ETSI TS 119 542 V1.1.1 (2025-10)

TECHNICAL SPECIFICATION

Electronic Signatures and Trust Infrastructures (ESI); Use of EU Digital Identity Wallets and electronic signatures for identification with Smart Contracts

Reference
DTS/ESI-0019542

Keywords
digital identity, digital signature, smart contract,
trust services

ETSI

650 Route des Lucioles
F-06921 Sophia Antipolis Cedex - FRANCE

Tel.: +33 4 92 94 42 00 Fax: +33 4 93 65 47 16

Siret N° 348 623 562 00017 - APE 7112B
Association à but non lucratif enregistrée à la
Sous-Préfecture de Grasse (06) N° w061004871

Contents

Intellectual Property Rights

Essential patents

IPRs essential or potentially essential to normative deliverables may have been declared to ETSI. The declarations pertaining to these essential IPRs, if any, are publicly available for **ETSI members and non-members**, and can be found in ETSI SR 000 314: *"Intellectual Property Rights (IPRs); Essential, or potentially Essential, IPRs notified to ETSI in respect of ETSI standards"*, which is available from the ETSI Secretariat. Latest updates are available on the ETSI IPR online database.

Pursuant to the ETSI Directives including the ETSI IPR Policy, no investigation regarding the essentiality of IPRs, including IPR searches, has been carried out by ETSI. No guarantee can be given as to the existence of other IPRs not referenced in ETSI SR 000 314 (or the updates on the ETSI Web server) which are, or may be, or may become, essential to the present document.

Trademarks

The present document may include trademarks and/or tradenames which are asserted and/or registered by their owners. ETSI claims no ownership of these except for any which are indicated as being the property of ETSI, and conveys no right to use or reproduce any trademark and/or tradename. Mention of those trademarks in the present document does not constitute an endorsement by ETSI of products, services or organizations associated with those trademarks.

DECT™, **PLUGTESTS™**, **UMTS™** and the ETSI logo are trademarks of ETSI registered for the benefit of its Members. **3GPP™**, **LTE™** and **5G™** logo are trademarks of ETSI registered for the benefit of its Members and of the 3GPP Organizational Partners. **oneM2M™** logo is a trademark of ETSI registered for the benefit of its Members and of the oneM2M Partners. **GSM®** and the GSM logo are trademarks registered and owned by the GSM Association.

Foreword

This Technical Specification (TS) has been produced by ETSI Technical Committee Electronic Signatures and Trust Infrastructures (ESI).

Modal verbs terminology

In the present document "**shall**", "**shall not**", "**should**", "**should not**", "**may**", "**need not**", "**will**", "**will not**", "**can**" and "**cannot**" are to be interpreted as described in clause 3.2 of the ETSI Drafting Rules (Verbal forms for the expression of provisions).

"**must**" and "**must not**" are **NOT** allowed in ETSI deliverables except when used in direct citation.

1 Scope

The present document specifies the use of EU Digital Identity Wallets, and advanced or qualified electronic signatures and seals conforming to the requirements of Regulation (EU) 2024/1183 [i.3], amending Regulation (EU) No 910/2014, (referred as eIDAS hereinafter in the present document). The advanced or qualified electronic signatures and seals in the present document are implemented using digital signatures.

The present document supports identification of natural or legal persons playing relevant roles in different stages of the life cycle of Smart Contracts, taking into account the needs for identification highlighted in ETSI TR 119 540 [i.1].

2 References

2.1 Normative references

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

Referenced documents which are not found to be publicly available in the expected location might be found in the ETSI docbox.

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

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

[1] ETSI EN 319 162-1: "Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 1: Building blocks and ASiC baseline containers".

[2] ETSI EN 319 132-1: "Electronic Signatures and Trust Infrastructures (ESI); XAdES digital signatures; Part 1: Building blocks and XAdES baseline signatures".

[3] ETSI EN 319 122-1: "Electronic Signatures and Infrastructures (ESI); CAdES digital signatures; Part 1: Building blocks and CAdES baseline signatures".

[4] ETSI EN 319 102-1: "Electronic Signatures and Trust Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation".

[5] ETSI TS 119 102-2: "Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 2: Signature Validation Report".

[6] ETSI TS 119 182-1: "Electronic Signatures and Infrastructures (ESI); JAdES digital signatures; Part 1: Building blocks and JAdES baseline signatures ".

[7] W3C® Recommendation 11 April 2013: "XML Signature Syntax and Processing. Version 1.1".

2.2 Informative references

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

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

The following referenced documents may be useful in implementing an ETSI deliverable or add to the reader's understanding, but are not required for conformance to the present document.

[i.1] ETSI TR 119 540: "Electronic Signatures and Trust Infrastructures (ESI); Standardisation requirements for Smart Contracts based on electronic ledgers".

[i.2] ISO/IEC 18013-5:2021: "Personal identification — ISO- compliant driving licence — Part 5: Mobile driving licence (mDL) application".

[i.3] Regulation (EU) 2024/1183 of the European Parliament and of the Council of 11 April 2024 amending Regulation (EU) No 910/2014 as regards establishing the European Digital Identity Framework.

[i.4] ETSI TS 119 472-2: "Electronic Signatures and Trust Infrastructures (ESI); Profiles for Electronic Attestation of Attributes; Part 2: Profiles for EAA/PID Presentations to Relying Party".

[i.5] Commission Implementing Regulation (EU) 2024/2977 of 28 November 2024 laying down rules for the application of Regulation (EU) No 910/2014 of the European Parliament and of the Council as regards person identification data and electronic attestations of attributes issued to European Digital Identity Wallets.

3 Definition of terms, symbols and abbreviations

3.1 Terms

For the purposes of the present document, the terms given in ETSI TR 119 540 [i.1] apply.

3.2 Symbols

Void.

3.3 Abbreviations

For the purposes of the present document, the following abbreviations apply:

ASiC	Associated Signature Container
CMS	Cryptographic Message Syntax
EAA	Electronic Attestation of Attributes
eIDAS	electronic IDentification, Authentication and trust Services

NOTE: As in Regulation 910/2014 amended by Regulation 2024/1183 [i.3].

EUDI	EUropean Digital Identity
ISO	International Standards Organization
JSON	JavaScript Object Notation
JWS	JSON Web Signature
MIME	Multipurpose Internet Mail Extensions
OID	Object IDentifier
PID	Personal Identification Data
SC	Smart Contract
SCP	Smart Contract Provider
SPO	Service Provision Option
VM	Virtual Machine
XML	eXtensible Markup Language

4 Roles, objects and signatories in Smart Contracts lifecycle

Table 1 of ETSI TR 119 540 [i.1]:

1) Identifies the Smart Contract life cycle phases.

2) Identifies a number of relevant roles for Smart Contract lifecycle. Entities may play one or several of them.

3) Identifies objects that the aforementioned roles generate during the mentioned lifecycle.

4) Identifies interactions between the different roles during the mentioned lifecycle.

5) For each of the mentioned interactions, it signals the identification needs (i.e. whether certain role needs to identify the other role before executing certain task), and potential identification mechanism (for instance, signing a data object).

The present document takes the mentioned table as starting point and specifies requirements for identification of the different roles during the Smart Contract lifecycle.

Below follows a list of objects that are managed during the Smart Contract lifecycle phases. The list specifies requirements on whether they have to be signed or not, and, in case they have to be signed, it identifies the signing entities:

1) **SC Production phase**. During this phase, the following objects shall be signed:

   - SC Language Specification and SC Language Specification Policy (see clause 8.2.2 of the present document). These data objects shall be signed by the SC Language Publisher.

   - Smart Contract compiler and Smart Contract Compiler Specification Policy (see clause 8.2.3 of the present document). These data objects shall be signed by the Smart Contract Compiler Publisher.

   - SC Virtual Machine and SC Virtual Machine Policy (see clause 8.2.4 of the present document). These data objects shall be signed by the SC Virtual Machine Publisher.

   - SC Package (see clause 8.3 of the present document). This data object shall be signed by the SC Publisher. The SC Package may include, among others the SC Source Code, SC Byte Code, SC Development Policy, SC Publication Policy, SC Legal Text, and SC Documentation, in support of the Smart Contract, signed by the SC Publisher.

   - Evidence of agreement by the Smart Contract parties (see clause 8.4 of the present document). Each evidence shall be signed by one Smart Contract party.

2) **SC Deployment phase** (see clause 9 of the present document). During this phase:

   - First, the SC Deployer shall validate the ASiC-E container generated by the SC Publisher.

   - The SC Deployer shall generate and sign a validation report of the mentioned ASiC-E container.

   - If the ASiC-E container validation succeeds, the SC Deployer shall deploy the Smart Contract and shall sign an evidence of the Smart Contract deployment.

3) **SC Execution phase:**

   - Before any execution of the Smart Contract, the SC Provider shall validate the signature on the evidence of the Smart Contract deployment generated by the SC Deployer.

   - The SC Provider and the SC User shall authenticate each other. The SC Provider may authenticate the Smart Contract requesting that the SC User signs some data object (see clause 10.3 of the present document).

   - After the execution of the Smart Contract, the SC Provider shall sign an Smart Contract execution report (see clause 10.4 of the present document).

5 AdES signatures profiles

5.1 Introduction

Clauses 6, 9, and 10 include digitally signing one or more data objects as an identification means.

In cases where the number of data objects to be signed is higher than 1 and they are detached from the digital signatures, then ASiC-E containers including CAdES or XAdES signatures shall be used, signing the mentioned data objects.

Otherwise, standalone CAdES, JAdES, or XAdES signatures, i.e. not included within ASiC-E containers, shall be used.

NOTE 1: At the moment when the present document was developed, there is not any specification standardizing the inclusion of JAdES signatures within ASiC containers. If at any time such specification is standardized, the present document will be reviewed and suitably updated.

The present clause defines profiles for (C/J/X)AdES signatures that may either individually sign one data object or be included in an ASiC-E container for signing several detached data objects.

These profiles shall be referenced, and suitably completed, throughout clauses 6, 9, and 10 whenever the identification of a certain actor is performed by signing one or more data objects.

NOTE 2: This will avoid repeating tables that are very similar several times in the present document.

5.2 Tables for defining AdES profiles

In order to minimize its size, the present document defines a format for tables that are able to define one profile for CAdES [3], one profile for JAdES [6], and one profile for XAdES [2] signatures.

These tables shall contain the following columns:

1) At least one column whose header shall be <AdES signature type> components/services, where <AdES signature type> shall identify either XAdES, or CAdES, or JAdES. There shall be as many columns of this type as AdES signatures that the table defines a profile for. Cells in this column may have different contents:

   a) In the case where the cell identifies a Service, the cell content starts with the keyword "Service" followed by the name of the service.

   b) In the case where the component provides a service, this cell contains "SPO" (for Service Provision Option), followed by the name of component given in the corresponding AdES signature format.

   c) Otherwise, this cell contains the name of the component given in the corresponding AdES signature format.

2) Column "Presence": This cell contains the specification of the presence of the component, or the provision of a service. Below follow the values that can appear in this column:

   - "shall be present": means that the component shall be incorporated to the signature with the cardinality indicated in column "Cardinality".

   - "may be present": means that the component may be incorporated to the signature with the cardinality indicated in column "Cardinality".

   - "shall be provided": means that the service identified in the first column of the row shall be provided as further specified in the SPO-related rows. This value only appears in rows that contain requirements for services. It does not appear in rows that contain requirements for components.

   - "conditioned presence": means that the incorporation to the signature of the component identified in the first column, with the cardinality indicated in column "Cardinality", is conditioned.

3) Column "Cardinality": This cell indicates the cardinality of the component. If the cardinality is the same for all the levels, only the values listed below appear. Otherwise the content specifies the cardinality for each level. See the example at the end of the present clause showing this situation. Below follow the values indicating the cardinality:

   - **0:** The signature shall not incorporate any instance of the component.
   - **1:** The signature shall incorporate exactly one instance of the component.
   - **0 or 1:** The signature shall incorporate zero or one instance of the component.
   - **≥ 0:** The signature shall incorporate zero or more instances of the component.
   - ≥ **1:** The signature shall incorporate one or more instances of the component.

4) Column "Additional notes and requirements": This cell contains numbers referencing notes and/or letters referencing additional requirements on the component different to the notes and requirements present in the corresponding AdES signature specification. Both notes and additional requirements are listed below the table. If the cell does not contain any number nor letter the note(s) and requirement(s) in the corresponding AdES specification shall apply.

Absence of a component defined in the corresponding AdES signature format shall be understood as a requirement "shall not be present" with a cardinality of 0.

NOTE: This will ensure that the size of the tables is the smallest possible.

For components of AdES signatures that have been defined in the documents on which each AdES signature has been built (i.e. elements defined in XML Signature W3C® Recommendation [7]-as `ds:Signature`-, elements defined in CMS -as `signatureValue`-, or members defined in JWS -as `signature`- member in a JWS JSON Serialization Syntax), the following rules shall apply:

1) Mandatory components shall obviously be present in the profiled AdES signatures regardless they are mentioned in the tables or not.

2) Optional components may be present in the profiled AdES signatures unless its presence is explicitly forbidden in the corresponding AdES specification, even if they are not mentioned in the tables.

The different AdES signature formats define a good number of components with similar semantics. A table defining requirements of presence and cardinality for an attribute/qualifying property which appears in different AdES signature formats, shall place these requirements in the same row, if these requirements are the same for the attribute/qualifying property in the different AdES signatures.

EXAMPLE 1: Below follows a row specifying requirements for the signingCertificateV2 qualifying property of XAdES and signing-certificate-v2 attribute of CAdES signatures.

XAdES Components/Services	CAdES Components/Services	Presence	Cardinality	Additional requirements and notes
`SigningCertificateV2`	`signing-certificate-v2`	shall be present	1	

If a certain component of one AdES signature format has a semantics that none of the components of another AdES signature format has, then the row defining requirements of presence and cardinality for this component, shall identify it in the column <AdES signature type> components/services corresponding to the AdES signature format where this component is specified, and the cell in the column corresponding to the AdES signature format that does not define any component offering this semantics, contain the legend N/A (not applicable).

Under these circumstances, the row shall define requirements of presence and cardinality for the AdES signature formats whose columns are different than N/A.

5.3 Profiles for standalone (C/J/X)AdES-B-B

Table 1 defines one profile for CAdES-B-B signatures, one profile for JAdES-B-B signatures, and one profile for XAdES-B-B signatures, which shall be used when the digital signature is standalone, i.e. is not included within an ASiC-E container.

In column "XAdES Components/Services", names of XML elements in the namespace whose URI is http://www.w3.org/2000/09/xmldsig# are preceded by prefix `ds`.

Table 1: Profile for AdES-B-B signature signing one data object

XAdES Components/Services	CAdES Components/Services	JAdES Components/Services	Presence	Cardinality	Additional requirements and notes
ds:KeyInfo/X509Data	SignedData.certificates	x5c	shall be present	1	
N/A	content-type	N/A	shall be present	1	
N/A	message-digest	N/A	shall be present	1	
ds:SignedInfo/ds:CanonicalizationMethod	N/A	N/A	shall be present	1	
SigningTime	signing-time	iat	shall be present	1	
SigningCertificateV2	signing-certificate-v2	N/A	shall be present	1	
CommitmentTypeIndication	N/A	N/A	may be present	≥ 0	
N/A	commitment-type-indication	srCms	may be present	0 or 1	
SignaturePolicyIdentifier	signature-policy-identifier	sigPId	may be present	0 or 1	
N/A	cms-algorithm-protection	N/A	may be present	0 or 1	
Service: signing properties, SC Byte Code and binding with SC Package	N/A	N/A	shall be present		
SPO: ds:Reference	N/A	N/A	shall be present	N+1	
N/A	encapContentInfo.econtentType	N/A	shall be present	1	
N/A	encapContentInfo.econtent	N/A	shall be present	1	
N/A	N/A	payload	shall be present	1	
DataObjectFormat	N/A	N/A	conditioned presence	≥ 0	
DataObjectFormat/Description	N/A	N/A	may be present	0 or 1	
DataObjectFormat/MimeType	N/A	N/A	shall be present	1	
DataObjectFormat/Encoding	N/A	N/A	may be present	0 or 1	
DataObjectFormat's ObjectReference attribute	N/A	N/A	shall be present	1	
N/A	Service: identifying the signed data type	N/A	should be present	0 or 1	
N/A	SPO: content-hints	N/A	conditioned presence	0 or 1	
N/A	SPO: mime-type	N/A	conditioned presence	0 or 1	
N/A	N/A	cty	should be present		

5.4 Profiles for standalone (C/J/X)AdES-B-LTA

Table 2 defines requirements for components that have to be added to the (C/J/X)AdES-B-B signatures specified in clause 5.3 to become (C/J/X)AdES-B-LTA standalone signatures.

NOTE: This is for keeping the size of the table to its minimum size.

In column "XAdES Components/Services", names of XML elements in the namespace whose URI is http://www.w3.org/2000/09/xmldsig# are preceded by prefix `ds`.

Table 2: Components to be added AdES-B-B standalone signature for becoming AdES-B-LTA signatures

XAdES Components/Services	CAdES Components/Services	JAdES Components/Services	Presence	Cardinality	Additional requirements and notes
`SignatureTimeStamp`	`N/A`	`sigTst`	shall be present	1	
`CertificateValues`	`N/A`	`xVals`	conditioned presence	0 or 1	
`AnyValidationData`	`N/A`	`anyValData`	conditioned presence	≥ 0	
`AttrAuthoritiesCertValues`	`N/A`	`axVals`	conditioned presence	0 or 1	
`N/A`	Service: revocation values in long-term validation	`N/A`	shall be provided		
`N/A`	SPO: `SignedData.crls.crl`	`N/A`	conditioned presence		
`N/A`	SPO: `SignedData.crls.other`	`N/A`	conditioned presence		
`RevocationValues`	`N/A`	`rVals`	conditioned presence	0 or 1	
`AttributeRevocationValues`	`N/A`	`arVals`	conditioned presence	0 or 1	
Service: Incorporation of validation data for electronic time-stamps	`N/A`		shall be provided	1	
SPO: `TimeStampValidationData`	`N/A`	SPO: `tstVD`	conditioned presence	≥ 0	
SPO: certificate and revocation values embedded in the electronic time-stamp itself	`N/A`	SPO: certificate and revocation values embedded in the electronic time-stamp itself	conditioned presence	≥ 0	
SPO: `AnyValidationData`	`N/A`	SPO: `anyValData`	conditioned presence	≥ 0	
`ArchiveTimeStamp` (defined in namespace whose URI is "http://uri.etsi.org/01903/v1.4.1#")	`Archive-timestamp-v3`	`arcTst`	shall be present	≥ 1	
`RenewedDigestsV2`	`N/A`	`N/A`	conditioned presence	≥ 0	

5.5 Profiles for (C/X)AdES-B-B signatures included in an ASiC-E container

Table 3 defines one profile for CAdES-B-B signatures, and one profile for XAdES-B-B signatures, which shall be included within an ASiC-E container for signing several detached data objects.

N is equal to the number of detached signed data objects in "Cardinality" column.

In column "XAdES Components/Services", names of XML elements in the namespace whose URI is http://www.w3.org/2000/09/xmldsig# are preceded by prefix `ds`.

Table 3: Profile for (C/X)AdES-B-B signature in ASiC-E container for signing several detached data objects

XAdES Components/Services	CAdES Components/Services	Presence	Cardinality	Additional requirements and notes
ds:KeyInfo/X509Data	SignedData.certificates	shall be present	1	
N/A	content-type	shall be present	1	
N/A	message-digest	shall be present	1	
ds:SignedInfo/ds:CanonicalizationMethod	N/A	shall be present	1	
ds:Reference	N/A	shall be present	N+1	
SigningTime	signing-time	shall be present	1	
SigningCertificateV2	signing-certificate-v2	shall be present	1	
DataObjectFormat	N/A	shall be present	N	
DataObjectFormat/Description	N/A	may be present	0 or º	
DataObjectFormat/MimeType	N/A	shall be present	1	
DataObjectFormat/Encoding	N/A	may be present	0 or 1	
DataObjectFormat's ObjectReference attribute	N/A	shall be present	1	
CommitmentTypeIndication	N/A	may be present	≥ 0 and ≤ N	
N/A	commitment-type-indication	may be present	0 or 1	
SignaturePolicyIdentifier	signature-policy-identifier	may be present	0 or 1	
N/A	cms-algorithm-protection	may be present		

5.6 Profile for XAdES-B-LTA long-term signatures included in an ASiC-E container

Table 4 defines a profile for XAdES-B-LTA long-term signatures, which shall be included within an ASiC-E container for signing several detached data objects.

In column "XAdES Components/Services", names of XML elements in the namespace whose URI is http://www.w3.org/2000/09/xmldsig# are preceded by prefix `ds`.

Table 4: Profile for XAdES-B-LTA signature to be included in an ASiC-E container for long-term

Elements/Qualifying properties/Services	Presence	Cardinality	Additional requirements and notes
`ds:KeyInfo/X509Data`	shall be present	1	
`ds:SignedInfo/ds:CanonicalizationMethod`	shall be present	1	
`ds:Reference`	shall be present	N+1	
`SigningTime`	shall be present	1	
`SigningCertificateV2`	shall be present	1	
`DataObjectFormat`	shall be present	N	See note
`DataObjectFormat/Description`	shall be present	1	
`DataObjectFormat/ObjectIdentifier`	shall not be present	0	
`DataObjectFormat/MimeType`	shall be present	1	
`DataObjectFormat/Encoding`	may be present	0 or 1	
`DataObjectFormat's ObjectReference` attribute	shall be present	1	
`CommitmentTypeIndication`	may be present	1	
`SignaturePolicyIdentifier`	may be present	0 or 1	
`SignatureTimeStamp`	shall be present	1	
`CertificateValues`	conditioned presence	0 or 1	
`AnyValidationData`	conditioned presence	≥ 0	
`AttrAuthoritiesCertValues`	conditioned presence	0 or 1	
`RevocationValues`	conditioned presence	0 or 1	
`AttributeRevocationValues`	conditioned presence	0 or 1	
Service: Incorporation of validation data for electronic time-stamps	shall be provided	-	
SPO: `TimeStampValidationData`	conditioned presence	≥ 0	
SPO: certificate and revocation values embedded in the electronic time-stamp itself	conditioned presence	≥ 0	
SPO: `AnyValidationData`	conditioned presence	≥ 0	
`ArchiveTimeStamp` (defined in namespace whose URI is "http://uri.etsi.org/01903/v1.4.1#")	shall be present	≥ 1	
`RenewedDigestsV2`	conditioned presence	≥ 0	
NOTE: Requirement for `DataObjectFormat`. There shall be one `DataObjectFormat` signed qualifying property fore each signed file.			

6 ASiC containers profiles

6.1 Introduction

In cases where the number of data objects to be signed is higher than 1 and they are detached from the digital signatures, then ASiC-E containers including CAdES or XAdES signatures shall be used, signing the mentioned data objects.

The present clause specifies general requirements for ASiC-E containers that the other clauses in the present document can reference whenever the identification of any entity requires their usage.

6.2 Profile for ASiC-E containers for short-term signatures

6.2.1 General requirements

The ASiC container shall be an ASiC-E, as specified in ETSI EN 319 162-1 [1], with one of the following types of AdES signatures: CAdES or XAdES.

6.2.2 Signing with XAdES

6.2.2.1 Requirements for signature file

The ASiC-E container shall contain one file called "signatures.xml" within its META-INF folder.

The "signatures.xml" file shall contain the `asic:XAdESSignatures` root element as specified in clause A.5 of ETSI EN 319 162-1 [1].

The `asic:XAdESSignatures` root element shall contain one `ds:Signature` containing a XAdES signature, whose profile is given in clause 5.5.

6.2.2.2 Specific requirements for the XAdES signature

The XAdES signature shall be a non-distributed XAdES signature.

The XAdES signature shall contain as many `ds:Reference` children of `ds:SignedInfo` as files forming the SC package within the SC-"<SCP Identifier> folder.

Each `ds:Reference` child shall reference one of the mentioned signed files forming the SC package as specified in clause A.6 of ETSI EN 319 162-1 [1].

The XAdES signature shall contain another `ds:Reference` child of `ds:SignedInfo` referencing the `xades:SignedProperties` of the XAdES signature itself.

The XAdES signature shall be a XAdES-B-B signature as specified in Table 3 of clause 5.5 of the present document.

6.2.3 Signing with CAdES

6.2.3.1 Requirements for ASICManifest file

The ASiC-E container shall contain the file named "signature.p7s", which shall contain the CAdES signature on the SC Package, within its META-INF folder.

The ASiC-E container shall contain within the file named "ASiCManifest.xml" an ASiCManifest, within its META-INF folder.

Within the "ASiCManifest.xml" file, its `ASiCManifest` root element shall contain one `SigReference` element.

The `URI` child of the `SigReference` element shall be a URI reference referencing the "signature.p7s" signature file as specified in clause A.6 of ETSI EN 319 162-1 [1].

The `ASiCManifest` root element shall contain as many `DataObjectReference` children elements as files that form the SC Package.

Each `DataObjectReference` child element shall contain an identifier of a digest algorithm and the digest value of one of the files of the SC Package, in its `ds:DigestMethod` and `ds:DigestValue` children elements, respectively.

The digest of one file of the SC Package shall appear within one and only one `DataObjectReference` child element.

Each `DataObjectReference` element shall contain a URI child element, which shall be a URI reference referencing one of the files that form the SC Package.

Each `DataObjectReference` child element shall contain the `MimeType` chid element.

Each file of the SC Package shall be referenced by one and only one `DataObjectReference`.

6.2.3.2 Specific requirements for the CAdES signature

The `SignerInfos` set shall contain only one instance of `SignerInfo` type.

The CAdES signature shall be a CAdES-B-B signature as specified in Table 3 of clause 5.5 of the present document.

6.3 Profile for ASiC-E containers for long-term signatures

6.3.1 General requirements

For ensuring the collective signature present within the ASiC-E containers specified in clause 6.2 of the present document in the long term, they shall either:

1) be preserved by a preservation trust service preserving electronic seals and electronic signatures; or

EXAMPLE: A preservation trust services in eIDAS [i.3].

2) be augmented as specified in clause 4.4.5 of ETSI EN 319 162-1 [1].

6.3.2 Signing with XAdES

When ASiC-E contains XAdES signatures, the long term availability and integrity shall be achieved by incorporating to the XAdES signatures unsigned qualifying properties, as required by clause 4.4.5 of ETSI EN 319 162-1 [1].

The XAdES signature shall be a XAdES-B-LTA signature as specified in Table 4 in clause 5.6.

6.3.3 Signing with CAdES

When ASiC-E contains CAdES signatures, the long term availability and integrity shall be achieved by adding to the ASiC container one ASiCArchiveManifest file for each time-stamp token added to the ASiC container.

7 Requirements on identity validation

7.1 Introduction

The present document requires the validation of electronic identities during the lifecycle of a Smart Contract.

The validation of electronic identities may be carried out either by:

1) validating standalone AdES signatures or ASiC-E containers generated by a certain entity; or

2) using the EUDI Wallet complying with Article 5a of Regulation (EU) 2024/1183 [i.3] amending Regulation (EU) No 910/2014 for requesting and presenting Electronic Attestation of Attributes (EAA hereinafter)/Personal Identification Data (PID hereinafter).

Validation of electronic identities by standalone AdES signatures or ASiC-E containers may include two steps: the actual validation of the standalone AdES signatures or ASiC-E containers (see clause 7.2.1), and the generation of a signed validation report (see clause 7.2.2).

Clause 7.3 specifies requirements for validating electronic identities using the EUDI Wallet.

7.2 Validation of electronic identities based on digital signatures

7.2.1 Requirements on validation of digital signatures

The validation of standalone AdES signatures or ASiC-E containers shall be carried out as specified in ETSI EN 319 102-1 [4].

7.2.2 Requirements on generation of signed validation reports

The entities validating standalone AdES signatures and ASiC-E container should generate validation reports as specified in ETSI TS 119 102-2 [5] and sign them with a standalone AdES-B-B signature as specified in clause 5.3 of the present document.

7.3 Validation of electronic identities based on EUDI Wallet

The present clause specifies requirements for the validation of an electronic identity based on the EUDI Wallet complying with Article 5a of [i.3].

The entities involved in the process, one of which may be an EAA/PID validation service, should use the protocol specified in ETSI TS 119 472-2 [i.4] for requesting and presenting the EAA/PID.

For validating electronic identities of natural persons using PIDs, the involved entities should support the mandatory PID attributes as specified in Table 1 of the Commission Implementing Regulation (EU) 2024/2977 [i.5], Annex a, for natural persons.

For validating electronic identities of legal persons using PIDs, the involved entities should support the mandatory PID attributes as specified in Table 3 of the Commission Implementing Regulation (EU) 2024/2977 [i.5], Annex a, for natural persons.

The entities involved in the validation of electronic identities through the EUDI wallet should support ISO/IEC 18013-5 [i.2] for interfacing to the Wallet.

8 Requirements for Production phase

8.1 Introduction

The present clause defines requirements for identification by means of their digital signatures, of the entities that, according to clause 4 of the present document, act during the SC provision phase.

8.2 Identification of SC Languages, SC Compilers and SC Virtual Machines providers

8.2.1 General requirements

During the SC Production phase, the providers of SC language(s), SC Compiler(s) and SC Virtual Machine(s), respectively, shall identify themselves by signing the products that they provide.

Therefore, the present clause specifies profiles for signing:

1) The SC Language Specification and the SC Language Specification Policy, generated by the SC Language Publisher.

2) The SC Compiler and the SC Compiler Specification Policy, generated by the SC Compiler Publisher.

3) The SC Virtual Machine and the SC Virtual Machine Policy, generated by the SC Virtual Machine Publisher.

The mentioned providers may sign their products with standalone AdES signatures (CAdES, XAdES, or JAdES) as specified in clause 5.3 of the present document.

The mentioned providers may also generate ASiC-E containers as specified in clause 6.2 of the present document for collectively signing several data objects.

The mentioned providers may also ensure the ASiC-E containers in the long-term as specified in clause 6.3 of the present document or using a preservation trust service preserving electronic signatures and electronic seals.

8.2.2 Identifying the SC Language Publisher by its signatures

The SC Language Publisher shall sign the SC language specification and the SC Language Specification Policy data objects.

If the SC Language Publisher generates an ASiC-E container for collectively signing these two data objects, the ASiC-E shall have one child folder of the root folder whose name shall follow the pattern "SCL-"<SC Language Publisher identifier>, where <SC Language Publisher identifier> shall be an identifier of the SC Language Publisher generated by the SC Language Publisher itself, and encoded as specified in bullet b) of clause 4.2 of ETSI EN 319 162-1 [1].

8.2.3 Identifying the SC Compiler Publisher by its signatures

The SC Compiler Publisher shall sign the SC Compiler and the SC Compiler policy data objects.

If the SC Compiler Publisher generates an ASiC-E container for collectively signing these two data objects, the ASiC-E shall have one child folder of the root folder whose name shall follow the pattern "SCC-"<SC Compiler Publisher identifier>, where <SC Compiler Publisher identifier> shall be an identifier of the SC Compiler Publisher generated by the SC Compiler Publisher itself, and encoded as specified in bullet b) of clause 4.2 of ETSI EN 319 162-1 [1].

8.2.4 Identifying the SC Virtual Machine Publisher by its signatures

The SC Virtual Machine Publisher shall sign the SC Virtual Machine and the SC Virtual Machine policy data objects.

If the SC Virtual Machine Publisher generates an ASiC-E container for collectively signing these two data objects, the ASiC-E shall have one child folder of the root folder whose name shall follow the pattern "SCVM-"<SC Virtual Machine Publisher identifier>, where <SC Virtual Machine Publisher identifier> shall be an identifier of the SC Virtual Machine Publisher generated by the SC Virtual Machine Publisher itself, and encoded as specified in bullet b) of clause 4.2 of ETSI EN 319 162-1 [1].

8.3 Identification of SC Publisher

The SC Publisher shall generate an ASiC-E container as specified in clause 6.2 of the present document, enclosing the SC package.

This ASiC-E container shall have one child folder of the root folder whose name shall follow the pattern "SC-"<SC Identifier>, where <SC Identifier> shall be an identifier of the Smart Contract generated by the SC Publisher, and encoded as specified in bullet b) of clause 4.2 of ETSI EN 319 162-1 [1].

The "SC-"<SC Identifier> folder shall contain the files that form the SC Package.

EXAMPLE 1: This folder can contain, for instance: one file containing the SC Source Code; one file containing the SC Byte Code; one file containing the SC legal text; and one file containing the SC Documentation.

For ensuring the ASiC-E container in the long term, it shall either:

1) be preserved by a preservation trust service preserving electronic seals and electronic signatures; or

EXAMPLE 2: A preservation trust services in eIDAS [i.3].

2) be augmented as specified in clause 6.3 of the present document.

8.4 Identification of SC parties

During the SC Production phase, the parties agreeing the Smart Contract shall identify themselves by signing an evidence that they accept the terms and conditions of the Smart Contract, and by signing an evidence that they accept the terms and conditions of the SC Provider.

Each party agreeing the Smart Contract shall sign the acceptance of the terms and conditions of the Smart Contract with an AdES-B-B digital signature as specified in Table 1 of clause 5.3 of the present document.

NOTE: AdES-B-B digital signatures can support Advanced Electronic Signatures/Seals and Qualified Electronic Signatures/Seals.

Each party agreeing the Smart Contract shall sign the acceptance of the terms and conditions of SC Provider with an AdES-B-B digital signature as specified in Table 1 of clause 5.3 of the present document.

For ensuring the mentioned signatures in the long term, they shall either:

1) be preserved by a preservation trust service preserving electronic seals and electronic signatures; or

2) be augmented as specified in clause 5.4 of the present document.

9 Requirements for Deployment phase

9.1 Validation of the ASiC-E enclosing the SC package by the SC Deployer

As it has been mentioned in clause 4 of the present document, the first action of this phase shall be the validation of the ASiC-E container enclosing the SC package by the SC Deployer, as specified in clause 7.2.1 of the present document.

After this validation the SC Deployer shall generate and sign a validation report as specified in clause 7.2.2 of the present document.

This validation report shall be signed with an AdES-B-B signature as specified in Table 1 of clause 5.3 of the present document.

9.2 Evidence of SC Deployment signed by the SC Deployer

9.2.1 Signature on deployed Smart Contract

If the validation of the ASiC-E container enclosing the SC package succeeds, then the SC Deployer shall deploy the Smart Contract on the Electronic Ledger.

After the SC deployment, the SC Deployer generates the SC Deployment Evidence, which shall be a digital signature on the Smart Contract deployed on the Electronic Ledger.

This signature:

1) Shall be an AdES-B-B signature in any of its formats (CAdES, JAdES or XAdES) as specified in Table 1 of clause 5.3 of the present document.

2) Shall indirectly sign the Smart Contract deployed.

3) Shall be bound to the SC Package. This binding shall be implemented by signing digests of the files that form the SC Package as specified in clause 9.2.2 of the present document.

4) And:

   - shall be placed within the same folder as the Smart Contract deployed;
   - shall have the same local name as the local name of the Smart Contract deployed file; and
   - shall have the extension ".xml" if the signature is XAdES, ".p7m" if the signature is CAdES, and ".json" if the signature is JAdES.

9.2.2 SC Package binding file

The file that allows binding with a SC Package shall be a Multipart MIME object with as many parts as files that form the SC Package signed in the ASiC-E container enclosing the SC package, specified in clause 8.3 of the present document.

All these parts shall have type `Text/Plain`.

The `Content-Description` header of the first part shall have as value a global locator of the ASiC container that signs the SC Package.

It is out of the scope of the present document to specify this global locator. It is expected that each SC Provider builds global locators suitable for locating any ASiC container in its systems.

The contents of each part shall be structured as follows:

1) Its first line shall contain the local name of one of the files that form the SC Package, and is placed within the child folder of the root folder "SC-"<SC Identifier>, child of the root folder in the ASiC container that signs the SC Package.

2) Its second line shall contain the OID of a digest algorithm as a sequence of numbers separated by ".".

3) Its third line shall contain the digest value of the file base64-encoded.

9.2.3 Signing with CAdES signature

The member `encapContentInfo.eContentType` shall have as value the following OID:

```
id-aa-ets-sc-package-signature OBJECT IDENTIFIER ::= { itu-t(0) identified-organization(4) etsi(0)
deliverable-domain (128) smart-contract-signatures (19542) 1 }
```

This OID identifies the signed object as a multipart MIME object that includes the digest of the Smart Contract deployed and the digest of the SC Package.

The member `encapContentInfo.eContent` shall be a multipart MIME object with N+1 parts, N being the number of files that form the SC Package.

All the parts shall have type `Text/Plain`.

The `Content-Description` header of the first part shall have the value "SC digest".

The first part shall be structured as follows:

1) Its first line shall contain the URI referencing the deployed Smart Contract file.

2) Its second line shall contain the OID of a digest algorithm as a sequence of numbers separated by ".".

3) Its third line shall contain the digest value of the Smart Contract base64-encoded.

The other N parts shall be the parts that form the Smart Contract binding file specified in clause 9.2.2 of the present document.

By signing this Multipart MIME file, the CAdES signature indirectly signs both the Smart Contract deployed and the SC Package binding file.

9.2.4 Signing with JAdES signature

The JAdES signature Payload shall have the multipart MIME object that has been defined as content of the `encapContentInfo.eContent` of a CAdES signature in clause 9.2.3 of the present document, base64url-encoded.

9.2.5 Signing with XAdES signature

The XAdES signature shall have 3 `ds:Reference` elements.

One of them shall be the one referencing the `ds:Object` containing the signed qualifying properties. Its contents shall be as specified in ETSI EN 319 132-1 [2].

Another `ds:Reference` element shall reference the Smart Contract file as deployed by the SC Deployer.

The `URI` attribute of this `ds:Reference` element shall have a value that allows to any XAdES application to properly retrieve the Smart Contract within the Electronic Ledger where it has been deployed.

Finally, the other `ds:Reference` element shall refer to a `ds:Object` that shall contain the contents of the SC Package binding file base64-encoded.

NOTE: The XAdES signature, therefore indirectly signs the detached Smart Contract file, and the files of the SC Package by signing the enveloped SC Package binding file.

10 Requirements for the SC Execution phase

10.1 Introduction

As it has been mentioned in clause 4 of the present document, the first action of this phase shall be the validation by the SC Provider of the evidence of the Smart Contract deployment signed by the SC Deployer (see clause 10.2 of the present document) before starting the first execution of the SC.

During this phase, before every execution of the Smart Contract, the SC Provider and the SC User shall mutually authenticate each other (see clause 10.3 of the present document).

After the execution of the Smart Contract on the Electronic Ledger, the SC Provider shall generate and sign a Smart Contract Execution Report (see clause 10.4 of the present document).

10.2 Requirements for validation of the signed SC Deployment evidence

The SC Provider, before any execution of the Smart Contract, shall validate the Smart Contract Deployment Evidence generated by the SC Deployer, as specified in clause 7.2.1 of the present document.

The SC Provider, once the mentioned validation has finalized, shall generate and sign a validation report as specified in clause 7.2.2 of the present document.

10.3 Requirements for validation of electronic identities of SC Provider and SC User

Before the execution of the Smart Contract, the SC Provider and the SC User shall mutually authenticate each other, validating their respective electronic identities.

If this validation is performed using the EUDI Wallet:

1) The SC Provider shall request the presentation of EAA/PID to the SC User, and the SC User shall present them to the SC Provider as specified in clause 7.3 of the present document.

2) The SC Provider shall generate an electronic identity validation report, and shall sign it with a standalone AdES-B-B signature as specified in clause 5.3 of the present document.

3) The SC User shall authenticate the SC Provider validating the AdES-B-B signature on the mentioned validation report.

NOTE: The specification of this electronic identity validation report when the validation is performed based on request and presentation of EAAs/PID is out of the scope of the present document.

If this validation is performed using digital signatures:

1) The SC User shall generate a standalone AdES-B-B signature as specified in clause 5.3 of the present document.

2) The SC Provider shall validate the mentioned AdES-B-B signature as specified in clause 7.2.1 of the present document.

3) The SC Provider shall generate and sign a validation report as specified in clause 7.2.2 of the present document.

4) The SC User shall authenticate the SC Provider validating the AdES-B-B signature on the mentioned validation report.

10.4 Requirements on the signed execution report by the SC Provider

After the execution of the Smart Contract, the SC Provider shall sign a report proving that this execution has taken place, with a standalone AdES-B-B signature as specified in clause 5.3 of the present document.

This signed execution report shall include details of the user that requested the execution of the Smart Contract, and whose identity has previously been validated by the SC Provider.

NOTE: The specification of the signed execution report is out of the scope of the present document.

History

Version	Date	Status
V1.1.1	October 2025	Publication