Federal Office
for Information Security

Technical Guideline TR-03130 eID-Server

Part 4: Conformance Test Specification

Version 1.2

17. March 2021

Federal Office for Information Security
Post Box 20 03 63
D-53133 Bonn

Phone: +49 22899 9582-0
E-Mail: eid@bsi.bund.de
Internet: https://www.bsi.bund.de

Table of Contents

1 Introduction 7

1.1 Terminology 8

2 Test Requirements 9

2.1 Test Environment 9
2.1.1 Overview 9
2.1.2 Server System 9
2.1.3 Testbed System 10
2.1.4 Network 10

2.2 Test Profiles 10
2.2.1 Basic Profiles 10
2.2.2 Optional/Recommended Profiles 11

3 Implementation Conformance Statement 12

3.1 Software Version 12

3.2 Supported API Versions 12

3.3 Support of Multiple eServices 12

3.4 Profiles 13

3.5 Server Address 14

3.6 Supported Cryptography 14
3.6.1 eCard API 14
3.6.2 eID-Interface 15
3.6.3 SAML Profile 15
3.6.4 EIDAS_MW profile 17
3.6.5 Certificates 18
3.6.6 Chip Authentication Algorithms 18

4 Definition of Configuration Data 20

4.1 Test Setup 20

4.2 Certificate Specification 21
4.2.1 CERT_SET_1 21
4.2.2 CERT_SET_2 31

4.3 eID-Cards 34
4.3.1 Card Profiles for [CERT_SET_1] 34
4.3.2 Card Profiles for [CERT_SET_2] 35

5 Definition for Test Cases 37

5.1 Test Case Notation 37

5.2 Commonly Used Elements 38
5.2.1 TC Token 39
5.2.2 StartPAOS 40
5.2.3 InitializeFramework 40
5.2.4 InitializeFrameworkResponse 40
5.2.5 DIDAuthenticate_EAC1InputType 41
5.2.6 DIDAuthenticateResponse_EAC1OutputType 42
5.2.7 DIDAuthenticate_EAC2InputType 43
5.2.8 DIDAuthenticateResponse_EAC2OutputType_A 44
5.2.9 DIDAuthenticateResponse_EAC2OutputType_B 44
5.2.10 Transmit 45

5.2.11 TransmitResponse 45
5.2.12 StartPAOSResponse 45
5.2.13 useIDRequest 46
5.2.14 useIDResponse 47
5.2.15 getResultRequest 47
5.2.16 getResultResponse 48
5.2.17 getServerInfoRequest 49
5.2.18 getServerInfoResponse 49
6 Test Specification 50
6.1 Module A: eID-Interface 50
6.1.1 Module A1: eID-Interface - Functional eID-Interface Operation 50
6.1.2 Module A2: eID-Interface – XML Security 54
6.1.3 Module A3: eID-Interface - TLS 55
6.2 Module B: SAML-Interface 55
6.2.1 Module B1: SAML-Interface – Functional Operational 55
6.2.2 Module B2: SAML-Interface - XML Security 61
6.2.3 Module B3: TLS – SAML 63
6.3 Module C: eCard-API Interface 63
6.3.1 Module C1: eCard-API Interface – Functional Operation 63
6.3.2 Module C2: TLS – eCard-API 68
6.4 Module D: eIDAS Middleware Mode 69
6.4.1 Module D1: eIDAS Middleware Mode – Functional eIDAS Operation 69
6.4.2 Module D2: eIDAS Middleware Mode - XML Security 70
6.4.3 Module D3: eIDAS Middleware Mode - TLS 71
6.5 Parameters 71
References 73

Figures

Figure 1: Test Environment 9
Figure 2: XML Schema Test Case 37
Figure 3: XML Schema ActionStep 38

Tables

Table 1: Mandatory test profiles 11
Table 2: Optional/Recommended test profiles 11
Table 3: Software Version 12
Table 4: Supported API Versions 12
Table 5: Multi-Client Capability 12
Table 6: Supported Basic Test Profiles 13
Table 7: Supported Optional/Recommended Test Profiles 14
Table 8: Addresses of the eID-Server 14
Table 9: Supported XML signature algorithms for SOAP communication 15
Table 10: Supported XML signature algorithms for SAML communication 16
Table 11: Supported XML encryption algorithms for SAML communication 17
Table 12: Supported XML signature algorithms for eIDAS communication 17
Table 13: Supported XML encryption algorithms for eIDAS communication 18
Table 14: Supported algorithms for Chip Authentication 18
Table 15: Variables referenced by the test cases 21
Table 16: Description of CERT_ECARD_CV_CVCA_1 22

Table 17: Description of CERT_ECARD_CV_DV_1_A 22
Table 18: Description of CERT_ECARD_CV_TERM_1_A 22
Table 19: Description of CERT_ECARD_CV_TERM_1_B 23
Table 20: Description of CERT_ECARD_CV_TERM_1_C 23
Table 21: Description of CERT_ECARD_CV_TERM_1_D 24
Table 22: Description of CERT_ECARD_CV_TERM_1_E_* 24
Table 23: Description of CERT_ECARD_CV_TERM_1_F 24
Table 24: Description of CERT_ECARD_CV_TERM_1_G 25
Table 25: Description of CERT_ECARD_TLS_EIDSERVER_1_* 25
Table 26: Description of CERT_ECARD_TLS_SAMLPROCESSOR_1_* 25
Table 27: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_<X> 26
Table 28: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_D 26
Table 29: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_E_* 26
Table 30: Description of CERT_EID_TLS_ESERVICE_1_<X> 27
Table 31: Description of CERT_EID_TLS_ESERVICE_1_E_* 27
Table 32: Description of CERT_EID_TLS_EIDSERVER_1_* 27
Table 33: Description of CERT_EID_XMLSIG_ESERVICE_1_<X> 27
Table 34: Description of CERT_EID_XMLSIG_ESERVICE_1_E_* 28
Table 35: Description of CERT_EID_XMLSIG_EIDSERVER_1_* 28
Table 36: Description of CERT_SAML_XMLSIG_ESERVICE_1_<X> 28
Table 37: Description of CERT_SAML_XMLSIG_ESERVICE_1_E_* 29
Table 38: Description of CERT_SAML_XMLSIG_EIDSERVER_1_* 29
Table 39: Description of CERT_SAML_XMLENC_ESERVICE_1_<X> 29
Table 40: Description of CERT_SAML_XMLENC_ESERVICE_1_E_* 29
Table 41: Description of CERT_SAML_XMLENC_EIDSERVER_1_* 30
Table 42: Description of CERT_ECARD_CSCA_1 30
Table 43: Description of CERT_ECARD_DS_1_A 30
Table 44: Description of CERT_ECARD_DS_1_B 31
Table 45: Description of CERT_ECARD_CV_CVCA_2_A 31
Table 46: Description of CERT_ECARD_CV_CVCA_2_B 31
Table 47: Description of CERT_ECARD_CV_CVCA_2_C 32
Table 48: Description of CERT_ECARD_CV_LINK_2_A 32
Table 49: Description of CERT_ECARD_CV_LINK_2_B 32
Table 50: Description of CERT_ECARD_CV_DV_2_A 33
Table 51: Description of CERT_ECARD_CV_TERM_2_A 33
Table 52: Description of CERT_TLS_EIDSERVER_2 33
Table 53: Description of CERT_ECARD_CSCA_2 34
Table 54: Description of CERT_ECARD_DS_2 34
Table 55: eID-Cards for [CERT_SET_1] used during the tests 35
Table 56: eID-Cards for [CERT_SET_2] used during the tests 36
Table 57: Structure of a TC Token 39
Table 58: Structure of a StartPAOS message 40
Table 59: Structure of an InitializeFrameworkResponse message 41
Table 60: Structure of DIDAuthenticate_EAC1InputType 42
Table 61: Structure of DIDAuthenticateResponse_EAC1OutputType 43
Table 62: Structure of DIDAuthenticate_EAC2InputType 44
Table 63: Structure of DIDAuthenticateResponse_EAC2OutputType_A 44
Table 64: Structure of DIDAuthenticateResponse_EAC2OutputType_B 45
Table 65: Structure of Transmit 45
Table 66: Structure of TransmitResponse 45
Table 67: Structure of StartPAOSResponse 46
Table 68: Structure of useIDRequest 46
Table 69: Structure of useIDResponse 47

Table 70: Structure of getResultRequest........48
Table 71: Structure of getResultResponse........48
Table 72: Structure of getServerInfoResponse........49
Table 73: Test Cases of Module A1_1........51
Table 74: Test Cases of Module A1_2........52
Table 75: Test Cases of Module A1_3........52
Table 76: Test Cases of Module A1_4........54
Table 77: Test Cases of Module A2........55
Table 78: Test cases of Module B1_1........56
Table 79: Test cases of Module B1_2........57
Table 80: Test Cases of Module B1_3........58
Table 81: Test cases of Module B1_4........59
Table 82: Test cases of Module B1_5........60
Table 83: Test Cases of Module B1_6........61
Table 84: Test Cases of Module B2_1........62
Table 85: Test Cases of Module B2_2........63
Table 86: Test Cases of Module C1_1........65
Table 87: Test Cases of Module C1_2........67
Table 88: Test Cases of Module C1_3........68
Table 89: Test cases of Module D1_1........69
Table 90: Test cases of Module D1_2........70
Table 91: Test Cases of Module D2........71
Table 92: Signature algorithms for XML Signature........72

1 Introduction

For Online-Authentication based on Extended Access Control Version 2 between an eService and an eID-Card (e.g. the German National Identity Card, the German electronic Residence Permit or the eID-Card for Union Citizens), the eService uses an eID-Server to integrate the corresponding functionality into the IT systems.

The eID-Server is specified in [TR-03130-1] and can be implemented by different vendors. It offers an interface to the eService and communicates with the eID-Client [TR-03124-1], the eID Card [TR-03127] and the corresponding Public Key Infrastructure (PKI) [CP-eID]. Both, eID-Client and eID-Server are based on the eCard-API-Framework [TR-03112] and support a subset of functions specified by this framework.

This Technical Guideline specifies conformity tests for eID-Servers according to [TR-03130-1]. The conformity tests are functional blackbox tests at the external interfaces of the eID-Server. The objective is to offer a basis for consistent and comparable quality assurance regarding the different eID-Server implementations. This shall guarantee conformity to the underlying specifications [TR-03130-1] and ensure interoperability with other components of the eID-Infrastructure.

The focus of the tests lies on the following functionalities and interfaces to the eService and the eID-Client, i.e.

- the eID-Interface:

  The eID-Interface is a direct communication interface between the Service Provider and the eID-Server as is specified in [TR-03130-1] that may be offered by the eID-Server. The tests verify the fulfilment of the requirements specified in [TR-03130-1]. This includes the SOAP messages and their correct usage as well as the cryptographic requirements for TLS and XML at this interface

- the SAML-Interface:

  SAML-Interface is a communication interface between the Service Provider and the eID-Server if Online- Authentication is embedded into a SAML-based authentication framework. In this case, the eID-Server contains a SAML-Processor that processes authentication requests and responses. The tests verify the correct implementation of the SAML profile according to [TR-03130-1]. This includes cryptographic requirements for TLS and XML at this interface

- the eCard-API:

  The eCard-API is the interface between the eID-Server and the eID-Client and is specified in [TR-03112]. The tests verify the correct implementation of this interface. This includes the correct execution of EAC as remote part of the Authentication Terminal according to [TR-03110], the PAOS communication with the eID-Client according to [TR-03112] and the fulfilment of cryptographic requirements for TLS according to [TR-03130-1].

- the attached mode:

  Besides, an eID-Server can be realised as a separate component or be attached to a Service Provider and/or SAML-Processor according to the attached model as described in [TR-03124-1]. The conformity tests cover both realisation scenarios.

- the eIDAS middleware mode:

  Additionally, the use of an eID-Server as eIDAS middleware according to [TR-03130-3] is covered by this test specification.

The following aspects are not within the scope of the conformance tests:

- Security checks which are not explicitly required by [TR-03130-1].

- Interface to the corresponding Public Key Infrastructures (for the retrieval and renewal of certificates, revocation lists[1], master lists or defect lists).
- Configuration of the eID-Server with the correct certificates and revocation lists in operational mode.
- Correct behaviour of the Service Provider, in particular in case the eID-Server is attached to the Service Provider.

This Technical Guideline is organized into 6 chapters. Chapter 2 describes the test environment to be used by test laboratories and the test profiles. The Implementation Conformance Statement that has to be filled by the applicant in order to determine the necessary test set is contained in Chapter 3. Subsequently, Chapter 4 specifies the configuration data for the tests and chapter 5 defines the structure of the test cases and commonly used elements. Finally, chapter 6 lists the test cases. The test cases themselves, including the necessary steps for test preparation, test execution and evaluation of the test results, are described in a set of XML files.

1.1 Terminology

The key words “MUST”, “MUST NOT”, “REQUIRED”, “SHALL”, “SHALL NOT”, “SHOULD”, “SHOULD NOT”, “RECOMMENDED”, “MAY”, and “OPTIONAL” in this document are to be interpreted as described in [RFC2119]. The key word "CONDITIONAL" is to be interpreted as follows:

CONDITIONAL: The usage of an item is dependent on the usage of other items. It is therefore further qualified under which conditions the item is REQUIRED or RECOMMENDED.

[1] In [TR-03130-1] also called blacklists.

2 Test Requirements

Each party willing to conduct the test series according to this document is going to need appropriate equipment and materials. This chapter introduces the general test requirements.

2.1 Test Environment

The set up consists of several parts that MUST be prepared prior to starting the test series. These parts will be introduced and explained in this section. The concrete implementation of the testing environment is not within the scope of this Technical Guideline.

2.1.1 Overview

In general, Figure 1 depicts the most important parts of the environment.

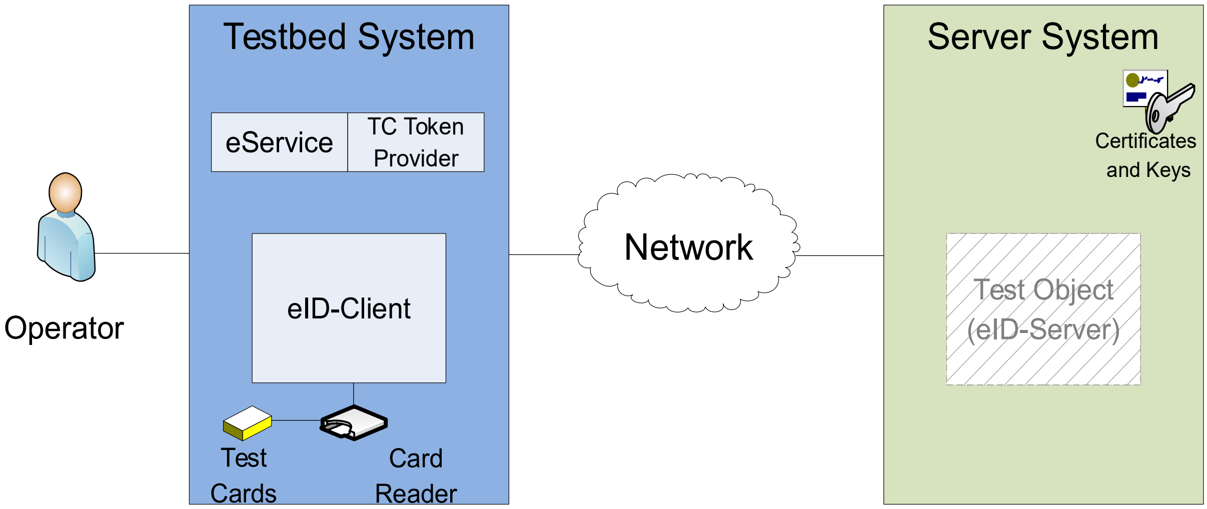


Figure 1: Test Environment

2.1.2 Server System

The eID-Server software, which is the test object, MAY either reside inside the test laboratory or at a remote location. In the latter case, the entire communication is transmitted over the internet. It is RECOMMENDED to supply a separate test instance of the eID-Server for the test procedure. The server MAY be run in a virtual environment.

In general, the used host requires at least two working network interfaces, since the communication with the eService (eID-Interface) and the eID-Client (eCard API Interface) is performed simultaneously.

Several tests require to configure the test object in an appropriate way. The manufacturer of the eID-Server MUST support the test laboratory to configure options for the tests as needed. In particular, the manufacturer MUST support the test laboratory by trustworthily storing the generated certificate material or revocation lists for different test settings on the test object in order to be able to perform the tests. Furthermore, the support also includes to set timeout values and the maximum number of simultaneous connections to reasonably small values unless the test object does not provide to configure these values.

A special communication model is the Attached eID-Server, cf. [TR-03124-1]. The scenario may arise if the eID-Server is combined with

- an eService, or

- the SAML Processor (including the TC Token Provider) into a single system.

If the eID-Server is attached to an eService, the communication between this eService and the eID-Server is carried out internally and the corresponding messages are out of scope. In this case, no TLS-1-2 and no pre-shared key is needed for communication between eID-Client and the corresponding eService, i.e. the corresponding channel TLS-2 is not based on a RSA_PSK cipher suite.

If the eID-Server supports the SAML profile, the eID-Server might be attached to the SAML Processor. In this case, TLS-1-2 is not necessary for communication with the SAML-Processor and no pre-shared key is needed, i.e. the corresponding channel TLS-2 is not based on a RSA_PSK cipher suite.

2.1.3 Testbed System

The testbed system comprises the necessary functionality of the eService, the eID-Client and the eID-Card interfaces to perform the test series. The tesbed system MUST support at least two network interfaces in order to be able to simulate the communication of the eID-Server with both, the eService (eID-Interface resp. SAML) and the eID-Client (eCard API Interface). Both interfaces SHOULD usually terminate in different domains. Furthermore, the testbed system SHOULD support a suitable interfaces to ease installation of certificate material on the eID-Server system, e.g. basic communication functionality according to [TR-03129].

The test laboratory MUST make all necessary provisions to exclude any negative effect on the test series. In particular, the test laboratory has to validate and confirm the test results after completion.

Because a number of tests require the usage of certificates created using different keys, it has to be ensured that no two keys are created equal.

2.1.4 Network

The test subject MAY be located either at a remote location (usually the site of its manufacturer) and only be accessible through the internet, or available in the test laboratory in a separate network. The latter configuration may be treated as a simplification of the online test. The preparation of the test laboratory therefore includes requesting the necessary IP addresses and DNS entries by the personnel.

2.2 Test Profiles

Test profiles describe preconditions and settings which may be enabled by the test laboratory in order to set up the environment for some of the test cases. Two separate profile categories exist: basic and optional.

2.2.1 Basic Profiles

The basic test profiles describe requirements defined in [TR-03130-1] and the underlying specifications which MUST be implemented by the manufacturer of conforming eID-Servers.

Profile ID	Description
PAOS	The eID-Server has implemented Reverse SOAP (=PAOS) communication.
EAC	The eID-Server has implemented Extended Access Control protocol according to [TR-03112].
EID_ACCESS	The eID-Server is able to manage access to requested attributes
REVOKED_CARD	The eID-Server has implemented revocation check and is capable to detect a revoked

Profile ID	Description
	eID-Card based on sector-specific identifiers.
EXPIRED_CARD	The eID-Server is capable to detect an expired eID-Card (via Document Validity Verification).
NONAUTH_CARD	The eID-Server is capable to recognise a non-authentic eID-Card.
CRYPTO	The eID-Server has implemented all requirements of [TR-03130-1] regarding the cryptography.
DG_VARIATIONS	The eID-Server is able to process eID-Cards with missing or empty data groups.

Table 1: Mandatory test profiles

2.2.2 Optional/Recommended Profiles

The different implementations may not match all optional specifications and recommendations. For this reason, optional test profiles describe specifications defined in [TR-03130-1] which MAY or SHOULD be implemented by the manufacturer of the eID-Server.

Profile ID	Description
SOAP	The eID-Server offers an (external) eID-Interface as described in [TR-03130-1].
SOAP_TLS	The eID-Server meets SOAP profile and supports TLS during the SOAP communication on the eID-Interface if communicating through an otherwise open network (e. g. Internet).
SAML	The eID-Server supports SAML-Profile Web Browser Single Sign-On (Web Browser SSO) profile specified in [SAML] Profiles, Section 4.1: Web Browser SSO Profile.
SAML_ATTACHED	The eID-Server meets SAML and is attached to the SAML-Processor and operates according to the Attached eID-Server Model as described in [TR-03124-1] for SAML.
ESER_ATTACHED	The eID-Server is attached to an eService and operates according to the Attached eID-Server Model as described in [TR-03124-1] for the eService.
EIDAS_MW	The eID-Server operates in eIDAS middleware mode as described in [TR-03130-3].
TLS_PSK	The eID-Server has implemented TLS-2 based on RSA_PSK cipher suites at the eCard-API, i.e. the eID-Server meets the SOAP or SAML profiles.
ALL_LINK	The eID-Server sends a complete link certificate chain to the eID-Client as recommended in [TR-03112].
RI_MIGRATION	The eID-Server is able to handle two sector public keys in the terminal's CV certificate and supports migration of sector-specific identifiers.
CONF_MAX_SE	The eID-Server meets SOAP and supports configuration of the maximum number of simultaneous sessions for one eService.
CONF_TM_OUT	The eID-Server meets SOAP and supports configuration of the time after which a session for SOAP communication at the eID-Interface expires.

Table 2: Optional/Recommended test profiles

Manufacturers are obliged to state which profiles are supported by their implementation through an Implementation Conformance Statement (ICS).

3 Implementation Conformance Statement

The purpose of the Implementation Conformance Statement is the declaration of supported functionality of the eID-Server to be approved by the applicant. The declarations of the applicant are used for the determination of the set of test cases to be performed.

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

3.1 Software Version

An applicant SHALL provide a declaration containing version of the software under test. Table 3 describes the required structure.

Element	Value
Name	
Vendor	
VersionMajor	
VersionMinor	
VersionSubminor	

Table 3: Software Version

3.2 Supported API Versions

An applicant SHALL also provide a declaration containing API versions supported by the software under test. Table 4 describes the required structure. This declaration MUST contain at least one version and provide the same information as the element `ServerVersion` sent in the `getServerInfoResponse` message to the eService if the eID-Interface (SOAP) is tested.

Element	Value
Major	
Minor	
Subminor (Bugfix)	

Table 4: Supported API Versions

3.3 Support of Multiple eServices

If the eID-Server is able to support different eServices and can utilize different terminal certificates respectively (multi-client capability), this MUST be declared here. In this case the vendor MUST support the testing laboratory in introducing the test certificates into the eID-Server and creating new clients.

Element	Yes / No
Support of multiple eServices	

Table 5: Multi-Client Capability

3.4 Profiles

An applicant SHALL provide a declaration containing information of the supported profiles. Table 6 describes the required basic test profiles.

Profile ID	Description	Yes / No
PAOS	The eID-Server has implemented Reverse SOAP (=PAOS) communication.	
EAC	The eID-Server has implemented Extended Access Control protocol according to [TR-03112].	
EID_ACCESS	The eID-Server is able to manage access to requested attributes	
REVOKED_CARD	The eID-Server has implemented revocation check and is capable to detect a revoked eID-Card based on sector-specific identifiers.	
EXPIRED_CARD	The eID-Server is capable to detect an expired eID-Card (via Document Validity Verification).	
NONAUTH_CARD	The eID-Server is capable to recognise a non-authentic eID-Card.	
CRYPTO	The eID-Server has implemented all requirements of [TR-03130-1] regarding the cryptography.	
DG_VARIATIONS	The eID-Server is able to process eID-Cards with missing or empty data groups.	

Table 6: Supported Basic Test Profiles

In addition, the applicant MUST specify which optional profiles are met. Table 7 describes the required structure of this declaration.

Profile ID	Description	Yes / No
SOAP	The eID-Server offers an (external) eID-Interface as described in [TR-03130-1].	
SOAP_TLS	The eID-Server meets SOAP and supports TLS during the SOAP communication on the eID-Interface if communicating through an otherwise open network (e. g. Internet).	
SAML	The eID-Server supports SAML-Profile Web Browser Single Sign-On (Web Browser SSO) profile specified in [SAML] Profiles, Section 4.1: Web Browser SSO Profile.	
SAML_ATTACHED	The eID-Server meets SAML and is attached to the SAML-Processor and operates according to the Attached eID-Server Model as described in [TR-03124-1] for SAML.	
ESER_ATTACHED	The eID-Server is attached to an eService and operates according to the Attached eID-Server Model as described in [TR-03124-1] for the eService.	
EIDAS_MW	The eID-Server operates in eIDAS middleware mode as described in [TR-03130-3].	
TLS_PSK	The eID-Server has implemented TLS-2 based on RSA_PSK cipher suites at the eCard-API, i.e. the eID-Server supports meets SOAP or not SAML_ATTACHED.	
ALL_LINK	The eID-Server sends a complete link certificate chain to the eID-Client as	

Profile ID	Description	Yes / No
	recommended in [TR-03112].	
RI_MIGRATION	The eID-Server is able to handle two sector public keys in the terminal's CV certificate and supports migration of sector-specific identifiers.	
CONF_MAX_SE	The eID-Server meets SOAP and supports configuration of the maximum number of simultaneous sessions for one eService.	
CONF_TM_OUT	The eID-Server meets SOAP and supports configuration of the time after which a session for SOAP communication at the eID-Interface expires.	

Table 7: Supported Optional/Recommended Test Profiles

If the eID-Server does not meet the profile CONF_MAX_SE, the applicant MUST provide a declaration confirming that the behaviour of the test object conforms with the requirements of [TR-03130-1] in case the maximum number of simultaneous sessions is exceeded.

If the eID-Server does not meet the profile CONF_TM_OUT, the applicant MUST provide a declaration confirming that the behaviour of the test object conforms with the requirements of [TR-03130-1] in case the session timeout occurs.

3.5 Server Address

The applicant MAY choose to perform the conformity tests with a test object placed at a remote location. In this case, the applicant SHALL provide a declaration of consent that data necessary to perform the test cases is exchanged over the internet. In order to be able to perform online tests, the manufacturer has to state the relevant addresses of the eID-Server instance to be tested. This structure contains four key-value pairs as depicted in Table 8.

Element	Value
eCard-API URI	
eID-Interface URI	
TC Token URI (if operating in attached mode)	
SAML-Processor-URI (if applicable)	

Table 8: Addresses of the eID-Server

3.6 Supported Cryptography

3.6.1 eCard API

An applicant SHALL provide a declaration containing information on the supported cryptography for TLS for communication between eID-Server and eID-Client. The declaration MUST be filled completely with all cryptographic parameters that are supported by the test object, i. e. other cryptographic parameters than listed in the ICS SHALL NOT be supported.

The applicant MUST use the ICS of [TR-03116-TS] to provide that declaration. The applicable parts of that ICS MUST be determined using the Mapping Document [TR-03116-TS-MD], chapter “eID-Server”, more precisely, the table “ICS data for eID-Servers”. The column ‘Mandatory ICS Data’ lists information that MUST be provided, the column ‘Optional ICS Data’ lists information that MAY be provided.

3.6.2 eID-Interface

If the profile SOAP is met, the applicant SHALL provide information about the supported cryptography for the eID-Interface.

3.6.2.1 TLS

If the profile SOAP_TLS is met, the applicant SHALL provide a declaration containing information on the supported cryptography for TLS. The declaration MUST be filled completely with all cryptographic parameters that are supported by the test object, i. e. other cryptographic parameters than listed in the ICS SHALL NOT be supported.

The applicant MUST use the ICS of [TR-03116-TS] to provide that declaration. The applicable parts of that ICS MUST be determined using the Mapping Document [TR-03116-TS-MD], chapter “eID-Server”, more precisely, the table “ICS data for eID-Servers”. The column ‘Mandatory ICS Data’ lists information that MUST be provided, the column ‘Optional ICS Data’ lists information that MAY be provided.

3.6.2.2 XML Signature

XML signatures are mandatory for SOAP communication with the eService. The applicant SHALL provide information about the supported signature and digest algorithms supported for XML signatures during SOAP communication. If the supported algorithms for generation and verification of signatures differ, the declaration MUST be filled for each method, separately.

Table 9 describes the structure of this declaration.

Signature method (URI)	Digest method	Canonicalization	Parameters (supported key lengths, elliptic curves, etc.)

Table 9: Supported XML signature algorithms for SOAP communication

Prior to the actual testing, the test laboratory MUST evaluate the statements of the ICS against the requirements of [TR-03130-1]. The result MUST be documented in the test report.

3.6.3 SAML Profile

If the profile SAML is met, the applicant SHALL provide information about the supported cryptography for the SAML profile. This includes transport protection via TLS supported by the SAML Processor and protection on content layer of the SAML communication via XML signatures and XML encryption.

3.6.3.1 TLS

The applicant SHALL provide a declaration containing information on the supported cryptography for TLS supported by the SAML-Processor. The declaration MUST be filled completely with all cryptographic parameters that are supported by the test object, i. e. other cryptographic parameters than listed in the ICS SHALL NOT be supported.

The applicant MUST use the ICS of [TR-03116-TS] to provide that declaration. The applicable parts of that ICS MUST be determined using the Mapping Document [TR-03116-TS-MD], chapter “eID-Server”, more precisely, the table “ICS data for eID-Servers”. The column ‘Mandatory ICS Data’ lists information that MUST be provided, the column ‘Optional ICS Data’ lists information that MAY be provided.

3.6.3.2 Signature Algorithms

The applicant SHALL provide information about the XML signature and digest algorithms supported for SAML messages. If the supported algorithms for generation and verification of signatures differ, the declaration MUST filled for each, separately.

Table 10 describes the structure of this declaration.

Signature method (URI)	**Digest method**	**Canonicalization**	**Parameters (supported key lengths, elliptic curves, etc.)**

Table 10: Supported XML signature algorithms for SAML communication

Prior to the actual testing, the test laboratory MUST evaluate the statements of the ICS against the requirements of [TR-03130-1]. The result MUST be documented in the test report.

3.6.3.3 XML Encryption

If the SAML Profile is supported by the test object, the applicant SHALL provide information about algorithms supported for SAML encryption. This includes key encryption, as well as content encryption of the data. If the supported algorithms for encryption and decryption differ, the declaration MUST be filled for each, separately. Each algorithm MUST be represented by the same URI that is used in the SAML message.

Table 11 describes the structure of this declaration.

<table>
<tr><td colspan="3"><b><i>Key Encryption algorithms</i></b></td></tr>
<tr><td><b>Key Transport algorithm (URI)</b></td><td colspan="2"><b>Parameters</b></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td><b>Key Agreement algorithm (URI)</b></td><td><b>Parameters</b></td><td><b>Key Wrapping algorithm (URI)</b></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3"><b><i>Content Encryption Algorithms (URI)</i></b></td></tr>
<tr><td colspan="3"></td></tr>
</table>

Table 11: Supported XML encryption algorithms for SAML communication

The test laboratory MUST verify that the declaration of the applicant is conform to the requirements of [TR-03130-1]. The result of the verification MUST be documented in the test report.

3.6.4 EIDAS_MW profile

If the profile EIDAS_MW is met, the applicant SHALL provide information about the supported cryptography. This includes protection on content layer of the eIDAS communication via XML signature and encryption.

3.6.4.1 TLS

As the eIDAS interface always operates as an attached eID-Server, it has to declare its supported TLS cryptography.

The declaration MUST be filled completely with all cryptographic parameters that are supported by the test object, i. e. other cryptographic parameters than listed in the ICS SHALL NOT be supported.

The applicant MUST use the ICS of [TR-03116-TS] to provide that declaration. The applicable parts of that ICS MUST be determined using the Mapping Document [TR-03116-TS-MD], chapter “eID-Server”, more precisely, the table “ICS data for eID-Servers”.The column ‘Mandatory ICS Data’ lists information that MUST be provided, the column ‘Optional ICS Data’ lists information that MAY be provided.

3.6.4.2 XML Signature

The applicant SHALL provide information about the XML signature and digest algorithms supported for SAML messages. If the supported algorithms for generation and verification of signatures differ, the declaration MUST filled for each, separately.

Table 12 describes the structure of this declaration.

Signature method (URI)	**Digest method**	**Canonicalization**	**Parameters (supported key lengths, elliptic curves, etc.)**

Table 12: Supported XML signature algorithms for eIDAS communication

Prior to the actual testing, the test laboratory MUST evaluate the statements of the ICS against the requirements of [TR-03130-3]. The result MUST be documented in the test report.

3.6.4.3 XML Encryption

The applicant SHALL provide information about algorithms supported for SAML encryption. This includes key encryption, as well as content encryption of the data. If the supported algorithms for encryption and decryption differ, the declaration MUST filled for each, separately. Each algorithm MUST be represented by the same URI that is used in the SAML message.

Table 13 describes the structure of this declaration.

<table>
<tr><td colspan="3"><b><i>Key Encryption algorithms</i></b></td></tr>
<tr><th>Key Transport algorithm (URI)</th><th colspan="2">Parameters</th></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><td></td><td colspan="2"></td></tr>
<tr><th>Key Agreement algorithm (URI)</th><th>Parameters</th><th>Key Wrapping algorithm (URI)</th></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td></td><td></td><td></td></tr>
<tr><td colspan="3"><b><i>Content Encryption Algorithms (URI)</i></b></td></tr>
<tr><td colspan="3"></td></tr>
<tr><td colspan="3"></td></tr>
<tr><td colspan="3"></td></tr>
</table>

Table 13: Supported XML encryption algorithms for eIDAS communication

The test laboratory MUST verify that the declaration of the applicant is conform to the requirements of [TR-03130-3]. The result of the verification MUST be documented in the test report.

3.6.5 Certificates

For XML Signatures and SAML Encryption, eService and eID-Server need to exchange appropriate X.509 certificates. The applicant MUST declare that only certificates are accepted that fulfil the cryptographic requirements of [TR-03130-1] and/or [TR-03130-3]. This applies for all external communication interfaces of the test object, like eID-Interface message signatures or SAML security.

3.6.6 Chip Authentication Algorithms

The applicant SHALL provide a declaration containing the supported algorithms for Chip Authentication and domain parameters according to [TR-03110]. The declaration MUST be filled completely with all parameters that are supported by the test object, i.e. other cryptographic parameters than listed in the Table 14 SHALL NOT be supported.

Table 14 describes the structure of this declaration.

Algorithms
Domain Parameters

Table 14: Supported algorithms for Chip Authentication

The test laboratory MUST verify that the declaration of the applicant is conform to the requirements of [TR-03116-2]. The result of the verification MUST be documented in the test report.

4 Definition of Configuration Data

4.1 Test Setup

This section presents the testing environment parameters to be used in the test setup of the conformance tests, see Table 15. These parameters are configured prior to the test begin and are constant for the complete test series.

Variable	Description
AGE	A valid year of life (Age) the owner of the eID-Card has completed.
AGE_NOK	A year of life (Age) the owner of the eID-Card has not yet completed.
ATTRIBUTE_A	Valid attribute present on the eID-Card.
ATTRIBUTE_ALL	Set of all data groups relevant for Online-Authentication, i.e. all attributes defined in the OperationsRequestorType.
ATTRIBUTE_SET_EIDAS	The eIDAS minimum dataset, mapped to the attributes of the German eID-Card as specified by [TR-03130-3]. This set contains all supported attributes.
ATTRIBUTE_SET_EIDAS_REQ	A subset of ATTRIBUTE_SET_EIDAS that only contains required attributes as specified by [TR-03130-3].
ATTRIBUTE_SET_EIDAS_OPT	A subset of ATTRIBUTE_SET_EIDAS that only contains optional attributes as specified by [TR-03130-3].
ATTRIBUTE_NONMINIMUM	The delta between the eIDAS minimum dataset and the attributes present on the German eID-Cards.
ATTRIBUTE_SET1	Set of the following attributes: Family Name, RestrictedID, Age Verification, CommunityID Verification.
ATTRIBUTE_DEN	Single attribute to which access is restricted by the user during the Online-Authentication.
ATTRIBUTE_OC	Set of attributes present on the eID-Card.
ATTRIBUTE_NOC	Set of attributes not present on the eID-Card.
COMMUNITYID	A valid CommunityID that is equal to the ID stored on the eID-Card.
COMMUNITYID_NOK	A valid CommunityID that is not equal to the ID stored on the eID-Card.
DGROUP_B	This is a valid data group that is present on the eID-Card and not equal to [ATTRIBUTE_DEN]. It is selected randomly from the set of remaining data groups available on the eID-Card.
DGROUP_5	Data group 5 (Family Name) of the eID application.
ESERVICEURL	Contains the exact valid URL of the eService which is used by the eID-Server to identify the eService.
ESERVICEURL_2	Another valid URL of the eService. It must differ from [ESERVICEURL].
ESERVICEURL_NOK	Contains an URL that differs from the correct eService URL. It is a syntactically valid URL that is not known to the eID-Server.
PSK	Variable containing a valid Pre Shared Key.
PSK_ODD	Variable containing [PSK] which, however, has been manipulated to have an odd number of HEX digits.
PSK_ID	Variable containing a unique random psk_identifier assigned to the PSK used in the

	current session. This value is encoded to the element SessionIdentifier in the TC Token.
PSK_INV	Variable containing [PSK] which, however, has been manipulated to have non HEX characters.
PSK_NOK	Variable containing random data with the same length as the expected [PSK].
PSK_SHORT	Variable containing less than 16 bytes, which is too short according to the schema definition shipped with [TR-03130-1].
RADDRESS	A https-URL including the default port number submitted to the eID-Client in the TC Token. It is used by the eID-Client to redirect the browser after conclusion of the Online-Authentication. This address is conform to the Same-origin policy according to [RFC6454] with the subjectURL contained in the CertificateDescription extension of the eService CV certificate.
RC	Counter variable that contains an integer. The selection of this number lies in the discretion of the sender. It is used to identify subsequent requests, which means that it is mandatory to properly increment it before sending a new message.
SID	Variable containing the valid session ID used by the eID-Server in the useID response and the eService in the getResult call to identify the Online-Authentication session.

Table 15: Variables referenced by the test cases

4.2 Certificate Specification

Due to the communication model specified in [TR-03130-1], sets of certificates and/or revocation lists for each supported interface are required. The required set of certificates includes CV as well as X.509 certificates.

This document defines unique names for certificates in order to present a clear description of the tests and the materials used thereby. Those names are placeholders and MAY differ from the ones used within a concrete testing environment.

The naming conventions are:

- `CERT_<interface>_<type>_<position>_<# of set>_{letter of variation}_[{subvariant of template}]`
- `BLACKLIST_<interface>_<type>_<position>_<# of set>_{letter of variation}_[{subvariant of template}]`

4.2.1 CERT_SET_1

This certificate set is the basic set of valid CV and X.509 certificates which are used in all test cases except EIDSERVER_C1_1_11.

To allow tests of all parameters listed in the ICS[2], it is necessary to perform the tests with differently configured eServices. For this purpose, the set contains template certificates. In order to perform all necessary test cases, the test laboratory MUST generate all relevant instances of these templates.

This certificate set defines several service providers that are used for different testing scenarios. If the test object is not multi-client capable, it may be necessary that the number of terminal certificates and revocation lists present in the test object is limited to one eService. In this case, only the certificates and revocation lists relevant for the particular test cases MUST be installed on the test object.

[2] E.g. for tests with varying crypto parameters.

4.2.1.1 Certificates for the eCard-API Interface and Card Communication

4.2.1.1.1 CERT_ECARD_CV_CVCA_1

Table 16 describes a CV certificate.

ID	CERT_ECARD_CV_CVCA_1
Purpose	This certificate is used as a regular CVCA certificate.
Description	This certificate is self-signed. It is a valid CV certificate stored as a Trust Point on the eID-Card. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_DV_1_A] of this set. The `CertificateHolderAuthorizationTemplate` contains the flags allowing to issue certificates for Authentication Terminals with authorization to get access to all the attributes listed in the OperationsRequestorType according to [TR-03130-1].

Table 16: Description of CERT_ECARD_CV_CVCA_1

4.2.1.1.2 CERT_ECARD_CV_DV_1_A

Table 17 describes a CV certificate.

ID	CERT_ECARD_CV_DV_1_A
Purpose	This certificate is used as a regular DV certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_CVCA_1]. It is a valid CV certificate accepted by the eID-Client. This certificate can be used to successfully verify Service CV certificates of this set. The `CertificateHolderAuthorizationTemplate` contains the flags allowing to issue certificates for Authentication Terminals with authorization to get access to all the attributes listed in the OperationsRequestorType according to [TR-03130-1].

Table 17: Description of CERT_ECARD_CV_DV_1_A

4.2.1.1.3 CERT_ECARD_CV_TERM_1_A

Table 18 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_A
Purpose	This certificate is used as a regular eService CV certificate issued for eService A.
Description	This certificate is issued for eService A and signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to get access to all the attributes listed in the OperationsRequestorType according to [TR-03130-1].

Table 18: Description of CERT_ECARD_CV_TERM_1_A

4.2.1.1.4 CERT_ECARD_CV_TERM_1_B

Table 19 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_B
Purpose	This certificate is used as a regular eService CV certificate for eService B.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to get access to the following attributes: Family Name; Restricted Identification, Age Verification and Community ID Verification.

Table 19: Description of CERT_ECARD_CV_TERM_1_B

4.2.1.1.5 CERT_ECARD_CV_TERM_1_C

Table 20 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_C
Purpose	This certificate is used as a regular eService CV certificate issued for eService C.
Description	This certificate is issued for eService C and signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` does not allow to read any of the 21 data groups or to perform any special function.

Table 20: Description of CERT_ECARD_CV_TERM_1_C

4.2.1.1.6 CERT_ECARD_CV_TERM_1_D

Table 21 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_D
Purpose	This certificate is used as a regular CV certificate for eService D.
Description	This certificate is issued for eService D and signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454].

	The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to perform Restricted Identifiaction. This certificate contains a second terminal sector for Restricted Identification (pseudonym).

Table 21: Description of CERT_ECARD_CV_TERM_1_D

4.2.1.1.7 CERT_ECARD_CV_TERM_1_E_*

Table 22 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_E_*
Purpose	This certificate is used as a regular CV certificate for eService E.
Description	This certificate is a template CV certificate representing all CV terminal certificates for eServices E_* used within the XML security tests according to the Modules Module A2: eID-Interface – XML Security and Module B2: SAML-Interface - XML Security. The certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to get access to the following attribute: Family Name.

*Table 22: Description of CERT_ECARD_CV_TERM_1_E_**

4.2.1.1.8 CERT_ECARD_CV_TERM_1_F

Table 23 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_F
Purpose	This certificate is used as a regular CV certificate for eService F.
Description	This certificate is issued for eServices F and signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to get access to [ATTRIBUTE_SET_EIDAS].

Table 23: Description of CERT_ECARD_CV_TERM_1_F

4.2.1.1.9 CERT_ECARD_CV_TERM_1_G

Table 24 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_1_G
Purpose	This certificate is used as a regular CV certificate for eService G
Description	This certificate is issued for eServices G and signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_1_A]. It is a valid CV certificate accepted by the eID-Client. The `CertificateDescription` of this eService certificate contains the hash of the TLS certificate defined within this set. The `RefreshAddress` [RADDRESS] given in the TC Token and the `subjectURL` contained in the `CertificateDescription` extension of the eService certificate conform to the Same-origin policy according to [RFC6454]. The hash of `CertificateDescription` is correctly stored in the appropriate extension. The `CertificateHolderAuthorizationTemplate` contains the authorization to get access to DG5.

Table 24: Description of CERT_ECARD_CV_TERM_1_G

4.2.1.1.10 CERT_ECARD_TLS_EIDSERVER_1_*

Table 25 describes a TLS certificate.

ID	CERT_ECARD_TLS_EIDSERVER_1_*
Purpose	This certificate is used for a regular TLS-2 connection establishment between the eID-Client and the eID-Server.
Description	This is a template certificate representing all X.509 certificate installed on the eID-Server[3]. The certificate is valid and accepted by the eID-Client. The hash of this certificate is contained in the `CertificateDescription` of the eService CV certificates defined within this set.

*Table 25: Description of CERT_ECARD_TLS_EIDSERVER_1_**

4.2.1.1.11 CERT_ECARD_TLS_SAMLPROCESSOR_1_*

Table 26 describes a TLS certificate if the eID-Server supports the profile SAML.

ID	CERT_ECARD_TLS_SAMLPROCESSOR_1_*
Purpose	This certificate is used for a regular TLS connection establishment between the eID-Client and the eID-Server's SAML-Processor.
Description	This is a template certificate respresenting all X.509 certificate installed on the eID-Server's SAML-Processor. Depending on the cipher suites supported by the SAML-Processor, this set of certificates MAY consist of several certificates, e.g. containing domain paramters for RSA, DSA, or ECDSA public keys. The certificate is valid and accepted by the eID-Clients. The hash of this certificate is contained in the `CertificateDescription` of the eService CV certificates defined within this set.

*Table 26: Description of CERT_ECARD_TLS_SAMLPROCESSOR_1_**

4.2.1.1.12 BLACKLIST_ECARD_EIDCARDS_TERM_1_<X>

Table 27 describes a revocation list of revoked eID-Cards for the eServices A, B, and C.

[3] If the eID-Server offers an eID-Interface (i.e. the eID-Server is not attached to the eService), this set usually consists of a single certificate based on RSA domain parameters. If the eID-Server is attached to the eService the set of certificates MAY consist of several certificates depending on the supported cipher suites, e.g. containing domain parameters for RSA, DSA, or ECDSA public keys.

ID	BLACKLIST_EIDCARDS_TERM_1_<X>
Purpose	This revocation list is as a regular revocation list of revoked eID-Cards for eService <X>.
Description	This is a template for a valid revocation list of revoked eID-Cards to enable the eID-Server to check the revocation status of the eID-Cards within the test series. It MUST be generated at least for eService B within this set. The revocation list contains the sector-specific revocation token of [EIDCARD_8].

Table 27: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_<X>

4.2.1.1.13 BLACKLIST_ECARD_EIDCARDS_TERM_1_D

Table 28 describes a revocation list of revoked eID-Cards for the eService D.

ID	BLACKLIST_EIDCARDS_TERM_1_D
Purpose	This revocation list is as a regular revocation list of revoked eID-Cards for eService D.
Description	This is a valid revocation list of revoked eID-Cards to enable the eID-Server to check the revocation status of the eID-Cards within the test series. The revocation list contains the first sector-specific revocation token of [EIDCARD_8], i.e. the current revocation token of this ID-Card. Additionally, this revocation list contains a revocation token that coincides with the old[4] revocation token of [EIDCARD_1].

Table 28: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_D

4.2.1.1.14 BLACKLIST_ECARD_EIDCARDS_TERM_1_E_*

Table 29 describes a revocation list of revoked eID-Cards for the eServices E_*.

ID	BLACKLIST_EIDCARDS_TERM_1_E_*
Purpose	This revocation list is as a regular revocation list of revoked eID-Cards for eService E_*.
Description	This is a template for a valid revocation list of revoked eID-Cards to enable the eID-Server to check the revocation status of the eID-Cards within the test series. It MUST be generated, if the eID-Server requires revocation lists for each eService. The revocation list is empty.

*Table 29: Description of BLACKLIST_ECARD_EIDCARDS_TERM_1_E_**

4.2.1.2 Certificates for eID-Interface

4.2.1.2.1 CERT_EID_TLS_ESERVICE_1_<X>

Table 30 describes TLS certificates for the eServices A, B, C, and D. They are used if the profile SOAP_TLS is met.

ID	CERT_EID_TLS_ESERVICE_1_<X>
Purpose	This certificate is used for a regular TLS connection establishment between eService <X> and the eID-Server for SOAP communication at the eID-Interface.
Description	This is a valid certificate accepted by the eID-Server for TLS at the eID-Interface, if SOAP_TLS is

[4] i.e. the revocation token that is calculated using the second sector public key in the Terminal sector extension of eService D.

	met.

Table 30: Description of CERT_EID_TLS_ESERVICE_1_<X>

4.2.1.2.2 CERT_EID_TLS_ESERVICE_1_E_*

Table 31 describes the TLS certificates for eService E_* that are used if the profile SOAP_TLS is met.

ID	CERT_EID_TLS_ESERVICE_1_E_*
Purpose	Removed
Description	Obsolete since version 1.1 (corresponding test cases were removed)

*Table 31: Description of CERT_EID_TLS_ESERVICE_1_E_**

4.2.1.2.3 CERT_EID_TLS_EIDSERVER_1_*

Table 32 describes a TLS certificate that is used if the profile SOAP_TLS is met.

ID	CERT_EID_TLS_EIDSERVER_1_*
Purpose	This certificate is used for regular TLS connection establishment between eService and the eID-Server for SOAP communication at the eID-Interface.
Description	This is a template certificate representing all TLS certificates with a private key used by the eID-Server for TLS at the eID-Interface. This is a valid certificate accepted by the eService for TLS at the eID-Interface, if SOAP_TLS is met. Depending on the cipher suites supported for TLS at the eID-interface, this set of certificates MAY consist of several certificates, e.g. containing domain parameters for RSA, DSA, or ECDSA public keys.

*Table 32: Description of CERT_EID_TLS_EIDSERVER_1_**

4.2.1.2.4 CERT_EID_XMLSIG_ESERVICE_1_<X>

Table 33 describes an X.509 certificate for the eServices A, B, C, and D. They are used if the profile SOAP is met.

ID	CERT_EID_XMLSIG_ESERVICE_1_<X>
Purpose	This certificate is used to verify regular XML signatures of eService <X> during SOAP communication with the eID-Server at the eID-Interface.
Description	This is a valid certificate accepted by the eID-Server for XML signatures at the eID-Interface.

Table 33: Description of CERT_EID_XMLSIG_ESERVICE_1_<X>

4.2.1.2.5 CERT_EID_XMLSIG_ESERVICE_1_E_*

Table 34 describes the X.509 certificates for eService E_* that are used if the profile SOAP is met.

ID	CERT_EID_XMLSIG_ESERVICE_1_E_*
Purpose	This certificate is used to verify regular XML signatures of eService E_* during SOAP communication with the eID-Server at the eID-Interface.
Description	This certificate is a template XML certificate representing all certificates of the eServices E_*. This is a valid c self-signed certificate accepted by the eID-Server for verification of XML signatures at the eID-Interface, if the profile SOAP is met. It is used within the XML security tests according to Module A2: eID-Interface – XML Security.

*Table 34: Description of CERT_EID_XMLSIG_ESERVICE_1_E_**

4.2.1.2.6 CERT_EID_XMLSIG_EIDSERVER_1_*

Table 35 describes X.509 certificates for XML Signature that are used if the profile SOAP is met.

ID	CERT_EID_XMLSIG_EIDSERVER_1_*
Purpose	This certificate is used for regular XML signatures of the eID-Server during SOAP communication with the eService at the eID-Interface.
Description	This is a template certificate representing all XML certificates with a private key used by the eID-Server for XML signatures at the eID-Interface. This is a valid certificate accepted by the eService for XML signatures, if SOAP is met. Depending on the signature algorithms supported by the eID-Server for generation, this set of certificates MAY consist of several certificates, e.g. containing domain parameters for RSA, DSA, or ECDSA public keys.

*Table 35: Description of CERT_EID_XMLSIG_EIDSERVER_1_**

4.2.1.3 Certificates for the SAML interface

4.2.1.3.1 CERT_SAML_XMLSIG_ESERVICE_1_<X>

Table 36 describes an X.509 certificate for the eServices A, B, C, and D. They are used if the profile SAML is met.

ID	CERT_SAML_XMLSIG_ESERVICE_1_<X>
Purpose	This certificate is used for regular XML signatures of eService <X> during communication with the eID-Server via SAML .
Description	This is a valid certificate accepted by the eID-Server's SAML-Processor for validation of XML signatures of eService <X> during SAML communication.

Table 36: Description of CERT_SAML_XMLSIG_ESERVICE_1_<X>

4.2.1.3.2 CERT_SAML_XMLSIG_ESERVICE_1_E_*

Table 37 describes an X.509 certificate for the eServices E. They are used if the profile SAML is met.

ID	CERT_SAML_XMLSIG_ESERVICE_1_E_*
Purpose	This certificate is used for regular XML signatures of eService E_* during communication with the eID-Server via SAML .
Description	This certificate is a template XML certificate representing all certificates of the eServices E_*. This is a valid certificate accepted by the eID-Server'S SAML-Processor for verification of XML signatures of SAML requests, if the profile SAML is met. It is used within the XML security tests according Module B2: SAML-Interface - XML Security.

*Table 37: Description of CERT_SAML_XMLSIG_ESERVICE_1_E_**

4.2.1.3.3 CERT_SAML_XMLSIG_EIDSERVER_1_*

Table 38 describes an X.509 certificate for SAML communication.

ID	CERT_SAML_XMLSIG_EIDSERVER_1_*
Purpose	This certificate is used for regular XML signatures of the eID-Server during SAML communication with the eService at the eID-Interface.
Description	This is a template certificate representing all XML certificates with a private key used by the eID-Server's SAML-Processor to sign the SAML response message for the eService. This is a valid certificate accepted by the eService for validation of XML signatures during SAML communications, if the profile SAML is met. Depending on the signature algorithms supported by the eID-Server for generation, this set of certificates MAY consist of several certificates, e.g. containing domain parameters for RSA, DSA, or ECDSA public keys.

*Table 38: Description of CERT_SAML_XMLSIG_EIDSERVER_1_**

4.2.1.3.4 CERT_SAML_XMLENC_ESERVICE_1_<X>

Table 39 describes an X.509 certificate for the eServices A, B, C, and D. They are used if the profile SAML is met.

ID	CERT_SAML_XMLENC_ESERVICE_1_<X>
Purpose	This certificate is used for regular XML encryption of eService <X> during SAML communication with the eID-Server.
Description	This valid certificate is used by the eID-Server's SAML-Processor to encrypt the SAML request for the eService <X> in case of SAML communication with the eID-Server's SAML-Processor.

Table 39: Description of CERT_SAML_XMLENC_ESERVICE_1_<X>

4.2.1.3.5 CERT_SAML_XMLENC_ESERVICE_1_E_*

Table 40 describes an X.509 certificate for SAML communication.

ID	CERT_SAML_XMLENC_ESERVICE_1_E_*
Purpose	This certificate is used for regular XML encryption of eService E_* during SAML communication with the eID-Server.
Description	This certificate is a template XML certificate representing all certificates of the eServices E_*. This is a valid certificate accepted by the eID-Server for encryption of SAML messages, if the profile SAML is met. It is used within the XML security tests according Module B2: SAML-Interface - XML Security.

*Table 40: Description of CERT_SAML_XMLENC_ESERVICE_1_E_**

4.2.1.3.6 CERT_SAML_XMLENC_EIDSERVER_1_*

Table 41 describes an X.509 certificate for SAML communication.

ID	CERT_SAML_XMLENC_EIDSERVER_1_*
Purpose	This certificate is used for regular XML encryption of the eID-Server during SAML communication with the eService
Description	This is a template certificate representing all XML certificates with a private key used by the eID-Server's SAML-Processor to decrypt the SAML request message for the eService in case of SAML communication. This is a valid certificate accepted by the eService for encryption of SAML messages, if the profile SAML is met. Depending on the algorithms supported by the eID-Server, this set of certificates MAY consist of several certificates, e.g. containing domain parameters for RSA public keys.

*Table 41: Description of CERT_SAML_XMLENC_EIDSERVER_1_**

4.2.1.4 Document Signer Certificates for eID-Cards

This section specifies the Document Signer certificate material used to define the card profiles for the test series.

4.2.1.4.1 CERT_ECARD_CSCA_1

Table 42 describes a CSCA certificate.

ID	CERT_ECARD_CSCA_1
Purpose	This certificate is used as a regular CSCA certificate.
Description	This certificate is self-signed. It is a valid X.509 certificate available on the eID-Server. This certificate can be used to successfully verify the certificate [CERT_ECARD_DS_1_A] and [CERT_ECARD_DS_1_B] of this set.

Table 42: Description of CERT_ECARD_CSCA_1

4.2.1.4.2 CERT_ECARD_DS_1_A

Table 43 describes a DS certificate.

ID	CERT_ECARD_DS_1_A
Purpose	This certificate is used as a regular DS certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CSCA_1]. It is a valid DS certificate accepted by the eID-Server. This certificate can be used to successfully verify the signature of EF.CardSecurity of the eID-Cards for this set, except for [EIDCARD_10] and [EIDCARD_11].

Table 43: Description of CERT_ECARD_DS_1_A

4.2.1.4.3 CERT_ECARD_DS_1_B

Table 44 describes a DS certificate.

ID	CERT_ECARD_DS_1_B

Purpose	This certificate is used as a revoked DS certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_EACRD_CSCA_1]. It is a valid DS certificate accepted by the eID-Server. The corresponding private key is used to sign the file EF.CardSecurity of [EIDCARD_10]. However, the certificate is revoked and contained on the corresponding CRL .

Table 44: Description of CERT_ECARD_DS_1_B

4.2.2 CERT_SET_2

This certificate set defines CV and TLS certificates which are used in [EIDSERVER_C1_1_11], where an alternative certificate chain is required. The CV chain is valid, but has been initiated by a newer root CA, not yet known to the eID-Card. Therefore, the eID-Card is not able to verify the eService CV certificate without a link certificate binding the new CVCA certificate to the old CVCA certificate. Note that if the eID-Card performs the update procedure, the state of the eID-Card will be changed afterwards.

This set is only used for tests at the eCard-API interface. Hence, certificates to be used at the remaining interfaces in order to fulfil the preconditions of the tests are out of scope of this document and is in responsibility of the testing laboratory.

4.2.2.1 Certificates for the eCard-API Interface and Card Communication

4.2.2.1.1 CERT_ECARD_CV_CVCA_2_A

Table 45 describes a CV certificate.

ID	CERT_ECARD_CV_CVCA_2_A
Purpose	This certificate is used as a regular root CVCA certificate in test cases where certificate handling is tested.
Description	This certificate is self-signed. It is a valid CV certificate stored as a Trust Point on the eID-Card. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_LINK_2_A] of this set.

Table 45: Description of CERT_ECARD_CV_CVCA_2_A

4.2.2.1.2 CERT_ECARD_CV_CVCA_2_B

Table 46 describes a CV certificate.

ID	CERT_ECARD_CV_CVCA_2_B
Purpose	This certificate is used as a regular root CVCA certificate in test cases where certificate handling is tested.
Description	This certificate is self-signed. It is a valid CV certificate stored as a Trust Point on the eID-Card. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_LINK_2_B] of this set.

Table 46: Description of CERT_ECARD_CV_CVCA_2_B

4.2.2.1.3 CERT_ECARD_CV_CVCA_2_C

Table 47 describes a CV certificate.

ID	CERT_ECARD_CV_CVCA_2_C
Purpose	This certificate is used as a regular root CVCA certificate in test cases where certificate handling is tested.
Description	This certificate is self-signed. It is a valid CV certificate stored as a Trust Point on the eID-Card. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_DV_2_A] of this set.

Table 47: Description of CERT_ECARD_CV_CVCA_2_C

4.2.2.1.4 CERT_ECARD_CV_LINK_2_A

Table 48 describes a CV certificate.

ID	CERT_ECARD_CV_LINK_2_A
Purpose	This certificate is used as regular CVCA link certificate in test cases where certificate handling is tested.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_CVCA_2_A]. It is a valid CV certificate accepted by the eID-Client. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_LINK_2_B] of this set.

Table 48: Description of CERT_ECARD_CV_LINK_2_A

4.2.2.1.5 CERT_ECARD_CV_LINK_2_B

Table 49 describes a CV certificate.

ID	CERT_ECARD_CV_LINK_2_B
Purpose	This certificate is used as regular CVCA link certificate in test cases where certificate handling is tested.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_LINK_2_A]. It is a valid CV certificate accepted by the eID-Client. This certificate can be used to successfully verify the certificate [CERT_ECARD_CV_DV_2_A] of this set.

Table 49: Description of CERT_ECARD_CV_LINK_2_B

4.2.2.1.6 CERT_ECARD_CV_DV_2_A

Table 50 describes a CV certificate.

ID	CERT_ECARD_CV_DV_2_A
Purpose	This certificate is used as regular DV certificate in test cases where the CVCA certificates stored on the eID-Card are outdated and cannot be validated without a corresponding Link-Certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_LINK_2_B]. It is a valid CV certificate accepted by the eID-Client. However, a link certificate needs to be sent first to bind the issuer of this certificate with the CVCA certificate

	stored on the eID-Card. This certificate can be used to successfully verify eService CV certificates of this set. The CertificateHolderAuthorizationTemplate contains full access rights of a non-official domestic Document Verifier.

Table 50: Description of CERT_ECARD_CV_DV_2_A

4.2.2.1.7 CERT_ECARD_CV_TERM_2_A

Table 51 describes a CV certificate.

ID	CERT_ECARD_CV_TERM_2_A
Purpose	This certificate is used as an eService CV certificate in test cases where the CVCA certificates stored on the eID-Card are outdated and cannot be validated without a corresponding Link-Certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CV_DV_2_A]. It is a valid CV certificate accepted by the eID-Client. The CertificateDescription of this eService certificate contains the hash of the TLS certificate defined within this set. The hash of CertificateDescription is correctly stored in the appropriate extension. The CertificateHolderAuthorizationTemplate contains the flags which enable the read all attributes relevant for Online-Authentication

Table 51: Description of CERT_ECARD_CV_TERM_2_A

4.2.2.1.8 CERT_TLS_EIDSERVER_2

Table 52 describes a TLS certificate.

ID	CERT_TLS_EIDSERVER_2
Purpose	This certificate is used for TLS-2 connection establishment between the eID-Client and the eID-Server in test cases where the hash of this certificate is not contained in the CertificateDescription of the eService CV certificate.
Description	This is a valid certificate initially accepted by the eID-Client. The hash of this certificate is not contained in the CertificateDescription of the eService CV certificate defined within this set.

Table 52: Description of CERT_TLS_EIDSERVER_2

4.2.2.2 Document Signer Certificates for eID-Cards

This section specifies the Document Signer certificate material used to define the card profiles for the test series.

4.2.2.2.1 CERT_ECARD_CSCA_2

Table 53 describes a CSCA certificate.

ID	CERT_ECARD_CSCA_2
Purpose	This certificate is used as a regular CSCA certificate.
Description	This certificate is self-signed. It is a valid X.509 certificate available on the eID-Server. This certificate can be used to successfully verify the certificate [CERT_ECARD_DS_2_A] of this set.

Table 53: Description of CERT_ECARD_CSCA_2

4.2.2.2.2 CERT_ECARD_DS_2

Table 54 describes a DS certificate.

ID	CERT_ECARD_DS_2
Purpose	This certificate is used as a regular DS certificate.
Description	This certificate is signed with the corresponding private key of the certificate [CERT_ECARD_CSCA_2]. It is a valid DS certificate accepted by the eID-Server. This certificate can be used to successfully verify the signature of EF.CardSecurity of the eID-Cards for this set.

Table 54: Description of CERT_ECARD_DS_2

4.3 eID-Cards

During Online-Authentication, particularities of the eID-Card will cause a different behaviour of the eID-Server. As a consequence, the test cases utilize multiple eID-Cards with different configurations and properties. The tests may be conducted using either physical or simulated cards.

4.3.1 Card Profiles for [CERT_SET_1]

Table 55 lists card profiles for [CERT_SET_1] as defined in Section 4.2.1 that have to be present in the laboratory in order to be able to run the corresponding test cases.

Card ID	Description
EIDCARD_1	This card profile represents a valid, default eID-Card accepting the certificate set [CERT_SET_1], i.e. containing the trust point [CERT_ECARD_CV_CVCA_1]. It further provides the access to all data groups and special functions defined within the tests cases. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_2	This card profile represents a valid eID-Card accepting the certificate set [CERT_SET_1], i.e. containing the trust point [CERT_ECARD_CV_CVCA_1]. DG6, DG10, DG13, DG19 and DG20 are not present. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_3	This card profile specifies a valid eID-Card based on profile [EIDCARD_1], but in DG8 day and month of birth are unknown, DG13 is empty, DG17 contains noPlaceInfo, and DG18 is empty. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_4_T	This card profile specifies a template of a valid eID-Card based on EIDCARD_1 where the supported algorithms for Chip Authentication varies. It is used to verify the support of algorithms for Chip Authentication. Each instance of the profile MUST contain exactly one algorithm of Table 14 in EF.CardSecurity, i. e. the eID-Server MUST NOT have a choice between different algorithms. It has either to accept the algorithm from EF.CardSecurity or to abort the EAC. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].

EIDCARD_5_T	This card profile specifies a template of a valid eID-Card based on [EIDCARD_1]. However, the keyID for its revocation Public Key is different from [EIDCARD_1]. It is used to verify that the eID-Server supports different key IDs for revocation check. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_6_T	This card profile specifies a template of a valid eID-Card based on [EIDCARD_1] where keyID for its pseudonym Public Key is different from [EIDCARD_1]. It is used to verify that the eID-Server supports different key IDs for the Restricted ID (pseudonym). The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_7	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the eID-Card is expired. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_8	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the eID-Card revocation token of the eID-Card is on the revocation list. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_9	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the signature of the card file EF.CardSecurity is manipulated. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_10	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the card file EF.CardSecurity is signed with revoked DS certificate [CERT_ECARD_DS_1_B].
EIDCARD_11	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the card file EF.CardSecurity is signed with a DS certificate not verifiable by the eID-Server.
EIDCARD_12_T	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the content of the card file EF.CardSecurity does not correspond with the respective content of the file EF.CardAccess. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A]
EIDCARD_13	This card profile specifies a template of a valid eID-Card that coincides with [EIDCARD_1] except that the public key for the Chip-Authentication stored in EF.CardSecurity is not a point of the corresponding elliptic curve. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A]
EIDCARD_14	This card profile represents a valid eID-Card accepting the certificate set [CERT_SET_1], i.e. containing the trust point [CERT_ECARD_CV_CVCA_1]. DG17 contains a non-German address. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].
EIDCARD_15	This card profile represents a valid eID-Card accepting the certificate set [CERT_SET_1], i.e. containing the trust point [CERT_ECARD_CV_CVCA_1]. It supports (at least) Chip Authentication Version 2 and Chip Authentication Version 3, indicated through the SecurityInfos in the files EF.CardAccess and EF.CardSecurity. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_1_A].

Table 55: eID-Cards for [CERT_SET_1] used during the tests

4.3.2 Card Profiles for [CERT_SET_2]

Table 56 lists card profiles for [CERT_SET_2] as defined in Section 4.2.2 that have to be present in the laboratory in order to be able to perform the corresponding test cases.

Card ID	Description
EIDCARD_101	This card profile represents a valid eID-Card based on the certificate set [CERT_SET_2] that contains the trust point [CERT_ECARD_CV_CVCA_2_A] as trust point.

	The card file EF.CardSecurity is signed with [CERT_ECARD_DS_2].
EIDCARD_102	This card profile represents a valid ID-Card based on the certificate set [CERT_SET_2] that contains two trust points, i.e. [CERT_ECARD_CV_CVCA_2_A] and [CERT_ECARD_CV_CVCA_2_B]. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_2].
EIDCARD_103	This card profile represents a valid eID-Card based on the certificate set [CERT_SET_2] that contains the trust point [CERT_ECARD_CV_CVCA_2_C] as trust point. The card file EF.CardSecurity is signed with [CERT_ECARD_DS_2].

Table 56: eID-Cards for [CERT_SET_2] used during the tests

5 Definition for Test Cases

This chapter explains the test case notation and commonly used elements.

5.1 Test Case Notation

All test cases are described within a set of XML files. An overview over the corresponding XML scheme is given in the following. The scheme is particularly designed to meet the requirements of eID-Server tests.

As depicted in Figure 2, each test is an object of the type `TestCase`. All test cases are organized hierarchically which is realized in XML using the abstract base type called `TestHierarchy`.

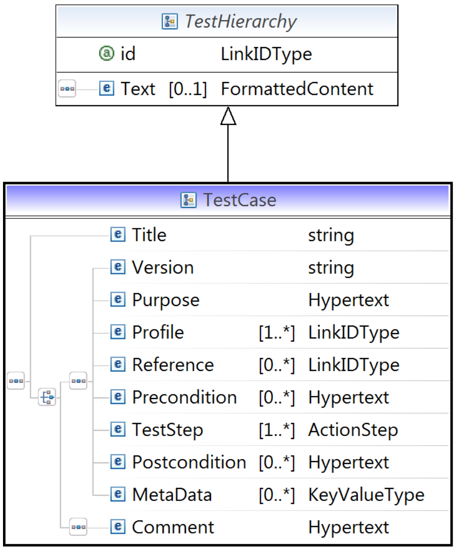


Figure 2: XML Schema Test Case

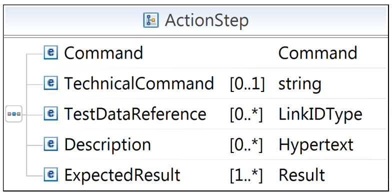
Each `TestCase` object has a unique *id* attribute and contains the following elements:

- `Title`
  title of the test case.
- `Version`
  current version of the test case.
- `Purpose`
  a short description of the intention of the test.
- `Profile`
  links to all relevant profiles.
- `Reference`
  optional reference to any kind of specification this test case is based on.
- `Precondition`
  all requirements which need to be fulfiled before running the test.
- `TestStep`
  this XML element is a complex type and consists of the different sub-elements addressed below.
- `Postcondition`
  the description of conditions which may be met after the test completion

- MetaData
  optional elements in form of key-value pairs containing meta information.

If a test has been moved or deleted, the body of TestCase only contains a Title and a respective description in the Comment element.

The TestStep object of type ActionStep is used at least once and contains the elements from Figure 3.



ActionStep

Command		Command
TechnicalCommand	[0..1]	string
TestDataReference	[0..*]	LinkIDType
Description	[0..*]	Hypertext
ExpectedResult	[1..*]	Result

Figure 3: XML Schema ActionStep

In particular, it consists of:

- Command
  represents the actual action that is performed within a single step.
- TechnicalCommand
  can optionally be used to specify a technical representation of the command to be able to process the step automatically by some testing suite.
- TestDataReference
  If the step refers to some predefined test data, such as certificates, the data element is referred using this element.
- Description
  adds further information about the command that is performed in the step.
- ExpectedResult
  denotes the behaviour of the test object in order to pass the test.

5.2 Commonly Used Elements

This chapter summarizes the messages (including their structure) which are used during the communication with an eID-Server. Each XML message coming from the eID-Server MUST be validated against the requirements of [TR-03130-1] and [TR-03112]. The result MUST be respected for the outcome of the corresponding test case. Furthermore, the conformity of HTTP messages to [RFC2616] and [RFC2818] MUST be evaluated and respected as well.

Each description of XML messages from the eID-Client or the eService contains a number of default values. The parameters deviating from the listed default values are described directly in the respective test cases.

Note that the Result elements exchanged during the test series must use the URIs defined in [TR-03130-1], e.g.

- http://www.bsi.bund.de/ecard/api/1.1/resultmajor#ok resp.
- http://www.bsi.bund.de/ecard/api/1.1/resultmajor#error.

For simplicity, the test case definitions provide a short form of this element, i. e. without the leading URI. However the testlaboratory MUST check the complete form of the Result element.

This chapter summarizes the messages including their structure which are used during the communication with an eID-Client.

5.2.1 TC Token

TC Token is an XML fragment defined in [TR-03124-1]. This section specifies default TC Token values utilized in the most test cases. Individual parameters deviating from these for a particular test case will be described directly in the respective test definition.

In general, the TC Token type presented in the Table 57 has the full length, which means it contains all parameters specified for a TC Token. In deviation from this, in case of an Attached eID-Server, i.e. the channel TLS-1-2 is not used, the elements `<PathSecurity-Protocol>` and `<PathSecurity-Parameters>` MUST be omitted.

<table>
<tr><td><b>ID</b></td><td>[TC_TOKEN]</td></tr>
<tr><td><b>Description</b></td><td>The TC Token entirely conforms to the specification provided in the [TR-03124-1]. Any deviations from the elements described below are provided within the respective test case description.</td></tr>
<tr><td><b>Content definition</b></td><td><pre>
<TCTokenType>
  <ServerAddress>{ServerAddress}</ServerAddress>
  <SessionIdentifier>{PSK_ID}</SessionIdentifier>
  <RefreshAddress>{RAddress}</RefreshAddress>
  <CommunicationErrorAddress>

   {CommunicationErrorAddress}

  </CommunicationErrorAddress>
  <Binding>urn:liberty:paos:2006-08</Binding>
  <PathSecurity-Protocol>urn:ietf:rfc:4279</PathSecurity-Protocol>
  <PathSecurity-Parameters>
   <PSK>{PSK}</PSK>
  </PathSecurity-Parameters>
</TCTokenType>
</pre></td></tr>
<tr><td><b>Default variable element values</b></td><td>
<table>
<tr><th>Variable</th><th>Value</th></tr>
<tr><td>{ServerAddress}</td><td>A https-URL submitted to the eID-Client in the TC Token. It is used by the eID-Client to connect to the eID-Server.</td></tr>
<tr><td>{PSK_ID}</td><td>A unique random <code>psk_identifier</code> identifying the PSK used for the current session.</td></tr>
<tr><td>{RAddress}</td><td>A https-URL including the default port number submitted to the eID-Client in the TC Token. It is used by the eID-Client to redirect the browser after conclusion of the Online-Authentication. This address is conform to the Same-origin policy according to [RFC6454] with the subjectURL contained in the CertificateDescription extension of the eService CV certificate.</td></tr>
<tr><td>{CommunicationErrorAddress}</td><td>A https-URL submitted to the eID-Client in the TC Token. It is used by the eID-Client to redirect the browser if a communication error occurred and no valid refreshURL could be determined.</td></tr>
<tr><td>{PSK}</td><td>A randomly set PSK (omitted for an Attached Server).</td></tr>
</table>
</td></tr>
</table>

Table 57: Structure of a TC Token

5.2.2 StartPAOS

`StartPAOS` is an XML message sent from the eID-Client to the eID-Server. It establishes a PAOS channel which is used until the eID-Server returns `StartPAOSResponse`. A typical content of this message is described in Table 58. The default values provided there have to be altered for some test cases. Any deviations are described directly in the respective test definition.

ID	StartPAOS
Description	The function `StartPAOS` is used for the establishment of a PAOS communication. The eID-Client SHALL incorporate information about connected card terminals and card applications in form of `ConnectionHandle`-elements into the `StartPAOS`-structure.

Content definition	`<StartPAOS>`
	` <SessionIdentifier>{SessionIdentifier}</SessionIdentifier>`
	` <ConnectionHandle>{ConnectionHandle}</ConnectionHandle>`
	` <UserAgent>`
	`  <Name>{Name}</Name>`
	`  <VersionMajor>{VersionMajor}</VersionMajor>`
	`  <VersionMinor>{VersionMinor}</VersionMinor>`
	`  <VersionSubminor>{VersionSubminor}</VersionSubminor>`
	` </UserAgent>`
	` <SupportedAPIVersions>`
	`  <Major>{Major}</Major>`
	`  <Minor>{Major}</Minor>`
	`  <Subminor>{Subminor}</Subminor>`
	` </SupportedAPIVersions>`
	` <SupportedDIDProtocols>`
	`  {SupportedDIDProtocols}`
	` </SupportedDIDProtocols>`
	`</StartPAOS>`

Table 58: Structure of a StartPAOS message

5.2.3 InitializeFramework

The `InitializeFramework` function initializes the eCard-API-Framework and can be used to query the version of the framework implementation. It has no invocation parameters and is sent from the eID-Server to the eID-Client.

5.2.4 InitializeFrameworkResponse

`InitializeFrameworkResponse` is a message returned to the eID-Server. It is described in Table 59. Afterwards, the eCard-API-Framework is initialized, and the functions available according to the APIACL can be invoked by the client application.

ID	InitializeFrameworkResponse
Description	The function `InitializeFrameworkResponse` is used to respond the `InitializeFramework`. The eID-Client SHALL incorporate the status information and the version of the eCard-API-Framework started with this function.

Content definition	`<InitializeFrameworkResponse>`
	` <Result>`
	`  {Result}`
	` </Result>`
	` <Version>`
	`  <Major>{Major}</Major>`
	`  <Minor>{Minor}</Minor>`
	`  <SubMinor>{SubMinor}</SubMinor>`
	` </Version>`
	`</InitializeFrameworkResponse>`

Table 59: Structure of an InitializeFrameworkResponse message

5.2.5 DIDAuthenticate_EAC1InputType

In the Phase 1 of the EAC, the eID-Server invokes `DIDAuthenticate` with the `DIDName` provided for PACE and `AuthenticationProtocolData` of the `EAC1InputType` explained in more detail below in Table 60.

ID	EAC1InputType
Description	The message EAC1InputType is used in the Phase 1 of the Extended PACE protocol and contain the necessary information to start the EAC.

<table>
<tr><td><b>Content definition</b></td><td>

```
<DIDAuthenticate>
 <ConnectionHandle type="ConnectionHandleType">
  {ConnectionHandle}
 </ConnectionHandle>
 <DIDName>PIN</DIDName>
 <AuthenticationProtocolData type="EAC1InputType">
  <Certificate>{Certificate}</Certificate>
   <CertificateDescription>
   {CertificateDescription}
  </CertificateDescription>
  <RequiredCHAT>{RequiredCHAT}</RequiredCHAT>
  <OptionalCHAT>{OptionalCHAT}</OptionalCHAT>
  <AuthenticatedAuxiliaryData>
   {AuthenticatedAuxiliaryData}
  </AuthenticatedAuxiliaryData>
  <TransactionInfo>{TransactionInfo}</TransactionInfo>
 </AuthenticationProtocolData>
</DIDAuthenticate>
```

</td></tr>
<tr><td><b>Default variable element values</b></td><td>

Variable	Value
{Certificate}	CV certificate of the eService.
{CertificateDescription}	A regular certificate description containing the required set of elements.
{RequiredCHAT}	Specifies the data, which are required by the eService.
{OptionalCHAT}	Specifies the data, which are requested by the eService, but which transmission may be suppressed by the user.
{AuthenticatedAuxiliaryData}	MAY contain additional data which are used to check the validity of the card, verify the age or check municipality citizenship.
{TransactionInfo}	This element is not sent by default.

</td></tr>
</table>

Table 60: Structure of DIDAuthenticate_EAC1InputType

5.2.6 DIDAuthenticateResponse_EAC1OutputType

This message is an XML data set in response to the `EAC1InputType` message received from the eID-Server.

ID	DIDAuthenticateResponse_EAC1OutputType
Description	This message is sent by the eID-Client in response to the EAC1InputType message received from the eID-Server.

Content definition	<DIDAuthenticateResponse>
	 <Result>
	  {Result}
	 </Result>
	 <AuthenticationProtocolData type="EAC1OutputType">
	  <CertificateHolderAuthorizationTemplate>
	   {CHAT}
	  </CertificateHolderAuthorizationTemplate>
	  <CertificationAuthorityReference>
	   {CAR}
	  </CertificationAuthorityReference>
	  <EFCardAccess>{EFCardAccess}</EFCardAccess>
	  <IDPICC>{IDPICC}</IDPICC>
	  <Challenge>{Challenge}</Challenge>
	 </AuthenticationProtocolData>
	</DIDAuthenticateResponse>

Table 61: Structure of DIDAuthenticateResponse_EAC1OutputType

5.2.7 DIDAuthenticate_EAC2InputType

Using the Chip Authentication domain parameters, the eID-Server generates a fresh key pair in the next step, forms an appropriate chain of additionally required certificates and finally, where required, signs the `Challenge` which has been transmitted.
The eID-Server then invokes `DIDAuthenticate` and relays `Authentication ProtocolData` of type `EAC2InputType`, which is described in more detail below in Table 62, to the eID-Client.

ID	DIDAuthenticate_EAC2InputType
Description	The message EAC2InputType defined here describes possible elements sent to the eID-Client.
Content definition	<DIDAuthenticate> <ConnectionHandle type="ConnectionHandleType"> {ConnectionHandle} </ConnectionHandle> <DIDName>PIN</DIDName> <AuthenticationProtocolData type="EAC2InputType"> <Certificate>{Certificate}</Certificate> <EphemeralPublicKey>{EphemeralPublicKey}</EphemeralPublicKey> <Signature>{Signature}</Signature> </AuthenticationProtocolData> </DIDAuthenticate>

Default variable element values	**Variable**	**Value**
	{Certificate}	This element is not sent by default.
	{EphemeralPublicKey}	A freshly generated correctly encoded ephemeral public key.
	{Signature}	A correctly signed challenge.

Table 62: Structure of DIDAuthenticate_EAC2InputType

5.2.8 DIDAuthenticateResponse_EAC2OutputType_A

This type specifies the structure of the `EAC2OutputType` which is used in the EAC protocol on the second request of `DIDAuthenticate`.

ID	DIDAuthenticateResponse_EAC2OutputType_A
Description	This message describes is sent of the eID-Client in response to the EAC2InputType message received from the eID-Server.

Content definition	`<DIDAuthenticateResponse>`
	` <Result> {Result} </Result>`
	` <AuthenticationProtocolData type="EAC2OutputType">`
	`  <EFCardSecurity>{EFCardSecurity}</EFCardSecurity>`
	`  <AuthenticationToken>{AT}</AuthenticationToken>`
	`  <Nonce>{Nonce}</Nonce>`
	` </AuthenticationProtocolData>`
	`</DIDAuthenticateResponse>`

Table 63: Structure of DIDAuthenticateResponse_EAC2OutputType_A

5.2.9 DIDAuthenticateResponse_EAC2OutputType_B

This type specifies the structure of the `EAC2OutputType` which is used in the EAC protocol on the second request of `DIDAuthenticate`. It is sent in case the eID-Clients repeats the challenge in DIDAuthenticateResponse_EAC2OutputType even after receiving DIDAuthenticate_EAC2InputType containing signature

ID	DIDAuthenticateResponse_EAC2OutputType_B
Description	This message describes is sent of the eID-Client in response to the EAC2InputType message received from the eID-Server.

Content definition	`<DIDAuthenticateResponse>`
	` <Result> {Result} </Result>`
	` <AuthenticationProtocolData type="EAC2OutputType">`
	`  <Challenge>{Challenge}</Challenge>`
	` </AuthenticationProtocolData>`
	`</DIDAuthenticateResponse>`

Table 64: Structure of DIDAuthenticateResponse_EAC2OutputType_B

5.2.10 Transmit

The `Transmit` function sends one or more APDU(s) to a connected eID-Card. In order to support the batch processing a set of `AcceptableStatusCode`-elements (9000 etc.) MAY be attached to each `InputAPDU`.

The test cases are formulated in a way that the eID-Server uses a single `Transmit` message to retrieve all data at once. However, it is possible also for the server to send multiple `Transmit` messages to fetch distinct attributes

<table>
<tr><td><b>ID</b></td><td>Transmit</td></tr>
<tr><td><b>Description</b></td><td>This message contains APDUs sent to the eID-Card.</td></tr>
<tr><td><b>Content definition</b></td><td><pre>
<Transmit>
 <SlotHandle>{SlotHandle}</SlotHandle>
 <InputAPDUInfo>
  <InputAPDU>{InputAPDU}</InputAPDU>
  <AcceptableStatusCode>{ASC}</AcceptableStatusCode>
 </InputAPDUInfo>
</Transmit>
</pre></td></tr>
<tr><td><b>Default variable element values</b></td><td>
<table>
<tr><th>Variable</th><th>Value</th></tr>
<tr><td>{InputAPDU}</td><td>Each APDU is described in the respective test case.</td></tr>
<tr><td>{ASC}</td><td>This element is not sent by default.</td></tr>
</table>
</td></tr>
</table>

Table 65: Structure of Transmit

5.2.11 TransmitResponse

The `TransmitResponse` function defines the return of the `Transmit` function.

<table>
<tr><td><b>ID</b></td><td>TransmitResponse</td></tr>
<tr><td><b>Description</b></td><td>This message describes the response to the <code>Transmit</code> function.</td></tr>
<tr><td><b>Content definition</b></td><td><pre>
<TransmitResponse>
 <Result>
  {Result}
 </Result>
 <OutputAPDU>{OutputAPDU}</OutputAPDU>
</TransmitResponse>
</pre></td></tr>
</table>

Table 66: Structure of TransmitResponse

5.2.12 StartPAOSResponse

The `StartPAOSResponse` command closes the PAOS channel and returns to the established TLS session.

<table>
<tr><td><b>ID</b></td><td>StartPAOSResponse</td></tr>
<tr><td><b>Description</b></td><td>This message finishes PAOS communication.</td></tr>
<tr><td><b>Content definition</b></td><td>

```
<StartPAOSResponse>
 <Result>{Result}</Result>
</StartPAOSResponse>
```

</td></tr>
</table>

Table 67: Structure of StartPAOSResponse

5.2.13 useIDRequest

The `useIDRequest` function can be used by the eService to communicate the data groups to be read to the eID-Server. Furthermore, it may also be used to request Age Verification and CommunityID Verification and to transport the necessary data for comparison. It is also possible for the eService to send a custom PSK that is to be used during the further communication.

<table>
<tr><td><b>ID</b></td><td>useIDRequest</td></tr>
<tr><td><b>Description</b></td><td>This message contains all data groups that the eService requests to be read by the eID-Server and optional comparison data for the Age Verification and the CommunityID Verification. It may also contain a PSK to be used.</td></tr>
<tr><td><b>Content definition</b></td><td>

```
<useIDRequest>
   <UseOperations>
      <{DATA_GROUP}>REQUIRED</{DATA_GROUP}>
   </UseOperations>
   <AgeVerificationRequest>
      <Age>{AGE}</Age>
   </AgeVerificationRequest>
   <PlaceVerificationRequest>
      <CommunityID>{COMMUNITYID}</CommunityID>
   </PlaceVerificationRequest>
   <PSK>{PSK}</PSK>
</useIDRequest>
```

</td></tr>
<tr><td><b>Default variable element values</b></td><td>
<table>
<tr><th>Variable</th><th>Value</th></tr>
<tr><td>{DATA_GROUP}</td><td>Data group to be read from the card.</td></tr>
<tr><td>{AGE}</td><td>As described in Section 4.1, Table 15.</td></tr>
<tr><td>{COMMUNITYID}</td><td>As described in Section 4.1, Table 15.</td></tr>
<tr><td>{PSK}</td><td>This element is not sent by default.</td></tr>
</table>
</td></tr>
</table>

Table 68: Structure of useIDRequest

5.2.14 useIDResponse

The `useIDResponse` function is used by the eID-Server to indicate the start of a new session. It also contains the PSK to be used for the communication with the eID-Client and may contain the URI of the eID-Servers eCard-API framework.

ID	useIDResponse
Description	This message contains the SessionID and the Pre Shared Key that is to be used during the communication.

Content definition	`<useIDResponse>`
	`   <Session>`
	`      <ID>{SID}</ID>`
	`   </Session>`
	`   <eCardServerAddress>{SERVER_ADDRESS}</eCardServerAddress>`
	`   <PSK>`
	`      <ID>{PSK_ID}</ID>`
	`      <Key>{PSK}</Key>`
	`   </PSK>`
	`   <Result>`
	`      {Result}`
	`   </Result>`
	`</useIDResponse>`

Default variable element values	<table><tr><th>Variable</th><th>Value</th></tr><tr><td>{SID}</td><td>As described in Section 4.1, Table 15.</td></tr><tr><td>{PSK_ID}</td><td>As described in Section 4.1, Table 15.</td></tr><tr><td>{PSK}</td><td>As described in Section 4.1, Table 15.</td></tr><tr><td>{SERVER_ADDRESS}</td><td>The address of the eID-Servers eCard-API framework as URI.</td></tr></table>

Table 69: Structure of useIDResponse

5.2.15 getResultRequest

The `getResultRequest` function is used by the eService to retrieve the results of the `useIDRequest` it has sent to the eID-Server. This type of request is continuously sent to the eID-Server until either the results of the computation are ready or a fatal error occurs. This is the request part of the result Polling Mechanism as referenced by the test cases.

ID	getResultRequest
Description	This message contains the session ID that matches a previous `useIDRequest` and a counter which is incremented for every subsequent request using this SID.

<table>
<tr><td><b>Content definition</b></td><td>&lt;getResultRequest>
   <Session>
      &lt;ID>{SID}&lt;/ID>
   </Session>
   <RequestCounter>{RC}</RequestCounter>
</getResultRequest&gt;</td></tr>
<tr><td><b>Default variable element values</b></td><td><table><tr><th>Variable</th><th>Value</th></tr><tr><td>{SID}</td><td>As described in Section 4.1, Table 15.</td></tr><tr><td>{RC}</td><td>As described in Section 4.1, Table 15.</td></tr></table></td></tr>
</table>

Table 70: Structure of getResultRequest

5.2.16 getResultResponse

The `getResultResponse` function is used by the eID-Server to return the current processing status or the result of the Online-Authentication to the eService. This is the response part of the Polling Mechanism as referenced by the test cases.

<table>
<tr><td><b>ID</b></td><td>getResultResponse</td></tr>
<tr><td><b>Description</b></td><td>This message contains the current status of the Online-Authentication and the actual authentication data once it is completely processed.</td></tr>
<tr><td><b>Content definition</b></td><td>&lt;getResultResponse>
   <PersonalData>
      {PersonalData}
   </PersonalData>
   <FulfilsAgeVerification>
      {FulfilsAgeVerification}
   </FulfilsAgeVerification>
   <FulfilsPlaceVerification>
      {FulfilsPlaceVerification}
   </FulfilsPlaceVerification>
   <OperationsAllowedByUser>
      {OperationsAllowedByUser}
   </OperationsAllowedByUser>
   <Result>
      {Result}
   </Result>
</getResultResponse&gt;</td></tr>
</table>

Table 71: Structure of getResultResponse

5.2.17 getServerInfoRequest

The `getServerInfoRequest` function can be used at any time during the Online-Authentication. It is a `nullType` message and is sent by the eService in order to obtain data about the eID-Server.

5.2.18 getServerInfoResponse

The `getServerInfoResponse` function is used by the eID-Server to return information about the version of the eID-Interface that the server currently implements and the operations that the eService may perform using it's current CV certificate.

ID	getServerInfoResponse
Description	This message contains the version of the eID-Interface that is currently implemented by the eID-Server and the operations that the eService may perform using its CV certificate.

Content definition	`<getServerInfoResponse>`
	`   <ServerVersion>`
	`      {ServerVersion}`
	`   </ServerVersion>`
	`   <DocumentVerificationRights>`
	`      {DocumentVerificationRights}`
	`   </DocumentVerificationRights>`
	`</getServerInfoResponse>`

Table 72: Structure of getServerInfoResponse

6 Test Specification

This chapter defines the test cases based on the specifications described in [TR-03130-1] and [TR-03112]. The test cases are grouped in modules according to the communication interfaces of the eID-Server, i.e.

- eID-Interface (Module A, Section 6.1)
- SAML-Profile (Module B, Section 6.2)
- eCard-API-Interface (Module C, Section 6.3)
- eIDAS middleware mode (Module D, Section 6.4)

The modules are divided into sub-modules. Each module has its own preconditions and goals, which are evaluated in the respective test cases. All test cases are described within a set of XML files. Each test is an object of the type `TestCase` (as defined in Chapter 5).

Both, the correctness and the completeness of the implementation are put under test. Furthermore, it is crucial to verify that the respective responses of the eID-Server contain the exact data read from the chip of the eID-Card. Therefore, the testing laboratory MUST ensure this consistency for each test case.

6.1 Module A: eID-Interface

The modules in this group evaluate the proper implementation of the eID-Interface as described in [TR-03130-1] by the test object.

6.1.1 Module A1: eID-Interface - Functional eID-Interface Operation

This module is divided into four parts which deal with the communication of the eService and the eID-Server at the eID-Interface.

6.1.1.1 Module A1_1: Processing of Regular Online-Authentication (eID-Interface)

This submodule tests the behaviour of the eID-Server at the eID-Interface in case of a regular Online-Authentication on application level. This includes the requests and responses for reading data groups and performing special functions. The test cases for this module are listed in Table 73.

ID	Purpose	Instruction	Profiles
EIDSERVER_A1_1_01_T	Positive test describing the eID-Interface interaction. The eService request requires to read a particular attribute.	The test has to be performed for each operation defined in the requester type. The test has to be performed with [EIDCARD_1] and [EIDCARD_14].	SOAP, EID_ACCESS
EIDSERVER_A1_1_02_T	Positive test describing the eID-Interface interaction. The eService request allows to read a particular attribute	The test has to be performed for each operation contained in the requester type.	SOAP, EID_ACCESS
EIDSERVER_A1_1_03_T	Positive test describing the eID-Interface interaction in case the user restricts access to a required attribute.	The test has to be performed for each operation contained in the requester type.	SOAP, EID_ACCESS

ID	Purpose	Instruction	Profiles
EIDSERVER_ A1_1_04_T	Positive test describing the eID-Interface interaction in case the user restricts access to an optional attribute.	The test has to be performed for each operation contained in the requester type.	SOAP, EID_ACCESS
EIDSERVER_ A1_1_05_T	Positive test describing the eID-Interface interaction in case of requested data groups being not present or empty.	The test has to be performed with [EIDCARD_2] and [EIDCARD_3].	SOAP, EID_ACCESS, DG_VARIATIONS
EIDSERVER_ A1_1_06	Positive test of the eID-Interface interaction in case the terminal CV certificate of the eService contains two terminal sectors.	-	SOAP, EID_ACCESS, RI_MIGRATION
EIDSERVER_ A1_1_07	Positive test describing the eID-Interface interaction in case Age and Place Verification have negative results.	-	SOAP, EID_ACCESS

Table 73: Test Cases of Module A1_1

6.1.1.2 Module A1_2: Processing of Irregular Online-Authentication (eID-Interface)

This submodule tests the behaviour of the eID-Server at the eID-Interface in case of an irregular Online-Authentication. Table 74 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ A1_2_01_T	This test case checks the behaviour of the eID-Server at the eID-Interface in case the eService requests to read a data group without having the corresponding authorization in the terminal certificate.	The test has to be performed for each operation contained in the requester type.	SOAP, EID_ACCESS
EIDSERVER_ A1_2_02_T	This test checks the behaviour of the eID-Server at the eID-Interface in case Passive Authentication failed.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_01], [EIDSERVER_C1_2_02], [EIDSERVER_C1_2_03], and [EIDSERVER_C1_2_04].	SOAP, NONAUTH_CARD,
EIDSERVER_ A1_2_03_T	This test case checks the behaviour of the eID-Server at the eID-Interface in case Chip Authentication failed.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_05], and [EIDSERVER_C1_2_06].	SOAP, NONAUTH_CARD
EIDSERVER_ A1_2_04	This test case checks the behaviour of the eID-Server at the eID-Interface in case of a revoked ID card.	-	SOAP, REVOKED_CARD
EIDSERVER_ A1_2_05	This test case checks the behaviour of the eID-Server at the eID-Interface in case the ID card is expired.	-	SOAP, EXPIRED_CARD

ID	Purpose	Instruction	Profiles
EIDSERVER_ A1_2_06_T	This test checks the behaviour of the eID-Server at the eID-Interface in case of a secure messaging error during Online-Authentication.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_07], and [EIDSERVER_C1_2_08].	SOAP, NONAUTH_CARD
EIDSERVER_ A1_2_07_T	This test checks the behaviour of the eID-Server at the eID-Interface in case Online-Authentication failed.	Perform the test with each of the underlying test cases [EIDSERVER_C1_3_08], and [EIDSERVER_C1_3_09].	SOAP
EIDSERVER_ A1_2_08	This test case checks the behaviour at the eID-Interface in case of sector migration of the terminal and a revoked eID-Card.	-	SOAP, REVOKED_CARD, RI_MIGRATION

Table 74: Test Cases of Module A1_2

6.1.1.3 Module A1_3: Processing of Correct eService Communication (eID-Interface)

This sub-module tests the correct processing of communication with the eService. This includes the initialization of a new session by the eService as well as the retrieval of the results. Table 75 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ A1_3_01	Positive test describing the retrieval of the information about the eID-Server before the Online-Authentication takes place.	-	SOAP
EIDSERVER_ A1_3_02	Positive test describing the retrieval of the information about the eID-Server during the Online-Authentication takes place.	-	SOAP
EIDSERVER_ A1_3_03	Positive test describing the retrieval of the information about the eID-Server after the Online-Authentication took place.	-	SOAP
EIDSERVER_ A1_3_04	Positive test describing the eID-Interface interaction utilizing a PSK predefined by the eService.	-	SOAP
EIDSERVER_ A1_3_05	Positive test that verifies that the eID-Server does not use static values as PSK or Session ID.	-	SOAP
EIDSERVER_ A1_3_06	Positive test describing the reoccurring call of the getResult function during the eID-Interface interaction process where the RequestCounter is incremented by more than 1.	-	SOAP

Table 75: Test Cases of Module A1_3

6.1.1.4 Module A1_4: Processing of Errors in eService Communication (eID-Interface)

This sub-module tests the correct processing of errors during the communication with the eService. Table 76 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ A1_4_01	This test case checks the behaviour of the eID-Server at the eID-Interface in case the eService requests to perform Age Verification, but the request does not contain the mandatory parameters for comparison.	-	SOAP
EIDSERVER_ A1_4_02	This test case checks the behaviour of the eID-Server at the eID-Interface in case the eService requests to perform Place Verification, but the request does not contain the mandatory parameters for comparison.	-	SOAP
EIDSERVER_ A1_4_03	This test case checks the behaviour of the eID-Server in case the "useOperations" element is missing entirely from the useIDRequest.	-	SOAP
EIDSERVER_ A1_4_04	This test case checks the behaviour of the eID-Server at the eID-Interface in case the PSK predefined by the eService has an odd number of characters.	-	SOAP
EIDSERVER_ A1_4_05	This test case checks the behaviour of the eID-Server in case the PSK predefined by the eService contains invalid characters.	-	SOAP
EIDSERVER_ A1_4_06	This test case checks the behaviour of the eID-Server in case the PSK predefined by the eService is too short.	-	SOAP
EIDSERVER_ A1_4_07	This test case checks the behaviour of the eID-Server in case the maximum number of parallel sessions exceeds.	-	SOAP, CONF_MAX_SE
EIDSERVER_ A1_4_08	This test case checks the behaviour of the eID-Server in case a session timeout occurs.	-	SOAP, CONF_TM_OUT
EIDSERVER_ A1_4_09	This test case checks the behaviour of the eID-Server in case the session ID in getResultRequest does not match the session ID submitted by the eID-Server.	-	SOAP

ID	Purpose	Instruction	Profiles
EIDSERVER_A1_4_10	This test case checks the behaviour of the eID-Server in case the session ID is missing entirely in getResultRequest.	-	SOAP
EIDSERVER_A1_4_11	This test case checks the behaviour of the eID-Server in case the RequestCounter is missing entirely in getResultRequest.	-	SOAP
EIDSERVER_A1_4_12	This test case checks the behaviour of the eID-Server in case the RequestCounter is reused.	-	SOAP
EIDSERVER_A1_4_13	This test case checks the behaviour of the eID-Server in case the RequestCounter is not an integer.	-	SOAP
EIDSERVER_A1_4_14	This test case checks the behaviour of the eID-Server in case the session ID is reused.	-	SOAP
EIDSERVER_A1_4_15	This test case checks the ability of the eID-Server to invalidate a session after an unsuccessful eID-Interface interaction due to a schema violation.	-	SOAP
EIDSERVER_A1_4_16	This test case checks the ability of the eID-Server to invalidate a session after an unsuccessful eID-Interface interaction due to wrong request counter.	-	SOAP
EIDSERVER_A1_4_17	This test case checks the ability of the eID-Server to invalidate a session after an unsuccessful eID-Interface interaction due to revoked eID-Card.	-	SOAP

Table 76: Test Cases of Module A1_4

6.1.2 Module A2: eID-Interface – XML Security

In order to assure authenticity and integrity of the SOAP messages on content level, XML signatures are employed. This module tests the correct behaviour of the eID-Server when dealing with these signatures. Table 77 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_A2_01_T	Positive test verifying the eID-Server to support the given signature algorithm and parameters XML signatures during communication at the eID-Interface.	Perform the test with all cryptographic algorithms and parameters for XML signatures (generation and verification) as listed in the ICS, table 9.	SOAP, CRYPTO

ID	Purpose	Instruction	Profiles
EIDSERVER_A2_02_T	This test case checks the behaviour of the eID-Server in case the eService uses an invalid XML signature during communication over the eID-Interface.	Perform the test with all cryptographic algorithms and parameters as listed in the ICS.	SOAP, CRYPTO
EIDSERVER_A2_03	This test case checks the behaviour of the eID-Server in case the eService does not use an XML signature during the communication at the eID-Interface.	-	SOAP, CRYPTO
EIDSERVER_A2_04_T	This test case checks the behaviour of the eID-Server in case the eService uses a signature algorithm during the communication over the eID-Interface which does not meet the requirements stated in [TR-03130-1].	Perform the test for each supported signature method using a hash function from Table 92 that does not meet the requirements of [TR-03130-1].	SOAP, CRYPTO

Table 77: Test Cases of Module A2

6.1.3 Module A3: eID-Interface - TLS

In case the communication of the eID-Server and the eID-Client uses an insecure channel (e.g. the Internet), [TR-03130-1] requires the usage of TLS to protect the transport layer. Several tests ascertain the correct behaviour in this optional scenario. These are only applicable if the eID-Server profiles SOAP_TLS and SOAP were set in the ICS. Otherwise this module is not applicable.

In order to comply with the requirements, the test cases from the [TR-03116-TS] Module B2 MUST be performed successfully. The applicable tests cases based on the supported profiles MUST be determined according to the Mapping Document [TR-03116-TS-MD], Section "*eID-Server*". The test cases belonging to the profiles mentioned in the 'Mandatory Profiles' column in table "eID-Server Profiles for the eID-Interface" of the Mapping Document [TR-03116-TS-MD] MUST be performed, performing the test cases belonging to the profiles in the 'Recommended Profiles' column of that same table is CONDITIONAL. Additionally, Module 0: ICS Checklist of [TR-03116-TS] MUST be performed successfully.

The specific requirements applicable to the eID-Interface are specified in the Mapping Document Section "*eID-Server*" as well.

6.2 Module B: SAML-Interface

The modules in this group evaluate the proper implementation of the communication via SAML as described in [TR-03130-1].

6.2.1 Module B1: SAML-Interface – Functional Operational

This module is divided into three parts which validate the functional operation of the eID-Server using the SAML profile.

6.2.1.1 Module B1_1: Processing of Regular Online-Authentication (SAML-Interface)

This sub-module tests the eID-Server on application level at the SAML interface in case of a regular Online-Authentication This includes the requests and responses for reading data groups and performing special functions. Table 78 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ B1_1_01_T	Positive test describing the SAML interaction during Online-Authentication. The eService requests to read a particular attribute, the field RequiredAttribute is absent.	Perform the test for each attribute of the RequestedAttributesType. The test has to be performed with [EIDCARD_1] and [EIDCARD_14].	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_ B1_1_02_T	Positive test describing the SAML interaction during Online-Authentication. The eService requests to read a particular attribute, the field RequiredAttribute is set to 'true'.	Perform the test for each attribute of the RequestedAttributesType.	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_ B1_1_03_T	Positive test describing the SAML interaction during Online-Authentication. The eService requests to read a particular attribute, the field RequiredAttribute is set to 'false'.	Perform the test for each attribute of the RequestedAttributesType.	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_ B1_1_04_T	Positive test describing the SAML interaction during Online-Authentication in case the user restricts access to a required attribute.	Perform the test for each attribute of the RequestedAttributesType.	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_ B1_1_05_T	Positive test describing the SAML interaction during Online-Authentication in case the user restricts access to an optional attribute.	Perform the test for each attribute of the RequestedAttributesType.	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_ B1_1_06_T	Positive test describing the SAML interaction during Online-Authentication in case of requested data groups are not present or empty.	The test has to be performed with [EIDCARD_2] and [EIDCARD_3].	SAML, EID_ACCESS, DG_VARIATIONS, TLS_PSK
EIDSERVER_ B1_1_07	Positive test describing the SAML interaction during Online-Authentication in case the terminal CV certificate contains two terminal sectors.	-	SAML, EID_ACCESS, RI_MIGRATION, TLS_PSK
EIDSERVER_ B1_1_08	Positive test describing the SAML interaction during Online-Authentication in case Age and Place Verification have negative results.	-	SAML, EID_ACCESS, TLS_PSK

Table 78: Test cases of Module B1_1

6.2.1.2 Module B1_2: Processing of Irregular Online-Authentication (SAML-Interface)

This sub-module tests the behaviour of the eID-Server at the SAML Interface in case of an irregular Online-Authentication. The test cases for this module are listed in Table 79.

ID	Purpose	Instruction	Profiles
EIDSERVER_B1_2_01_T	This test case checks the behaviour of the eID-Server at the SAML-Interface in case the eService requests to read an attributes without having the corresponding authorization in the terminal certificate.	The test has to be performed each attribute of the RequestedAttributesType.	SAML, EID_ACCESS, TLS_PSK
EIDSERVER_B1_2_02_T	This test checks the behaviour of the eID-Server at the SAML-Interface in case Passive Authentication fails.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_01], [EIDSERVER_C1_2_02], [EIDSERVER_C1_2_03], and [EIDSERVER_C1_2_04].	SAML, NONAUTH_CARD, TLS_PSK
EIDSERVER_B1_2_03_T	This test checks the behaviour of the eID-Server at the SAML-Interface in case Chip Authentication fails.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_05], and [EIDSERVER_C1_2_06].	SAML, NONAUTH_CARD, TLS_PSK
EIDSERVER_B1_2_04	This test case checks the behaviour of the eID-Server at the SAML interface in case a revoked eID-Card is used during EAC.	-	SAML, REVOKED_CARD, TLS_PSK
EIDSERVER_B1_2_05	This test case checks the behaviour of the eID-Server at the SAML-Interface in case the eID-Card is expired.	-	SAML, EXPIRED_CARD, TLS_PSK
EIDSERVER_B1_2_06_T	This test checks the behaviour of the eID-Server at the SAML-Interface in case of a secure messaging error during Online-Authentication.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_07], and [EIDSERVER_C1_2_08].	SAML, NONAUTH_CARD, TLS_PSK
EIDSERVER_B1_2_07_T	This test case checks the behaviour of the eID-Server at the SAML-Interface in case Online-Authentication failed.	Perform the test with each of the underlying test cases [EIDSERVER_C1_3_08], and [EIDSERVER_C1_3_09].	SAML, TLS_PSK
EIDSERVER_B1_2_08	This test case checks the behaviour of the eID-Server at the SAML-Interface in case of sector migration of the terminal and a revoked eID-Card.	-	SAML, REVOKED_CARD, RI_MIGRATION, TLS_PSK

Table 79: Test cases of Module B1_2

6.2.1.3 Module B1_3: Processing of SAML Communication (SAML-Interface)

This sub-module tests the correct processing of SAML messages. The test cases for this module are listed in Table 80.

ID	Purpose	Instruction	Profiles
EIDSERVER_ B1_3_01	Positive test describing the SAML interaction during Online-Authentication in case the attribute "AssertionConsumerServiceURL" has an address deviating from the default.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_02	This test case checks the behaviour of the eID-Server at the SAML interface in case the element "Issuer" in AuthnRequest is [ESERVICEURL_NOK].	-	SAML, TLS_PSK
EIDSERVER_ B1_3_03	This test case checks the behaviour of the eID-Server at the SAML interface in case the mandatory element "Issuer" in AuthnRequest is entirely missing.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_04	This test case checks the behaviour of the eID-Server at the SAML interface in case the mandatory element "Extensions" in AuthnRequest is entirely missing.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_05	This test case checks the behaviour of the eID-Server at the SAML interface in case the element "Extensions" in AuthnRequest is empty.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_06	This test case checks the behaviour of the eID-Server at the SAML interface in case the element "EncryptedAuthnRequestExtension" in AuthnRequest is empty.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_07	This test case checks the behaviour of the eID-Server at the SAML interface the Age Verification requested by the eService do not contain the mandatory parameters for comparison.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_08	This test case checks the behaviour of the eID-Server at the SAML interface the Place Verification requested by the eService do not contain the mandatory parameters for comparison.	-	SAML, TLS_PSK
EIDSERVER_ B1_3_09	This test case checks the behaviour of the eID-Server at the SAML interface in case the maximum number of parallel sessions is exceeded	-	SAML, TLS_PSK

Table 80: Test Cases of Module B1_3

6.2.1.4 Module B1_4: Processing of Regular Online-Authentication (Attached SAML)

This sub-module tests the eID-Server on application level at the SAML interface in case of a regular Online-Authentication, with a SAML processor that is attached to the eID-Server. This includes the requests and responses for reading data groups and performing special functions. Table 81 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ B1_4_01_T	This test case is almost equal to EIDSERVER_B1_1_01_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_01_T.	SAML, EID_ACCESS, SAML_ATTACHED
EIDSERVER_ B1_4_02_T	This test case is almost equal to EIDSERVER_B1_1_02_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_02_T.	SAML, EID_ACCESS, SAML_ATTACHED
EIDSERVER_ B1_4_03_T	This test case is almost equal to EIDSERVER_B1_1_03_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_03_T.	SAML, EID_ACCESS, SAML_ATTACHED
EIDSERVER_ B1_4_04_T	This test case is almost equal to EIDSERVER_B1_1_04_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_04_T.	SAML, EID_ACCESS, SAML_ATTACHED
EIDSERVER_ B1_4_05_T	This test case is almost equal to EIDSERVER_B1_1_05_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_05_T.	SAML, EID_ACCESS, SAML_ATTACHED
EIDSERVER_ B1_4_06_T	This test case is almost equal to EIDSERVER_B1_1_06_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_06_T.	SAML, EID_ACCESS, DG_VARIATIONS, SAML_ATTACHED
EIDSERVER_ B1_4_07	This test case is almost equal to EIDSERVER_B1_1_07, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_07.	SAML, EID_ACCESS, RI_MIGRATION, SAML_ATTACHED
EIDSERVER_ B1_4_08	This test case is almost equal to EIDSERVER_B1_1_08, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_1_08.	SAML, EID_ACCESS, SAML_ATTACHED

Table 81: Test cases of Module B1_4

6.2.1.5 Module B1_5: Processing of Irregular Online-Authentication (Attached SAML)

This sub-module tests the behaviour of the eID-Server at the SAML interface in case of an irregular Online-Authentication while operating in attached SAML processor mode. The test cases for this module are listed in Table 82.

ID	Purpose	Instruction	Profiles
EIDSERVER_	This test case is almost equal to	Refer to the instructions for	SAML, EID_ACCESS,

B1_5_01_T	EIDSERVER_B1_2_01_T, except it does not expect a PSK for the eCard-API communication.	EIDSERVER_B1_2_01_T.	SAML_ATTACHED
EIDSERVER_B1_5_02_T	This test case is almost equal to EIDSERVER_B1_2_02_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_02_T.	SAML, NONAUTH_CARD, SAML_ATTACHED
EIDSERVER_B1_6_03_T	This test case is almost equal to EIDSERVER_B1_2_03_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_03_T.	SAML, NONAUTH_CARD, SAML_ATTACHED
EIDSERVER_B1_5_04	This test case is almost equal to EIDSERVER_B1_2_04, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_04.	SAML, REVOKED_CARD, SAML_ATTACHED
EIDSERVER_B1_5_05	This test case is almost equal to EIDSERVER_B1_2_05, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_05.	SAML, EXPIRED_CARD, SAML_ATTACHED
EIDSERVER_B1_5_06_T	This test case is almost equal to EIDSERVER_B1_2_06_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_06_T.	SAML, NONAUTH_CARD, SAML_ATTACHED
EIDSERVER_B1_5_07_T	This test case is almost equal to EIDSERVER_B1_2_07_T, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_07_T.	SAML, SAML_ATTACHED
EIDSERVER_B1_5_08	This test case is almost equal to EIDSERVER_B1_2_08, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_2_08.	SAML, REVOKED_CARD, RI_MIGRATION, SAML_ATTACHED

Table 82: Test cases of Module B1_5

6.2.1.6 Module B1_6: Processing of SAML Communication (Attached SAML)

This sub-module tests the correct processing of SAML messages with a SAML processor that is attached to the eID-Server. The test cases for this module are listed in Table 83.

ID	Purpose	Instruction	Profiles
EIDSERVER_B1_6_01	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_01.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_02	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_02.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_03	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_03.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_04	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_04.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_05	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_05.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_06	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_06.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_07	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_07.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_08	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_08.	SAML, SAML_ATTACHED
EIDSERVER_B1_6_09	This test case is almost equal to EIDSERVER_B1_3_01, except it does not expect a PSK for the eCard-API communication.	Refer to the instructions for EIDSERVER_B1_3_09.	SAML, CONF_MAX_SE, SAML_ATTACHED

Table 83: Test Cases of Module B1_6

6.2.2 Module B2: SAML-Interface - XML Security

If SAML is used for communication between eID-Server and eService, [TR-03130-1] mandates that person related data is cryptographically protected on content layer using XML signatures and XML encryption.

This test module is divided into two parts, both dealing with the cryptographic security elements of the XML data used during the SAML communication.

6.2.2.1 Module B2_1: XML Signature – SAML

In case the SAML profile is used, each message has to contain a valid XML signature to ensure the authenticity and the integrity of the transmitted data. The tests in this module evaluate the correct behaviour of the eID-Server when evaluating this type of information. The test cases for this module are listed in Table 84.

ID	Purpose	Instruction	Profiles
EIDSERVER_ B2_1_01_T	Positive test verifying the eID-Server to support all algorithms and parameters listed in the ICS for verification of XML signatures during communication using the SAML-Profile.	Perform the test for each signature algorithm and parameters supported for verification of XML signatures for the SAML profile as listed in the ICS.	SAML, CRYPTO
B2_1_02_T	Positive test verifying the eID-Server to support all algorithms and parameters listed in the ICS for generation of XML signatures during communication using the SAML-Profile.	Perform the test for each signature algorithm and parameters supported for generation of XML signatures for the SAML profile as listed in the ICS.	SAML, CRYPTO
EIDSERVER_ B2_1_03_T	This test case checks the behaviour of the eID-Server in case the SAML message contains an invalid XML signature.	Perform the test for each supported signature method.	SAML, CRYPTO
EIDSERVER_ B2_1_04	This test case checks the behaviour of the eID-Server in case the SAML message does not contain any XML signature.	-	SAML, CRYPTO
EIDSERVER_ B2_1_05_T	This test case checks the behaviour of the eID-Server in case the SAML message is signed using a signature algorithm which does not meet the requirements of [TR-03130-1].	Perform the test for each supported signature method using a hash function that does not meet the requirements of [TR-03130-1].	SAML, CRYPTO

Table 84: Test Cases of Module B2_1

6.2.2.2 Module B2_2: XML Encryption – SAML

This module evaluates the correct treatment of the encrypted XML data and the keys by the eID-Server. Table 85 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ B2_2_01_T	Positive test verifying the eID-Server to support all algorithms and parameters listed in the ICS for XML decryption during communication using the SAML-Profile.	Perform the test for each algorithm and parameters as listed in the ICS.	SAML, CRYPTO

ID	Purpose	Instruction	Profiles
EIDSERVER_B2_2_02_T	Positive test verifying the eID-Server to support all algorithms and parameters listed in the ICS for XML encryption during communication using the SAML-Profile.	Perform the test for each algorithm and parameters as listed in the ICS.	SAML, CRYPTO

Table 85: Test Cases of Module B2_2

6.2.3 Module B3: TLS – SAML

To secure the transport layer of the SAML communication between the SAML-Processor of the eID-Server and the eID-Client, TLS is used. This test module evaluates the correct implementation and usage of TLS for this communication. It is only applicable if the eID-Server profile SAML was set in the ICS. Otherwise this module is not applicable.

In order to comply with the requirements, the test cases from the [TR-03116-TS] Module B1 MUST be performed successfully. The applicable tests cases based on the supported profiles MUST be determined according to the Mapping Document [TR-03116-TS-MD], Section "*eID-Server*". The test cases belonging to the profiles mentioned in the 'Mandatory Profiles' column in table "eID-Server Profiles for the SAML" of the Mapping Document [TR-03116-TS-MD] MUST be performed, performing the test cases belonging to the profiles in the 'Recommended Profiles' column of that same table is CONDITIONAL. Additionally, Module 0: ICS Checklist of [TR-03116-TS] MUST be performed successfully.

The specific requirements applicable to the SAML-Interface are specified in the Mapping Document Section "*eID-Server*" as well.

6.3 Module C: eCard-API Interface

This test module evaluates the correct implementation of EAC, and the eCard-API communication with the eID-Client as specified [TR-03112]. This is step 4 in Section 4.1.2 of [TR-03130-1]. In particular, the proper implementation of the Attached eID-Server Model is evaluated here.

Since [TR-03130-1] and [TR-03112] allow variations in the behaviour of the eID-Server (e.g. in which order data groups are read or checks are made by the eID-Server), the evaluations during the tests need to be flexible.

6.3.1 Module C1: eCard-API Interface – Functional Operation

This module is divided into three parts which deal with the communication of the eService and the eID-Server at the eID-Interface.

6.3.1.1 Module C1_1: Processing of Regular Online-Authentication (eCard-API)

This sub-module checks the correct behaviour of the eID-Server at the eCard-API interface during Online-Authentication. This includes performing different algorithms, reading of data groups and performing special functions. Table 86 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_ C1_1_01_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client. The eService request requires to read a particular attribute.	The test has to be performed for each attribute.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_02_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client. The eService request allows to read a particular attribute.	The test has to be performed for each attribute.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_03_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case the user restricts access to an attribute. The eService request requires to read a particular attribute.	The test has to be performed for each attribute.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_04_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case the user restricts access to an attribute. The eService request allows to read a particular attribute.	The test has to be performed for each attribute.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_05_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case requested data groups are not present or empty.	The test has to be performed with [EIDCARD_2] and [EIDCARD_3].	PAOS, EAC, EID_ACCESS, DG_VARIATIONS
EIDSERVER_ C1_1_06	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case the terminal CV certificate contains two terminal sectors.	-	PAOS, EAC, EID_ACCESS, RI_MIGRATION
EIDSERVER_ C1_1_07_T	Positive test verifying that the eID-Server supports the given algorithm for the Chip Authentication.	Perform the test for each supported Chip Authentication algorithm and each supported set of standardized domain parameters as listed in the ICS.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_08	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case Age and Place Verification have negative results.	-	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_09_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client checking that the eID-Server is capable to use different keyIDs for Revocation check.	The test has to be performed several times using 2 different key IDs.	PAOS, EAC, EID_ACCESS

ID	Purpose	Instruction	Profiles
EIDSERVER_ C1_1_10_T	Positive test describing the PAOS communication between the eID-Server and the eID-Client checking that the eID-Server is capable to use different keyIDs for the RestrictedID (Pseudonym).	The test has to be performed several times using 2 different key IDs.	PAOS, EAC, EID_ACCESS
EIDSERVER_ C1_1_11	Positive test verifying that the eID-server's response EAC1InputType contains all link certificates known to the eID-Server.	-	PAOS, EAC, EID_ACCESS, ALL_LINK
EIDSERVER_ C1_1_12	Positive test describing the retrieval to the TC Token in case of in case of an Attached Server without SAML interaction.	-	PAOS, EAC, ESER_ATTACHED
EIDSERVER_ C1_1_13	This test verifies that an eID-Server operating in attached mode does not accept new TLS connections for the same application session.	-	PAOS, EAC, ESER_ATTACHED
EIDSERVER_ C1_1_14	This test verifies that an eID-Server operating in SAML attached mode does not accept new TLS connections for the same application session.	-	PAOS, EAC, SAML_ATTACHED
EIDSERVER_ C1_1_15	Positive test describing the PAOS communication between the eID-Server and the eID-Client. The eIDAS request requires to read the minimum dataset.		PAOS, EAC, EID_ACCESS, EIDAS_MW
EIDSERVER_ C1_1_16	Positive test verifying the behaviour of the eID-Server during PAOS communication between the eID-Server and the eID-Client in case an eID-Card additionally supports a higher version of Chip Authentication than the eCard-API (i.e. CAv3).	The test has to be performed with [EIDCARD_15].	PAOS, EAC

Table 86: Test Cases of Module C1_1

6.3.1.2 Module C1_2: Processing of EAC-Related Errors

This module focuses on tests that deal with EAC2-related errors during Online-Authentication. This includes failing of CA, PA and SM-errors during card communication. Table 87 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_C1_2_01	This test checks the behaviour of the eID-Server during PAOS communication in case "EFCardSecurity" contains a manipulated signature.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_02	This test checks the behaviour of the eID-Server during PAOS communication in case the card file EF.CardSecurity is signed with a revoked DS certificate.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_03	This test checks the behaviour of the eID-Server during PAOS communication in case the card file EF.CardSecurity is signed with a DS certificate not verifiable by the eID-Server.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_04_T	This test checks the behaviour of the eID-Server during PAOS communication in case the card file "EFCardSecurity" contains content that does not correspond with "EFCardAccess".	The test has to be performed several times applying the following deviations: 1 The OIDs for PACE do not match. 2 The PACE domain parameters do not match 3 The OIDs for CA do not match	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_05	This test checks the behaviour of the eID-Server during the PAOS communication in case the card file "EFCardSecurity" contains a public key for Chip Authentication that is not on the elliptic curve.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_06	This test checks the behaviour of the eID-Server during the PAOS communication in case the DIDAuthenticateResponse_EAC2OutputType contains an incorrect "AuthenticationToken" element.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_07	This test checks the behaviour of the eID-Server during the PAOS communication in case the TransmitResponse message contains "OutputAPDU" elements without MAC value.	-	PAOS, EAC, NONAUTH_CARD
EIDSERVER_C1_2_08	This test checks the behaviour of the eID-Server during the PAOS communication in case the TransmitResponse message contains "OutputAPDU" elements with invalid MAC values.	-	PAOS, EAC, NONAUTH_CARD

ID	Purpose	Instruction	Profiles
EIDSERVER_C1_2_09	Positive test checking the behaviour of the eID-Server during the PAOS communication in case the document is on the revocation list.	-	PAOS, EAC, REVOKED_CARD
EIDSERVER_C1_2_10	Positive test checking the behaviour of the eID-Server during the PAOS communication in case the eID-Card used has expired.	-	PAOS, EAC, EXPIRED_CARD

Table 87: Test Cases of Module C1_2

6.3.1.3 Module C1_3: Processing of PAOS Communication

The tests in this module ascertain the correctness of the messages of the eCard-API framework.

ID	Purpose	Instruction	Profiles
EIDSERVER_C1_3_01	Positive test describing the PAOS communication between the eID-Server and the eID-Client in case the message StartPAOS contains additional optional elements.	-	PAOS, EAC
EIDSERVER_C1_3_02	This test checks the behaviour of the eID-Server in case the element "IDPICC" is missing in the DIDAuthenticateResponse_EAC1OutputType.	-	PAOS, EAC
EIDSERVER_C1_3_03	This test checks the behaviour of the eID-Server in case the element "Challenge" is missing in the DIDAuthenticateResponse_EAC1OutputType.	-	PAOS, EAC
EIDSERVER_C1_3_04	This test checks the behaviour of the eID-Server in case the element "EFCardSecurity" is missing in the DIDAuthenticateResponse_EAC2OutputType.	-	PAOS, EAC
EIDSERVER_C1_3_05	This test checks the behaviour of the eID-Server in case the DIDAuthenticateResponse_EAC2OutputType contains two valid "EFCardSecurity" elements.	-	PAOS, EAC
EIDSERVER_C1_3_06	This test checks the behaviour of the eID-Server in case the element "AuthenticationToken" is missing in the DIDAuthenticateResponse_EAC2OutputType.	-	PAOS, EAC

ID	Purpose	Instruction	Profiles
EIDSERVER_C1_3_07	This test checks the behaviour of the eID-Server in case the DIDAuthenticateResponse_EAC2OutputType contains two valid "AuthenticationToken" elements.	-	PAOS, EAC
EIDSERVER_C1_3_08	This test checks the behaviour of the eID-Server in case the element "Nonce" is missing in the DIDAuthenticateResponse_EAC2OutputType.	-	PAOS, EAC
EIDSERVER_C1_3_09	This test checks the behaviour of the eID-Server in case the DIDAuthenticateResponse_EAC2OutputType contains two valid "Nonce" elements.	-	PAOS, EAC
EIDSERVER_C1_3_10	This test checks the behaviour of the eID-Server during the PAOS communication in case the eID-Clients repeats the challenge in DIDAuthenticateResponse_EAC2OutputType even after receiving DIDAuthenticate_EAC2InputType containing signature.	-	PAOS, EAC
EIDSERVER_C1_3_11	This test checks the behaviour of the eID-Server in case the DIDAuthenticateResponse_EAC1OutputType signalizes an error in the element "Result".	-	PAOS, EAC
EIDSERVER_C1_3_12	This test checks the behaviour of the eID-Server in case the DIDAuthenticateResponse_EAC2OutputType signalizes an error in the element "Result".	-	PAOS, EAC

Table 88: Test Cases of Module C1_3

6.3.2 Module C2: TLS – eCard-API

To secure the communication between eID-Server and eID-Client, [TR-03130-1] requires to utilize TLS. This test module evaluates the correct implementation and usage of TLS at the eCard-API interface. The tests are the functional counterparts of module E in [TR-03124-2]. They are only applicable if the eID-Server profile TLS_PSK was set in the ICS. Otherwise this module is not applicable.

In order to comply with the requirements, the test cases from the [TR-03116-TS] Module B1 MUST be performed successfully. The applicable tests cases based on the supported profiles MUST be determined according to the Mapping Document [TR-03116-TS-MD], Section “*eID-Server*”. The test cases belonging to the profiles mentioned in the ‘Mandatory Profiles’ column in table “eID-Server Profiles for the eCard-API interface” of the Mapping Document [TR-03116-TS-MD] MUST be performed, performing the test cases belonging to the profiles in the ‘Recommended Profiles’ column of that same table is CONDITIONAL. Additionally, Module 0: ICS Checklist of [TR-03116-TS] MUST be performed successfully.

The specific requirements applicable to the eCard-API Interface are specified in the Mapping Document Section "*eID-Server*" as well.

6.4 Module D: eIDAS Middleware Mode

The modules in this group evaluate the proper implementation of the communication with the eIDAS infrastructure via SAML as described in [TR-03130-3].

6.4.1 Module D1: eIDAS Middleware Mode – Functional eIDAS Operation

This module is divided into two parts which validate the functional operation of the eID-Server using the EIDAS_MW profile.

6.4.1.1 Module D1_1: Processing of Regular Online-Authentication (eIDAS Middleware Mode)

This sub-module tests the eID-Server on application level in eIDAS middleware mode in case of a regular Online-Authentication. This includes the requests and responses for reading data groups and performing special functions. Table 89 lists all test cases in this module.

ID	Purpose	Instruction	Profiles
EIDSERVER_D1_1_01_T	Positive test describing the eIDAS interaction during Online-Authentication. The eIDAS connector requests the eIDAS minimum dataset.	The test has to be performed with [EIDCARD_1] and [EIDCARD_14].	EIDAS_MW, EID_ACCESS
EIDSERVER_D1_1_02_T	Positive test describing the eIDAS interaction during Online-Authentication. The eIDAS connector requests a reduced minimum dataset by skipping one attribute.	Perform the test for each attribute of the minimum dataset.	EIDAS_MW, EID_ACCESS
EIDSERVER_D1_1_03_T	Positive test describing the eIDAS interaction during Online-Authentication in case of requested data groups are not present or empty.	The test has to be performed with [EIDCARD_2] and [EIDCARD_3].	EIDAS_MW, EID_ACCESS, DG_VARIATIONS
EIDSERVER_D1_1_04	Negative test describing the eIDAS interaction during Online-Authentication. The eIDAS connector requests to read the eIDAS minimum dataset and an unknown attribute.	-	EIDAS_MW, EID_ACCESS
EIDSERVER_D1_1_05	Positive test describing the eIDAS interaction during Online-Authentication. The eIDAS connector requests the eIDAS minimum dataset. The user denies access to optional attributes.	-	EIDAS_MW, EID_ACCESS

Table 89: Test cases of Module D1_1

6.4.1.2 Module D1_2: Processing of eIDAS Communication

This sub-module tests the behaviour of the eID-Server in eIDAS middleware mode in case of an irregular Online-Authentication. The test cases for this module are listed in Table 90.

ID	Purpose	Instruction	Profiles
EIDSERVER_ D1_2_01	This test checks the behaviour of the eID-Server at the eIDAS-Interface in case the eService requests to read a data group without having the corresponding authorization in the terminal certificate.	-	EIDAS_MW, EID_ACCESS
EIDSERVER_ D1_2_02_T	This test checks the behaviour of the eID-Server at the eIDAS-Interface in case Passive Authentication fails.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_01], [EIDSERVER_C1_2_02], [EIDSERVER_C1_2_03], and [EIDSERVER_C1_2_04].	EIDAS_MW, NONAUTH_CARD
EIDSERVER_ D1_2_03_T	This test checks the behaviour of the eID-Server at the eIDAS-Interface in case Chip Authentication fails.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_05], and [EIDSERVER_C1_2_06].	EIDAS_MW, NONAUTH_CARD
EIDSERVER_ D1_2_04	This test case checks the behaviour of the eID-Server at the eIDAS-Interface in case a revoked eID-Card is used during EAC.	-	EIDAS_MW, REVOKED_CARD
EIDSERVER_ D1_2_05	This test case checks the behaviour of the eID-Server at the eIDAS-Interface in case the eID-Card is expired.	-	EIDAS_MW, EXPIRED_CARD
EIDSERVER_ D1_2_06_T	This test checks the behaviour of the eID-Server at the eIDAS-Interface in case of a secure messaging error during Online-Authentication.	Perform the test with each of the underlying test cases [EIDSERVER_C1_2_07], and [EIDSERVER_C1_2_08].	EIDAS_MW, NONAUTH_CARD
EIDSERVER_ D1_2_07_T	This test case checks the behaviour of the eID-Server at the eIDAS-Interface in case Online-Authentication failed.	Perform the test with each of the underlying test cases [EIDSERVER_C1_3_08], and [EIDSERVER_C1_3_09].	EIDAS_MW
EIDSERVER_ D1_2_08	This test case checks the behavior of the eID-Server at the eIDAS-Interface in case the user denies access to required attributes.		EIDAS_MW, EID_ACCESS

Table 90: Test cases of Module D1_2

6.4.2 Module D2: eIDAS Middleware Mode - XML Security

In case the EIDAS_MW profile is used, the eIDAS request message has to contain a valid XML signature to ensure the authenticity and the integrity of the transmitted data. The tests in this module evaluate the correct behaviour of the eID-Server when a message is not correctly signed. The test cases for this module are listed in Table 91.

ID	Purpose	Instruction	Profiles
EIDSERVER_D2_01_T	This test case checks the behaviour of the eID-Server in case the eIDAS request message contains an invalid XML signature.	Perform the test for each supported signature method.	EIDAS_MW, CRYPTO
EIDSERVER_D2_02	This test case checks the behaviour of the eID-Server in case the eIDAS request message does not contain any XML signature.	-	EIDAS_MW, CRYPTO

Table 91: Test Cases of Module D2

6.4.3 Module D3: eIDAS Middleware Mode - TLS

To secure the transport layer of the communication between the eIDAS-Interface of the eID-Server and the eID-Client, TLS is used. This test module evaluates the correct implementation and usage of TLS for this communication. It is only applicable if the eID-Server profile EIDAS_MW was set in the ICS. Otherwise this module is not applicable.

In order to comply with the requirements, the test cases from the [TR-03116-TS] Module B1 MUST be performed successfully. The applicable tests cases based on the supported profiles MUST be determined according to the Mapping Document [TR-03116-TS-MD], Section "*eID-Server*". The test cases belonging to the profiles mentioned in the 'Mandatory Profiles' column in table "eID-Server Profiles for the eIDAS-MW" of the Mapping Document [TR-03116-TS-MD] MUST be performed, performing the test cases belonging to the profiles in the 'Recommended Profiles' column of that same table is CONDITIONAL. Additionally, Module 0: ICS Checklist of [TR-03116-TS] MUST be performed successfully.

The specific requirements applicable to the eIDAS-MW-Interface are specified in the Mapping Document Section "*eID-Server*" as well.

6.5 Parameters

This section defines parameters which are referred by several test cases.

The following table defines signature algorithms for XML Signature according to [RFC4051].

Supported signature algorithms
http://www.w3.org/2000/09/xmldsig#rsa-sha1
http://www.w3.org/2001/04/xmldsig-more#rsa-md5
http://www.w3.org/2001/04/xmldsig-more#rsa-sha256
http://www.w3.org/2001/04/xmldsig-more#rsa-sha384
http://www.w3.org/2001/04/xmldsig-more#rsa-sha512
http://www.w3.org/2001/04/xmldsig-more/rsa-ripemd160
http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha1
http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha224
http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha256
http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha384
http://www.w3.org/2001/04/xmldsig-more#ecdsa-sha512

Table 92: Signature algorithms for XML Signature

References

[TR-03130-1] BSI, Technical Guideline TR-03130 eID-Server - Part 1: Functional Specifications

[TR-03124-1] BSI, Technical Guideline TR-03124 eID-Client - Part 1: Specifications

[TR-03127] BSI, Technische Richtlinie TR-03127 eID-Karten mit eID- und eSign-Anwendung basierend auf Extended Access Control

[CP-eID] BSI, Certicicate Policy für die eID-Anwendung des ePA

[TR-03112] BSI, Technical Guideline TR-03112 eCard-API-Framework

[TR-03110] BSI, Technical Guideline TR-03110 - Advanced Security Mechanisms for Machine Readable Travel Documents and eIDAS Token

[TR-03130-3] BSI, Technical Guideline TR-03130-3 eID-Server - Part 3: eIDAS-Middleware-Service for eIDAS-Token

[TR-03129] BSI, TR-03129 PKIs for Machine Readable Travel Documents-Protocols for the Management of Certificates and CRLs

[TR-03116-TS] BSI, Technical Guideline TR-03116-TS: TLS Test-Specification

[TR-03116-TS-MD] BSI, Annex to BSI TR-03116-TS: Mapping of application-specific requirements

[TR-03116-2] BSI, Technische Richtlinie TR-03116 Kryptographische Vorgaben für Projekte der Bundesregierung - Teil 2: Hoheitliche Ausweisdokumente

[RFC6454] A. Barth, RFC 6454: The Web Origin Concept

[RFC2616] R. Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, T. Berners-Lee, Hypertext Transfer Protocol -- HTTP/1.1

[RFC2818] E. Rescorla, HTTP Over TLS

[TR-03124-2] BSI, Technical Guideline TR-03124 eID-Client - Part 2: Conformance Test Specifications

[RFC4051] D. Eastlake 3rd, Additional XML Security Uniform Resource Identifiers (URIs)