response APDU to External Authenticate. The return codes to all other commands sent by reader are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct.

	When the reader receives the negative response to External Authenticate command, the Terminal Authentication must be aborted. The expected result code is '0xF8 08 69 85' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

6.2.3.4 R_TA_4 – Abort because of Secure Messaging Error

Test ID	R_TA_4.1.1
Purpose	Check that the reader aborts terminal authentication if it does not receive SM data objects from LT in response APDU to command Get Challenge.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT.. LT receives command APDUs (with SM) from the reader. LT sends response APDUs (**with SM**) back to the reader for the commands MSE: Set DST, PSO: Verify Certificate for all certificates of the certificate chain and MSE: Set AT. LT sends response APDUs **without SM** back to the reader for the command Get Challenge.
Expected results	Each command APDU received at LT is correctly coded and secured by SM The SM in the command APDU generated by the reader and received by LT is correct. The LT receive the command APDUs: • MSE: Set DST • PSO: Verify Certificate (for all certificates in the certificate chain) • MSE: Set AT and • Get ChallengeIt is expected that the reader detects the missing SM in

	response APDU to Get Challenge received from LT and aborts command execution. The Terminal Authentication must be aborted. UT receives result code 'F0 10 00 01'.
Post processing	Reset of eCard and reader

Test ID	R_TA_4.2.1
Purpose	Check that the reader aborts terminal authentication if it receives incorrect SM data objects from LT in response APDU to command MSE: Set DST (wrong tag '98' instead of '99' for the processing status).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT.LT receives command MSE: Set DST (with SM) from the reader. LT sends response APDU with incorrect SM (wrong tag '98' instead of '99' for the processing status) back to the reader for this command. The return code in response APDU to MSE: Set DST is positive ('90 00').
Expected results	The command APDU to MSE: Set DST received at LT is correctly coded and secured by SM. It is expected that the reader detects the incorrect SM in response APDU to MSE: Set DST received from LT and aborts command execution. The Terminal Authentication must be aborted. UT receives result code 'F0 10 00 01'.
Post processing	Reset of eCard and reader

Test ID	R_TA_4.2.2

Purpose	Check that the reader aborts terminal authentication if it receives incorrect SM data objects from LT in response APDU to command MSE: Set DST (wrong cryptogram in the MAC data object).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	R_PACE AND R_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the reader. After PACE protocol especially the following eCard data are available in the reader: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: All command steps for Terminal Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDU MSE: Set DST (with SM) from the reader. LT sends response APDU with incorrect SM (wrong cryptogram in the MAC data object) back to the reader for this command. The return code in response APDU to MSE: Set DST is positive ('90 00').
Expected results	The command APDU to MSE: Set DST received at LT is correctly coded and secured by SM. It is expected that the reader detects the incorrect SM in response APDU to MSE: Set DST received from LT and aborts command execution. The Terminal Authentication must be aborted. UT receives result code 'F0 10 00 01'.
Post processing	Reset of eCard and reader

Test ID	R_TA_4.3.1
Purpose	The reader aborts terminal authentication, if it receives that SM data objects are missing from LT in response APDU to command MSE: Set DST (wrong tag '98' instead of '99' for the processing status). Check that the session keys are deleted after error in secure messaging handling (69 87). The PACE protocol is executed with password CAN

References	[TR-03110]
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to Table 30, No. 1; CA certificates according to Table 33, No. 1, 2 Make certificates and the password CAN available in UT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (CAN) is entered via PINPad of reader 2. Entering the correct CAN into the reader 3. LT receives command APDUs without SM from the test object reader and sends response APDUs without SM back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. LT receives command MSE: Set DST with SM as described in chapter 5.3.2 and sends response APDUs with incorrect SM (wrong tag '98' instead of '99' for the processing status) back to the reader for this command. Response APDU: 98 02 <SW1SW2> 8E 08 <MAC> <SW1SW2> The return code in response APDU to MSE: Set DST is positive ('90 00'). 5. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (CAN) is entered via PINPad of reader 6. Entering the correct CAN into the reader 7. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1 (without SM), 5.3.2 (with SM) and 5.3.3 (with SM). The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 8. UT receives OutBuffer of ScardControl (see chapter 5.2.4).

Expected results	1. – 2. – 3. The command APDUs for the PACE protocols which LT receives from the reader are built up as described in chapters 5.3.1. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in Table 30, No. 1. 4. UT receives a returncode from Terminal, that shows, that an expected SM Data object is missing • <Result_Code>: The result code is negative (i.e. 'XX YY 69 87'). 5. – 6. – 7. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1 (without SM), 5.3.2 (with SM) and 5.3.3 (with SM). For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in Table 30, No. 1. 8. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard and reader

Test ID	R_TA_4.3.2
Purpose	The reader aborts terminal authentication if it receives an incorrect SM data objects from LT in response APDU to command MSE: Set DST (tag '8E' has false length, '07' instead if '08'). Check that the session keys are deleted after error in secure messaging

	handling (6988). The PACE protocol is executed with password CAN
References	[TR-03110]
Profiles	R_PACE
Preconditions	Certificate with role signature terminal and access rights according to Table 30, No. 1; CA certificates according to Table 33, No. 1, 2 Make certificates and the password CAN available in UT. The PACE protocol has been executed successfully with password-IDs CAN in the reader. After PACE protocol especially the following eCard data are available in UT: -PK_PICC (public key of the eCard) -D_PICC (static domain parameters of the eCard) -ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol.
Test scenario	1. LT receives command MSE: Set DST with SM as described in chapter 5.3.2 and sends response APDUs with incorrect SM (MAC tag '8E' has only 7 bytes data) back to the reader for this command. Response APDU: 99 02 <SW1SW2> 8E 07 <MAC> <SW1SW2> The return code in response APDU to MSE: Set DST is positive ('9000'). 2. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT for terminal certificate as specified in Profiles • <PIN>: value of the password 3. LT receives command APDUs without SM from the test object reader and sends response APDUs without SM back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. UT received an error code 'F0 10 00 01' (Communication abort). 2. – 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '02', CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in Table 29, No. 1. 4. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

6.2.4 Chip Authentication

For all test cases for chip authentication the precondition is to successful establish a PACE channel. Where no terminal role and password type is defined, these parameters can be chosen from these which are supported by the reader (see chapter 4.3 Terminal type).

If no terminal type and/or password is defined, the priority of the terminal type to use in the test cases are. AT, IS and ST. The priority of the password to use in the in the test cases are CAN, PIN and MRZ. That does mean, first select the first supported terminal type then select the first supported password type which is supported in combination with the terminal type.
The used terminal type and password must be documented in the test report.

6.2.4.1 R_CA_1 – Correct Execution of Chip Authentication Protocol

Test ID	R_CA_1.1.1
Purpose	Check reader for correct execution of chip authentication protocol. Use certificate with role inspection system.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with CAN. After execution of these protocols especially the following eCard and terminal data are available in UT:

	- ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the UT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_CA_1.1.2
Purpose	Check reader for correct execution of chip authentication protocol. Use certificate with role authentication terminal.
References	[TR-03110], 4.3, B.2, C.4.2
Profiles	R_PACE AND R_CA
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with PIN. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT

	after PACE protocol. Terminal Authentication has been successful executed by the UT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_CA_1.1.3
Purpose	Check reader for correct execution of chip authentication protocol. Use certificate with role signature terminal.
References	[TR-03110], 4.3, B.2, C.4.3
Profiles	R_PACE AND R_CA
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with CAN. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3:

	All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return codes in all response APDUs are positive ('90 00'). Moreover Reader and LT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Test ID	R_CA_1.2.1
Purpose	Check reader for correct execution of chip authentication protocol. Use of several algorithms for the chip authentication protocol. The values for ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm and one static key pair. This means that only one data object ChipAuthenticationPublicKeyInfo is available in EF.CardSecurity. The test has to be executed for each chip authentication algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the

	following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. Moreover the reader and LT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. Chip Authentication performs correct and reader returns positive Result code to UT.
Post processing	Reset of eCard and reader

Test ID	R_CA_1.2.2
Purpose	Check reader for correct execution of chip authentication protocol. Use of several algorithms and multiple keys for the chip authentication protocol. LT stores two static key pairs for chip authentication. This means that in EF.CardAccess two SecurityInfos with ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo are available and that EF.CardSecurity contains two public keys. For this the related algorithms are identical but the static key pairs are different. The test has to be executed for one chip authentication algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights according.

	The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC_1 and PK_PICC_2 (public keys of the eCard) - keyId_1 and keyId_2 of the related static key pairs - D_PICC_1 and D_PICC_2 (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return codes in all response APDUs are positive ('90 00'). Moreover the reader and LT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. Chip Authentication performs correct and reader returns positive Result code to UT.
Post processing	Reset of eCard and reader

6.2.4.2 R_CA_2 – Abort because of Internal LT Error

Test ID	R_CA_2.1.1
Purpose	Check that reader aborts chip authentication when LT returns error code to command MSE: Set AT (transmitting parameters for chip authentication to LT).
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the

	following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The LT sends negative return code ('6A 88') in response APDU to the first MSE: Set AT. The return codes to all other commands sent by the reader are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to MSE: Set AT, the Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xF8 0B 6A 88' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_CA_2.2.1
Purpose	Check that reader aborts chip authentication when LT returns error code to command General Authenticate when verifying the ephemeral public key of the reader.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights according. The PACE and terminal authentication protocols have been executed successfully with i the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard)

	- D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return code to MSE: Set AT command sent by the reader is positive ('90 00'). The LT sends negative return code ('6A 80') in response APDU to the General Authenticate command.
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. When the reader receives the negative response to General Authenticate command, the Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xF8 0C 6A 80' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_CA_2.3.1
Purpose	Check that reader aborts chip authentication when LT computes incorrect key data for SM.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.

Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader.. The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC) correctly, but derives a wrong K_MAC', i. e. different from KDF_MAC(K, r2_PICC), and computes a wrong T_PICC' = MAC(K_MAC', PKeph_PCD). The reader receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. The reader derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. The reader checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD). The Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xE0 00 00 0A' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

6.2.4.3 R_CA_3 – Abort because of Incorrect LT Data

Test ID	R_CA_3.1.1
Purpose	Check that reader aborts chip authentication when LT returns an incorrect response APDU to command General Authenticate.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard)

	A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC = MAC(K_MAC, PKeph_PCD) correctly. LT transmits an incorrect r2_PICC', i. e. different from r2_PICC, to the reader. The reader receives random number r2_PICC' and authentication token T_PICC in response APDU to General Authenticate. The reader derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC' = KDF_MAC(K, r2_PICC') and computes MAC(K_MAC', PKeph_PCD).
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. The reader checks that the received T_PICC is different from the computed MAC(K_MAC', PKeph_PCD). The Chip Authentication must be aborted. UT receives negative result code The expected result code is '0xE0 00 00 0A' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_CA_3.2.1
Purpose	Check that reader aborts chip authentication when LT returns an incorrect cryptogram in response APDU to command General Authenticate.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD)

	- PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC = MAC(K_MAC, PKeph_PCD) correctly. LT transmits an incorrect T_PICC', i. e. different from T_PICC, to the reader. The reader receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. The reader derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).
Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. The reader checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD). The Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xE0 00 00 0A' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_CA_3.3.1
Purpose	Check that reader aborts chip authentication when LT returns an incorrect cryptogram generated with a wrong algorithm in response APDU to command General Authenticate. The values for ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. But for the generation of T_PICC LT uses another algorithm. The test has to be executed for one algorithm specified in the manufacturer's conformance statement (see chapter 4.2 Cryptographic algorithms). The

	following substitutions have to be used (algorithm indicated in EF.CardAccess, algorithm used by LT): - id-CA-DH 1, id-CA-DH 2 - id-CA-DH 2, id-CA-DH 1 - id-CA-DH 3, id-CA-DH 1 - id-CA-DH 4, id-CA-DH 1 - id-CA-ECDH 1, id-CA-ECDH 2 - id-CA-ECDH 2, id-CA-ECDH 1 - id-CA-ECDH 3, id-CA-ECDH 1 - id-CA-ECDH 4, id-CA-ECDH 1
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), key K_MAC = KDF_MAC(K, r2_PICC) correctly and T_PICC' = MAC(K_MAC, PKeph_PCD) also correctly, but with a wrong algorithm. LT transmits this incorrect T_PICC' to the reader. The reader receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. The reader derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).

Expected results	Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. The reader checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD). The Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xE0 00 00 0A' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

Test ID	R_CA_3.4.1
Purpose	Check that reader aborts chip authentication when LT returns an incorrect cryptogram in response APDU to command General Authenticate. The generation of the cryptogram is based on the correct algorithm but LT uses a wrong static key pair. LT stores two static key pairs for chip authentication. This means that in EF.CardAccess two SecurityInfos with ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo are available and that EF.CardSecurity contains two public keys. For this the related algorithms are identical but the static key pairs are different. The test has to be executed for one chip authentication algorithm specified in the manufacturer's conformance statement (see chapter 4.2 Cryptographic algorithms).
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	R_PACE AND R_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully in the reader. After execution of these protocols especially the following eCard and terminal data are available in the reader: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC_1 and PK_PICC_2 (public key of the eCard) - keyId_1 and keyId_2 of the related static key pairs - D_PICC_1 and D_PICC_2 (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between reader and LT after PACE protocol. Terminal Authentication has been successful executed by the reader.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3:

	All command steps for Chip Authentication have to be executed directly by the reader without sending commands from the UT. LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader.. The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC_2, PKeph_PCD, D_PICC), if UT identifies keyId_1, or K = KA(SK_PICC_1, PKeph_PCD, D_PICC), if UT identifies keyId_2 key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC' = MAC(K_MAC, PKeph_PCD) correctly. LT transmits T_PICC' to the reader. The reader receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. UT derives key material K = KA(SKeph_PCD, PK_PICC_1, D_PICC), if the reader identified keyId_1, or K = KA(SKeph_PCD, PK_PICC_2, D_PICC), if the reader identified keyId_2, keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM. The SM in the command APDU generated by the reader and received by LT is correct. The reader checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD). The Chip Authentication must be aborted. UT receives negative result code. The expected result code is '0xE0 00 00 0A' (see chapter 5.2.4).
Post processing	Reset of eCard and reader

6.2.5 Access to the eID Application

6.2.5.1 R_eID_1 – Correct Reading Access to eID Data with EAC

Test ID	R_eID_1.1.1
Purpose	Check reader for correct reading access to eID data with EAC. Use certificate with role inspection system.
References	[TR-03110], C.4.1, E.1
Profiles	R_PACE AND R_eID

Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with MRZ-Password via InBuffer for EstablishPACEChannel. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there is subsequently performed with SFIs '01', ..., '0C' and '11', ..., '15' (referencing data groups DG1,..., DG12 and DG17,..., DG21). UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_1.2.1
Purpose	Check reader for correct reading access to eID data with EAC. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2, E.1
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1, 2; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there is subsequently performed with SFIs '01', ..., '0C' and '11', ..., '15' (referencing data groups DG1,..., DG12 and DG17,..., DG21). UT sends command APDUs to the test object reader via PC/SC with SM. LT

	receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_1.3.1_template
Purpose	Check reader for correct reading access to eID data with EAC. Use certificate with role authentication terminal. In compliance to the Implementation Conformance Statement different algorithms for secure messaging have to be used. The algorithms - 3DES - AES – 128 - AES – 192 - AES – 256 have to be tested if supported by the reader The test has to be performed for each supported algorithm.
References	[TR-03110], A.4, C.4.2, E.1, F.2
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1, 2; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with password CAN or PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there is subsequently performed with SFIs '01', ..., '0C' and '11', ..., '15' (referencing data groups DG1,..., DG12 and DG17,..., DG21). UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are

	positive ('90 00').
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Testcase ID	Algorithm
R_eID_1.3.1a	3DES
R_eID_1.3.1b	AES – 128
R_eID_1.3.1c	AES – 192
R_eID_1.3.1d	AES – 256

Table 55: Test case R_eID_1.3.1

6.2.5.2 R_eID_2 – Correct Writing Access to eID Data with EAC

Test ID	R_eID_2.1.1
Purpose	Check reader for correct writing access to eID data with EAC. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2, E.1
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN or PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. The contents of data groups DG17,..., DG21 of the eCard at LT are known in UT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.6: The command sequence described there is subsequently performed with SFIs '11', ..., '15' (referencing data groups DG17,..., DG21). Use new data for each DG. Check the correct execution of the UPDATE BINARY by reading the DG.

	UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before. Check that the new data in the updated DGs corresponds with the data that was used in the UPDATE BINARY command.
Post processing	The original contents of DG17, ..., DG21 in the eCard at LT is restored using the command sequence from chapter 5.3.6. Reset of eCard and reader

6.2.5.3 R_eID_3 – Correct Execution of Internal eID Functions

Test ID	R_eID_3.1.1
Purpose	Check reader for correct execution of restricted identification for eCards. Use certificate with role authentication terminal.
References	[TR-03110], 4.5, C.4.2
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with password-IDs CAN or PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.7: UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.

Post processing	Reset of eCard and reader

Test ID	R_eID_3.2.1
Purpose	Check reader for correct age verification for card holder. Use certificate with role authentication terminal. The test case is performed in several variants for the date of birth to be checked: - date of birth to be checked is smaller than date of birth in eCard – 1 - date of birth to be checked coincides with date of birth in eCard – 1 - date of birth to be checked coincides with date of birth - date of birth to be checked coincides with date of birth in eCard + 1 - date of birth to be checked is greater than date of birth in eCard + 1 This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], C.4.2
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with password-IDs CAN or PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. In the MSE: Set AT command of the terminal authentication protocol (see chapter 5.3.2) the field <aux_data> must be mapped to the following value: 73 <L_73> 06 <L_06> <OID> 53 <L_53> <disc_data> OID: OID for age verification disc_data: date of birth to be checked (the variants to be checked are defined under Purpose)
Test scenario	The execution steps have to be performed as described in chapter 5.3.8 (one execution for each variant as defined under Purpose): UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM

	and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_3.3.1
Purpose	Check reader for correct verification of card holder's community ID. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2
Profiles	R_PACE AND R_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN or PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. In the MSE: Set AT command of the terminal authentication protocol (see chapter 5.3.2) the field `<aux_data>` must be mapped to the following value: `73 <L_73> 06 <L_06> <OID> 53 <L_53> <disc_data>` OID: OID for community ID verification disc_data: community ID to be checked
Test scenario	The execution steps have to be performed as described in chapter 5.3.8: UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

6.2.5.4 R_eID_4 – Password Management Functions for Authenticated Terminals

The test case in this chapter are only for readers which can handle the PIN management function direct and accept AT certificates which have enabled the bit “PIN Management”.

Test ID	R_eID_4.1.1
Purpose	Check that an authenticated terminal supports the password management function to change the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	R_PACE AND R_eID AND (R_Chg_PIN OR R_PIN_MGT_AT)
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '03' (for changing PIN): UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard and reader

Test ID	R_eID_4.2.1
Purpose	Check that an authenticated terminal supports the password management function to change the CAN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	R_PACE AND R_eID AND R_Chg_CAN

Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with password-ID CAN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '02' (for changing CAN): UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	The original CAN is restored in the eCard. Reset of eCard and reader

Test ID	R_eID_4.3.1
Purpose	Check that an authenticated terminal supports the password management function to unblock the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	R_PACE AND R_eID AND R_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PIN is blocked in the eCard. The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.2: UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the

	reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_4.4.1
Purpose	Check that an authenticated terminal supports the password management function to activate the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	R_PACE AND R_eID AND R_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PIN is deactivated in the eCard. The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.3, where the command Activate is performed with (Byte INS = '44'): UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_4.5.1
Purpose	Check that an authenticated terminal supports the password management

	function to deactivate the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	R_PACE AND R_eID AND R_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with PIN as user input via PIN-Pad. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.3, where the command Deactivate is performed with (Byte INS = '04'): UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The deactivated PIN is activated again. Reset of eCard and reader

6.2.5.5 R_eID_5 – Password Management Functions for Unauthenticated Terminals after PACE

Test ID	R_eID_5.1.1
Purpose	Check that an unauthenticated terminal supports the password management function to change the PIN. The current PIN is *not* a transport PIN.
References	[TR-03110], 3.5.1; [PCSC10]
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	The PACE protocol has been executed successfully with PIN as user input via PIN-Pad, where the PIN is not a transport PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.

Test scenario	UT sends command FEATURE_MODIFY_PIN_DIRECT to the test object reader via PC/SC. Since the SM channel is a established with the current active PIN, only the new PIN shall be requested as user input. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard and reader

Test ID	R_eID_5.1.2
Purpose	Check that an unauthenticated terminal supports the password management function to change the PIN. The current PIN is a transport PIN.
References	[TR-03110], 3.5.1; [PCSC10]
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	The PACE protocol has been executed successfully with PIN as user input via PIN-Pad, where the PIN is a transport PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.
Test scenario	UT sends command FEATURE_MODIFY_PIN_DIRECT to the test object reader via PC/SC. Since the SM channel is a established with the current active PIN, only the new PIN shall be requested as user input. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_5.2.1

Purpose	Check that an unauthenticated terminal supports the password management function to reset the retry counter for the PIN using the PUK.
References	[TR-03110], 3.5.1; [PCSC10]
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	The PIN is blocked in the eCard. The PACE protocol has been executed successfully with PUK as user input via PIN-Pad. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.
Test scenario	UT sends command APDU “Reset Retry Counter” without SM to the test object reader. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_eID_5.3.1
Purpose	Check that an unauthenticated terminal supports the password management function to set a new PIN using the PUK.
References	[TR-03110], 3.5.1; [PCSC10]
Profiles	R_PACE AND R_eID AND R_Chg_PIN_PUK
Preconditions	The PACE protocol has been executed successfully with PUK as user input via PIN-Pad. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.
Test scenario	UT sends command FEATURE_MODIFY_PIN_DIRECT to the test object reader via PC/SC. Since the SM channel is a established with the PUK, only the new PIN shall be requested as user input. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before.

	Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard and reader

Test ID	R_eID_5.4.1
Purpose	Check that an unauthenticated terminal resumes temporarily a PIN using the CAN. The test is executed with the CAN. It calls an extension of the PC/SC function SCardControl where the CAN is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 3.5.1
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	The PIN is suspended (retry counter = 1).
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (since CAN is used) • <CHAT>: empty • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: empty 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty) . 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00').

	• <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Test ID	R_eID_5.5.1
Purpose	Check that an unauthenticated terminal resumes a temporarily resumed PIN by using it in PACE protocol. This PACE protocol is performed with Secure Messaging (SM) where the SM keys have been derived by execution of another PACE protocol with the CAN. The test case assumes, as specified in the Precondition, that the PACE protocol with CAN has been executed before test start. The test is executed with the PIN. It calls an extension of the PC/SC function SCardControl where the PIN is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 3.5.1
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	First a suspended PIN has been used (i. e. retry counter = 1). Then the PACE protocol has been executed successfully with password-ID CAN to temporarily resume the PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (since PIN is used) • <CHAT>: empty • <PIN>:empty, password is entered via PINPad of reader • <CERT_DESC>: empty 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). All command and response APDUs are secured with Secure Messaging where SM keys are used that have have been derived in the PACE protocol with CAN

	according to Precondition. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03', since PIN is used. • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty). • All command APDUs are secured correctly with Secure Messaging. 3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '63 C1'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Test ID	R_eID_5.5.2
Purpose	Check that an unauthenticated terminal resumes a temporarily resumed PIN by using it in PACE protocol. This PACE protocol is performed with Secure Messaging (SM) where the SM keys have been derived by execution of another PACE protocol with the CAN. The test case assumes, as specified in the Precondition, that the PACE protocol with CAN has been executed before test start. In the test case a first attempt to resume the temporarily resumed PIN is answered by an error code from LT. A second attempt to resume the PIN shall be performed with the SM keys derived in the PACE protocol with the CAN The test is executed with the PIN. It calls an extension of the PC/SC function SCardControl where the PIN is transmitted as user input via the PINPad of the reader.
References	[TR-03110], 3.5.1

Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	First a suspended PIN has been used (i. e. retry counter = 1). Then the PACE protocol has been executed successfully with password-ID CAN to temporarily resume the PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows (first attempt to resume PIN): • <PIN-ID>: '03' (since PIN is used) • <CHAT>: empty • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: empty 2. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return code in the response APDU to MSE: Set AT is negative ('6A 00'). The command and response APDU to MSE: Set AT are secured with Secure Messaging where SM keys are used that have have been derived in the PACE protocol with CAN according to Precondition. 3. UT receives OutBuffer of SCardControl (see chapter 5.2.4). 4. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows (second attempt to resume PIN): • <PIN-ID>: '03' (since PIN is used) • <CHAT>: empty • <PIN>: empty, password is entered via PINPad of reader • <CERT_DESC>: empty 5. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. All command and response APDUs are secured with Secure Messaging where SM keys are used that have have been derived in the PACE protocol with CAN according to Precondition. 6. UT receives OutBuffer of SCardControl (see chapter 5.2.4).
Expected results	1. - 2. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The command APDU to MSE: Set AT is sent by the reader. • The command APDU to MSE: Set AT is secured correctly with Secure Messaging.

	3. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is negative. • <data_card_acc>, <CAR1>, <CAR2> and <IDPICC>: The fields are empty. 4. - 5. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03', since PIN is used. • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty). • All command APDUs are secured correctly with Secure Messaging where the SM keys are the same as in step 2. 6. For the OutBuffer of SCardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '63 C1'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC.
Post processing	Reset of eCard and reader

Test ID	R_eID_5.6.1
Purpose	Check that an unauthenticated terminal aborts the password management function to change the PIN if the new PIN was submitted by the UT.
References	[TR-03110], 3.5.1
Profiles	R_PACE AND R_eID AND R_PIN_MGT_uT
Preconditions	The PACE protocol has been executed successfully with eID-PIN as user input via PIN-Pad. After execution of this protocol, a Secure Messaging Channel (SM) is established between reader and LT.

Test scenario	UT sends command APDU “Reset Retry Counter” with new PIN as described in chapter 5.3.9.1 (P2 = '03') without SM to the test object reader. The reader receives the command APDU sends response APDU back to the UT.
Expected results	The reader returns error code 0x6982
Post processing	Reset of eCard and reader

6.2.6 Access to Biometric Data

6.2.6.1 R_bio_1 – Correct Reading Access to Biometric Data with EAC

Test ID	R_bio_1.1.1
Purpose	Check reader for correct reading access to biometric data Fingerprint (DG 3) and Iris (DG 4) of the ePassport application with EAC. Use certificate with role inspection system.
References	[TR-03110], C.4.1
Profiles	R_PACE AND R_bio
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.10: The command sequence described there is subsequently performed with SFIs referencing data groups DG 3 (Fingerprint) and DG 4 (Iris). UT sends command APDUs to the test object reader via PC/SC with SM. LT receives command APDUs with SM from the reader and sends response APDUs with SM back to the reader. UT receives response APDUs from the reader via PC/SC with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Use of the Digital Signature Application

6.2.6.2 R_Sig_1 – Successful Key Pair Generation

Test ID	R_Sig_1.1.1
Purpose	Check correct execution of authentication protocols PACE, TA and CA with authentication terminal of the QCA and LT. PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The test for the PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data requested from the QCA (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends

	response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard and reader

Test ID	R_Sig_1.2.1
Purpose	Check access with the authentication terminal of the QCA to the necessary identification data of the eCard-User (DG1 to DG21). PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03110], E.1, E.1.1; [TR-03117], 4.3.2
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all

	response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends command SELECT to the eCard: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with 'E80704007F00070302' = AID of DF.eID 8. UT sends READ BINARY commands to the eCard to retrieve data DG1 to DG21 from the eID application: READ BINARY: '0C B0 P1 00 <L_c> 97 01 00 8E 08 <MAC> 00' with P1 = '8 <SFI>'. Read all Data Groups which are available on the chip (usually DG1 to 12 and 17 to 21).
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader

	and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response APDU of the command SELECT is positive. Response APDU for SELECT is: 99 02 90 00 8E 08 <MAC> 90 00 8. The response APDUs of the READ BINARY commands are all positive. Response APDU for READ BINARY DG_x is: 87 <L_{87}> 01 <Cryptogram of DG_x> 99 02 90 00 8E 08 <MAC> 90 00 The decrypted and reformatted data coincide with the related data of DG_x in LT.
Post processing	Reset of eCard and reader

Test ID	R_Sig_1.3.1
Purpose	Check access with the authentication terminal of the QCA to the signature application of the eCard for key generation. PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated”
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles

	2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends SELECT to the eCard: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with 'A0 00 00 01 67 45 53 49 47 4E' = AID of DF.eSign 8. UT sends the command GENERATE ASYMMETRIC KEY PAIR to the eCard Command APDU GENERATE ASYMMETRIC KEY PAIR: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional)
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The `<PIN-ID>` in command APDU to MSE: Set AT is '03' (PIN). • The role in `<OID-Role>` in command APDU to MSE: Set AT is '02' (authentication terminal). • The `<access rights>` in command APDU to MSE: Set AT are as defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4),

	the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response of the command SELECT is positive. Response APDU for SELECT is: '99 02 90 00 8E 08 <MAC> 90 00' 8. The response of the command GENERATE ASYMMETRIC KEY PAIR is positive:. Response APDU for GENERATE ASYMMETRIC KEY PAIR is: 87 <L_{87}> 01 <Cryptogram of Public Key Data Object> 99 02 90 00 8E 08 <MAC> 90 00 Public Key Data Object: '7F 49' <$L_{7F\ 49}$> <Public Key> The decrypted and reformatted data coincide with the public key data object of the signature key in the eSign application in LT.
Post processing	Termination of the signature key in the eCard Reset of eCard and reader

6.2.6.3 R_Sig_2 – Abort Key Pair Generation

Test ID	R_Sig_2.1.1

Purpose	Check that the reader aborts key pair generation if it the authentication terminal does **not** have the access right “Install Qualified Certificate”. PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal **without** access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 3; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated”
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least the eCard-User related data (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response

	APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends the command SELECT to the eCard: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with 'A0 00 00 01 67 45 53 49 47 4E' = AID of DF.eSign 8. UT sends command GENERATE ASYMMETRIC KEY PAIR to the eCard: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional)
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 3. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.

	6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response of the command SELECT is positive. Response APDU for SELECT is: '99 02 90 00 8E 08 <MAC> 90 00' 8. The response APDU of the command GENERATE ASYMMETRIC KEY PAIR is negative ('69 82', Security Status not satisfied). Response APDU for GENERATE ASYMMETRIC KEY PAIR is: 99 02 69 82 8E 08 <MAC> 69 82
Post processing	Reset of eCard and reader

Test ID	R_Sig_2.1.2
Purpose	Check that the reader aborts key pair generation if the authentication terminal have the access right “Install Qualified Certificate” but the PACE protocol is performed with the CAN (instead of the PIN). PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID **CAN**.
References	[TR-03117], 4.3.2, A.2.5
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated”
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data (DG1 to DG21)

	• <PIN>: empty, because password (CAN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct CAN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends command SELECT to the eCard: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with 'A0 00 00 01 67 45 53 49 47 4E' = AID of DF.eSign 8. UT sends command GENERATE ASYMMETRIC KEY PAIR to the eCard: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional)
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as

	defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response of the command SELECT is positive. Response APDU for SELECT is: '99 02 90 00 8E 08 <MAC> 90 00' 8. The response of the command GENERATE ASYMMETRIC KEY PAIR is negative ('69 82', Security Status not satisfied). Response APDU for GENERATE ASYMMETRIC KEY PAIR is: 99 02 69 82 8E 08 <MAC> 69 82
Post processing	Reset of eCard and reader

Test ID	R_Sig_2.2.1
Purpose	Check that the reader aborts key pair generation if the eSign application could not be selected. PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”.

	The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated”
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends command SELECT to the eCard:

	'0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with '4E 47 49 53 45 67 01 00 00 A0' = Incorrect AID of DF.eSign
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response APDU of the command SELECT is negative('6A 82', File or application not found). Response APDU for SELECT is: '99 02 6A 82 8E 08 <MAC> 6A 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_2.2.2
Purpose	Check that the reader aborts key pair generation if the LT returns an error code to the command Generate Asymmetric Key Pair. PACE between reader and LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	R_PACE AND R_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all

	response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3. 7. UT sends command SELECT to the eCard: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with 'A0 00 00 01 67 45 53 49 47 4E' = AID of DF.eSign 8. UT sends command GENERATE ASYMMETRIC KEY PAIR to the eCard: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 9. The LT returns an error code to the reader, the response APDU is built up as follows: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found)
Expected results	1. - 2. - 3. The command APDUs which LT receives from the reader are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader

	and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 7. The response APDU of the command SELECT is positive. Response APDU for SELECT is: '99 02 90 00 8E 08 <MAC> 90 00' 8. The LT receives the secured command GENERATE ASYMMETRIC KEY PAIR as sent by the UT. 9. The response APDU of the command GENERATE ASYMMETRIC KEY PAIR is negative. Response APDU for GENERATE ASYMMETRIC KEY PAIR as sent by LT.
Post processing	Reset of eCard and reader

6.2.6.4 R_Sig_3 removed in version 1.1

6.2.6.5 R_Sig_4 – Successful Signature Generation

Test ID	R_Sig_4.1.1
Purpose	Check correct execution of the PACE protocol, TA and CA with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in

	InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (CAN) is entered via PINPad of reader 2. Entering the correct CAN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4).
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard and reader

Test ID	R_Sig_4.2.1
Purpose	Check correct execution of the eSign-PIN verification with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate

	with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.1
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign-PIN is valid, in status “operational” and not blocked Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 2. UT calls PC/SC function FEATURE_VERIFY_PIN_DIRECT of the signature terminal. 3. The signature terminal requests for entering the password 4. The correct eSign-PIN is entered successfully into the signature terminal 5. The signature terminal generates the secured command APDU for the Verify command and transmits it to the LT 6. The LT verifies the secured command APDU for the Verify command 7. The LT sends the secured response APDU back to the signature terminal; Secured Response APDU for Verify: '99 02 90 00 8E 08 <MAC> 90 00' 8. The signature terminal sends the unsecured response APDU for the VERIFY command back to the UT
Expected results	1. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.

	2. - 3. The signature terminal requests the password entry 4. - 5. - 6. The secured command APDU for the VERIFY which LT receives from the reader is built up as follows; Secured command APDU Verify: '0C 20 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): eSign-PIN 7. - 8. The unsecured response APDU for the VERIFY which UT receives from the reader is built up as follows: Unsecured response APDU Verify: '90 00'
Post processing	Reset of eCard and reader

Test ID	R_Sig_4.3.1
Purpose	Check correct execution of the signature generation with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.4
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign application is selected and the eSign-PIN was successfully verified Private signature key is in status “operational”
Test scenario	1. UT sends unsecured command APDU for the PSO : Compute Digital Signature command to the signature terminal: Unsecured command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>'

	2. The signature terminal generates the secured command APDU for the PSO : Compute Digital Signature command and transmits it to the LT 3. The LT verifies the secured command APDU for the PSO : Compute Digital Signature command 4. The LT sends the secured response APDU for the PSO : Compute Digital Signature command with the generated signature back to the signature terminal; Secured response APDU for PSO : Compute Digital Signature: '87 <L_{87}> 01 <Cryptogram> 99 02 90 00 8E 08 <MAC> 90 00'; Cryptogram: Encrypted Signature 5. The signature terminal sends the unsecured response APDU for the PSO : Compute Digital Signature command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the PSO : Compute Digital Signature command which LT receives from the reader is built up as follows; Secured command APDU PSO : Compute Digital Signature: Select: '0C 2A 9E 9A <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted hash value of the data to be signed 4. - 5. The unsecured response APDU for the PSO : Compute Digital Signature which UT receives from the reader is built up as follows: Unsecured response APDU PSO : Compute Digital Signature: '<Digital Signature> 90 00';
Post processing	Reset of eCard and reader

6.2.6.6 R_Sig_5 – Abort Signature Generation

Test ID	R_Sig_5.1.1
Purpose	Check that the reader aborts eSign-PIN verification if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.1

Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign-PIN is valid, in status “operational” and not blocked Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 2. UT calls PC/SC function FEATURE_VERIFY_PIN_DIRECT of the signature terminal. 3. The signature terminal requests for entering the password 4. The correct eSign-PIN is entered successfully into the signature terminal 5. The signature terminal generates the secured command APDU for the VERIFY command and transmits it to the LT 6. The LT verifies the secured command APDU for the VERIFY command 7. The LT sends the secured response APDU indicating an error back to the signature terminal; Secured Response APDU for VERIFY with status code '69 82': '99 02 69 82 8E 08 <MAC> 69 82' 8. The signature terminal sends the unsecured response APDU for the VERIFY command back to the UT
Expected results	1. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 2. - 3. The signature terminal requests the password entry 4. - 5. -

	6. The secured command APDU for the VERIFY which LT receives from the reader is built up as follows; Secured command APDU Verify: '0C 20 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): eSign-PIN 7. - 8. The unsecured response APDU for the Verify which UT receives from the reader is built up as follows: Unsecured response APDU Verify: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_5.1.2
Purpose	Check that the reader aborts signature generation if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.4
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign application is selected and the eSign-PIN was successfully verified (status have to be set in LT) Private signature key is in status “operational”
Test scenario	1. UT sends unsecured command APDU for the PSO : Compute Digital Signature command to the signature terminal: Unsecured command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>' 2. The signature terminal generates the secured command APDU for the PSO : Compute Digital Signature command and transmits it to the LT 3. The LT verifies the secured command APDU for the PSO : Compute

	Digital Signature command 4. The LT sends the secured response APDU for the PSO : Compute Digital Signature command with the an error code back to the signature terminal; Secured response APDU for PSO : Compute Digital Signature: '99 02 69 82 8E 08 <MAC> 69 82 5. The signature terminal sends the unsecured response APDU for the PSO : Compute Digital Signature command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the PSO : Compute Digital Signature command which LT receives from the reader is built up as follows; Secured command APDU PSO : Compute Digital Signature: Select: '0C 2A 9E 9A <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted hash value of the data to be signed 4. - 5. The unsecured response APDU for the PSO : Compute Digital Signature which UT receives from the reader is built up as follows: Unsecured response APDU PSO : Compute Digital Signature: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_5.1.3
Purpose	Check that the reader aborts eSign-PIN verification if the password which is used for the PACE protocol doesn't provide the required permits. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID **PIN** which is transmitted as user input via the Pinpad.
References	[TR-03117], 4.4.2, A.2.1
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with **PIN**, TA and CA between signature terminal and LT with establishment of a trusted channel

	eSign-PIN is valid, in status “operational” and not blocked Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 2. UT calls PC/SC function FEATURE_VERIFY_PIN_DIRECT of the signature terminal. 3. The signature terminal requests for entering the password 4. The correct eSign-PIN is entered successfully into the signature terminal 5. The signature terminal generates the secured command APDU for the VERIFY command and transmits it to the LT 6. The LT verifies the secured command APDU for the VERIFY command 7. The LT sends the secured response APDU indicating an error back to the signature terminal; Secured Response APDU for VERIFY with status code '69 82': '99 02 69 82 8E 08 <MAC> 69 82' 8. The signature terminal sends the unsecured response APDU for the VERIFY command back to the UT
Expected results	1. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 2. - 3. The signature terminal requests the password entry 4. - 5. - 6. The secured command APDU for the VERIFY which LT receives from the reader is built up as follows; Secured command APDU Verify: '0C 20 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): eSign-PIN 7. -

	8. The unsecured response APDU for the VERIFY which UT receives from the reader is built up as follows: Unsecured response APDU Verify: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_5.1.4
Purpose	Check that the reader aborts **signature generation** if the password which is used for the PACE protocol doesn't provide the required permits. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID **PIN** which is transmitted as user input via the Pinpad.
References	[TR-03117], 4.4.2, A.2.4
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with **PIN**, TA and CA between signature terminal and LT with establishment of a trusted channel eSign application is selected and the eSign-PIN was successfully verified (status have to be set in LT) Private signature key is in status “operational”
Test scenario	1. UT sends unsecured command APDU for the PSO : Compute Digital Signature command to the signature terminal: Unsecured command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>' 2. The signature terminal generates the secured command APDU for the PSO : Compute Digital Signature command and transmits it to the LT 3. The LT verifies the secured command APDU for the PSO : Compute Digital Signature command 4. The LT sends the secured response APDU for the PSO : Compute Digital Signature command with the an error code back to the signature terminal; Secured response APDU for PSO : Compute Digital Signature: '99 02 69 82 8E 08 <MAC> 69 82 5. The signature terminal sends the unsecured response APDU for the PSO : Compute Digital Signature command back to the UT

Expected results	1. - 2. - 3. The secured command APDU for the PSO : Compute Digital Signature command which LT receives from the reader is built up as follows; Secured command APDU PSO : Compute Digital Signature: Select: '0C 2A 9E 9A <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted hash value of the data to be signed 4. - 5. The unsecured response APDU for the PSO : Compute Digital Signature which UT receives from the reader is built up as follows: Unsecured response APDU PSO : Compute Digital Signature: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_5.2.1
Purpose	Check that the reader aborts eSign-PIN verification if the LT sends an error code. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.1
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign-PIN is valid, in status “operational” and not blocked Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive

	('90 00'). 2. UT calls PC/SC function FEATURE_VERIFY_PIN_DIRECT of the signature terminal. 3. The signature terminal requests for entering the password 4. The correct eSign-PIN is entered successfully into the signature terminal 5. The signature terminal generates the secured command APDU for the VERIFY command and transmits it to the LT 6. The LT verifies the secured command APDU for the VERIFY command 7. The LT sends the secured response APDU indicating an error back to the signature terminal; Secured Response APDU for Verify with status code SW1 SW2: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' and SW1 SW2 = '63C2' (Verification failed) or SW1 SW2 = '69 83' (Authentication method blocked) or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found) 8. The signature terminal sends the unsecured response APDU for the VERIFY command back to the UT
Expected results	1. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 2. - 3. The signature terminal requests the password entry 4. - 5. - 6. The secured command APDU for the VERIFY which LT receives from the reader is built up as follows; Secured command APDU VERIFY: '0C 20 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): eSign-PIN 7. - 8. The unsecured response APDU for the VERIFY which UT receives from the reader is built up as follows: Unsecured response APDU Verify: SW1 SW2 with: SW1 SW2 = '63C2' (Verification failed) or SW1 SW2 = '69 83' (Authentication method blocked) or

	SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found)
Post processing	Reset of eCard and reader

Test ID	R_Sig_5.2.2
Purpose	Check that the reader aborts signature generation if the LT sends an error code. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID CAN which is transmitted by the UT to the signature terminal.
References	[TR-03117], 4.4.2, A.2.4
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel eSign application is selected and the eSign-PIN was successfully verified (status have to be set in LT) Private signature key is in status “operational”
Test scenario	1. UT sends unsecured command APDU for the PSO : Compute Digital Signature command to the signature terminal: Unsecured command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>' 2. The signature terminal generates the secured command APDU for the PSO : Compute Digital Signature command and transmits it to the LT 3. The LT verifies the secured command APDU for the PSO : Compute Digital Signature command 4. The LT sends the secured response APDU for the PSO : Compute Digital Signature command with the an error code back to the signature terminal; Secured response APDU for PSO : Compute Digital Signature: '99 02 SW1 SW2 8E 08 <MAC> 69 82 with or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 80' (Incorrect parameters in data field) 5. The signature terminal sends the unsecured response APDU for the PSO : Compute Digital Signature command back to the UT

Expected results	1. - 2. - 3. The secured command APDU for the PSO : Compute Digital Signature command which LT receives from the reader is built up as follows; Secured command APDU PSO : Compute Digital Signature: Select: '0C 2A 9E 9A <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted hash value of the data to be signed 4. - 5. The unsecured response APDU for the PSO : Compute Digital Signature which UT receives from the reader is built up as follows: Unsecured response APDU PSO : Compute Digital Signature: '69 84' (Reference data not usable) or '6A 80' (Incorrect parameters in data field)
Post processing	Reset of eCard and reader

6.2.6.7 R_Sig_6 – Successful Password Management Functions

Test ID	R_Sig_6.1.1
Purpose	Check correct execution of setting the eSign-PIN with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “terminated” Status of the private signature key is “terminated” Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used)

	• <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (PIN) is entered via PINPad of reader 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 7. The signature terminal requests for entering the new password 8. The correct new eSign-PIN is entered successfully into the signature terminal 9. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 10. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 11. The LT sends the secured response APDU back to the signature terminal; Secured Response APDU for Change Reference Data: '99 02 90 00 8E 08 <MAC> 90 00' 12. The signature terminal sends the unsecured response APDU for the Change Reference Data command back to the UT
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03'

	(signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. - 7. The signature terminal requests the entry for the new password 8. - 9. - 10. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU Change Reference Data: '0C 24 01 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): new eSign-PIN 11. - 12. The unsecured response APDU for the CHANGE REFERENCE DATA which UT receives from the reader is built up as follows: Unsecured response APDU Change Reference Data: '90 00'
Post processing	Reset of eCard and reader

Test ID	R_Sig_6.2.1
Purpose	Check correct execution of changing the eSign-PIN with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”.

	The test for the PACE protocol is executed with password-ID CAN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and not blocked Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '02' (CAN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (CAN) is entered via PINPad of reader 2. Entering the correct CAN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 7. The signature terminal requests for entering the old and the new password 8. The correct old and the new eSign-PIN are entered successfully into the signature terminal 9. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT

	10. The LT verifies the secured command APDU for the Change Reference Data command 11. The LT sends the secured response APDU back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 90 00 8E 08 <MAC> 90 00' 12. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. - 7. The signature terminal requests the entry for the old and new password 8. - 9. - 10. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN

	with Data (in plaintext): old eSign-PIN || new eSign-PIN 11. - 12. The unsecured response APDU for the CHANGE REFERENCE DATA which UT receives from the reader is built up as follows: Unsecured response APDU Change Reference Data: '90 00'
Post processing	Reset of eCard and reader

Test ID	R_Sig_6.3.1
Purpose	Check correct execution of resetting the retry counter of the eSign-PIN with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PUK which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.3
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and blocked Use counter of the PUK is not zero, i.e. the processing of a further reset retry counter is possible Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '04' (PUK is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (PUK) is entered via PINPad of reader 2. Entering the correct PUK into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2

	and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. UT sends the unsecured command RESET RETRY COUNTER to the signature terminal. Unsecured command APDU for Reset Retry Counter: '00 2C 03 P2 <L_c> <data is empty>'; P2: Reference to eSign-PIN 7. The signature terminal generates the secured command APDU for the RESET RETRY COUNTER command and transmits it to the LT 8. The LT verifies the secured command APDU for the RESET RETRY COUNTER command 9. The LT sends the secured response APDU back to the signature terminal; Secured Response APDU for RESET RETRY COUNTER: '99 02 90 00 8E 08 <MAC> 90 00' 10. The signature terminal sends the unsecured response APDU for the RESET RETRY COUNTER command back to the UT
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '04' (PUK). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i. e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'.

	• <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. - 7. - 8. The secured command APDU for the RESET RETRY COUNTER which LT receives from the reader is built up as follows; Secured command APDU RESET RETRY COUNTER: '0C 2C 03 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN 9. - 10. The unsecured response APDU for the RESET RETRY COUNTER which UT receives from the reader is built up as follows: Unsecured response APDU RESET RETRY COUNTER: '90 00'; The error counter of the eSign-PIN in LT is set to its initial value.
Post processing	Reset of eCard and reader

6.2.6.8 R_Sig_7 – Abort Password Management Functions

Test ID	R_Sig_7.1.1
Purpose	Check that the reader aborts setting of the eSign-PIN if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2

	Make certificates available in UT Status of the eSign-PIN is “terminated” Status of the private signature key is “terminated” Successful PACE protocol with PIN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the new password 3. The correct new eSign-PIN is entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA COMMAND 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 69 82 8E 08 <MAC> 69 82' 7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the new password 3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 01 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.1.2
Purpose	Check that the reader aborts **setting of the eSign-PIN** if the password which is used for the PACE protocol doesn't provide the required permits. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID **CAN** which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “terminated” Status of the private signature key is “terminated” Successful PACE protocol with **CAN**, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the new password 3. The correct new eSign-PIN is entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 69 82 8E 08 <MAC> 69 82' 7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the new password

	3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 01 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.2.1_template
Purpose	Check that the reader aborts password management function if the eSign application could not be selected. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with different password-IDs: PIN, CAN and PUK. The specific password is entered via the PINPad of the signature terminal. So this test case covers three variants: - PIN: abort in case of setting the eSign-PIN - CAN: abort in case of changing the eSign-PIN - PUK: abort in case of resetting the retry counter of the eSign-PIN
References	[TR-03117], 4.4.3
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “terminated” Status of the private signature key is “terminated” Successful PACE protocol with PIN, CAN or PUK (see table 56), TA and CA between signature terminal and LT with establishment of a trusted channel
Test scenario	1. UT starts function (see table 56) by sending unsecured command SELECT

	to the signature terminal: Unsecured command APDU for Select: '00 A4 04 0C <L_c> <AID of eSign application>'; 'A0 00 00 01 67 45 53 49 47 4E' = AID of eSign application 2. The LT receives the secured command APDU for the command SELECT. 3. The LT sends the secured response APDU with an error code to the reader. The secured response APDU for the SELECT is built up as follows: '99 02 6A 82 8E 08 <MAC> 6A 82'; '6A 82' : File or application not found 4. The UT receives the response code to the command SELECT.
Expected results	1. - 2. The secured command APDU for the SELECT command which LT receives from the reader is built up as follows; Secured command APDU SELECT: '0C A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with Data (in plaintext): AID = 'A0 00 00 01 67 45 53 49 47 4E' AID of eSign application 3. - 4. The unsecured response APDU for the SELECT command which UT receives from the reader is built up as follows: Unsecured response APDU SELECT: '6A 82'
Post processing	Reset of eCard and reader

Testcase	***Password***	***Function***
R_Sig_7.2.1a	PIN	Setting the eSign-PIN.
R_Sig_7.2.1b	CAN	Changing the eSign-PIN.
R_Sig_7.2.1c	PUK	Resetting the retry counter of the eSign-PIN.

Table 56: Test case R_Sig_7.2.1

Test ID	R_Sig_7.2.2
Purpose	Check that the reader aborts setting of the eSign-PIN if the LT returns an error code to the command Change Reference Data. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”.

	The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the new password 3. The correct new eSign-PIN is entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 83' (Authentication method blocked) or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 80' (Incorrect parameters in data field) or SW1 SW2 = '69 88' (Referenced data not found) 7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the new password 3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 01 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00';

	P2: Reference of the eSign-PIN with Data (in plaintext): new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: SW1 SW2 with SW1 SW2 = '69 83' (Authentication method blocked) or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 80' (Incorrect parameters in data field) or SW1 SW2 = '69 88' (Referenced data not found)
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.3.1
Purpose	Check that the reader aborts changing the eSign-PIN if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and not blocked Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the old and the new password 3. The correct old and the new eSign-PIN are entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the

	CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 69 82 8E 08 <MAC> 69 82' 7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the old and new password 3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): old eSign-PIN || new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA with an error code which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.3.2
Purpose	Check that the reader aborts **changing the eSign-PIN** if the password which is used for the PACE protocol doesn't provide the required permits. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID **PIN** which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig

Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and not blocked Successful PACE protocol with **PIN**, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the old and the new password 3. The correct old and the new eSign-PIN are entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 69 82 8E 08 <MAC> 69 82' 7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the old and new password 3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): old eSign-PIN || new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA with an error code which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: '69 82'

Post processing	Reset of eCard and reader

Test ID	R_Sig_7.4.1
Purpose	Check that the reader aborts changing the eSign-PIN if the LT returns an error code to the command Change Reference Data. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.2
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and not blocked Successful PACE protocol with PIN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls PC/SC function FEATURE_MODIFY_PIN_DIRECT of the signature terminal. 2. The signature terminal requests for entering the old and the new password 3. The correct old and the new eSign-PIN are entered successfully into the signature terminal 4. The signature terminal generates the secured command APDU for the CHANGE REFERENCE DATA command and transmits it to the LT 5. The LT verifies the secured command APDU for the CHANGE REFERENCE DATA command 6. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for CHANGE REFERENCE DATA: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 83' (Authentication method blocked) or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 80' (Incorrect parameters in data field) or SW1 SW2 = '69 88' (Referenced data not found)

	7. The signature terminal sends the unsecured response APDU for the CHANGE REFERENCE DATA command back to the UT
Expected results	1. - 2. The signature terminal requests the entry for the old and new password 3. - 4. - 5. The secured command APDU for the CHANGE REFERENCE DATA which LT receives from the reader is built up as follows; Secured command APDU CHANGE REFERENCE DATA: '0C 24 00 P2 <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN with Data (in plaintext): old eSign-PIN || new eSign-PIN 6. - 7. The unsecured response APDU for the CHANGE REFERENCE DATA with an error code which UT receives from the reader is built up as follows: Unsecured response APDU CHANGE REFERENCE DATA: SW1 SW2 with SW1 SW2 = '69 83' (Authentication method blocked) or SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 80' (Incorrect parameters in data field) or SW1 SW2 = '69 88' (Referenced data not found)
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.5.1
Purpose	Check that the reader aborts resetting the retry counter of the eSign-PIN if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PUK which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.3; [TR-03110] B6.1, B.11.9
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2 Make certificates available in UT

	Status of the eSign-PIN is “operational” and blocked Use counter of the PUK is not zero, i.e. the processing of a further reset retry counter is possible Successful PACE protocol with PUK, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT sends the unsecured command RESET RETRY COUNTER to the signature terminal. Unsecured command APDU for Reset Retry Counter: '00 2C 03 P2 <L_c> <data is empty>'; P2: Reference to eSign-PIN 2. The signature terminal generates the secured command APDU for the RESET RETRY COUNTER command and transmits it to the LT 3. The LT verifies the secured command APDU for the RESET RETRY COUNTER command 4. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for RESET RETRY COUNTER: '99 02 69 82 8E 08 <MAC> 69 82' 5. The signature terminal sends the unsecured response APDU for the RESET RETRY COUNTER command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the RESET RETRY COUNTER which LT receives from the reader is built up as follows; Secured command APDU RESET RETRY COUNTER: '0C 2C 03 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN 4. - 5. The unsecured response APDU for the RESET RETRY COUNTER with an error code which UT receives from the reader is built up as follows: Unsecured response APDU RESET RETRY COUNTER: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.5.2
Purpose	Check that the reader aborts **resetting the retry counter** of the eSign-PIN if

	the password which is used for the PACE protocol doesn't provide the required permits. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID **CAN** which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.3; [TR-03110] B.6.1, B.11.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Status of the eSign-PIN is “operational” and blocked Use counter of the PUK is not zero, i.e. the processing of a further reset retry counter is possible Successful PACE protocol with **CAN**, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT sends the unsecured command RESET RETRY COUNTER to the signature terminal.: Unsecured command APDU for RESET RETRY COUNTER: '00 2C 03 P2 <L_c> <data is empty>'; P2: Reference to eSign-PIN 2. The signature terminal generates the secured command APDU for the RESET RETRY COUNTER command and transmits it to the LT 3. The LT verifies the secured command APDU for the RESET RETRY COUNTER command 4. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for RESET RETRY COUNTER: '99 02 69 82 8E 08 <MAC> 69 82' 5. The signature terminal sends the unsecured response APDU for the RESET RETRY COUNTER command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the RESET RETRY COUNTER which LT receives from the reader is built up as follows;

	Secured command APDU RESET RETRY COUNTER: '0C 2C 03 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN 4. - 5. The unsecured response APDU for the RESET RETRY COUNTER with an error code which UT receives from the reader is built up as follows: Unsecured response APDU RESET RETRY COUNTER: '69 82'
Post processing	Reset of eCard and reader

Test ID	R_Sig_7.6.1
Purpose	Check that the reader aborts resetting the retry counter of the eSign-PIN if the LT returns an error code to the command Reset Retry Counter. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PUK which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.3, A.2.3; [TR-03110] B.6.1, B.11.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PUK, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT sends the unsecured command RESET RETRY COUNTER to the signature terminal.: Unsecured command APDU for RESET RETRY COUNTER: '00 2C 03 P2 <L_c> <data is empty>'; P2: Reference to eSign-PIN 2. The signature terminal generates the secured command APDU for the RESET RETRY COUNTER command and transmits it to the LT 3. The LT verifies the secured command APDU for the RESET RETRY COUNTER command 4. The LT sends the secured response APDU with an error code back to the

	signature terminal; Secured Response APDU for RESET RETRY COUNTER: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 84' (Referenced data invalidated) or SW1 SW2 = '6A 88' (Referenced data not found) 5. The signature terminal sends the unsecured response APDU for the RESET RETRY COUNTER command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the RESET RETRY COUNTER which LT receives from the reader is built up as follows; Secured command APDU RESET RETRY COUNTER: '0C 2C 03 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN 4. - 5. The unsecured response APDU for the RESET RETRY COUNTER with an error code which UT receives from the reader is built up as follows: Unsecured response APDU RESET RETRY COUNTER: SW1 SW2 with SW1 SW2 = '69 84' (Referenced data invalidated) or SW1 SW2 = '6A 88' (Referenced data not found)
Post processing	Reset of eCard and reader

6.2.6.9 R_Sig_8 – Successful Termination of the Signature Function

Test ID	R_Sig_8.1.1
Purpose	Check correct execution of the termination of the eSign-PIN with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.4, A.2.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Reference eSign-PIN is retrieved from DF.CIA stored in LT

Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (PIN) is entered via PINPad of reader 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. UT sends unsecured command TERMINATE to the signature terminal: Unsecured command APDU for Terminate: '00 E6 10 P2 <L_c> <data empty>'; P2: Reference to eSign-PIN 7. The signature terminal generates the secured command APDU for the TERMINATE command and transmits it to the LT 8. The LT verifies the secured command APDU for the TERMINATE command 9. The LT sends the secured response APDU back to the signature terminal; Secured Response APDU for TERMINATE: '99 02 90 00 8E 08 <MAC> 90 00' 10. The signature terminal sends the unsecured response APDU for the TERMINATE command back to the UT
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. - 7. - 8. The secured command APDU for the TERMINATE command which LT receives from the reader is built up as follows; Secured command APDU TERMINATE: '0C E6 10 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN 9. - 10. The unsecured response APDU for the TERMINATE command which UT receives from the reader is built up as follows: Unsecured response APDU TERMINATE: '90 00'
Post processing	Reset of eCard and reader

Test ID	R_Sig_8.1.2
Purpose	Check correct execution of the termination of the private signature key with a signature terminal. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”.

	The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.4, A.2.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT
Test scenario	1. UT calls PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Generate qualified electronic signature” • <PIN>: empty, because password (PIN) is entered via PINPad of reader 2. Entering the correct PIN into the reader 3. LT receives command APDUs from the test object reader and sends response APDUs back to the reader as described in chapters 5.3.1, 5.3.2 and 5.3.3. The return codes in all response APDUs are positive ('90 00'). In case of a signature terminal all authentication protocols (PACE, TA and CA) are completely performed within EstablishPACEChannel between the reader and LT. 4. UT receives OutBuffer of ScardControl (see chapter 5.2.4). 5. The execution steps for the selection of the eSign application have to be performed as described in chapter 5.3.4: UT sends command APDUs to the test object reader via PC/SC (without SM). LT receives command APDUs (with SM) from the reader and sends response APDUs (with SM) back to the reader. UT receives response APDUs from the reader via PC/SC (without SM). The return codes in all response APDUs are positive ('90 00'). 6. UT sends unsecured command TERMINATE to the signature terminal: Unsecured command APDU for Terminate: '00 E6 21 00 <L_c> <DST with reference to the signature key>'; DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference> 7. The signature terminal generates the secured command APDU for the TERMINATE command and transmits it to the LT 8. The LT verifies the secured command APDU for the TERMINATE command 9. The LT sends the secured response APDU back to the signature terminal;

	Secured Response APDU for TERMINATE: '99 02 90 00 8E 08 <MAC> 90 00' 10. The signature terminal sends the unsecured response APDU for the TERMINATE command back to the UT
Expected results	1. - 2. - 3. The command APDUs for the authentication protocols which LT receives from the reader are built up as described in chapters 5.3.1, 5.3.2 and 5.3.3. For the PACE protocol with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' (PIN). • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 4. For the OutBuffer of ScardControl, that UT receives (see chapter 5.2.4), the following holds: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the reader and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. 6. - 7. - 8. The secured command APDU for the TERMINATE command which LT receives from the reader is built up as follows; Secured command APDU TERMINATE: '0C E6 21 00 <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted DST with reference to the private signature key: DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference> 9. - 10. The unsecured response APDU for the TERMINATE command which UT receives from the reader is built up as follows:

	Unsecured response APDU TERMINATE: '90 00'
Post processing	Reset of eCard and reader

6.2.6.10 R_Sig_9 – Abort Termination of the Signature Function

Test ID	R_Sig_9.1.1_template
Purpose	Check that the reader aborts termination of the signature function if the signature terminal does **not** have a certificate with the access right “Generate qualified electronic signature”. The test case covers the two variants - termination of the eSign-PIN and - termination of the private signature key. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal **without** access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.4, A.2.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 2; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function (see table 57) bysending unsecured command TERMINATE. 2. The signature terminal generates the secured command APDU for the TERMINATE command and transmits it to the LT 3. The LT verifies the secured command APDU for the TERMINATE command 4. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for TERMINATE:

	'99 02 69 82 8E 08 <MAC> 69 82' 5. The signature terminal sends the unsecured response APDU for the TERMINATE command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the TERMINATE command which LT receives from the reader is built up as described in table 57. 4. - 5. The unsecured response APDU with an error code for the TERMINATE command which UT receives from the reader is built up as follows: Unsecured response APDU TERMINATE: '69 82'
Post processing	Reset of eCard and reader

Test case	*Function*	*Test scenario (Step 1)*	*Expected results (Step 3)*
R_Sig_9.1.1a	Termination of the eSign-PIN.	Unsecured command APDU for TERMINATE the eSign-PIN: '00 E6 10 P2 <Lc> <data empty>' with P2: Reference to eSign-PIN;	Secured command APDU TERMINATE the eSign-PIN: '0C E6 10 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN
R_Sig_9.1.1b	Termination of the private signature key.	Unsecured command APDU for TERMINATE the private signature key: '00 E6 21 00 <L_c> <DST with reference to the signature key>' with DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>	Secured command APDU TERMINATE the private signature key: '0C E6 21 00 <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted DST with reference to the private signature key: DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>

Table 57: Test case R_Sig_9.1.1

Test ID	R_Sig_9.1.2_template
Purpose	Check that the reader aborts **termination of the signature function** if the

	password which is used for the PACE protocol doesn't provide the required permits. The test case covers the two variants - termination of the eSign-PIN and - termination of the private signature key. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID **CAN** which is entered via the PINPad of the signature terminal.
References	[TR-03117], 4.4.4, A.2.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with **CAN**, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function (see table 58) by sending unsecured command TERMINATE to the signature terminal. 2. The signature terminal generates the secured command APDU for the TERMINATE command and transmits it to the LT 3. The LT verifies the secured command APDU for the TERMINATE command 4. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for TERMINATE: '99 02 69 82 8E 08 <MAC> 69 82' 5. The signature terminal sends the unsecured response APDU for the TERMINATE command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the TERMINATE command which LT receives from the reader is built up as described in table 58. 4. -

	5. The unsecured response APDU with an error code for the TERMINATE command which UT receives from the reader is built up as follows: Unsecured response APDU TERMINATE: '69 82'
Post processing	Reset of eCard and reader

Test case	*Function*	*Test scenario (Step 1)*	*Expected results (Step 3)*
R_Sig_9.1.2a	Termination of the eSign-PIN.	Unsecured command APDU for Terminate the eSign-PIN: '00 E6 10 P2 <L_c> <data empty>' with P2: Reference to eSign-PIN;	Secured command APDU TERMINATE the eSign-PIN: '0C E6 10 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN
R_Sig_9.1.2b	Termination of the private signature key.	Unsecured command APDU for TERMINATE the private signature key: '00 E6 21 00 <L_c> <DST with reference to the signature key>' with DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>	Secured command APDU TERMINATE the private signature key: '0C E6 21 00 <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted DST with reference to the private signature key: DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>

Table 58: Test case R_Sig_9.1.2

Test ID	R_Sig_9.2.1_template
Purpose	Check that the reader aborts termination of the signature function if the LT returns an error code to the command Terminate. The test case covers the two variants - termination of the eSign-PIN and - termination of the private signature key. PACE, TA and CA between signature terminal and LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is entered via the PINPad of the signature terminal.

References	[TR-03117], 4.4.4, A.2.6
Profiles	R_PACE AND R_TA AND R_CA AND R_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA between signature terminal and LT with establishment of a trusted channel The eSign application was successfully selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function (see table 59) by sending unsecured command TERMINATE to the signature terminal. 2. The signature terminal generates the secured command APDU for the TERMINATE command and transmits it to the LT 3. The LT verifies the secured command APDU for the TERMINATE command 4. The LT sends the secured response APDU with an error code back to the signature terminal; Secured Response APDU for TERMINATE: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found) 5. The signature terminal sends the unsecured response APDU for the TERMINATE command back to the UT
Expected results	1. - 2. - 3. The secured command APDU for the Terminate command which LT receives from the reader is built up as described in table 59. 4. - 5. The unsecured response APDU with an error code for the TERMINATE command which UT receives from the reader is built up as follows: Unsecured response APDU TERMINATE: SW1 SW2 with SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found)
Post processing	Reset of eCard and reader

Test case	*Function*	*Test scenario (Step 1)*	*Expected results (Step 3)*
R_Sig_9.2.1a	Termination of the eSign-PIN.	Unsecured command APDU for Terminate the eSign-PIN: '00 E6 10 P2 <L_c> <data empty>' with P2: Reference to eSign-PIN;	Secured command APDU TERMINATE the eSign-PIN: '0C E6 10 P2 <L_c> 8E 08 <MAC> 00'; P2: Reference of the eSign-PIN
R_Sig_9.2.1b	Termination of the private signature key.	Unsecured command APDU for TERMINATE the private signature key: '00 E6 21 00 <L_c> <DST with reference to the signature key>' with DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>	Secured command APDU TERMINATE the private signature key: '0C E6 21 00 <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted DST with reference to the private signature key: DST with reference to the signature key: B6 <L_{B6}> 84 <L_{84}> <private key reference>

Table 59: Test case R_Sig_9.2.1

6.3 Test Cases for Test Object Terminal Software

6.3.1 PACE

6.3.1.1 TS_PACE_1 – Correct Execution of PACE Protocol

Test ID	TS_PACE_1.1.1_template
Purpose	Check correct execution of PACE protocol in the terminal software. Use certificate with role inspection system. The test is executed with the passwords CAN and MRZ-Password.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates and the password (CAN resp. MRZ-Password) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 60) and certificates (according to Profile) to terminal

	software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 60. • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (inspection system). • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Passwort	<PIN-ID>
TS_PACE_1.1.1a	Use PACE with MRZ	'01'
TS_PACE_1.1.1b	Use PACE with CAN	'02'

Table 60: Test case TS_PACE_1.1.1

Test ID	TS_PACE_1.1.2_template
Purpose	Check correct execution of PACE protocol in the terminal software. Use certificate with role authentication terminal. The test is executed with the passwords PIN and CAN.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE

Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password (PIN, CAN) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 61) and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 61. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Passwort	<PIN-ID>
TS_PACE_1.1.2a	Use PACE with CAN	'02'
TS_PACE_1.1.2b	Use PACE with PIN	'03'

Table 61: Test case TS_PACE_1.1.2

Test ID	TS_PACE_1.1.3_template
Purpose	Check correct execution of PACE protocol in the terminal software. Use certificate with role signature terminal. The test is executed with the passwords CAN, PIN and PUK.
References	[TR-03110], 4.2, C.4.3
Profiles	TS_PACE
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates and the password (CAN, PIN resp. PUK) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 62) and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 62. • The role in <OID-Role> in command APDU to MSE: Set AT is '03' (signature terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 30, No. 1. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Passwort	<PIN-ID>
TS_PACE_1.1.3a	Use PACE with CAN	'02'
TS_PACE_1.1.3b	Use PACE with PIN	'03'
TS_PACE_1.1.3c	Use PACE with PUK	'04'

Table 62: Test case TS_PACE_1.1.3

Test ID	TS_PACE_1.2.1_template
Purpose	Check correct execution of PACE protocol in the terminal software. Use an unauthenticated terminal. The test is executed with the passwords CAN, PIN and PUK.
References	[TR-03110], 3.5.1, 4.2
Profiles	TS_PACE
Preconditions	
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 63) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT as defined in table 63 • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty) . 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Passwort	<PIN-ID>
TS_PACE_1.2.1a	Use PACE with CAN	'02'
TS_PACE_1.2.1b	Use PACE with PIN	'03'
TS_PACE_1.2.1c	Use PACE with PUK	'04'

Table 63: Test case TS_PACE_1.2.1

Test ID	TS_PACE_1.3.1_template
Purpose	Check correct execution of PACE protocol in the terminal software. Use exact one PACEInfo with standardized domain parameter which is supported by the TS (see Implementation Conformance Statement). Don't use a PACEDomainParameterInfo within EF.CardAccess The test has to be executed for each supported PACE algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully. The test is executed with the password CAN and PIN.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 64. Make certificates and the password CAN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 64) and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '02', because CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 64 • The <access rights> in command APDU to MSE: Set AT as described in table 64 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_1.3.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_1.3.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_1.3.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 64: Test case TS_PACE_1.3.1

Test ID	TS_PACE_1.3.2_template
Purpose	Check correct execution of PACE protocol in the terminal software, if LT supports several algorithms for PACE. Use three different PACEInfo in EF.CardAccess with different algorithms and standardized domain parameters. Use algorithms and domain parameters which are supported by the TS (chapter 4.2 Cryptographic algorithms). Don't use a PACEDomainParameterInfo within EF.CardAccess. The test is executed with the password CAN and PIN.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 65. Make certificates and the password CAN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password

	(according to table 65) and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', because CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 65. • The <access rights> in command APDU to MSE: Set AT as described in table 65. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_1.3.2a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_1.3.2b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN.	'02'
TS_PACE_1.3.2c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 65: Test case TS_PACE_1.3.2

Test ID	TS_PACE_1.3.3_template

Purpose	Check correct execution of PACE protocol in the terminal software, if LT supports proprietary domain parameters for PACE. Use one PACEInfo with standardized domain parameters and one PACEInfo with proprietary domain parameters in PACEDomainParameterInfo within EF.CardAccess. The test is executed with the password CAN and PIN.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 66. Make certificates and the password CAN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password (according to table 66) and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', because CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 66. • The <access rights> in command APDU to MSE: Set AT as described in table 66. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_1.3.3a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_1.3.3b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN.	'02'
TS_PACE_1.3.3c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 66: Test case TS_PACE_1.3.3

6.3.1.2 TS_PACE_2 – Abort of PACE Protocol because of Internal LT Error

Test ID	TS_PACE_2.1.1_template
Purpose	Check that terminal software aborts PACE protocol when LT derives cryptographic key from CAN incorrectly. The test is executed with the passwords CAN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 67. Make certificates and the password (CAN) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT derives an *incorrect encryption key* K_pi from the password stored in the eCard. It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '02', CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 67. • The <access rights> in command APDU to MSE: Set AT as described in table 67. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_2.1.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
TS_PACE_2.1.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 67: Test case TS_PACE_2.1.1

Test ID	TS_PACE_2.1.2
Purpose	Check that terminal software aborts PACE protocol when LT derives cryptographic key from PIN incorrectly. The test is executed with the passwords PIN.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the password PIN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT derives an *incorrect encryption key* K_pi from the password stored in the eCard. It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response

	APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03', PIN is used • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_PACE_2.2.1_template
Purpose	Check that terminal software aborts PACE protocol when LT returns error code to command MSE: Set AT. The test is executed with the passwords CAN and PIN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 68. Make certificates and the password password (according to table 68) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT sends the *negative* return code '6A 88' in answer message to MSE: Set AT back to the terminal software. It is expected that the terminal software aborts protocol execution after receiving this return code. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. -

	2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 68. • The <access rights> in command APDU to MSE: Set AT as described in table 68. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the terminal software must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_2.2.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_2.2.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN.	'02'
TS_PACE_2.2.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 68: Test case TS_PACE_2.2.1

Test ID	TS_PACE_2.3.1_template
Purpose	Check that terminal software aborts PACE protocol when LT transmits incorrect data for mapping function Map in answer to card command General Authenticate (Step 2). The test is executed with the CAN and PIN.
References	[TR-03110], 4.2, B.1, C.4.2
Profiles	TS_PACE
Preconditions	Certificates as described in table 69. Make certificates and the password (according to table 69) available in UT

Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT receives card command General Authenticate (Step 2) with Mapping Data of the terminal from the terminal software via CLI. LT sends *incorrect* Mapping data (incremented by 1) of the eCard (D_PICC) in the response APDU to card command General Authenticate (Step 2) back to the terminal software. It is accepted that the terminal software aborts protocol execution after receiving the incorrect mapping data in response APDU to General Authenticate (Step 2). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 69. • The <access rights> in command APDU to MSE: Set AT as described in table 69. • The command APDUs for General Authenticate (Step 3, 4) may be missing. This is the case if the terminal software aborts communication to LT when detecting the specified error. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_2.3.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_2.3.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN	'02'
TS_PACE_2.3.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 69: Test case TS_PACE_2.3.1

Test ID	TS_PACE_2.4.1 deleted in Version 1.1

Test ID	TS_PACE_2.5.1_template
Purpose	Check that terminal software aborts PACE protocol when LT generates an incorrect ephemeral key pair. The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificates as described in table 70. Make certificates and the password (according to table 70) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT generates an *incorrect* ephemeral key pair (SKeph_PICC, PKeph_PICC). It is accepted that the terminal software aborts protocol execution after detecting the erroneous PKeph_PICC in response APDU to General Authenticate (Step 3). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. -

	2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 70. • The <access rights> in command APDU to MSE: Set AT as described in table 70. • The command APDU for General Authenticate (Step 4) may be missing. This is the case if the terminal software aborts communication to LT when detecting the specified error. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_2.5.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_2.5.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN.	'02'
TS_PACE_2.5.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 70: Test case TS_PACE_2.5.1

Test ID	TS_PACE_2.6.1_template
Purpose	Check that terminal software aborts PACE protocol when LT computes key data for secure messaging (SM) incorrectly. The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificates as described in table 71. Make certificates and the password (according to table 71) available in UT

Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT computes key material *incorrectly*, i. e. *different from* KA(SKeph_PICC, PKeph_PCD, Deph), extracts Kmac from key material and checks that T_PCD is *different from* MAC(Kmac, PKeph_PICC). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 71. • The <access rights> in command APDU to MSE: Set AT as described in table 71. • The command APDU for General Authenticate (Step 4) may be missing. This is the case if the terminal software aborts communication to LT when detecting the specified error. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_2.6.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using password CAN.	'01'
TS_PACE_2.6.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using password CAN.	'02'
TS_PACE_2.6.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using password PIN.	'02'

Table 71: Test case TS_PACE_2.6.1

6.3.1.3 TS_PACE_3 – Abort of PACE Protocol because of Incorrect LT Data

Test ID	TS_PACE_3.1.1_template
Purpose	Check that terminal software aborts PACE protocol when LT transmits incorrect PACE parameters (inconsistent data in these parameters). The test is executed with the passwords CAN and PIN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 72. Make certificates and the password (according to table 72) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT sends EF.CardAccess in response APDU to command Read Binary with the following change: In SecurityInfo *PACEInfo* change length byte of tag "version" from '01' to '02'. 30 0F 06 0A 04 00 7F 00 07 02 02 04 02 02 02 **02** 01 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The Terminal software must abort communication to LT after receiving the inconsistent data in response APDU to Read Binary. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.1.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.1.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_3.1.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 72: Test case TS_PACE_3.1.1

Test ID	TS_PACE_3.1.2_template
Purpose	Check that terminal software aborts PACE protocol when LT transmits incorrect PACE parameters (standardized domain parameter identifier contained in these parameters which is not supported by the terminal software). The test is executed with the passwords CAN and PIN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 73. Make certificates and the password (according to table 73) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT sends data from EF.CardAccess with standardized domain parameters in PACEInfo in response APDU to command Read Binary. As parameterID in PACEInfo use a domain parameter identifier which is not supported by the terminal software (see Implementation Conformance Statement) 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The command APDUs for MSE: Set AT, General Authenticate (Step 1, 2, 3, 4) are missing, since the terminal software must abort communication to LT after receiving the response APDU to Read Binary. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.1.2a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.1.2b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_3.1.2c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 73: Test case TS_PACE_3.1.2

Test ID	TS_PACE_3.2.1_template
Purpose	Check that terminal software aborts PACE protocol when LT transmits an incorrect cryptogram in response message to card command General Authenticate (Step 1). The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, B.1, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table74. Make certificates and the password (according to table 74) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT computes the encrypted nonce encNonce *incorrectly,* i. e. *different from* E(K_pi, s). It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are

	built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table74. • The <access rights> in command APDU to MSE: Set AT as described in table74. • The command APDUs for General Authenticate (Step 2, 3, 4) may be missing. This is the case if the terminal software aborts communication to LT when detecting the specified error. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.2.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.2.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_3.2.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 74: Test case TS_PACE_3.2.1

Test ID	TS_PACE_3.3.1_template
Purpose	Check that terminal software aborts PACE protocol when LT transmits an ephemeral public key in response message to card command General Authenticate (Step 3) that coincides with the ephemeral public key transmitted by the terminal software in the command APDU to this command. The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, B.1, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 75. Make certificates and the password (according to table 75) available in UT

Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT generates the ephemeral key pair correctly but transmits in response message to card command General Authenticate (Step 3) the *ephemeral public key received from the terminal software*. It is expected that the terminal software aborts protocol execution after receiving the ephemeral public key in response APDU to General Authenticate (Step 3). 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 75. • The <access rights> in command APDU to MSE: Set AT as described in table 75. • The command APDU for General Authenticate (Step 4) must be missing. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.3.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.3.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_3.3.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 75: Test case TS_PACE_3.3.1

Test ID	TS_PACE_3.4.1_template

Purpose	Check that terminal software aborts PACE protocol when LT transmits an incorrect cryptogram, that it has generated with derived SM key data, in response message to card command General Authenticate (Step 4). The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table76. Make certificates and the password (according to table 76) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT computes authentication token T_PICC *incorrectly*, i. e. *different from* MAC(Kmac, PKeph_PCD). The return codes in the response APDUs of all commands are positive ('90 00'). 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table76. • The <access rights> in command APDU to MSE: Set AT as described in table76. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.4.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.4.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'
TS_PACE_3.4.1c	Certificate with role authentication terminal and access rights according to 29, No. 4; CA certificates according to 32, No. 1, 2 using the PIN.	'02'

Table 76: Test case TS_PACE_3.4.1

Test ID	TS_PACE_3.5.1_template
Purpose	Check that terminal software aborts PACE protocol when EF.CardAccess indicate an unsupported algorithm but LT use a supported one. The values for PACEInfo and PACEDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. But for the generation of T_PICC LT uses another algorithm. The test is executed with the CAN and the PIN.
References	[TR-03110], 4.2, A.1.1.1, A.1.2, A.3, B.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 77. Make certificates and the password (according to table 77) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits CAN and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. EF.CardAccess should indicate only one unsupported algorithm. LT computes authentication token T_PICC *with a supported algorithm*. The return codes in the response APDUs of all commands are positive ('90 00'). 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions:

	• The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used or '03' for PIN. • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 77. • The <access rights> in command APDU to MSE: Set AT as described in table 77. 3. UT receives an information from terminal software about protocol abort because of an unsupported algorithm
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_3.5.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 using the CAN.	'01'
TS_PACE_3.5.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 using the CAN.	'02'

Table 77: Test case TS_PACE_3.5.1

6.3.1.4 TS_PACE_4 – Abort of PACE Protocol because of Incorrect UT Data

Test ID	TS_PACE_4.1.1_template
Purpose	Check that terminal software aborts PACE protocol when an incorrect password (CAN) is transmitted.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificates as described in table 78. Make certificates and the incorrect password (CAN) available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the wrong password in the terminal software. It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are

	positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' (CAN is used) • The role in <OID-Role> in command APDU to MSE: Set AT as described in table 78. • The <access rights> in command APDU to MSE: Set AT as described in table 78. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Terminal type	<OID-Role>
TS_PACE_4.1.1a	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2	'01'
TS_PACE_4.1.1b	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2	'02'

Table 78: Test case TS_PACE_4.1.1

Test ID	TS_PACE_4.1.2
Purpose	Check that terminal software aborts PACE protocol when an incorrect password PIN is transmitted. Use certificate with role authentication terminal. The test is executed with the incorrect password (PIN).
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates and the incorrect password PIN available in UT

Test scenario	1. UT starts PACE protocol in terminal software and transmits an incorrect password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the wrong password in the terminal software. It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (Authentication Terminal). • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 1. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_PACE_4.1.3
Purpose	Check that terminal software aborts PACE protocol when an incorrect MRZ is transmitted. Use certificate with role inspection system. The test is executed with an incorrect MRZ.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 Make certificates and an incorrect MRZ available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits an incorrect MRZ and certificates (according to Profile) to terminal software.

	2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT derives an encryption key K_pi from the password stored in the eCard that differs from the encryption key derived from the wrong password in the terminal software. It is accepted that the terminal software aborts protocol execution after receiving the encrypted nonce in response APDU to General Authenticate (Step 1). The return codes in the response APDUs of all commands apart from General Authenticate (Step 4) are positive ('90 00'). The return code in the response APDU to General Authenticate (Step 4) (if this message is sent) is an error code ('63 XX') indicating that the authentication has failed. Moreover, it is accepted that the terminal software aborts protocol execution after it has received the transport PIN. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions (only if these commands have been transmitted by the terminal software): • The <PIN-ID> in command APDU to MSE: Set AT is '01 (MRZis used). • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (Inspection System). • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_PACE_4.2.1
Purpose	Check that terminal software aborts PACE protocol when an incorrect combination of password-ID and CHAT is transmitted by UT. Use certificate with role authentication terminal, that is *not authorized to use the CAN*. The test is executed with the password CAN.
References	[TR-03110], 4.2, C.4.2
Profiles	TS_PACE
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 2 (*not authorized to use CAN*); CA certificates according to 32, No. 1, 2

	Make certificates and the CAN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT sends the *negative* return code '6A 80', indicating that the terminal type referenced by the CHAT is not authorized to use the referenced password (CAN), in answer message to MSE: Set AT back to the terminal software. It is expected that the terminal software aborts protocol execution after receiving this return code. Alternatively, the reader may detect the inconsistence between CHAT and password and thus does not send MSE: Set AT. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02' since CAN is used, if this command is sent. • The role in <OID-Role> in command APDU to MSE: Set AT is '02' (authentication terminal), if this command is sent. • The <access rights> in command APDU to MSE: Set AT are as defined in 29, No. 2, if this command is sent. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the terminal software must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_PACE_4.2.2
Purpose	Check that terminal software aborts PACE protocol when an incorrect combination of password-ID and CHAT is transmitted by UT. Use certificate with role inspection system (not authorized to use PIN) The test is executed with the PIN.
References	[TR-03110], 4.2, C.4.1
Profiles	TS_PACE
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2

	Make certificates and the PIN available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits password and certificates (according to Profile) to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. LT sends the *negative* return code '6A 80', indicating that the terminal type referenced by the CHAT is not authorized to use the referenced password (PIN), in answer message to MSE: Set AT back to the terminal software. It is expected that the terminal software aborts protocol execution after receiving this return code. Alternatively, the reader may detect the inconsistence between CHAT and password and thus does not send MSE: Set AT. 3. UT receives an information from terminal software about protocol abort.
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '03' since PIN is used, if this command is sent. • The role in <OID-Role> in command APDU to MSE: Set AT is '01' (inspection system), if this command is sent. • The <access rights> in command APDU to MSE: Set AT are as defined in 28, No. 1, if this command is sent. • The command APDUs for General Authenticate (Step 1, 2, 3, 4) are missing, since the terminal software must abort communication to LT after receiving the specified error code to MSE: Set AT. 3. UT receives an information from terminal software about protocol abort.
Post processing	Reset of eCard, reader and terminal software

6.3.2 Terminal Authentication

For all test cases for terminal authentication the precondition is to successful establish a PACE channel. Where no terminal role and password type is defined, these parameters can be chosen from these which are supported by the terminal software (see chapter 4.3 Terminal type).

If no terminal type and/or password is defined, the priority of the terminal type to use in the test cases are. AT, IS and ST. The priority of the password to use in the in the test cases are CAN, PIN and MRZ. That does mean, first select the first supported terminal type then select the first supported password type which is supported in combination with the terminal type.
The used terminal type and password must be documented in the test report.

If this test unit is used for testing eID clients according to BSI TR-03124-1, then all certificates with role authentication terminal and access rights according to table table 29, No. **1** SHALL be replaced with the certificates with access rights according to table 29, No. **4**.

6.3.2.1 TS_TA_1 – Correct Execution of Terminal Authentication Protocol

Test ID	TS_TA_1.1.1
Purpose	Check terminal software for correct execution of terminal authentication protocol. Use certificate with role inspection system.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE protocol has been executed successfully with CAN in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_1.1.2
Purpose	Check terminal software for correct execution of terminal authentication protocol. Use certificate with role authentication terminal.

References	[TR-03110], 4.4, C.4.2
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE protocol has been executed successfully with PIN in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_1.1.3
Purpose	Check terminal software for correct execution of terminal authentication protocol. Use certificate with role signature terminal.
References	[TR-03110], 4.4, C.4.3
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 The PACE protocol has been executed successfully with CAN in the terminal software. After PACE protocol especially the following eCard data are available in UT:

	- PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_1.2.1
Purpose	Check terminal software for correct execution of terminal authentication protocol. Use of several algorithms for terminal authentication. The value for TerminalAuthenticationInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. The test has to be executed for each supported terminal authentication algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.4, A.1.1.3, A.6, B.3
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard)

	A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00').
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

6.3.2.2 TS_TA_2 – Abort because of Inconsistent Data in Terminal Software

Test ID	TS_TA_2.1.1
Purpose	Check that terminal software aborts terminal authentication if the OID in CHAT of the terminal certificate used in the terminal authentication protocol is different from the OID in CHAT used in the PACE protocol.
References	[TR-03110], 4.4, B.3, C.4.1; [TR-03119], D.3
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. Use OID for authentication terminal in PACE CHAT. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software

	and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The OID (inspection system) in the CHAT of the terminal certificate submitted by UT in the command APDU to PSO: Verify Certificate is different from the OID (authentication terminal) in CHAT used in the PACE protocol.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. UT receives response APDUs to the commands MSE: Set DST and PSO: Verify Certificate for all certificates of the certificate chain which are correctly coded without SM. The response APDUs to MSE: Set DST and PSO: Verify Certificate (for the CV certificates) coincide apart from SM with the response APDU sent by LT directly before. The response APDU to PSO: Verify for the terminal certificate received by UT consists of the negative return code '69 85'. It is expected that the terminal software aborts protocol execution after sending this return code.
Post processing	Reset of eCard, reader and terminal software

6.3.2.3 TS_TA_3 – Abort because of Internal LT Error

Test ID	TS_TA_3.1.1
Purpose	Check that terminal software aborts terminal authentication when LT returns error code to command MSE: Set DST (setting public key for certificate verification)
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without

	SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The only command APDU sent by UT is MSE: Set DST. The return code in response APDU to MSE: Set DST is negative ('6A 80').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The only command APDU received at LT and the only response APDU received at UT are to MSE: Set DST.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_3.2.1
Purpose	Check that terminal software aborts terminal authentication when LT returns error code to command PSO: Verify Certificate when verifying the terminal certificate.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The command APDUs sent by UT are MSE: Set DST and the commands PSO: Verify Certificate for all certificates of the certificate chain. The return code in response APDU to PSO: Verify Certificate for the terminal certificate

	is negative ('6A 80'), the return codes to all other commands sent by UT are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The only command APDUs received at LT and the only response APDUs received at UT are to MSE: Set DST and to PSO: Verify Certificate (for all certificates in the certificate chain).
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_3.3.1
Purpose	Check that terminal software aborts terminal authentication when LT returns error code to command MSE: Set AT (transmitting parameters for terminal authentication to LT).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The command APDUs sent by UT are MSE: Set DST, the commands PSO: Verify Certificate for all certificates of the certificate chain and MSE: Set AT. The return code in response APDU to MSE: Set AT is negative ('6A 88'), the return codes to all other commands sent by UT are positive ('90 00').

Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The command APDUs received at LT and the response APDUs received at UT are to MSE: Set DST, to PSO: Verify Certificate (for all certificates in the certificate chain) and to MSE: Set AT.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_3.4.1
Purpose	Check that terminal software aborts terminal authentication when LT returns error code to command Get Challenge (random number for terminal authentication).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The command APDUs sent by UT are MSE: Set DST, the commands PSO: Verify Certificate for all certificates of the certificate chain, MSE: Set AT and Get Challenge. The return code in response APDU to Get Challenge is negative ('ZZ ZZ', the actual value depends on the eCard operating system), the return codes to all other commands sent by UT are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly

	before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The command APDUs received at LT and the response APDUs received at UT are to MSE: Set DST, to PSO: Verify Certificate (for all certificates in the certificate chain), to MSE: Set AT and to Get Challenge.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_3.5.1
Purpose	Check that terminal software aborts terminal authentication when LT returns error code to command External Authenticate when checking signed data from UT in terminal authentication protocol.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The command APDUs sent by UT are MSE: Set DST, the commands PSO: Verify Certificate for all certificates of the certificate chain, MSE: Set AT, Get Challenge and External Authenticate. The return code in response APDU to External Authenticate is negative ('69 85'), the return codes to all other commands sent by UT are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct.

	Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The command APDUs received at LT and the response APDUs received at UT are to MSE: Set DST, to PSO: Verify Certificate (for all certificates in the certificate chain), to MSE: Set AT, to Get Challenge and to External Authenticate.
Post processing	Reset of eCard, reader and terminal software

6.3.2.4 TS_TA_4 – Abort because of Secure Messaging Error

Test ID	TS_TA_4.1.1
Purpose	Check that the terminal software aborts terminal authentication if it does not receive SM data objects from LT in response APDU to command Get Challenge.
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software. LT sends response APDUs (with SM) back to the terminal software for the commands MSE: Set DST, PSO: Verify Certificate for all certificates of the certificate chain and MSE: Set AT. LT sends response APDUs without SM back to the terminal software for the command Get Challenge. UT receives response APDUs from the terminal software (without SM). The return codes in response APDU to all commands sent by UT are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct.

	Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The command APDUs received at LT are to MSE: Set DST, to PSO: Verify Certificate (for all certificates in the certificate chain), to MSE: Set AT and to Get Challenge. The response APDUs received at UT are to MSE: Set DST, to PSO: Verify Certificate (for all certificates in the certificate chain) and to MSE: Set AT. It is expected that the terminal software detects the missing SM in response APDU to Get Challenge received from LT and aborts command execution. Thus UT does not receive any response APDU to Get Challenge.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_4.2.1
Purpose	Check that the terminal software aborts terminal authentication if it receives incorrect SM data objects from LT in response APDU to command Get Challenge (wrong tag '98' instead of '99' for the processing status).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal and access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDU Get Challenge to the test object terminal software (without SM). LT receives this command APDU (with SM) from the terminal software. LT sends response APDUs with incorrect SM (wrong tag '98' instead of '99' for the processing status) back to the terminal software for this command. UT does not receive any response APDU from the terminal software. The return code in response APDU to Get Challenge is positive ('90 00').
Expected results	The command APDU to Get Challenge received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the

	terminal software and received by LT is correct. It is expected that the terminal software detects the incorrect SM in response APDU to Get Challenge received from LT and aborts command execution. Thus UT does not receive a response APDU to any command in terminal authentication.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_TA_4.2.2
Purpose	Check that the terminal software aborts terminal authentication if it receives incorrect SM data objects from LT in response APDU to command Get Challenge (wrong cryptogram in the MAC data object).
References	[TR-03110], 4.4, B.3, C.4.1
Profiles	TS_PACE AND TS_TA
Preconditions	Certificate with universal access rights. The PACE protocol has been executed successfully in the terminal software. After PACE protocol especially the following eCard data are available in UT: - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) - ID_PICC (identification of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.2: UT sends command APDU Get Challenge to the test object terminal software (without SM). LT receives this command APDU (with SM) from the terminal software. LT sends response APDUs with incorrect SM (wrong cryptogram in the MAC data object) back to the terminal software for this command. UT does not receive any response APDU from the terminal software. The return code in response APDU to Get Challenge is positive ('90 00').
Expected results	The command APDU to Get Challenge received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. It is expected that the terminal software detects the incorrect SM in response APDU to Get Challenge received from LT and aborts command execution. Thus UT does not receive a response APDU to any command in terminal authentication.

Post processing	Reset of eCard, reader and terminal software

6.3.3 Chip Authentication

For all test cases for chip authentication the precondition is to successful establish a PACE channel. Where no terminal role and password type is defined, these parameters can be chosen from these which are supported by the terminal software (see chapter 4.3 Terminal type).

If no terminal type and/or password is defined, the priority of the terminal type to use in the test cases are. AT, IS and ST. The priority of the password to use in the in the test cases are CAN, PIN and MRZ. That does mean, first select the first supported terminal type then select the first supported password type which is supported in combination with the terminal type.
The used terminal type and password must be documented in the test report.

If this test unit is used for testing eID clients according to BSI TR-03124-1, then all certificates with role authentication terminal and access rights according to table table 29, No. **1** SHALL be replaced with the certificates with access rights according to table 29, No. **4**.

6.3.3.1 TS_CA_1 – Correct Execution of Chip Authentication Protocol

Test ID	TS_CA_1.1.1
Purpose	Check for correct execution of chip authentication protocol in the terminal software. Use certificate with role inspection system.
References	[TR-03110], 4.3, C.4.1
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with password-ID CAN. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software . UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure

	messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_1.1.2
Purpose	Check for correct execution of chip authentication protocol in the terminal software. Use certificate with role authentication terminal.
References	[TR-03110], 4.3, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with password-IDs PIN. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software . UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct.

	Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_1.1.3
Purpose	Check for correct execution of chip authentication protocol in the terminal software. Use certificate with role signature terminal.
References	[TR-03110], 4.3, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 The PACE and terminal authentication protocols have been executed successfully with password-IDs CAN. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software . UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_1.2.1
Purpose	Check for correct execution of chip authentication protocol in the terminal software. Use of several algorithms for the chip authentication protocol. The values for ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm and one static key pair. This means that only one data object ChipAuthenticationPublicKeyInfo is available in EF.CardSecurity. The test has to be executed for each supported chip authentication algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.

Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_1.2.2
Purpose	Check for correct execution of chip authentication protocol in the terminal software. Use of several algorithms and multiple keys for the chip authentication protocol. This means that in EF.CardAccess two SecurityInfos with ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo are available and that EF.CardSecurity contains two public keys. For this the related algorithms are identical but the static key pairs are different. The test has to be executed for each supported chip authentication algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC_1 and PK_PICC_2 (public keys of the eCard) - keyId_1 and keyId_2 of the related static key pairs - D_PICC_1 and D_PICC_2 (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly

	before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

6.3.3.2 TS_CA_2 – Abort because of Internal LT Error

Test ID	TS_CA_2.1.1
Purpose	Check that terminal software aborts chip authentication when LT returns error code to command MSE: Set AT (transmitting parameters for chip authentication to LT).
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The only command APDU sent by UT is MSE: Set AT. The return code in response APDU to MSE: Set AT is negative ('6A 88').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The only command APDU received at LT and the only response APDU

	received at UT are to MSE: Set AT.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_2.2.1
Purpose	Check that terminal software aborts chip authentication when LT returns error code to command General Authenticate when verifying the ephemeral public key of the terminal software.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The command APDUs sent by UT are MSE: Set AT and General Authenticate. The return code in response APDU to General Authenticate is negative ('6A 80'), the return codes to MSE: Set AT is positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. The command APDUs received at LT and the response APDUs received at UT are to MSE: Set AT and to General Authenticate.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_2.3.1 deleted in version 1.1

6.3.3.3 TS_CA_3 – Abort because of Incorrect LT Data

The test cases in this chapter are not applicable if the the DUT is a eID-Client.

Test ID	TS_CA_3.1.1
Purpose	Check that terminal software aborts chip authentication when LT returns an incorrect random number in the response APDU to command General Authenticate.
References	[TR-03110], 4.3, B.2, C.4.1
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC = MAC(K_MAC, PKeph_PCD) correctly. LT transmits an incorrect r2_PICC', i. e. different from r2_PICC, to the terminal software. UT receives random number r2_PICC' and authentication token T_PICC in response APDU to General Authenticate. UT derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC' = KDF_MAC(K, r2_PICC') and computes MAC(K_MAC', PKeph_PCD).
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly

	before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. UT checks that the received T_PICC is different from the computed MAC(K_MAC', PKeph_PCD).
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_3.2.1 deleted in version 1.1

Test ID	TS_CA_3.3.1
Purpose	Check that terminal software aborts chip authentication when LT returns an incorrect cryptogram generated with a wrong algorithm in response APDU to command General Authenticate. The values for ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo within EF.CardAccess of the LT indicates that LT supports exactly one algorithm. But for the generation of T_PICC LT uses another algorithm. The test has to be executed for each algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully. The following substitutions have to be used (algorithm indicated in EF.CardAccess, algorithm used by LT): - id-CA-DH 1, id-CA-DH 2 - id-CA-DH 2, id-CA-DH 1 - id-CA-DH 3, id-CA-DH 1 - id-CA-DH 4, id-CA-DH 1 - id-CA-ECDH 1, id-CA-ECDH 2 - id-CA-ECDH 2, id-CA-ECDH 1 - id-CA-ECDH 3, id-CA-ECDH 1 - id-CA-ECDH 4, id-CA-ECDH 1
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed

	successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC (public key of the eCard) - D_PICC (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(SK_PICC, PKeph_PCD, D_PICC), key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC = MAC(K_MAC, PKeph_PCD) correctly. LT transmits an incorrect T_PICC', i. e. different from T_PICC, to the terminal software. UT receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. UT derives key material K = KA(SKeph_PCD, PK_PICC, D_PICC), keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. UT checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD).
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_CA_3.4.1
Purpose	Check that terminal software aborts chip authentication when LT returns an incorrect cryptogram in response APDU to command General Authenticate. The generation of the cryptogram is based on the correct algorithm but LT uses a wrong key pair. Use of several algorithms and multiple keys for the chip authentication

	protocol. LT stores two static key pairs for chip authentication. This means that in EF.CardAccess two data objects SecurityInfo with ChipAuthenticationInfo and ChipAuthenticationDomainParameterInfo are available and that EF.CardSecurity contains two public keys. For this the related algorithms are identical but the static key pairs are different. The test has to be executed for each algorithm specified in the manufacturer's conformance statement (chapter 4.2 Cryptographic algorithms). This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], 4.3, A.1.1.2, A.4, B.2
Profiles	TS_PACE AND TS_TA AND TS_CA
Preconditions	Certificate with universal access rights. The PACE and terminal authentication protocols have been executed successfully. After execution of these protocols especially the following eCard and terminal data are available in UT: - ephemeral terminal key pair (SKeph_PCD, PKeph_PCD) - PK_PICC_1 and PK_PICC_2 (public keys of the eCard) - keyId_1 and keyId_2 of the related static key pairs - D_PICC_1 and D_PICC_2 (static domain parameters of the eCard) A Secure Messaging Channel (SM) is established between terminal software and LT after PACE protocol.
Test scenario	The execution steps have to be performed as described in chapter 5.3.3: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (with SM) from the terminal software and sends response APDUs (with SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). LT generates random number r2_PICC and – according to step 7 in chapter 5.3.3 – computes key material K = KA(**SK_PICC_2**, PKeph_PCD, D_PICC), if UT identifies **keyID_1**, or K = KA(**SK_PICC_1**, PKeph_PCD, D_PICC), if UT identifies **keyID_2** key K_MAC = KDF_MAC(K, r2_PICC) and T_PICC' = MAC(K_MAC, PKeph_PCD) correctly. LT transmits T_PICC' to the terminal software. UT receives random number r2_PICC and authentication token T_PICC' in response APDU to General Authenticate. UT derives key material K = KA(SKeph_PCD, **PK_PICC_1**, D_PICC), if UT identified **keyId_1**, or K = KA(SKeph_PCD, **PK_PICC_2**, D_PICC), if UT identified **keyId_2**,

	keys K_MAC = KDF_MAC(K, r2_PICC) and computes MAC(K_MAC, PKeph_PCD).
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides apart from SM with the command APDU sent by UT directly before. The SM in the command APDU generated by the terminal software and received by LT is correct. Each response APDU received at UT is correctly coded without SM and coincides apart from SM with the response APDU sent by LT directly before. UT checks that the received T_PICC' is different from the computed MAC(K_MAC, PKeph_PCD).
Post processing	Reset of eCard, reader and terminal software

6.3.4 Access to the eID Application

If this test unit is used for testing eID clients according to BSI TR-03124-1, then all certificates with role authentication terminal and access rights according to table table 29, No. **1** SHALL be replaced with the certificates with access rights according to table 29, No. **4**.

6.3.4.1 TS_eID_1 – Correct Reading Access to eID Data with EAC

Test ID	TS_eID_1.1.1
Purpose	Check terminal software for correct reading access to eID data with EAC. Use certificate with role inspection system.
References	[TR-03110], C.4.1, E.1
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there shall be subsequently performed for all data groups (using the SFIs) with read access in the CHAT. UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').

Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_1.2.1
Purpose	Check terminal software for correct reading access to eID data with EAC. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2, E.1
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1, 2; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with PIN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there shall be subsequently performed for all data groups (using the SFIs) with read access in the CHAT. UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_1.3.1_template
Purpose	Check terminal software for correct reading access to eID data with EAC. Use certificate with role authentication terminal. In compliance to the manufacturer's Implementation Conformance Statement

	(see chapter 4.2)different algorithms for secure messaging have to be used. The algorithms - 3DES - AES – 128 - AES – 192 - AES - 256 have to be tested, if support by the reader is stated in the manufacturer's conformance statement. The test has to be performed for each supported algorithm.
References	[TR-03110], A.4, C.4.2, E.1, F.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1, 2; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with password CAN or PIN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.5: The command sequence described there shall be subsequently performed for all data groups (using the SFIs) with read access in the CHAT. UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	The following results are expected: Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Algorithm
TS_eID_1.3.1a	3DES
TS_eID_1.3.1b	AES – 128
TS_eID_1.3.1c	AES – 192
TS_eID_1.3.1d	AES – 256

Table 79: Test case TS_eID_1.3.1

6.3.4.2 TS_eID_2 – Correct Writing Access to eID Data with EAC

Test ID	TS_eID_2.1.1
Purpose	Check terminal software for correct writing access to eID data with EAC. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2, E.1
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND Write_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. The contents of data groups DG17,..., DG21 of the eCard at LT are known in UT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.6: The command sequence described there shall be subsequently performed for all data groups (using the SFIs) with write access in the CHAT. Check the correct execution of the UPDATE BINARY by reading the DG. UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before. Check that the new data in the updated DGs corresponds with the data that was used in the UPDATE BINARY command.

Post processing	The original contents of DG17, ..., DG21 in the eCard at LT is restored using the command sequence from chapter 5.3.6. Reset of eCard, reader and terminal software

6.3.4.3 TS_eID_3 – Correct Execution of Internal eID Functions

Test ID	TS_eID_3.1.1_template
Purpose	Check terminal software for correct execution of restricted identification for eCards. Use certificate with role authentication terminal.
References	[TR-03110], 4.5, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.7: UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_3.1.1a	CAN
TS_eID_3.1.1b	PIN

Table 80: Test case TS_eID_3.1.1

Test ID	TS_eID_3.2.1_template

Purpose	Check terminal software for correct age verification for card holder. Use certificate with role authentication terminal. The test case is performed in several variants for the date of birth to be checked: - date of birth to be checked is smaller than date of birth in eCard – 1 - date of birth to be checked coincides with date of birth in eCard – 1 - date of birth to be checked coincides with date of birth - date of birth to be checked coincides with date of birth in eCard + 1 - date of birth to be checked is greater than date of birth in eCard + 1This test case is only rated as a PASS if all passes are completed successfully.
References	[TR-03110], C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. In the MSE: Set AT command of the terminal authentication protocol (see chapter 5.3.2) the field `<aux_data>` must be mapped to the following value: `73 <L_73> 06 <L_06> <OID> 53 <L_53> <disc_data>` OID: OID for age verification disc_data: date of birth to be checked (the variants to be checked are defined under Purpose)
Test scenario	The execution steps have to be performed as described in chapter 5.3.8 (one execution for each variant as defined under Purpose): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_3.2.1a	CAN
TS_eID_3.2.1b	PIN

Table 81: Test case TS_eID_3.2.1

Test ID	TS_eID_3.3.1_template
Purpose	Check terminal software for correct verification of card holder's community ID. Use certificate with role authentication terminal.
References	[TR-03110], C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT. In the MSE: Set AT command of the terminal authentication protocol (see chapter 5.3.2) the field `<aux_data>` must be mapped to the following value: `73 <L_73> 06 <L_06> <OID> 53 <L_53> <disc_data>` OID: OID for community ID verification disc_data: community ID to be checked
Test scenario	The execution steps have to be performed as described in chapter 5.3.8: UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_3.3.1a	CAN
TS_eID_3.3.1b	PIN

Table 82: Test case TS_eID_3.3.1

6.3.4.4 TS_eID_4 – Password Management Functions for Authenticated Terminals

Test ID	TS_eID_4.1.1_template
Purpose	Check that an authenticated terminal supports the password management function to change the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND (TS_Chg_PIN OR TS_PIN_MGT_AT)
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '03' (for changing PIN): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_4.1.1a	CAN
TS_eID_4.1.1b	PIN

Table 83: Test case TS_eID_4.1.1

Test ID	TS_eID_4.2.1_template
Purpose	Check that an authenticated terminal supports the password management function to change the CAN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND TS_Chg_CAN
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '02' (for changing CAN): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The original CAN is restored in the eCard. Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_4.2.1a	CAN
TS_eID_4.2.1b	PIN

Table 84: Test case TS_eID_4.2.1

Test ID	TS_eID_4.3.1
Purpose	Check that an authenticated terminal supports the password management function to unblock the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND TS_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PIN is blocked in the eCard. The PACE, terminal authentication and chip authentication protocols have been executed successfully with password-ID CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.2: UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_4.4.1
Purpose	Check that an authenticated terminal supports the password management function to activate the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND TS_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PIN is deactivated in the eCard.

	The PACE, terminal authentication and chip authentication protocols have been executed successfully with password-ID CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.3, where the command Activate is performed with (Byte INS = '44'): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_4.5.1_template
Purpose	Check that an authenticated terminal supports the password management function to deactivate the PIN. Use certificate with role authentication terminal.
References	[TR-03110], 3.5.2, C.4.2
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_eID AND TS_PIN_MGT_AT
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.3, where the command Deactivate is performed with (Byte INS = '04'): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before.

	Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The deactivated PIN is activated again. Reset of eCard, reader and terminal software

Testcase ID	Password for PACE
TS_eID_4.5.1a	CAN
TS_eID_4.5.1b	PIN

Table 85: Test case TS_eID_4.5.1

6.3.4.5 TS_eID_5 – Password Management Functions for Unauthenticated Terminals after PACE

Test ID	TS_eID_5.1.1
Purpose	Check that an unauthenticated terminal supports the password management function to change the PIN. The current PIN is *not* a transport PIN.
References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_PIN_MGT_uT
Preconditions	The PACE protocol has been executed successfully with password-ID PIN, where the PIN is not a transport PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between terminal software and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '03' (for changing PIN): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard, reader and terminal software

Test ID	TS_eID_5.1.2
Purpose	Check that an unauthenticated terminal supports the password management function to change the PIN. The current PIN is a transport PIN.
References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_PIN_MGT_uT
Preconditions	The PACE protocol has been executed successfully with password-ID PIN, where the PIN is a transport PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between terminal software and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '03' (for changing PIN): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_5.2.1
Purpose	Check that an unauthenticated terminal supports the password management function to reset the retry counter for the PIN using the PUK.
References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_PIN_MGT_uT
Preconditions	The PIN is blocked in the eCard. The PACE protocol has been executed successfully with password-ID PUK. After execution of this protocol, a Secure Messaging Channel (SM) is established between terminal software and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.2: UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends

	response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_5.3.1
Purpose	Check that an unauthenticated terminal supports the password management function to set a new PIN using the PUK.
References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_Chg_PIN_PUK
Preconditions	The PACE protocol has been executed successfully with password-ID PUK. After execution of this protocol, a Secure Messaging Channel (SM) is established between terminal software and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.9.1, where the command Reset Retry Counter is performed with P2 = '03' (for changing PIN): UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	The original PIN is restored in the eCard. Reset of eCard, reader and terminal software

Test ID	TS_eID_5.4.1
Purpose	Check that an unauthenticated terminal resumes temporarily a PIN using the CAN.

References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_PIN_MGT_uT
Preconditions	The PIN is suspended (i. e. retry counter = 1).
Test scenario	1. UT starts PACE protocol in terminal software and transmits password to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The <PIN-ID> in command APDU to MSE: Set AT is '02', since CAN is used. • The data object with tag 7F 4C is missing in command APDU to MSE: Set AT (since CHAT is empty) . 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_eID_5.5.1
Purpose	Check that an unauthenticated terminal resumes a temporarily resumed PIN by using it in PACE protocol. This PACE protocol is performed with Secure Messaging (SM) where the SM keys have been derived by execution of another PACE protocol with the CAN. The test case assumes, as specified in the Precondition, that the PACE protocol with CAN has been executed before test start. The test is executed with the PIN.
References	[TR-03110], 3.5.1
Profiles	TS_PACE AND TS_eID AND TS_PIN_MGT_uT
Preconditions	First a suspended PIN has been used (i. e. retry counter = 1). Then the PACE protocol has been executed successfully with password-ID

	CAN to temporarily resume the PIN. After execution of this protocol, a Secure Messaging Channel (SM) is established between terminal software and LT.
Test scenario	1. UT starts PACE protocol in terminal software and transmits password to terminal software. 2. LT receives command APDUs from the test object terminal software and sends response APDUs back to the terminal software as described in chapter 5.3.1. The return codes in all response APDUs are positive ('90 00'). All command and response APDUs are secured with Secure Messaging where SM keys are used that have have been derived in the PACE protocol with CAN according to Precondition. 3. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard)
Expected results	1. - 2. The command APDUs which LT receives from the terminal software are built up as described in chapter 5.3.1 with the following additions: • The `<PIN-ID>` in command APDU to MSE: Set AT is '03', since PIN is used. • The data object with tag `7F 4C` is missing in command APDU to MSE: Set AT (since CHAT is empty). • All command APDUs are secured correctly with Secure Messaging. 3. UT checks that the received data coincide with the data of EF.CardAccess in LT.
Post processing	The original PIN is restored in the eCard. Reset of eCard, reader and terminal software

Test ID	TS_eID_5.5.2 deleted in version 1.2

6.3.5 Access to Biometric Data

6.3.5.1 TS_bio_1 – Correct Reading Access to Biometric Data with EAC

Test ID	TS_bio_1.1.1
Purpose	Check terminal software for correct reading access to biometric data Fingerprint (DG 3) and Iris (DG 4) of the ePassport application with EAC. Use certificate with role inspection system.

References	[TR-03110], C.4.1
Profiles	TS_PACE AND TS_TA AND TS_CA AND TS_bio
Preconditions	Certificate with role inspection system and access rights according to 28, No. 1; CA certificates according to 31, No. 1, 2 The PACE, terminal authentication and chip authentication protocols have been executed successfully with CAN. After execution of these protocols, a Secure Messaging Channel (SM) is established between UT and LT.
Test scenario	The execution steps have to be performed as described in chapter 5.3.10: The command sequence described there is subsequently performed with SFIs referencing data groups DG 3 (Fingerprint) and DG 4 (Iris). UT sends command APDUs to the test object terminal software with SM. LT receives command APDUs with SM from the terminal software and sends response APDUs with SM back to the terminal software. UT receives response APDUs from the terminal software with SM. The return codes in all response APDUs are positive ('90 00').
Expected results	Each command APDU received at LT is correctly coded and secured by SM and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and secured by SM and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

6.3.6 Use of the Digital Signature Application

6.3.6.1 TS_Sig_1 – Successful Key Pair Generation

Test ID	TS_Sig_1.1.1
Purpose	Check for correct execution of authentication protocols PACE, TA and CA via the terminal software of an authentication terminal of the QCA and LT. It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The test for the PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2

	Make certificates available in UT
Test scenario	1. UT starts PACE protocol in terminal software and transmits certificates (according to Profile) to terminal software. 2. LT receives PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3). 3. LT sends OutBuffer of ScardControl (see chapter 5.2.4), where the positions in OutBuffer are defined as follows: • <Result_Code>: The result code is positive (i.e. '00 00 00 00'). • <status_mse>: The status code for command MSE: Set AT is '90 00'. • <data_card_acc>: The transmitted eCard data for card access coincide with the data of EF.CardAccess in LT. • <CAR1> and <CAR2>: The current and previous CAR are as defined in the eCard. • <IDPICC>: The field contains the correct ID_PICC. 4. UT receives the following data from terminal software: • PK_ICC (public key of the eCard) • D_PICC (static domain parameters of the eCard) • ID_PICC (identification of the eCard) 5. The execution steps for terminal authentication have to be performed as described in chapter 5.3.2 with the following properties: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (**without** SM) from the terminal software and sends response APDUs (**without** SM) back to the terminal software. UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). 6. The execution steps for chip authentication have to be performed as described in chapter 5.3.3 with the following properties: UT sends command APDUs to the test object terminal software (without SM). LT receives command APDUs (**without** SM) from the terminal software and sends response APDUs (**without** SM) back to the terminal software . UT receives response APDUs from the terminal software (without SM). The return codes in all response APDUs are positive ('90 00'). Moreover UT generates the keys K_MAC and K_ENC for subsequent secure messaging as described in step 10 of chapter 5.3.3.
Expected results	1. - 2. LT receives PC/SC function SCardControl with InBuffer for EstablishPACEChannel (see chapter 5.2.3), where the positions in

	InBuffer are defined as follows: • <PIN-ID>: '03' (PIN is used) • <CHAT>: Restricted CHAT: at least “Install Qualified Certificate” and the eCard-User related data requested from the QCA (DG1 to DG21) • <PIN>: empty, because password (PIN) is entered via PINPad of reader • <CERT_DESC>: List of certificates as specified in Profiles 3. - 4. UT checks that the received data coincide with the data of EF.CardAccess in LT. 5. Each command APDU received at LT is correctly coded and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before. 6. Each command APDU received at LT is correctly coded and coincides with the command APDU sent by UT directly before. Each response APDU received at UT is correctly coded and coincides with the response APDU sent by LT directly before.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_1.2.1
Purpose	Check access via the terminal software of an authentication terminal of the QCA to the necessary identification data of the eCard-User (DG1 to DG21).It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03110], E.1, E.1.1; [TR-03117], 4.3.2
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel

Test scenario	1. UT calls function to transmit the command Select to the LT Select: '00 A4 04 0C <L_c> <AID of DF.eID>' with 'E80704007F00070302' = AID of DF.eID 2. LT receives PC/SC function call with a secured command APDU for Select. 3. LT sends PC/SC response with a secured response APDU for Select: '99 02 90 00 8E 08 <MAC> 90 00' 4. UT receives response with response APDU for Select 5. UT calls functions to transmit a sequence of 21 Read Binary commands to the LT to retrieve data DG1 to DG21 from the eID application: Read Binary: '00 B0 P1 00 <L_c> 00' with P1 = '8 <SFI>'. Read all Data Groups which are available on the chip (usually DG 1 to 12 and 17 to 21). 6. LT receives PC/SC function call with a secured command APDU for Read Binary. 7. LT sends PC/SC response with a secured response APDU for Read Binary: '87 <L87> <Cryptogram> 99 02 90 00 8E 08 <MAC> 90 00' with Cryptogram: Encrypted data of the correspondent DG 8. UT receives response with response APDU for Read Binary.
Expected results	1. - 2. The secured command APDU for Select is built up as follows: Select: '0C A4 04 0C <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted AID of DF.eID with 'E80704007F00070302' = AID of DF.eID 3. - 4. UT receives response with response APDU for Select: '90 00' 5. - 6. The secured command APDU for Read Binary is built up as follows: '0C B0 P1 00 <L_c> 97 01 00 8E 08 <MAC> 00' with P1 as in the previous step of the test scenario defined. 7. - 8. UT receives response with response APDU for Read Binary: '<Data of correspondent DG>90 00'
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_1.3.1

Purpose	Check access via the terminal software of an authentication terminal of the QCA to the signature application of the eCard for key generation. It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated”
Test scenario	1. UT calls function to transmit the command Select to the LT: Select: '00 A4 04 0C <L_c> <AID of DF.eSign>' with 'A0 00 00 01 67 45 53 49 47 4E' = AID of DF.eSign 2. LT receives PC/SC function call with a secured command APDU for Select. 3. LT sends PC/SC response with a secured response APDU for Select: '99 02 90 00 8E 08 <MAC> 90 00' 4. UT receives response with response APDU for Select 5. UT calls function to transmit the command APDU Generate Asymmetric Key Pair to the LT; Key Generation with return of the public key; Command APDU Generate Asymmetric Key Pair: '00 47 82 00 <L_c> <Data> <L_e>' with Data: B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 6. LT receives PC/SC function call with a secured command APDU for Generate Asymmetric Key Pair. 7. LT sends PC/SC response with a secured response APDU for Generate Asymmetric Key Pair: '87 <L_{87}> <Cryptogram> 99 02 90 00 8E 08 <MAC> 90 00' with Cryptogram: Encrypted public key DO '7F 49 <$L_{7F\ 49}$> public key data

	8. UT receives response with response APDU for Generate Asymmetric Key Pair
Expected results	1. - 2. The secured command APDU for Select is built up as follows: Select: '0C A4 04 0C <L_c> 87 <L_{87}> <Cryptogram> 8E 08 <MAC> 00'; Cryptogram: Encrypted AID of DF.eID with 'E80704007F00070302' = AID of DF.eID 3. - 4. UT receives response with response APDU for Select: '90 00' 5. - 6. The secured command APDU for Generate Asymmetric Key Pair is built up as follows: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 7. - 8. UT receives response with response APDU for Generate Asymmetric Key Pair: '7F 49 <$L_{7F\ 49}$> <Public Key Data> 90 00'; The received public key data coincide with the public key data object of the signature key in the eSign application in LT.
Post processing	Termination of the signature key in the eCard Reset of eCard, reader and terminal software

6.3.6.2 TS_Sig_2 – Abort Key Pair Generation

Test ID	TS_Sig_2.1.1
Purpose	Check that the terminal software of an authentication terminal of the QCA aborts key pair generation if it the authentication terminal does **not** have the access right “Install Qualified Certificate”. It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal without access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5

Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 3; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated” Successful PACE protocol with PIN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel eSign application is selected
Test scenario	1. UT calls function to transmit the command APDU Generate Asymmetric Key Pair to the LT; Key Generation with return of the public key; Command APDU Generate Asymmetric Key Pair: '00 47 82 00 <L_c> <Data> <L_e>' with Data: B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 2. LT receives PC/SC function call with a secured command APDU for Generate Asymmetric Key Pair. 3. LT sends PC/SC response with a secured response APDU for Generate Asymmetric Key Pair with an error code: '99 02 69 82 8E 08 <MAC> 69 82' 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code.
Expected results	1. - 2. The secured command APDU for Generate Asymmetric Key Pair is built up as follows: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 3. - 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code: '69 82'
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_2.1.2

Purpose	Check that the terminal software of an authentication terminal of the QCA aborts key pair generation if the authentication terminal have the access right “Install Qualified Certificate” but the PACE protocol is performed with the CAN (instead of the PIN). It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID **CAN**.
References	[TR-03117], 4.3.2, A.2.5
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated” Successful PACE protocol with CAN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel eSign application is selected
Test scenario	1. UT calls function to transmit the command APDU Generate Asymmetric Key Pair to the LT; Key Generation with return of the public key; Command APDU Generate Asymmetric Key Pair: '00 47 82 00 <L_c> <Data> <L_e>' with Data: B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 2. LT receives PC/SC function call with a secured command APDU for Generate Asymmetric Key Pair. 3. LT sends PC/SC response with a secured response APDU for Generate Asymmetric Key Pair with an error code: '99 02 69 82 8E 08 <MAC> 69 82' 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code.
Expected results	1. - 2. The secured command APDU for Generate Asymmetric Key Pair is built up as follows: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08

	<MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 3. - 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code: '69 82'
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_2.2.1
Purpose	Check that the terminal software of an authentication terminal of the QCA aborts key pair generation if the eSign application could not be selected. It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Status of eSign-PIN in LT is “operational” Status of signature key in LT is “terminated” Successful PACE protocol with PIN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel Use LT **without** installed eSign application.
Test scenario	1. UT calls function to transmit the command APDU Select to the LT: '00 A4 04 0C <L_c> <AID of DF.eSign>' 2. LT receives PC/SC function call with a secured command APDU for Select. 3. LT sends PC/SC response with a secured response APDU for Select with an error code: '99 02 6A 82 8E 08 <MAC> 6A 82' 4. UT receives response with response APDU for Select with an error code.

Expected results	1. - 2. The secured command APDU for Select is built up as follows: '00 A4 04 0C <L_c> 87 <L_{87}> 01 <Cryptogram> 8E 08 <MAC> 00' with Data (in plaintext): AID = '4E 47 49 53 45 67 01 00 00 A0' incorrect AID of DF.eSign 3. - 4. UT receives response with response APDU for Select with an error code: '6A 82'
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_2.2.2
Purpose	Check that the terminal software of an authentication terminal of the QCA aborts key pair generation if the LT returns an error code to the command Generate Asymmetric Key Pair. It is assumed that the terminal software is installed at the ZDA. PACE within LT, TA and CA between authentication terminal of the QCA and LT. Use of certificate with role authentication terminal with access right “Install Qualified Certificate”. The PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.3.2, A.2.5
Profiles	TS_Sig
Preconditions	Certificate with role authentication terminal and access rights according to 29, No. 1; CA certificates according to 32, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN within LT as well as TA and CA between authentication terminal of the QCA and LT with establishment of a trusted channel eSign application is selected
Test scenario	1. UT calls function to transmit the command APDU Generate Asymmetric Key Pair to the LT; Key Generation with return of the public key; Command APDU Generate Asymmetric Key Pair: '00 47 82 00 <L_c> <Data> <L_e>' with Data: B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 2. LT receives PC/SC function call with a secured command APDU for Generate Asymmetric Key Pair.

	3. LT sends PC/SC response with a secured response APDU for Generate Asymmetric Key Pair with an error code: '99 02 SW1 SW2 8E 08 <MAC> SW1 SW2' with SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found) 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code.
Expected results	1. - 2. The secured command APDU for Generate Asymmetric Key Pair is built up as follows: '0C 47 82 00 <L_c> 87 <L_{87}> 01 <Cryptogram> 97 <L_{97}> <L_e> 8E 08 <MAC> 00' with Data (in plaintext): B6 <L_{B6}> 84 <L_{84}> <private key reference> || '7F 49': data related to the key to be generated (optional) 3. - 4. UT receives response with response APDU for Generate Asymmetric Key Pair with an error code: 'SW1 SW2' with SW1 SW2 = '69 84' (Reference data not usable) or SW1 SW2 = '6A 88' (Referenced data not found)
Post processing	Reset of eCard, reader and terminal software

6.3.6.3 TS_Sig_3 – Entering the CAN deleted in version 1.1

6.3.6.4 TS_Sig_4 – Successful Signature Generation

Test ID	TS_Sig_4.1.1
Purpose	Check correct execution of Select eSign application via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT

	Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel
Test scenario	1. UT calls function to transmit the command APDU for Select to the terminal software: Command APDU for Select: '00 A4 04 0C <L_c> <AID of eSign application>' with AID = 'A0 00 00 01 67 45 53 49 47 4E' AID of eSign application 2. LT receives PC/SC function call with a command APDU for Select. 3. LT sends PC/SC response with a response APDU for Select to the terminal software: Response APDU for Select: '90 00'. 4. UT receives response APDU for Select.
Expected results	1. - 2. The command APDU for Select which LT receives from the terminal software is built up as sent by UT. 3. - 4. The response APDU for Select which UT receives from the terminal software is built up as sent by LT.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_4.2.1
Purpose	Check correct execution of the eSign-PIN verification via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.1
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel eSign application is selected eSign-PIN is valid, in status “operational” and not blocked

	Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to transmit the command APDU for Verify (without the eSign-PIN) to the terminal software: Command APDU for Verify: '00 20 00 P2 <L_c> <eSign-PIN: empty>'; P2: Reference to eSign-PIN 2. LT receives PC/SC function call with a command APDU for Verify. 3. LT sends PC/SC response with a response APDU for Verify to the terminal software: Response APDU for Verify: '90 00'. 4. UT receives response APDU for Verify.
Expected results	1. - 2. The command APDU for Verify which LT receives from the terminal software is built up as sent by UT. 3. - 4. The response APDU for Verify which UT receives from the terminal software is built up as sent by LT.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_4.3.1
Purpose	Check correct execution of a signature generation via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.1
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel eSign application is selected eSign-PIN is successfully verified

Test scenario	1. UT calls function to transmit the command APDU for PSO : Compute Digital Signature to the terminal software: Command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>' 2. LT receives PC/SC function call with a command APDU for PSO : Compute Digital Signature. 3. LT sends PC/SC response with a response APDU for PSO : Compute Digital Signature to the terminal software: Response APDU for PSO : Compute Digital Signature: '<Digital Signature> 90 00'. 4. UT receives response APDU for PSO : Compute Digital Signature.
Expected results	1. - 2. The command APDU for PSO : Compute Digital Signature which LT receives from the terminal software is built up as sent by UT. 3. - 4. The response APDU for PSO : Compute Digital Signature which UT receives from the terminal software is built up as sent by LT.
Post processing	Reset of eCard, reader and terminal software

6.3.6.5 TS_Sig_5 – Abort Signature Generation

Test ID	TS_Sig_5.1.1
Purpose	Check that the execution of Select eSign application via the terminal software of a signature terminal is aborted, if the eSign application could not be selected. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel
Test scenario	1. UT calls function to select the eSign application.

	2. LT receives PC/SC function call with a command APDU for Select. 3. LT sends PC/SC response with a response APDU for Select with an error code to the terminal software: Response APDU for Select: '6A 82'. 4. UT receives response.
Expected results	1. - 2. The command APDU for Select which LT receives from the terminal software is Command APDU for Select: '00 A4 04 0C <L_e> <AID of eSign application>' with AID = 'A0 00 00 01 67 45 53 49 47 4E' AID of eSign application 3. - 4. UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_5.2.1
Purpose	Check that the execution of the eSign-PIN verification via the terminal software of a signature terminal is aborted if the password verification fails. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.1
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel eSign application is selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to verify the eSign-PIN(e.g. starting signature funtion in TS) 2. LT receives PC/SC function call with a command APDU for Verify. 3. LT sends PC/SC response with a response APDU for Verify with an error

	code to the terminal software: Response APDU for Verify: '69 82' or '63 CX', with X= remaining retries. 4. UT receives response.
Expected results	1. - 2. The command APDU for Verify which LT receives from the terminal software is PC/SC command FEATURE_VERIFY_PIN_DIRECT 3. - 4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_5.3.1
Purpose	Check that the execution of a signature generation via the terminal software of a signature terminal is aborted if the signature generation fails. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.1
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel eSign application is selected
Test scenario	1. UT calls function to start signature function. 2. LT receives PC/SC function call with a command APDU for PSO : Compute Digital Signature. 3. LT sends PC/SC response with a response APDU with an error code for PSO : Compute Digital Signature to the terminal software: Response APDU for PSO : Compute Digital Signature: '69 82'. 4. UT receives response.
Expected results	1. - 2. The command APDU for PSO : Compute Digital Signature which LT

	receives from the terminal software is: Command APDU for PSO : Compute Digital Signature: '00 2A 9E 9A <L_c> <Hash value of the data to be signed> <L_e>' 3. - 4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

6.3.6.6 TS_Sig_6 – Successful Password Management Functions

Test ID	TS_Sig_6.1.1
Purpose	Check correct execution of setting the eSign-PIN via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.2
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected Status of eSign-PIN is “terminated” Status of private signature key is “terminated” Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to set a new eSign-PIN. 2. LT receives PC/SC function call with a command for Change Reference Data. 3. LT sends PC/SC response with a response APDU for Change Reference Data to the terminal software: Response APDU for Change Reference Data: '90 00'. 4. UT receives response.
Expected results	1. -

	2. The command for Change Reference Data which LT receives from the terminal software is PC/SC command FEATURE_MODIFY_PIN_DIRECT. 3. - 4. The UT receives response of successfully setting the eSign-PIN from the terminal.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_6.2.1
Purpose	Check correct execution of changing the eSign-PIN via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID CAN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.2
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with CAN, TA and CA within LT with establishment of a trusted channel eSign application is selected Status of eSign-PIN is “operational” Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to change the eSign-PIN. 2. LT receives PC/SC function call with a command for Change Reference Data. 3. LT sends PC/SC response with a response APDU for Change Reference Data to the terminal software: Response APDU for Change Reference Data: '90 00'. 4. UT receives response.
Expected results	1. - 2. The command for Change Reference Data which LT receives from the terminal software is PC/SC command

	FEATURE_MODIFY_PIN_DIRECT. 3. - 4. The UT receives response of successfully changing the eSign-PIN from the terminal.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_6.3.1
Purpose	Check correct execution of resetting the retry counter of the eSign-PIN via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PUK which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.2
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PUK, TA and CA within LT with establishment of a trusted channel eSign application is selected Status of eSign-PIN is “blocked” Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to retting the retry counter of the eSign-PIN. 2. LT receives PC/SC function call with a command for Reset Retry Counter. 3. LT sends PC/SC response with a response APDU for Reset Retry Counter to the terminal software: Response APDU for Reset Retry Counter: '90 00'. 4. UT receives response.
Expected results	1. - 2. The command for Reset Retry Counter which LT receives from the terminal software is PC/SC command FEATURE_VERIFY_PIN_DIRECT. 3. -

	4. The UT receives response of successfully resetting the eSign-PIN from the terminal software.
Post processing	Reset of eCard, reader and terminal software

6.3.6.7 TS_Sig_7 – Abort Password Management Functions

Test ID	TS_Sig_7.1.1
Purpose	Check that the execution of setting or changing the eSign-PIN via the terminal software of a signature terminal is aborted if the password management function fails. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN.
References	[TR-03117], 4.4.2, A.2.3
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to set the eSign-Pin. 2. LT receives PC/SC function call with a command for Change Reference Data. 3. LT sends PC/SC response with a response APDU with an error code for Change Reference Data to the terminal software: Response APDU for Change Reference Data: 'SW1 SW2' with SW1 SW2 = '69 82' (Security Status not satisfied) or SW1 SW2 = '69 83' (Authentication method blocked) 4. UT receives response.
Expected results	1. - 2. The command for Change Reference Data which LT receives from the terminal software is PC/SC command FEATURE_MODIFY_PIN_DIRECT. 3. -

	4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_7.1.2
Purpose	Check that the execution of resetting the retry counter of the eSign-PIN via the terminal software of a signature terminal is aborted if the password management function fails. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PUK.
References	[TR-03117], 4.4.2, A.2.3
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PUK, TA and CA within LT with establishment of a trusted channel eSign application is selected Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to reset the retry counter of the eSign-PIN. 2. LT receives PC/SC function call with a command for Reset Retry Counter. 3. LT sends PC/SC response with a response APDU with an error code for Reset Retry Counter to the terminal software: Response APDU for Reset Retry Counter: 'SW1 SW2' with SW1 SW2 = '69 82' (Security status not satisfied) or SW1 SW2 = '69 84' (Referenced data invalidated) 4. UT receives response.
Expected results	1. - 2. The command for Reset Retry Counter which LT receives from the terminal software is PC/SC command FEATURE_VERIFY_PIN_DIRECT. 3. - 4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

6.3.6.8 TS_Sig_8 – Successful Termination of the Signature Function

Test ID	TS_Sig_8.1.1
Purpose	Check correct execution of the termination of the eSign-PIN via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.6
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected. Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to terminate the eSign-PIN. 2. LT receives PC/SC function call with a command APDU for Terminate. 3. LT sends PC/SC response with a response APDU for Terminate to the terminal software: Response APDU for Terminate: '90 00'. 4. UT receives response.
Expected results	1. - 2. The command APDU for Terminate which LT receives from the terminal software is: Command APDU for Terminate: '00 E6 10 P2 <L_c> <data is empty>' with P2: reference to the eSign-PIN 3. - 4. The UT receives response of successfully terminate the eSign-PIN from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_8.1.2

Purpose	Check correct execution of the termination of the private signature key via the terminal software of a signature terminal. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.6
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected. eSign-PIN must be terminated.
Test scenario	1. UT calls function to terminate the private signature key. 2. LT receives PC/SC function call with a command APDU for Terminate. 3. LT sends PC/SC response with a response APDU for Terminate to the terminal software: Response APDU for Terminate: '90 00'. 4. UT receives response.
Expected results	1. - 2. The command APDU for Terminate which LT receives from the terminal software is: Command APDU for Terminate: '00 E6 21 00 <L_c> <DST with reference to the private signature key>' with DST: 'B6 <L_{B6}> 84 <L_{84}> <private key reference>' 3. - 4. The UT receives response of successfully terminate the private signature key from the terminal software.
Post processing	Reset of eCard, reader and terminal software

6.3.6.9 TS_Sig_9 – Abort Termination of the Signature Function

Test ID	TS_Sig_9.1.1
Purpose	Check that the execution of the termination of the eSign-PIN via the terminal

	software of a signature terminal is aborted if the termination fails. PACE, TA and CA within LT. Use of certificate with role signature terminal with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.6
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected. Reference eSign-PIN is retrieved from DF.CIA stored in LT
Test scenario	1. UT calls function to terminate the eSign-PIN. 2. LT receives PC/SC function call with a command APDU for Terminate. 3. LT sends PC/SC response with a response APDU for Terminate with an error code to the terminal software: Response APDU for Terminate: '69 82'. 4. UT receives response.
Expected results	1. - 2. The command APDU for Terminate which LT receives from the terminal software is: Command APDU for Terminate: '00 E6 10 P2 <L_c> <data is empty>' with P2: reference to the eSign-PIN 3. - 4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Test ID	TS_Sig_9.1.2
Purpose	Check that the execution of the termination of the private signature key via the terminal software of a signature terminal is aborted if the termination fails. PACE, TA and CA within LT. Use of certificate with role signature terminal

	with access right “Generate qualified electronic signature”. The test for the PACE protocol is executed with password-ID PIN which is transmitted by the UT to the LT.
References	[TR-03117], 4.4.2, A.2.6
Profiles	TS_Sig
Preconditions	Certificate with role signature terminal and access rights according to 30, No. 1; CA certificates according to 33, No. 1, 2 Make certificates available in UT Successful PACE protocol with PIN, TA and CA within LT with establishment of a trusted channel eSign application is selected. eSign-PIN is terminated.
Test scenario	1. UT calls function to terminate the private signature key. 2. LT receives PC/SC function call with a command APDU for Terminate. 3. LT sends PC/SC response with a response APDU for Terminate with an error code to the terminal software: Response APDU for Terminate: '6A 88'. 4. UT receives response.
Expected results	1. - 2. The command APDU for Terminate which LT receives from the terminal software is: Command APDU for Terminate: '00 E6 21 00 <L_c> <DST with reference to the private signature key>' with DST: 'B6 <L_{B6}> 84 <L_{84}> <private key reference>' 3. - 4. The UT receives an error from the terminal software.
Post processing	Reset of eCard, reader and terminal software

Annex

Bibliography

TR-03110	BSI: Technische Richtlinie TR-03110, Advanced Security Mechanisms for Machine Readable Travel Documents – Extended Access Control (EAC), Version 2
TR-03117	BSI: Technische Richtlinie TR-03117, eCards mit kontaktloser Schnittstelle als sichere Signaturerstellungseinheit;
TR-03119	BSI: Technische Richtlinie TR-03119, Anforderungen an Chipkartenleser mit ePA Unterstützung;
TR-03112-1	BSI: Technische Richtlinie BSI TR-03112-1, eCard-API-Framework
PCSC10	Apple etc.:Interoperability Specifications for ICCs and Personal Computer Systems, Part 10 IFDs with Secure PIN Entry Capabilities, Revision 2.02.06, April 2009
SigG	Gesetz über die Rahmenbedingungen für elektronische Signaturen (Signaturgesetz – SigG) vom 16. Mai 2001 (BGBl. I S. 876), zuletzt geändert durch Artikel 4 des Gesetzes vom 17. Juli 2009
SigV	Verordnung zur elektronischen Signatur (Signaturverordnung – SigV) vom 16. November 2001 (BGBl. I S. 3074), zuletzt geändert durch Verordnung vom 17. Dezember 2009