Federal Office
for Information Security

Annex to BSI TR-03129

Cryptographic Message Syntax profile for content signature

Version 1.0

Document history

Version	Date	Description
0.9	2022	Initial Draft
1.0	2022	First Version

Federal Office for Information Security
Post Box 20 03 63
D-53133 Bonn
E-Mail: eid@bsi.bund.de
Internet: https://www.bsi.bund.de

Table of Contents

Tables

1 Introduction

In the eID infrastructure, data exchange takes place between the entities of the associated PKI. This data exchange, which uses the SOAP messaging protocol, is specified in the BSI TR-03129 series [TR-03129-1], [TR-03129-2] and [TR-03129-3]. Certain messages within this data exchange include a Cryptographic Message Syntax (CMS) container, which contains a signed data structure. The respective content placed in these CMS containers is thereby signed by a dedicated private key. Now the receiving entity of the CMS container is able to verify the authenticity of the respective content if it possesses the corresponding certificate belonging to the sending entity.

In the context of the eID infrastructure, the receiving entity refers to the DV (which in this case is called "BerCA"). The sending entity refers to the individual eID servers. A dedicated Request Signer Certificate (RSC) is used to sign the CMS container.

This document specifies the profile of the CMS containers which are used for this purpose. The specification of the Request Signer Certificates for eID applications is defined in [CP-eID].

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 [RFC 2119]. 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.

When used in tables (profiles), the key words are abbreviated as shown in Table 1.

Key word	**Equivalent**	**Abbrev.**
MUST / SHALL	REQUIRED	m
MUST NOT / SHALL NOT	–	x
SHOULD	RECOMMENDED	r
MAY	OPTIONAL	o
–	CONDITIONAL	c

Table 1: Key Words

2 CMS Container Profiles

CMS and the general structure of a CMS container is described in [RFC 5652]. A CMS container gets defined using the ASN.1 data structure described in [X.208-88]. In the following, the general profile of the applied CMS containers will be specified concretely.

2.1 Signed Data Container

Table 2 specifies the general profile of the applied CMS container with the type Signed Data. Using this CMS container type, it is possible to embed data in a CMS container and then sign that data digitally.

According to [RFC 5652],the `SignedData` structure is defined within the sequence `ContentInfo`,

```
ContentInfo ::= SEQUENCE {
      contentType ContentType,
      content [0] EXPLICIT ANY DEFINED BY contentType
}
```

The `ContentType` is application-specific and defined in chapter 3.1. The content itself is the `SignedData` structure according to table 2.

Field	Comment	Type	Value
`SignedData`	MUST	`SEQUENCE`	
`version`	MUST	`CMSVersion (= INTEGER)`	'3'
`digestAlgorithms`	MUST	`DigestAlgorithmIdentifiers (= SET OF DigestAlgorithmIdentifier; = AlgorithmIdentifier)`	Covered in chapter 3.4.1
`encapContentInfo`	MUST	`EncapsulatedContentInfo (= SEQUENCE)`	Contains the content to be signed.
`eContentType`	MUST	`ContentType (= OBJECT IDENTIFIER)`	Covered in chapter 3.2
`eContent`	MUST	`OCTET STRING`	Contains the specific content.
`certificates`	MAY	`CertificateSet (= SET OF CertificateChoices; = CHOICE { certificate Certificate, extendedCertificate ExtendedCertificate, v1AttrCert AttributeCertificateV1, v2AttrCert AttributeCertificateV2, other OtherCertificateFormat })`	The field is optional. It can contain the self-signed X.509 certificate with which the content is signed (i.e. the RSC). If the certificate is included, a cross-check must be performed by the recipient with the

Field	Comment	Type	Value
			stored certificate.
crls	MUST NOT	RevocationInfoChoices (= SET OF RevocationInfoChoice; = CHOICE { crl CertificateList, other OtherRevocationInfoFormat })	*(Not applicable)*
signerInfos	MUST	SignerInfos (= SET OF SignerInfo; = SEQUENCE)	Contains necessary information regarding the signature. Only one SignerInfo shall be provided within this field.
version	MUST	CMSVersion (= INTEGER)	'1'
sid	MUST	SignerIdentifier (= CHOICE { issuerAndSerialNumber IssuerAndSerialNumber, subjectKeyIdentifier SubjectKeyIdentifier })	Contains the DN of the certificate issuer and an issuer-specific certificate serial number to identify the respective certificate.
digestAlgorithm	MUST	DigestAlgorithmIdentifier (= AlgorithmIdentifier)	Covered in chapter 3.4.1
signedAttrs	MUST	SignedAttributes (= SET SIZE (1..MAX) OF Attribute; = SEQUENCE)	Contains a collection of attributes which get signed additionally. The specific attributes are covered in chapter 3.3.
attrType	MUST	OBJECT IDENTIFIER	Contains the type of the respective attribute.
attrValues	MUST	SET OF AttributeValue (= ANY)	Contains the value of the respective attribute.
signatureAlgorithm	MUST	SignatureAlgorithmIdentifier (= AlgorithmIdentifier)	Covered in chapter 3.4.2
signature	MUST	SignatureValue (= OCTET STRING)	Contains the result of the signature generation.

Field	Comment	Type	Value
unsignedAttrs	MUST NOT	UnsignedAttributes (= SET SIZE (1..MAX) OF Attribute: = SEQUENCE)	*(Not applicable)*

Table 2: General CMS Container

3 Additional Configurations

This section specifies additional configurations, which are referred by table 2.

3.1 Content Types

The `ContentType` and the object identifier of the `SignedData` element depends on the use case

ContentType	Object Identifier	Description
id-eIDServer-webserverCertificate	*itu-t(0) identified-organization(4) etsi(0) reserved(127) etsi-identified-organization(0) bsi-de(7) applications (3) eID (2) id-eID-PKI (4) id-eID-PKI-certificates (1) id-eIDServer-certificates (1) id-eIDServer-webserverCertificate(1)*	TLS server certificate for the entanglement
id-eIDServer-PKIComTLSCertificateRequest	*itu-t(0) identified-organization(4) etsi(0) reserved(127) etsi-identified-organization(0) bsi-de(7) applications (3) eID (2) id-eID-PKI (4) id-eID-PKI-certificates (1) id-eIDServer-certificates (1) id-eIDServer-PKIComTLSCertificate(2) id-eIDServer-PKIComTLSCertificateRequest(1)*	Certificate Signing Request (CSR) for the PKI communication
id-eIDServer-requestSignerCertificate	*itu-t(0) identified-organization(4) etsi(0) reserved(127) etsi-identified-organization(0) bsi-de(7) applications (3) eID (2) id-eID-PKI (4) id-eID-PKI-certificates (1) id-eIDServer-certificates (1)id-eIDServer-requestSignerCertificate(3)*	Request Signer Certificate (RSC)

Tabelle 3: ContentType and Object Identifiers

3.2 Encapsulated Content Types

The `ContentType` and the object identifier of the `eContentType` element depends on the use case

eContentType	Object Identifier	Description
pkix(7)	*{iso(1) identified-organization(3) dod(6) internet(1) security(5)*	X.509 certificate according to [RFC 5280]

eContentType	Object Identifier	Description
	mechanisms(5) pkix(7)}	
PKCS-10	*{iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-10(10) }*	Certificate Signing Request (CSR)according to [RFC 2986]

Tabelle 4: ContentType and Object Identifiers

3.3 Signed Attributes

Within the set of attributes to be signed, the following attributes have to be present.

The object identifiers of these attributes were obtained from [RFC 5652].

Type	Value
content-type attribute	
attrType	`{ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) id-contentType(3) }`
attrValues	Contains the '`eContentType`' of the respective container as type OBJECT IDENTIFIER.
message-digest attribute	
attrType	`{ iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs-9(9) id-messageDigest(4) }`
attrValues	Contains the message digest of the '`eContent`' of the respective container as type OCTET STRING.

Table 5: Specific Attributes

3.4 Cryptographic Details

Request Signer Certificates are self-signed X.509 certificates according to [RFC 5280]. The domain parameters of the public key of a Request Signer Certificate must correspond to the domain parameters of CVCA certificates. The domain parameters of CVCA certificates are specified in [TR-03116-2]. Depending on this, the matching cryptographic algorithms and parameters must be used within the CMS container.

3.4.1 Digest Algorithms

The digests algorithms listed in table 6 have to be applied.

The object identifiers of these algorithms are obtained from [RFC 5754].

Algorithm	Length	Object Identifier
SHA-256	256 Bit	`{ joint-iso-itu-t(2) country(16) us(840) organization(1) gov(101) csor(3) nistAlgorithms(4) hashalgs(2) sha256(1) }`

Table 6: Specific Digest Algorithms

3.4.2 Signature Algorithms

The signature algorithms listed in table 7 have to be applied.

The object identifiers of these algorithms are obtained from [TR-03111].

Algorithm	Length	Object Identifier
ECDSA	256 Bit	`{ itu-t(0) identified-organization(4) etsi(0) reserved(127) etsi-identified-organization(0) bsi-de(7) algorithms(1) id-ecc(1) signatures(4) ecdsa-plain-signatures(1) ecdsa-plain-SHA256(3) }`

Table 7: Specific Signature Algorithms

3.4.3 Domain Parameters

The domain parameters for elliptic curves listed in table 8 have to be applied.

The object identifiers of these specific curves are obtained from [RFC 5639].

Curve	Length	Object Identifier
brainpoolP256r1	256 Bit	`{ iso(1) identified-organization(3) teletrust(36) algorithm(3) signatureAlgorithm(3) ecSign(2) ecStdCurvesAndGeneration(8) ellipticCurve(1) versionOne(1) brainpoolP256r1(7) }`

Table 8: Specific Domain Parameters

Reference Documentation

TR-03129-1	BSI: Protocols for the Management of Certificates and CRLs in Public-Key-Infrastructures (PKIs), Part 1: Common Specifications, 2022
TR-03129-2	BSI: PKIs for Machine Readable Travel Documents, Part 2: Supplemental specifications for public and official authorities, 2017
TR-03129-3	BSI: Protocols for the Management of Certificates and CRLs in Public-Key-Infrastructures (PKIs), Part 3: Electronic Identity (eID) documents based on Extended Access Control (EAC), 2022
CP-eID	BSI: Certificate Policy für die Country Verifying Certification Authority, eID-Anwendung, 2021
RFC 2119	S. Bradner: Key words for use in RFCs to Indicate Requirement Levels, 1997
RFC 5652	R. Housley: Cryptographic Message Syntax (CMS), 2009
X.208-88	CCITT: Recommendation X.208: Specification of Abstract Syntax Notation One (ASN.1), 1988
RFC 5280	D. Cooper, et al.: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile, 2008
RFC 2986	M. Nystrom, B. Kaliski: PKCS #10: Certification Request Syntax Specification Version 1.7, 2000
TR-03116-2	BSI: Kryptographische Vorgaben für Projekte der Bundesregierung, Teil 2: Hoheitliche und eID-Dokumente, 2022
RFC 5754	S. Turner: Using SHA2 Algorithms with Cryptographic Message Syntax, 2010
TR-03111	BSI: Elliptic Curve Cryptography, 2018
RFC 5639	M. Lochter, et al.: Elliptic Curve Cryptography (ECC) Brainpool Standard Curves and Curve Generation, 2010