Federal Office
for Information Security

Basics of Digital Signature Techniques and Trust Services

Designation:	**Legal Framework** **Technical Aspects**
Abbrevation:	BSI DSig
Version:	2.0
Stand:	18th April 2023

Version	Date	Author	Description
2.0	18th April 2023	BSI	New Version on base of eIDAS [(EU)910/2014]
1.0	3rd March 2006	BSI	Initial Version on base of German Digital Signature Act (Deutsches Signaturgesetz 2001) [SigG]

Federal Office for Information Security
Postfach 20 03 63, 53133 Bonn, Germany
E-Mail: tresor@bsi.bund.de
Internet: https://www.bsi.bund.de

Acknowledgements

This publication was commissioned by Mario Engel and Dr. Ulrike Korte and produced in cooperation with the Federal Office for Information Security. We would like to express our sincere thanks to all who supported the creation of this document and provided us with valuable advice and corrections.

Dr. Detlef Hühnlein (ecsec GmbH), Steffen Schwalm (msg systems ag), Dr. Chrisitan Berghoff, Dr. Ute Gebhard, Dr. Ulf Löckmann, Dr. Rainer Plaga

Contents

List of Figures

List of Tables

Abstract

The eIDAS-regulation, which came fully into force in July 2016, provides a Europe wide mandatory legal framework for digital identities and trust services. It enables trustworthy digital transaction between public administrations, companies and citizens of the European Member States. In this context, eIDAS contains two main parts: digital identities and trust services. This publication focuses on these trust services, introduced by eIDAS and explains in detail

- the legal framework on the eIDAS-Regulation (EU) No 910/2014 [(EU)910/2014],
- the eIDAS/ETSI- framework based on [M460] of the European Commission for standardisation of electronic signatures, seals and timestamps and trust services,
- the technical principles concerning the cryptographic foundations of digital signature techniques,
- the formats of signatures, seals, timestamps and evidence records on base of the eIDAS/ETSI-framework for standardisation and the RFC-Standards [RFC3161, RFC4998, RFC6283],
- the trust services in practice, based on the eIDAS-Regulation.

Concerning the trust services, this publication deals in detail with the generation and validation of (qualified) electronic signatures, seals and timestamps as well as with preservation services and additionally mentions further trust services and the trusted lists. (Qualified) electronic signatures or seals make the authenticity and integrity of electronic records evident against 3rd parties, a (qualified) time stamp gives a valid Proof of Existence (PoE) and evidence for the time of transactions. Therefore, digital signature techniques and trust services are of particular importance here because they make any manipulation or falsification of the original electronic data immediately recognisable and therefore represent the technical basis for the secure and legally binding execution of electronic business transactions.

1 Legal Framework

1.1 The eIDAS-Regulation (EU) No 910/2014

Against the background of the experiences gathered with the implementation of the European Signature Directive [1999/93/EC] in the different Member States of the European Union and in cross-border scenarios, the European Commission in *June 2012* started the legislative procedure [2012/0146/COD] and published the ”Proposal for a regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market” [COM(2012)238]. In 2014 the ”Regulation (EU) No 910/2014 of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC” [(EU)910/2014] was finally published, which is commonly known as ”eIDAS-Regulation”.

The eIDAS-Regulation has entered into force on 17 September 2014[1] and according to *Article 52 Nr. 2* it has become fully applicable on *1 July 2016*, which is also the day on which the Signature Directive [1999/93/EC] was repealed[2].

1.1.1 Overview and Timeline of the eIDAS-Regulation

The eIDAS-Regulation [(EU)910/2014] contains 77 recitals, which explain the background of this regulation and the motivation behind it, as well as 52 articles organised in six chapters and four annexes.

The six chapters of the eIDAS-Regulation address the following topics:

1.) General Provisions (*Articles 1-5*, see Section 1.1.2),

2.) Electronic Identification (eID) (*Articles 6-12*, see Section 1.1.3),

3.) Trust Services (*Articles 13-45*, see Section 1.1.4),

4.) Electronic Documents (*Article 46*, see Section 1.1.5),

5.) Delegation of Power and Implementing Provisions (*Articles 47-48*, see Section 1.1.6) and

6.) Final Provisions (*Articles 49-52*, see Section 1.1.7).

[1] This is the twentieth day after its publication in the Official Journal of the European Union on *28 August 2014* (cf. [(EU)910/2014, Article 52 Nr. 1])

[2] See [(EU)910/2014, Article 52 Nr. 2 and Article 50].

An overview with respect to the implementing acts related to the eIDAS-Regulation [(EU)910/2014] is provided in Table 1.1.

Regulation	Article	Subject	in Force
[(EU)2015/296]	12 (7)	Collaboration of Member States with respect to eID	2015/03/17
[(EU)2015/806]	23 (3)	EU Trust Mark for Qualified Trust Services	2015/06/12
[(EU)2015/1501]	12 (8)	eID Interoperability Framework	2015/09/29
[(EU)2015/1502]	8 (3)	Stipulations for eID Assurance Levels	2015/09/29
[(EU)2015/1505]	22 (5)	Technical Specification and Format for Trusted Lists	2015/09/29
[(EU)2015/1506]	27 (5), 37 (5)	Formats for Advanced Electronic Signatures and Seals	2015/09/29
[(EU)2015/1984]	9 (5)	Details for Notification of eID Schemes	2015/11/05
[(EU)2016/650]	30 (3), 39 (2)	Standards for QSCD Security Assessment	2016/05/16

Table 1.1: Overview of the Implementing Acts of the eIDAS-Regulation

The eIDAS-Regulation introduces an interoperability framework for *electronic identification* (eID)[3], which is explained in Section 1.1.3 in more detail. The Member States collaborate with respect to the implementation of this interoperability framework based on the principles and procedures set out in [(EU)2015/296]. This includes the exchange of information (*Articles 4-6*), a peer review process (*Articles 7-11*) and the establishment of a ”Cooperation Network” (Articles 12 et seq). According to [(EU)910/2014, Article 8] an eID scheme may have assurance level ”low”, ”substantial” or ”high”, whereas the technical and organization details with respect to assigning the assurance levels are subject to [(EU)2015/1502]. [(EU)2015/1984] defines the details with respect to the notification of eID schemes according to [(EU)910/2014, Article 9]. While a Member State may decide to recognise a notified eID scheme earlier[4], the mutual recognition of notified eID schemes according to [(EU)910/2014, Article 6] became obligatory on *29 September 2018*[5].

With respect to *trust services* it is important to note, that Secure Signature Creation Devices (SSCD), Qualified Certificates (QC) and Certification-Service-Providers (CSP) according to [1999/93/EC] are compliant to [(EU)910/2014], albeit a CSP needs to submit a Conformity Assessment Report (CAR) to its Supervisory Body (SB) until *1 July 2017*[6]. The qualified Trust Services (TS) according to the eIDAS-Regulation are listed in Trusted Lists (TL) according to [(EU)2015/1505, ETSI-119612(v2.2.1)] and the qualified Trust Service Providers (TSP) may use the EU Trust Mark according to [(EU)2015/806], as depicted in Figure 1.1[7]. Furthermore [(EU)2015/1506] specifies the formats for advanced electronic signatures and seals, accepted by public sector bodies in the EU Member States. [(EU)2016/650] refers to protection profiles for the security assessment of qualified electronic signature/seal creation device (QSCD)[8].

[3] See [(EU)910/2014, Article 12] and [(EU)2015/1501].

[4] See [(EU)910/2014, Article 52 (4)].

[5] See [(EU)910/2014, Article 52 (1. c)] and the entry into force of the implementing acts according to Article 8 (3) (i.e. [(EU)2015/1502]) and Article 12 (8) (i.e. [(EU)2015/296]) on *29 September 2015*.

[6] See [(EU)910/2014, Article 51].

[7] The EU Trust Mark can be downloaded from `https://ec.europa.eu/digital-single-market/en/news/eu-trust-mark-logos-ready-download`.

[8] See [(EU)910/2014, Article 29]

Figure 1.1: EU Trust Mark according to [(EU)2015/806]

1.1.2 Chapter I - General Provisions - Articles 1-5

The *first Chapter* of the eIDAS-Regulation provides general provisions regarding

- subject matter (Article 1),
- scope (Article 2),
- definitions (Article 3),
- internal market principle (Article 4) and
- data processing and protection (Article 5).

Article 1 - Subject Matter

According to *Article 1*, the eIDAS-Regulation

(a) addresses the cross-border recognition of notified electronic identification (eID) schemes for natural and legal persons (see Section 1.1.3),

(b) ”lays down rules for trust services, in particular for electronic transactions” (see Section 1.1.4) and

(c) establishes a legal framework for
   - electronic signatures (see Section 1.1.4.4),
   - electronic seals (see Section 1.1.4.5),
   - electronic time stamps (see Section 1.1.4.6),
   - electronic documents (see Section 1.1.5),
   - electronic registered delivery services (see Section 1.1.4.7) and
   - certificate services for website authentication (see Section 1.1.4.8).

Article 2 - Scope

Article 2 clarifies that the eIDAS-Regulation only applies to (1) notified eID schemes and trust service providers, established in the European Union, which are (2) not exclusively used in closed systems and the regulation (3) does not address national law with respect to the conclusion and validity of contracts and legal obligations with respect to form. (The eIDAS-regulation was adopted by the EFTA-members Norway, Iceland and Liechtenstein so that it applies for eID-schemes and trust service there too. [9492/21])

Article 3 - Definitions

Article 3 defines the most important terms with respect to the eIDAS-Regulation including electronic identification, authentication, electronic signature and trust service for example.

Article 4 - Internal market principle

Article 4 underlines that there shall be no restriction with respect to the cross-border provisioning of trust services and products as well and that trust services, which comply with the eIDAS-Regulation, may freely circulate in the internal market of the European Union (see [WeKa14]) and in the EFTA-states Norway, Iceland and Liechtenstein.

Article 5 - Data processing and protection

Article 5 refers to the Data Protection Directive [95/46/EC], replaced by the General Data Protection Regulation [(EU)2016/679], and states that the use of pseudonyms in electronic transactions shall not be prohibited.

1.1.3 Chapter II - Electronic Identification - Articles 6-12

The *second Chapter* of the eIDAS-Regulation is dedicated to electronic identification. It defines under which circumstances electronic identification schemes and electronic identification means are mutually recognised for cross-border authentication among the different EU Member States.

For this purpose, the eIDAS-Regulation defines the assurance levels ”low”, ”substantial” and ”high” (Article 8), which correspond to varying levels of confidence in a claimed or asserted identity of a person and defines prerequisites (Article 7) and procedures (Article 9) for the notification of electronic identity schemes, which enjoy the mutual recognition (Article 6) in a cross-border setting. To ensure an adequate level of trust in the pan-European electronic identification framework, there are obligations for the EU Member States which are to be fulfilled in case of a security breach (Article 10), stipulations for liability (Article 11) and rules governing the cooperation of EU Member States and the interoperability of the notified electronic identification schemes (Article 12).

Article 6 - Mutual recognition

According to *Article 6* electronic identification means shall be recognised in a cross-border authentication, provided that the corresponding scheme is included in the list of notified schemes, published by the European Commission pursuant to Article 9 (1.a). The provided assurance level ”corresponds to an assurance level equal or higher is equal to or higher than” the required assurance level ”provided that the assurance level of that electronic identification means corresponds” to the level ”substantial” or ”high”[9](1. b) and the required assurance level of the service provided by a public sector body is at least ”substantial” (1. c).

As already mentioned above, the obligatory mutual recognition started on *29 September 2018*.

Article 7 - Eligibility for notification of electronic identification schemes

A prerequisite for the mutual recognition of electronic identification means is that an EU Member State notifies an electronic identification scheme, which fulfills the requirements defined in Article 7, to the European Commission.

[9] The recognition of electronic identification means, which only fulfill the assurance level ”low” is optional (see Article 6 (1)).

Among the requirements for an electronic identification scheme to be eligible for notification according to *Article 7* it is requested that ”(a) the electronic identification means under the electronic identification scheme are issued: (i) by the notifying Member State; (ii) under a mandate from the notifying Member State; or (iii) independently of the notifying Member State and are recognised by that Member State”. The ”notifying Member State ensures that the person identification data uniquely representing the person in question is attributed, in accordance with the technical specifications, standards and procedures for the relevant assurance level set out in the implementing act referred to in Article 8(3), to the natural or legal person referred to in point 1 of Article 3 at the time the electronic identification means under that scheme is issued” and ensures the availability of cross-border authentication, which is free of charge for services online provided by public sector bodies (see *Article 7 (f)*).

Article 8 - Assurance Levels of Electronic Identification Schemes

An important attribute of an electronic identification scheme is the level of assurance provided by it, which shall be specified within the notification (*Article 8 (1)*).

Article 8 (2) of [(EU)910/2014] defines the assurance levels ”low”, ”substantial” and ”high”, which ”depend on the *degree of confidence* that electronic identification means provides in claimed or asserted identity of a person” and which is equipped with controls, which differently decrease the *risk of misuse or alteration* of the identity as summarised in Table 1.2.

Assurance level	Degree of confidence	Risk of misuse or alteration
low	limited degree of confidence	decrease the risk
substantial	substantial degree of confidence	decrease substantially the risk
high	higher degree of confidence	prevent misuse or alteration

Table 1.2: Assurance levels of electronic identification schemes

Further details with respect to the determination of the assurance level of an electronic identification scheme are specified in the implementing act [(EU)2015/1502] referred to in *Article 8 (3)* of the eIDAS-Regulation. According to *Article 1* and the Annex of the implementing act [(EU)2015/1502] the graded set of requirements for the three assurance levels address the reliability and quality of aspects like

(a) Enrolment
  - Application and registration
  - Identity proofing and verification (natural person)
  - Identity proofing and verification (legal person)
  - Binding between the electronic identification means of natural and legal persons

(b) Electronic identification means management
  - Electronic identification means characteristics and design
  - Issuance, delivery and activation
  - Suspension, revocation and reactivation
  - Renewal and replacement

(c) Authentication

(d) Management and organisation

- General provisions
- Published notices and user information
- Information security management
- Record keeping
- Facilities and staff
- Technical controls
- Compliance and audit

Article 9 - Notification

As specified in [(EU)910/2014, Article 9 (1)] the notification includes ”

(a) a description of the electronic identification scheme, including its assurance levels according to *Article 8* and the issuer(s) of electronic identification means under the scheme;

(b) the applicable supervisory regime and information on the liability regime with respect to

(i) the party *issuing* the electronic identification means; and

(ii) the party *operating* the authentication procedure;

(c) the authority or authorities responsible for the electronic identification scheme;

(d) information on the entity or entities which manage the registration of the unique person identification data;

(e) a description of how the requirements set out in the implementing act [(EU)2015/1502] referred to in *Article 12 (8)* are met;

(f) a description of the authentication referred to in point (f) of *Article 7*;

(g) arrangements for suspension or revocation of either the notified electronic identification scheme or authentication or the compromised parts concerned.”

According to *Article 9 (2)* the European Commission has been obliged to publish the list[10] of notified electronic identification schemes according to *Article 9* since *29 September 2016*[11] and to publish according amendments within two month after the receipt of further notifications (*Article 9 (3)*). According to (*Article 9 (4)*), a Member State may request ”to remove an electronic identification scheme notified by that Member State from the list referred to in paragraph 2” and all notifications shall be made electronically in the format defined in the Annex of [(EU)2015/1984] (*Article 9 (5)*).

[10] https://ec.europa.eu/digital-building-blocks/wikis/display/EIDCOMMUNITY/Overview+of+pre-notified+and+notified+eID+schemes+under+eIDAS

[11] See [(EU)910/2014, Article 9 (2)] together with [(EU)2015/1501] and [(EU)2015/1502].

Article 10 - Security breach

In case the notified eID scheme or the cross-border authentication ”is breached or partly compromised in a manner that affects the reliability of the cross-border authentication of that scheme”, the notifying Member State shall according to *Article 10 (1)* immediately suspend or revoke the cross-border authentication or compromised parts concerned, and shall inform other Member States and the European Commission. In a similar manner, ”the notifying Member State shall re-establish the cross-border authentication and shall inform other Member States and the Commission”, when a breach or compromise is remedied (*Article 10 (2)*). If a breach or compromise will not be remedied within three months after the suspension or revocation, the notifying Member State shall notify the other Member States and the European Commission of the withdrawal of the eID scheme (*Article 10 (3)*).

Article 11 - Liability

According to *Article 11 (1)*, the notifying Member State shall be liable for damages caused to any person due to a failure related to the correct assignment of identity attributes to a person (*Article 7 (d)*) and problems related to the cross-border authentication procedure (*Article 7 (f)*). Furthermore, the party issuing the electronic identification means shall be liable for damages caused by failures related to the issuing of the electronic identification means (*Article 11 (2)*) and the party operating the authentication procedure is ”liable for damage caused intentionally or negligently to any natural or legal person due to a failure to ensure the correct operation of the authentication referred to in point (f) of *Article 7* in a cross-border transaction” (see *Article 11 (3)*).

Article 12 - Cooperation and interoperability

In order to implement the envisioned cross-border authentication and identification in practice, *Article 12* introduces an interoperability framework, which according to *Article 12 (3)* ”meet the following criteria:

(a) it aims to be technology neutral and does not discriminate between any specific national technical solutions for electronic identification within a Member State;

(b) it follows European and international standards, where possible;

(c) it facilitates the implementation of the principle of privacy by design; and

(d) it ensures that personal data is processed in accordance with Directive [95/46/EC].[12]

According to *Article 12 (4)* the ”interoperability framework shall consist of

(a) a reference to minimum technical requirements related to the assurance levels under *Article 8*;

(b) a mapping of national assurance levels of notified electronic identification schemes to the assurance levels under *Article 8*;

(c) a reference to minimum technical requirements for interoperability;

(d) a reference to a minimum set of person identification data uniquely representing a natural or legal person, which is available from electronic identification schemes;

[12] The Directive [95/46/EC] replaced by the new General Data Protection Regulation [(EU)2016/679] from *May 25th, 2018* onwards.

(e) rules of procedure;

(f) arrangements for dispute resolution; and

(g) common operational security standards."

Further dewtails with respect to the interoperability framework are defined in the implementing act [(EU)2015/1501] pursuant to *Article 12 (8)* and the related technical specifications [eIDAS-IA, eIDAS-SAML-MF, eIDAS-SAML-AP, eIDAS-CR].

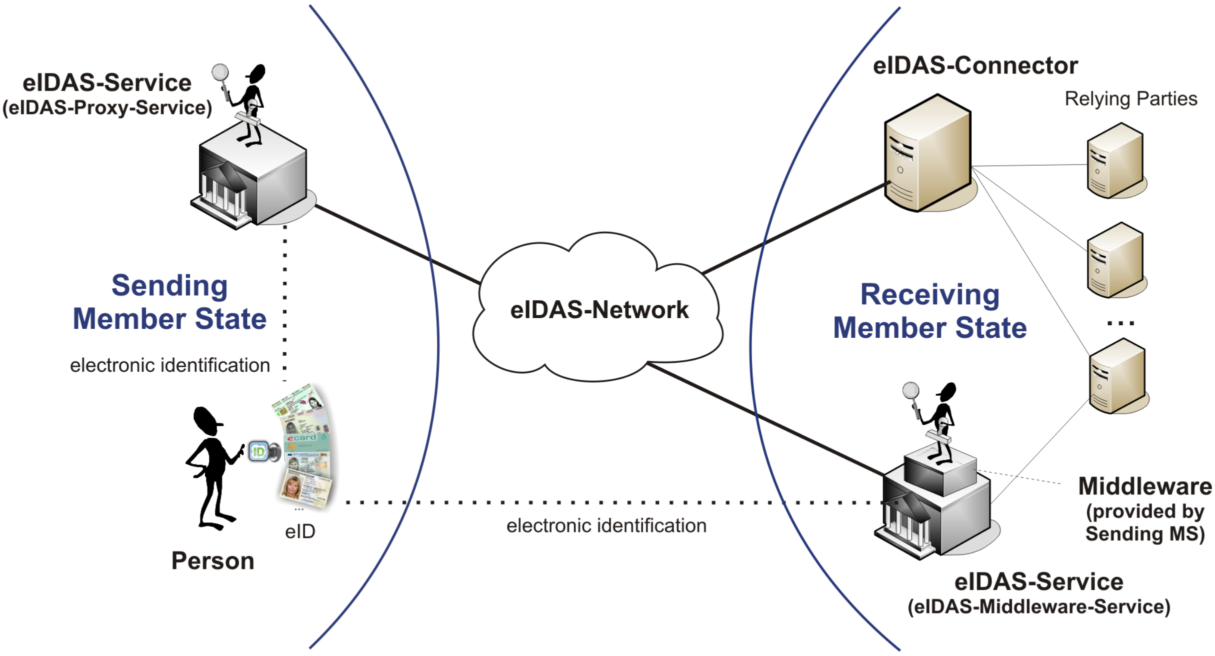


Figure 1.2: eIDAS-Network according to [eIDAS-IA]

As defined in [eIDAS-IA] and depicted in Figure 1.2, the interoperability framework is realised by a network ("eIDAS-Network") of connection points ("eIDAS-Nodes"), which are involved in the cross-border authentication process. Considering the role within the authentication process one may distinguish between an "eIDAS-Connector", which is *requesting* a cross-border authentication and an "eIDAS-Service", which is *providing* cross-border authentication. An eIDAS-Service may be realised as "eIDAS-Proxy-Service", which is operated by the Sending Member State, or as an "eIDAS-Middleware-Service", which is operated by the Receiving Member State using software, called "Middleware", provided by the Sending Member State.

Article 6 of [(EU)2015/1501] requires that the "protection of privacy and confidentiality of the data exchanged and the maintenance of data integrity between[13] the nodes shall be ensured by using best available technical solutions and protection practices."

Furthermore *Article 10* of [(EU)2015/1501] requires that operators of eIDAS-Services fulfill the "requirements of standard ISO/IEC 27001 by certification, or by equivalent methods of assessment, or by complying with national legislation."

Article 11 together with the Annex of [(EU)2015/1501] define a minimum set of person identification data uniquely representing a natural or legal person in a cross-border context.

[13] It should be noted that the secure transmission of the SAML-based messages *between* the eIDAS-Nodes may not be sufficient to provide adequate security against (e.g. man-in-the-middle) attacks (cf. [EHS09]) mounted between the eIDAS-Connector and the Relying Parties. As noted in [eIDAS-IA, Section 3.1] this interface is beyond the scope of the interoperability framework and "must be secure enough to meet the requirements of the maximum level of assurance transmitted by the Connector, in order to ensure the authenticity and confidentiality of the transmitted personal identification data."

Article 12 of [(EU)2015/1501] states that the Cooperation Network established by [(EU)2015/296], which is explained below, may adopt opinions on the need to develop technical specifications, which shall be developed by the European Commission in cooperation with the EU Member States as part of the digital service infrastructure according to [(EU)2021/1153], which establishes the ”Connecting Europe Facilty” (CEF).

Article 12 (5) of [(EU)910/2014] requires that the Member States shall cooperate with respect to (a) the interoperability of electronic identification schemes, which are (meant to be) notified pursuant to *Article 9 (1)* and (b) the security of electronic identification schemes.

According to *Article 12 (6)* the ”cooperation between the Member States shall consist of

(a) the exchange of information, experience and good practice as regards electronic identification schemes and in particular technical requirements related to interoperability and assurance levels;

(b) the exchange of information, experience and good practice as regards working with assurance levels of electronic identification schemes under *Article 8*;

(c) peer review of electronic identification schemes falling under this Regulation; and

(d) examination of relevant developments in the electronic identification sector.”

More details with respect to the cooperation between the Member States have been specified in [(EU)2015/296], which in particular includes the establishment of

- ”points of single contact” within the Member States (*Article 3*),
- ”the Cooperation Network” (see [eIDAS-CN]) (*Article 12*), which consists of representatives of the EU Member States and countries of the European Economic Area and is chaired by the European Commission (*Articles 15-16*).

According to *Article 14* of [(EU)2015/296] the Cooperation Network is for example mandated to adopt opinions[14] on developments relating to the interoperability framework according to [(EU)2015/1501] and the minimum technical specifications, standards and procedures regarding assurance levels according to [(EU)2015/1502] and support the peer review (*Articles 7-10*) and notification process.

1.1.4 Chapter III - Trust Services - Articles 13-45

The *third chapter* of the eIDAS-Regulation addresses trust services, which are according to *Article 3 (16)* defined to be ”an electronic service normally provided for remuneration which consists of:

(a) the creation, verification, and validation of electronic signatures, electronic seals or electronic time stamps, electronic registered delivery services and certificates related to those services, or

(b) the creation, verification and validation of certificates for website authentication; or

(c) the preservation of electronic signatures, seals or certificates related to those services”.

1.1.4.1 Section 1 - General Provisions - Articles 13-16

The *first section* within this chapter is dedicated to general provisions, which include rules for liability (Article 13), international aspects (Article 14), accessibility (Article 15) and penalties (Article 16).

[14] See [eIDAS-CNO-16-01, eIDAS-CNO-16-02].

Article 13 - Liability and burden of proof

Article 13 defines the general rules with respect to liability and the burden of proof. Within possible liability limitations (*Article 13 (2)*) and with application of the national rules on liability *(Article 13 (3)*), trust service providers (TSP) are liable for any damage caused by a failure to comply with the obligations under the eIDAS-Regulation (*Article 13 (1)*). While in case of a non-qualified TSP, the burden of proof lies with the damaged person, a qualified trust service provider (QTSP) shall be presumed to be liable ”unless that qualified trust service provider proves that the damage referred to in the first subparagraph occurred without the intention or negligence of that qualified trust service provider.”

Article 14 - International aspects

Article 14 regulates international aspects and in particular states that TSP established in a third country shall be recognised as legally equivalent to TSP established in the EU, if there is an applicable agreement between the EU and the third country or international organisation according to *Article 218* of the Treaty of the Functioning of the European Union [TFEU] (1). Such an agreement shall ensure in particular that the foreign TSP fulfills the applicable requirements (2a) and that European TSP are on the other hand also recognised in the third country or international organisation under consideration (2b).

Article 15 - Accessibility for persons with disabilities

Article 15 stipulates that, where feasible, trust services and related end-user products shall be made accessible for persons with disabilities.

Article 16 - Penalties

Article 16 regulates that penalties applicable to infringements of the eIDAS-Regulation shall be defined by the Member States, such that they are effective, proportionate and dissuasive.

1.1.4.2 Section 2 - Supervision - Articles 17-19

The *second section* defines requirements for supervision, which includes regulations for the supervisory body (SB) (Article 17), rules related to mutual assistance (Article 18) and the definition of general security requirements applicable to TSP (Article 19).

Article 17 - Supervisory Body

According to *Article 17*, Member States shall (1) designate an appropriately equipped SB established in their territory or, upon mutual agreement with another Member State, a SB established in that other Member State and (2) notify to the Commission the names and addresses of their respective designated SB. The role of the SB (3) shall be to supervise (a) qualified trust service providers through ex ante[15] and ex post[16] supervisory activities that they meet the requirements of the eIDAS-Regulation and (b) non-qualified trust service providers through ex post supervisory activities, when informed that those do not meet the stipulated requirements.

According to *Article 17 (4)* the tasks of the SB ”shall in particular include

[15] ”Ex ante” based on forecasts rather than actual results.

[16] ”Ex post” based on actual results rather then forecasts.

(a) to cooperate with other SB and provide them with assistance in accordance with *Article 18*;

(b) to analyse the Conformity Assessment Reports (CAR) referred to in *Articles 20 (1) and 21 (1)*;

(c) to inform other supervisory bodies and the public about breaches of security or loss of integrity in accordance with *Article 19 (2)*;

(d) to report to the Commission about its main activities in accordance with *Article 17 (6)*;

(e) to carry out audits or request a Conformity Assessment Body (CAB) to perform a conformity assessment of the qualified trust service providers in accordance with *Article 20 (2)*;

(f) to cooperate with the data protection authorities, in particular, by informing them without undue delay, about the results of audits of qualified trust service providers, where personal data protection rules appear to have been breached;

(g) to grant qualified status to trust service providers and to the services they provide and to withdraw this status in accordance with *Articles 20 and 21*;

(h) to inform the body responsible for the national trusted list referred to in *Article 22 (3)* about its decisions to grant or to withdraw qualified status, unless that body is also the supervisory body;

(i) to verify the existence and correct application of provisions on termination plans in cases where the qualified trust service provider ceases its activities, including how information is kept accessible in accordance with point (h) of *Article 24 (2)*;

(j) to require that trust service providers remedy any failure to fulfil the requirements laid down in this Regulation."

According to *Article 17 (5)*, the "Member States may require SB to establish, maintain and update a trust infrastructure in accordance with the conditions under national law." As stipulated in *Article 17 (6)*, each SB is required to submit an annual activity report[17] to the European Commission, which in particular contains a summary of the breach notifications received from the trust service providers according to *Article 19 (2)* and the European Commission will make the annual reports available to Member States (*Article 17 (7)*).

Article 18 - Mutual assistance

According to *Article 18*, the SB "shall cooperate with a view to exchanging good practice." This cooperation is in particular supported by the informal "Forum of European Supervisory Authorities" (FESA)[18].

Article 19 - Security requirements applicable to trust service providers

Article 19 specifies security requirements applicable to trust service providers. According to *Article 19 (1)* qualified and non-qualified TSP "shall take appropriate technical and organisational measures to manage risks posed to the security of the trust services they provide".

[17] According to *Article 17 (8)* the Commission may, by means of implementing acts, define formats and procedures for this report.

[18] See `http://www.fesa.eu/`.

Under *Article 19 (2)* the qualified and non-qualified trust service providers shall immediately, but in any event within 24 hours, inform the SB in charge, and other relevant stakeholders[19], in case of a security breach (*Article 19 (2)*). Each SB to provide ENISA once a year a summary of notifications received from TSP (*Article 19 (3)*).

According to *Article 19 (4)* the Commission may, by means of implementing acts, (a) further specify the required security measures referred to in paragraph (1) and (b) define formats and procedures for the breach notification, including applicable deadlines, according to paragraph (2). While these implementing acts are not yet existing, or known to be developed, there are drafts of corresponding guidelines on appropriate security levels of TSPs [ENISA-SL], on supervision of trust service providers [ENISA-STS] and a proposal for incident reporting [ENISA-IR] developed by ENISA. However, they have no legally binding character.

1.1.4.3 Section 3 - Qualified Trust Services - Articles 20-24

The *third section* is dedicated to qualified trust services and contains rules for

- the supervision of qualified trust service providers (QTSP) (Article 20),
- the initiation of a qualified trust service (QTS) (Article 21),
- trusted lists (TL) (Article 22),
- the EU Trust Mark for qualified trust services (Article 23) and
- requirements for QTSP (Article 24).

Article 20 - Supervision of qualified trust service providers

Under *Article 20 (1)*, QTSP ”shall be audited at their own expense at least every 24 months by a conformity assessment body” (CAB), which needs to be accredited according to [(EC)765/2008] and be competent with respect to [(EU)910/2014]. The purpose of the audit shall be to confirm that the QTSP and the qualified trust services (QTS) provided by them fulfil the requirements laid down in the eIDAS-Regulation. The QTSP shall submit the resulting conformity assessment report (CAR) to the SB within the period of three working days after receiving it.

Article 20 (2) entitles the SB to audit a QTSP at any time, or request a CAB to audit a QTSP at the expense of this QTSP.

Furthermore, where the SB requires the QTSP to remedy any failure to fulfil requirements of the eIDAS-Regulation, the SB may in the end withdraw the qualified status as explained in *Article 20 (3)*.

According to *Article 20 (4)* the ”Commission may, by means of implementing acts, establish reference numbers for standards” of (a) the accreditation of CAB and for the content of CAR and (b) auditing rules under which CAB will carry out their conformity assessment of the QTSP.

Article 21 - Initiation of a qualified trust service

As explained in *Article 21*, a TSP which intends to start providing qualified trust services shall (1) submit to the SB in charge a corresponding notification together with the CAR issued by a CAB. The SB will verify whether the TSP and the trust services under consideration fulfill the requirements of the eIDAS-Regulation and, in case of a positive evaluation ”inform the body referred to in *Article 22(3)* for the purposes of updating the the according trusted lists”. The QTSP may begin to provide

[19] Depending on the details of a security incident, this may include the affected natural or legal person, other SB, other relevant bodies, such as the competent national body for information security or the data protection authority, ENISA (see [ENISA-IR]) or even the public.

the QTS, as soon as the qualified status has been indicated in the trusted list (3). According to (4) the European Commission may, by means of implementing acts, define formats and procedures for the initialisation of the process, the CAR and related processes.

Article 22 - Trusted lists

According to *Article 22*, the EU Member States are obliged to establish, maintain and publish, in a machine readable format, trusted lists, which include information with respect to the QTSP and the qualified trust services they provide (1-2). The Member States shall inform the European Commission about the body which is responsible for the publication of the trusted list and the location where the trusted list is published (3), such that the Commission is able to publish a compound ”List of the Lists” (LotL) with this information (4)[20]. Further details with respect to the handling of the trusted lists are defined within the implementing act [(EU)2015/1505], referred to in paragraph (5), which in particular specifies details with respect to the format of the trusted list based on [ETSI-119612(v2.2.1)].

Article 23 - EU trust mark for qualified trust services

Article 23 states, that QTSP which are included in the trusted list, may use the ”EU Trust Mark” defined in the implementing act [(EU)2015/806] and depicted in Figure 1.1, provided that the website of the QTSP contains a link to the applicable trusted list.

Article 24 - Requirements for qualified trust service providers

Article 24 is of central importance with respect to the eIDAS-Regulation, as it defines general requirements for QTSP. This article contains stipulations with respect to

(1) the issuance of qualified certificates,

(2) general requirements for QTSPs,

(3) revocation of certificates and

(4) provision of certificate status information.

In *Article 24 (1)* it is regulated that a QTSP, which is ”issuing a qualified certificate for a trust service”[21], or a third party acting on behalf of the QTSP, ”shall verify, by appropriate means and in accordance with national law, the identity and, if applicable, any specific attributes of the natural or legal person to whom the qualified certificate is issued”. For this purpose the QTSP has the following options:

(a) ”by the physical presence of the natural person or of an authorised representative of the legal person; or

(b) remotely, using electronic identification means, for which prior to the issuance of the qualified certificate, a physical presence of the natural person or of an authorised representative of the legal person was ensured and which meets the requirements set out in *Article 8* with regard to the assurance levels ’substantial’ or ’high’; or

[20] See `https://ec.europa.eu/information_society/policy/esignature/trusted-list/tl-mp.xml`

[21] While the eIDAS-Regulation here obviously only regulates the case in which a qualified certificate is issued for a *trust service* (see *Article 3 (16)*), and not for any other *signatory* (see *Article 3 (9)*) or *creator of a seal* (see *Article 3 (24)*) for example, this requirement seems to be applied in practice, at least in a similar fashion, for the issuance of any qualified certificate.

(c) by means of a certificate of a qualified electronic signature or of a qualified electronic seal issued in compliance with point (a) or (b); or

(d) by using other identification methods recognised at national level which provide equivalent assurance in terms of reliability to physical presence. The equivalent assurance shall be confirmed by a conformity assessment body."

According to *Article 24 (2)* a "qualified trust service provider providing qualified trust services shall:

(a) inform the supervisory body of any change in the provision of its qualified trust services and an intention to cease those activities;

(b) employ staff and, if applicable, subcontractors who possess the necessary expertise, reliability, experience, and qualifications and who have received appropriate training regarding security and personal data protection rules and shall apply administrative and management procedures which correspond to European or international standards;

(c) with regard to the risk of liability for damages in accordance with *Article 13*, maintain sufficient financial resources and/or obtain appropriate liability insurance, in accordance with national law;

(d) before entering into a contractual relationship, inform, in a clear and comprehensive manner, any person seeking to use a qualified trust service of the precise terms and conditions regarding the use of that service, including any limitations on its use;

(e) use trustworthy systems and products that are protected against modification and ensure the technical security and reliability of the processes supported by them;

(f) use trustworthy systems to store data provided to it, in a verifiable form so that:

    (i) they are publicly available for retrieval only where the consent of the person to whom the data relates has been obtained,

    (ii) only authorised persons can make entries and changes to the stored data,

    (iii) the data can be checked for authenticity;

(g) take appropriate measures against forgery and theft of data;

(h) record and keep accessible for an appropriate period of time, including after the activities of the qualified trust service provider have ceased, all relevant information concerning data issued and received by the qualified trust service provider, in particular, for the purpose of providing evidence in legal proceedings and for the purpose of ensuring continuity of the service. Such recording may be done electronically;

(i) have an up-to-date termination plan to ensure continuity of service in accordance with provisions verified by the supervisory body under point (i) of *Article 17 (4)*;

(j) ensure lawful processing of personal data in accordance with Directive [95/46/EC];

(k) in case of qualified trust service providers issuing qualified certificates, establish and keep updated a certificate database.

Article 24 (3) stipulates that "if a qualified trust service provider issuing qualified certificates decides[22] to revoke a certificate, it shall register such revocation in its certificate database and publish

[22] Note, that the eIDAS-Regulation does not impose a strict obligation for QTSPs to support revocation of qualified certificates, which could in theory lead to a system based on qualified certificates, which would not even reach assurance level low (cf. [(EU)2015/1502, Annex, Section 2.2.3]).

the revocation status of the certificate in a timely manner, and in any event within 24 hours after the receipt of the request. The revocation shall become effective immediately upon its publication."

According to *Article (4)* "qualified trust service providers issuing qualified certificates shall provide to any relying party information on the validity or revocation status of qualified certificates issued by them. This information shall be made available at least on a per certificate basis at any time and beyond[23] the validity period of the certificate in an automated manner that is reliable, free of charge and efficient."

As defined in *Article 24 (5)* the Commission may, by means of implementing acts, establish reference numbers of standards for trustworthy systems and products, which comply with the requirements under points (e) and (f) of *Article 24 (2).* "Compliance with the requirements laid down in this Article shall be presumed where trustworthy systems and products meet those standards."

Summary of the eIDAS Trust System

The "eIDAS Trust System" defined in *Article 17 through 24* of the eIDAS-Regulation is summarised in Figure 1.3.

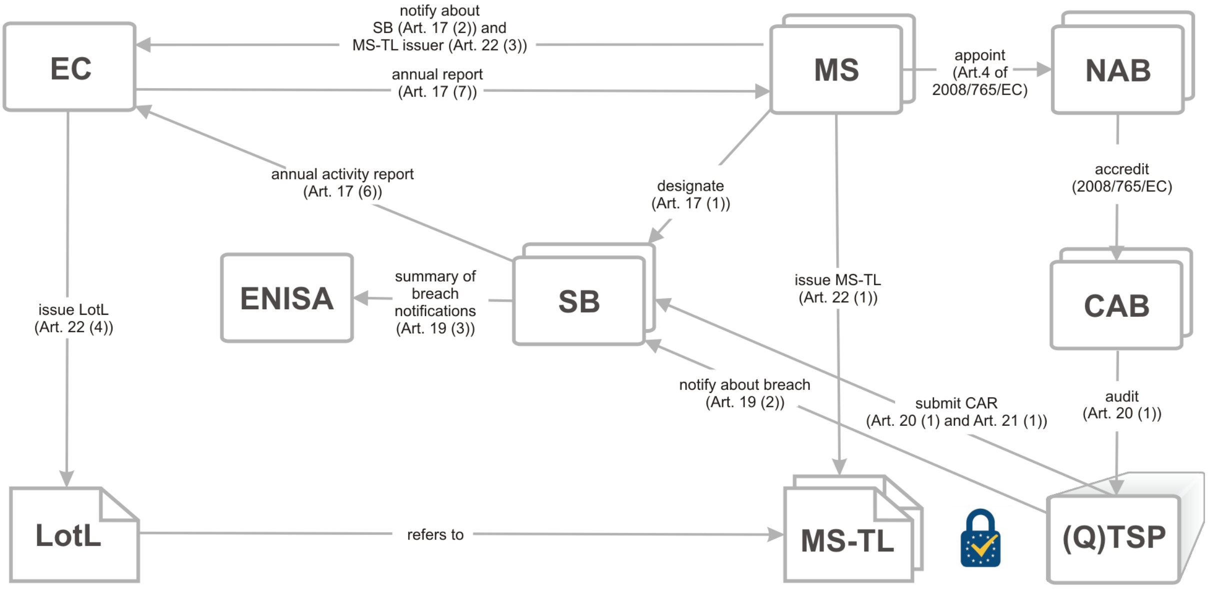

EC
notify about SB (Art. 17 (2)) and MS-TL issuer (Art. 22 (3))
annual report (Art. 17 (7))
MS
appoint (Art.4 of 2008/765/EC)
NAB
annual activity report (Art. 17 (6))
designate (Art. 17 (1))
accredit (2008/765/EC)
ENISA
summary of breach notifications (Art. 19 (3))
SB
issue MS-TL (Art. 22 (1))
CAB
issue LotL (Art. 22 (4))
notify about breach (Art. 19 (2))
submit CAR (Art. 20 (1) and Art. 21 (1))
audit (Art. 20 (1))
LotL
refers to
MS-TL
(Q)TSP

Figure 1.3: eIDAS Trust System

As depicted in Figure 1.3, the "eIDAS Trust System" in particular comprises the European Commission (EC), the EU Member States (MS), the Supervisory Bodies (SB), the European Union Agency for Network and Information Security (ENISA), the Conformity Assessment Bodies (CAB) according to [(EU)910/2014, Article 3 (18)], which are accredited by the national Accreditation Body (NAB) according to [(EC)765/2008] for performing eIDAS-specific conformity assessments and last but not least the (Qualified) Trust Service Providers ((Q)TSP), which provide one or more trust services.

The different organisations interact with each other in order to maintain trust-related aspects of the eIDAS ecosystem and finally provide a sequence of XML-based lists, which provide information with respect to the trustworthiness of the eIDAS related trust services and the certificates and keys used within, whereas the technical format for the "Trusted Lists" is defined in [(EU)2015/1505], which in turn refers to [ETSI-119612(v2.2.1)].

[23] As the eIDAS-Regulation does not clearly specify what "beyond" precisely means here, it is likely that there will be differences with respect to the certification practices of the QTSP. [ETSI-319411-2(v2.4.1), Section 6.3.10, Note 2] states, that the obligation to support OCSP vanishes with the expiry of the certificate, and according to [ETSI-319411-2(v2.4.1), Section 6.3.10, Note 3] "there are plans for further " activities for handling revocation status beyond the validity period of the certificate."

According to [(EU)910/2014, Article 22 (3)] MS notify the EC ”on the body responsible for establishing, maintaining and publishing national trusted lists, and details of where such lists are published, the certificates used to sign or seal the trusted lists and any changes thereto.” According to [(EU)910/2014, Article 22 (4)] ((EU)910/2014) the EC uses this information to publish the ”List of the List” (LotL), which in turn points to the Trust Lists of the individual Member States (MS-TL).

In order to be listed in the MS-TL and thus be allowed to provide a qualified trust service, a TSP needs to engage a CAB which audits the TSP's trust service and creates a Conformity Assessment Report (CAR), which is submitted to the SB in charge with a corresponding notice according to [(EU)910/2014, Article 21 (1)]. The SB verifies the CAR and includes the information related to the Trust Service under consideration in the corresponding MS-TL and ”grant[s] qualified status to the trust service provider and the trust services it provides” [(EU)910/2014, Article 21 (2)] .

1.1.4.4 Section 4 - Electronic Signatures - Articles 25-34

The fourth section of this chapter is dedicated to electronic signatures, which have been subject of the electronic signature directive [1999/93/EC]. This section contains stipulations with respect to

- legal effects of electronic signatures (Article 25),
- requirements for advanced electronic signatures (AdES) (*Article 26*),
- electronic signatures in public services (*Article 27*),
- qualified certificates for electronic signatures (*Article 28*),
- requirements (*Article 29*), certification (*Article 30*) and listing (*Article 31*) of qualified electronic signature creation devices
- requirements for the validation of qualified electronic signatures (*Article 32*) and related qualified validation services (*Article 33*), and
- qualified preservation service (*Article 34*)

Article 25 - Legal effects of electronic signatures

Article 25 stipulates that

(1) ”An electronic signature shall not be denied legal effect and admissibility as evidence in legal proceedings solely on the grounds that it is in an electronic form or that it does not meet the requirements for qualified electronic signatures.

(2) A qualified electronic signature shall have the equivalent legal effect of a handwritten signature.

(3) A qualified electronic signature based on a qualified certificate issued in one Member State shall be recognised as a qualified electronic signature in all other Member States.”

While the regulations in paragraphs (1) and (2) correspond with [1999/93/EC, Article 5], the third paragraph addresses an important additional aspect, which was not clearly regulated on a European level before the eIDAS Regulation came into effect.

Article 26 - Requirements for advanced electronic signatures

In *Article 26* it is defined that ”an advanced electronic signature shall meet the following requirements:

(a) it is uniquely linked to the signatory;

(b) it is capable of identifying the signatory;

(c) it is created using electronic signature creation data that the signatory can, with a high level of confidence, use under his sole control; and

(d) it is linked to the data signed therewith in such a way that any subsequent change in the data is detectable.”

These requirements are almost[24] literally copied from [1999/93/EC, Article 2 Nr. 2].

Article 27 - Electronic signatures in public services

Article 27 introduces an important new obligation for public services in Europe, which aims at fostering the practical interoperability of electronic signatures in Europe and beyond. This regulation forces public sector bodies, which require an advanced electronic signature or an advanced electronic signature based on a qualified certificate for electronic signatures or a qualified electronic signature[25] to use an online service, to at least recognise the formats or methods defined in the implementing act [(EU)2015/1506] referred to in *Article 27 (5)*.

Furthermore, *Article 27 (3)* stipulates, that ”Member States shall not request for cross-border use in an online service offered by a public sector body an electronic signature at a higher security level than the qualified electronic signature” and the European Commission ”may, by means of implementing acts, establish

- reference numbers of standards for advanced electronic signatures” (*Article 27 (4)* and
- reference formats of advanced electronic signatures or reference methods where alternative formats are used (*Article 27 (5)*).

The implementing act according to *Article 27 (5)* is [(EU)2015/1506] and its *Article 1* clarifies that the recognition comprises XML, CMS or PDF advanced electronic signatures at conformance level B, T or LT level[26] (see Chapter 4.3) or using an associated signature container according to the following standards (see [(EU)2015/1506, Annex]):

- XAdES Baseline Profile [ETSI-103171(v2.1.1)]
- CAdES Baseline Profile [ETSI-103173(v2.2.1)]
- PAdES Baseline Profile [ETSI-103172(v2.2.2)]
- Associated Signature Container Baseline Profile [ETSI-103174(v2.2.1)]

[24] There are minor differences in (c) (”maintain” vs. ”use”), which are due to the objective of the eIDAS-Regulation to allow ”remote electronic signatures” as mentioned in recital (52) of [(EU)910/2014].

[25] The advanced electronic signature, or in a similar manner a seal (see *Article 37*), may (see *Article 27 (2)*) or may not (see *Article 27 (1)*) be based on a qualified certificate.

[26] It is important to note, that the LTA level, which is part of clause 9 of the different AdES baseline profiles, has been excluded because of the ongoing standardisation work, which has for example in the meantime produced first AdES-formats (see [ETSI-119122-3(v1.1.1)]) based on evidence records according to [RFC4998].

According to [(EU)2015/1506, Article 2 (1)] Member States shall recognise other formats of electronic signatures, ”provided that the Member State where the trust service provider used by the signatory is established offers other Member States signature validation possibilities, suitable, where possible, for automated processing.” In this case the ”signature validation possibilities shall (see [(EU)2015/1506, Article 2 (2)])

(a) allow other Member States to validate the received electronic signatures online, free of charge and in a way that is understandable for non-native speakers;

(b) be indicated in the signed document, in the electronic signature or in the electronic document container; and

(c) confirm the validity of an advanced electronic signature provided that:

   (1) the certificate that supports the advanced electronic signature was valid at the time of signing, and when the advanced electronic signature is supported by a qualified certificate, the qualified certificate that supports the advanced electronic signature was, at the time of signing, a qualified certificate for electronic signature complying with Annex I of Regulation (EU) No 910/2014 and that it was issued by a qualified trust service provider;

   (2) the signature validation data corresponds to the data provided to the relying party;

   (3) the unique set of data representing the signatory is correctly provided to the relying party;

   (4) the use of any pseudonym is clearly indicated to the relying party if a pseudonym was used at the time of signing;

   (5) when the advanced electronic signature is created by a qualified electronic signature creation device, the use of any such device is clearly indicated to the relying party;

   (6) the integrity of the signed data has not been compromised;

   (7) the requirements provided for in *Article 26* of Regulation (EU) No 910/2014 were met at the time of signing;

   (8) the system used for validating the advanced electronic signature provides to the relying party the correct result of the validation process and allows the relying party to detect any security relevant issues.”

Note, that these requirements almost literally mirror the requirements for the validation of qualified electronic signatures according to *Article 32*.

Validation services offered by Member States are known to exist in Austria[27], Estonia[28] and Italy[29] for example, but it is not entirely clear whether these services fulfill the requirements of [(EU)910/2014, Article 32] or [(EU)2015/1506, Article 2].

Article 28 - Qualified certificates for electronic signatures

Article 28 (1) and (2) of the eIDAS-Regulation are dedicated to qualified certificates for electronic signatures and stipulates that they shall meet the requirements laid down in Annex I (1.1.8) and may only include non-mandatory additional attributes, which do not affect the interoperability and recognition of qualified certificates (*Article 28 (3)*). *Article 28 (4)* makes it clear that ”if a qualified certificate for electronic signatures has been revoked after initial activation, it shall lose its validity from the moment of its revocation, and its status shall not in any circumstances be reverted.”

[27] See `https://www.signatur.rtr.at/de/vd/Pruefung.html`.

[28] See `https://siva-arendus.eesti.ee/`.

[29] See `http://dss.agid.gov.it/validation`.

Member States may - within the limits of *Article 28 (5)* - ”lay down national rules on temporary suspension of a qualified certificate for electronic signature” and the European Commission may, by means of implementing acts, establish reference numbers of standards (see table 1.1) for qualified certificates for electronic signature.

Article 29 - Requirements for qualified signature creation devices

With respect to qualified electronic signature creation devices (QSCD) *Article 29 (1)* refers to Annex II and *Article 29 (2)* empowers the European Commission to issue implementing acts, which establish reference numbers of standards for qualified electronic signature creation devices. As such an implementing act does not exist yet, there is the closely related implementing act [(EU)2016/650] persuant to *Articles 30 (3) and 39 (2)* of the eIDAS-Regulation, which addresses standards for the security assessment of QSCD.

Article 30 - Certification of qualified signature creation devices

Article 30 (1) stipulates that the conformity of QSCD with the requirements laid down in Annex II shall be certified by appropriate public or private bodies, which are designated by Member States (*Article (1)*) and notified to the European Commission, which will make that information available to Member States (*Article (2)*). According to *Article 30 (3)* the Commission shall, by means of implementing acts, establish standards for the security assessment of QSCD, which have to be applied for new evaluation processes, as soon as they are available and suitable for the designated purpose. According to *Article 30 (4)* the ”Commission shall be empowered to adopt delegated acts in accordance with *Article 47* concerning the establishment of specific criteria to be met by the designated bodies referred to in paragraph 1 of this Article.”

The implementing act [(EU)2016/650][30] persuant to *Article 30 (3)* differentiates in *Article 1* between QSCD for

1. *local* signature creation in a user-managed environment and
2. *remote* signature creation, where a qualified trust service provider manages the electronic signature creation data on behalf of a signatory.

While [(EU)2016/650, Article 1.1] for the local signature creation refers to the Annex, which in turn refers to

- ISO/IEC 15408 – Information technology – Security techniques – Evaluation criteria for IT security (Part 1-3) [ISO15408-1, ISO15408-2, ISO15408-3],
- ISO/IEC 18045:2008: Information technology – Security techniques – Methodology for IT security evaluation [ISO18045] and
- EN 419 211 – Protection profiles for secure signature creation device (Part 1-6, as appropriate) [EN419211-1, EN419211-2, EN419211-3, EN419211-4, EN419211-5, EN419211-6],

[(EU)2016/650, Article 1.2] clarifies that for the remote signing case one may, until further notice, pursue the less formal peer review process according to *Article 30 (3) (b)* of the eIDAS-Regulation.

[30] Note, that [(EU)2016/650] also covers the case of seal creation devices according to *Article 39*.

Article 31 - Publication of a list of certified qualified electronic signature creation devices

According to *Article 31 (1)*, ”Member States shall notify to the Commission without undue delay and no later than one month after the certification is concluded, information on qualified electronic signature creation devices that have been certified by the bodies referred to in *Article 30 (1)*. They shall also notify to the Commission, without undue delay and no later than one month after the certification is cancelled, information on electronic signature creation devices that are no longer certified.”

”On the basis of the information received, the Commission shall establish, publish and maintain a list of certified qualified electronic signature creation devices.” (see *Article 31 (2)*). This list has been published by the European Commission on [EU-QSCD-2021]. According to *Article 31 (3)* ”The Commission may, by means of implementing acts, define formats and procedures applicable for the purpose of paragraph 1.” Such implementation acts do not yet exist and their developments are not yet known.

Article 32 - Requirements for the validation of qualified electronic signatures

According to *Article 32 (1)* ”The process for the validation of a qualified electronic signature shall confirm the validity of a qualified electronic signature provided that:

(a) the certificate that supports the signature was, at the time of signing, a qualified certificate for electronic signature complying with Annex I;

(b) the qualified certificate was issued by a qualified trust service provider and was valid at the time of signing;

(c) the signature validation data corresponds to the data provided to the relying party;

(d) the unique set of data representing the signatory in the certificate is correctly provided to the relying party;

(e) the use of any pseudonym is clearly indicated to the relying party if a pseudonym was used at the time of signing;

(f) the electronic signature was created by a qualified electronic signature creation device;

(g) the integrity of the signed data has not been compromised;

(h) the requirements provided for in *Article 26* were met at the time of signing.”

As stipulated in *Article 32 (2)* ”the system used for validating the qualified electronic signature shall provide to the relying party the correct result of the validation process and shall allow the relying party to detect any security relevant issues.”

Finally, *Article 32 (3)* empowers the European Commission, by means of implementing acts, to establish reference numbers of standards for the validation of qualified electronic signatures.

Article 33 - Qualified validation service for qualified electronic signatures

As defined in *Article 33 (1)* ”a qualified validation service for qualified electronic signatures may only be provided by a qualified trust service provider who:

(a) provides validation in compliance with *Article 32(1)*; and

(b) allows relying parties to receive the result of the validation process in an automated manner, which is reliable, efficient and bears the advanced electronic signature or advanced electronic seal of the provider of the qualified validation service."

According to *Article 33 (2)*, the European "Commission may, by means of implementing acts, establish reference numbers of standards for qualified validation service" referred to in *Article 33 (1)*.

Article 34 - Qualified preservation service for qualified electronic signatures

According to *Article 34 (1)* "a qualified preservation service for qualified electronic signatures may only be provided by a qualified trust service provider that uses procedures and technologies capable of extending the trustworthiness of the qualified electronic signature beyond the technological validity period."

As defined in *Article 34 (2)*, the European Commission may, by means of implementing acts, establish reference numbers of standards for qualified preservation service compliant to in *Article 34 (1)*.

1.1.4.5 Section 5 - Electronic Seals - Articles 35-40

Article 35 - Legal effects of electronic seals

The legal effects of electronic seals are defined in *Article 35*. The main difference is that "a qualified electronic signature shall have the equivalent legal effect of a handwritten signature" (*Article 25 (2)*), while "a qualified electronic seal shall enjoy the presumption of integrity of the data and of correctness of the origin of that data to which the qualified electronic seal is linked" (*Article 35 (2)*). According to *Article 35 (3)* "A qualified electronic seal based on a qualified certificate issued in one Member State shall be recognised as a qualified electronic seal in all other Member States."

Article 36 - Requirements for advanced electronic seals

Article 36 defines for advanced electronic seals "the following requirements:

(a) it is uniquely linked to the creator of the seal;

(b) it is capable of identifying the creator of the seal;

(c) it it is created using electronic seal creation data that the creator of the seal can, with a high level of confidence under its control, use for electronic seal creation; and

(d) it is linked to the data to which it relates in such a way that any subsequent change in the data is detectable."

whereas the differences to the corresponding specification for advanced electronic signatures in *Article 26* is that

- the term "signatory" is replaced by "creator of the seal" and
- an advanced electronic signature requires that "it is created using electronic signature creation data that the signatory can, with a high level of confidence, use under his *sole* control;" (*Article 26 (c)*), while for an advanced electronic seal it is only required that "it is created using electronic seal creation data that the creator of the seal can, with a high level of confidence under its control, use for electronic seal creation" (*Article 36 (c)*.

Article 37 - Electronic seals in public services

Article 37 is almost[31] a literal copy of *Article 27*, in which ”signature” is replaced by ”seal” and, as the implementing act [(EU)2015/1506] equally refers to electronic signatures as well as electronic seals, all stipulations from *Article 27* apply in an analogous manner also to electronic seals.

Article 38 - Qualified certificates for electronic seals

Article 38 is again a literal copy of *Article 28*, which addresses the topic of electronic seals instead of electronic signatures and hence refers to Annex III, instead of Annex I.

The only material difference between Annex III and Annex I is that a qualified certificate for electronic signatures shall contain ”at least the name of the signatory, or a pseudonym; if a pseudonym is used, it shall be clearly indicated” (Annex I (c)), while qualified certificates for electronic seals shall contain ”at least the name of the creator of the seal and, where applicable, registration number as stated in the official records” (Annex III (c)).

Article 39 - Qualified electronic seal creation devices

Article 39 states that Article 29, Article 30 and Article 31 shall apply mutatis mutandis to qualified electronic seal creation devices.

Article 40 - Validation and preservation of qualified electronic seals

In a similar manner *Article 40* states that Article 32, Article 33 and Article 34 shall apply mutatis mutandis to the validation and preservation of qualified electronic seals.

1.1.4.6 Section 6 - Electronic Time Stamps - Articles 41-42

Article 41 - Legal effects of electronic time stamps

The legal effects of electronic time stamps are defined in *Article 41* corresponding to the legal effects for electronic signatures in Article 25 and electronic seals in Article 35.

The specific aspect for electronic time stamps is, that ”a qualified electronic time stamp shall enjoy the presumption of the accuracy of the date and the time it indicates and the integrity of the data to which the date and time are bound.”

According to *Article 41 (3)* ”A qualified electronic time stamp issued in one Member State shall be recognised as a qualified electronic time stamp in all Member States.”

Article 42 - Requirements for qualified electronic time stamps

Article 42 (1) states that ”a qualified electronic time stamp shall meet the following requirements:

(a) it binds the date and time to data in such a manner as to reasonably preclude the possibility of the data being changed undetectably;

(b) it is based on an accurate time source linked to Coordinated Universal Time; and

(c) it is signed using an advanced electronic signature or sealed with an advanced electronic seal of the qualified trust service provider, or by some equivalent method.”

[31] A necessary difference is, that *Article 27 (4)* refers to *Article 26*, while *Article 37 (4)* refers to *Article 36*. One editorial difference is, that *Article 27 (5)* mentions ”Union legal acts”, while *Article 37 (5)* refers to ”legal acts of the Union”.

According to *Article 42 (2)*, ”the Commission may, by means of implementing acts, establish reference numbers of standards for the binding of date and time to data and for accurate time sources. Compliance with the requirements laid down in paragraph 1 shall be presumed where the binding of date and time to data and the accurate time source meets those standards.”

1.1.4.7 Section 7 - Electronic Registered Delivery Services - Articles 43-44

The seventh section of the eIDAS-Regulation addresses electronic registered delivery services.

Article 43 - Legal effects of an electronic registered delivery service

The legal effects of an electronic registered delivery service are defined in *Article 43*. Corresponding to the stipulations for electronic signatures in Article 25, electronic seals in Article 35 and electronic time stamps in Article 41 *Article 43 (1)* states, that ”data sent and received using an electronic registered delivery service shall not be denied legal effect and admissibility as evidence in legal proceedings solely on the grounds that it is in an electronic form or that it does not meet the requirements of the qualified electronic registered delivery service”.

Article 43 (2) stipulates that ”Data sent and received using a qualified electronic registered delivery service shall enjoy the presumption

- of the integrity of the data,
- the sending of that data by the identified sender,
- its receipt by the identified addressee and
- the accuracy of the date and time of sending and receipt indicated by the qualified electronic registered delivery service.”

Article 44 - Requirements for qualified electronic delivery services

According to *Article 44 (1)*, ”qualified electronic registered delivery services shall meet the following requirements:

(a) they are provided by one or more qualified trust service provider(s);

(b) they ensure with a high level of confidence the identification of the sender;

(c) they ensure the identification of the addressee before the delivery of the data;

(d) the sending and receiving of data is secured by an advanced electronic signature or an advanced electronic seal of a qualified trust service provider in such a manner as to preclude the possibility of the data being changed undetectably;

(e) any change of the data needed for the purpose of sending or receiving the data is clearly indicated to the sender and addressee of the data;

(f) the date and time of sending, receiving and any change of data are indicated by a qualified electronic time stamp.

In the event of the data being transferred between two or more qualified trust service providers, the requirements in points (a) to (f) shall apply to all the qualified trust service providers.”

Under *Article 43 (2)*, ”the Commission may, by means of implementing acts, establish reference numbers of standards for processes for sending and receiving data. Compliance with the requirements laid down in paragraph 1 shall be presumed where the process for sending and receiving data meets those standards.”

1.1.4.8 Section 8 - Website Authentication - Article 45

Article 45 - Requirements for qualified certificates for website authentication

Article 45 (1) refers to Annex IV, which defines the requirements for qualified certificates for website authentication and *Article 45 (2)* states that ”the Commission may, by means of implementing acts, establish reference numbers of standards for qualified certificates for website authentication. Compliance with the requirements laid down in Annex IV shall be presumed where a qualified certificate for website authentication meets those standards.”

1.1.5 Chapter IV - Electronic Documents - Article 46

Article 46 - Legal effects of electronic documents

Article 46 states that ”an electronic document shall not be denied legal effect and admissibility as evidence in legal proceedings solely on the grounds that it is in electronic form”, which is a generalisation of the corresponding regulation in *Article 5 (2)* of the by now repealed [1999/93/EC], which only addressed electronic signatures.

1.1.6 Chapter V - Delegations of Power and Implementing Provisions - Articles 47-48

Article 47 - Exercise of the delegation

Article 47 empowers the European Commission to adopt delegated acts according to *Article 30 (4)* in consultation with the European Parliament and the Council, which establish specific criteria to be met by the designated bodies, which certify the conformity of qualified electronic signature creation devices.

Article 48 - Committee procedure

The eIDAS-regulation contains many articles, which empower the European Commission to adopt implementing acts, which have been developed with the assistance of a committee, which is composed of representatives of the Member States (cf. *Article 3 (2)* of [(EU)182/2011]) and which in particular provides opinions to draft implementing acts (cf. *Article 5* of [(EU)182/2011]).

The currently existing implementing acts are listed in Table 1.1. The topics for which the European Commission may adopt further implementing acts are listed in the following table.

1.1.7 Chapter VI - Final Provisions - Articles 49-52

Article 49 - Review

Article 49 states that ”the Commission shall review the application of this Regulation and shall report to the European Parliament and to the Council no later than *1 July 2020*”, whereas special attention will be paid to the newly introduced aspects related to the mutual recognition and the cross-border use of eID (*Article 6 and 7 (f)*), preservation services (*Article 34*), registered electronic delivery services (*Article 44*) and qualified certificates for website authentication (*Article 45*). ”The report [...] shall be accompanied, where appropriate, by legislative proposals” and there will be similar reports every four years.

Article	Subject
17 (8)	Annual report of Supervisory Bodies
19 (4)	Security measures and breach notification
20 (4)	Accreditation of Conformity Assessment Bodies and auditing rules
21 (4)	Formats and procedures for initiation of qualified trust services
24 (5)	Requirements for Trust Service Providers
28 (6)	Standards for qualified electronic signatures
29 (2)	Standards for qualified electronic signature creation devices (QSCD)
31 (3)	Formats and procedures for notification of certified QSCDs
32 (3)	Standards for the validation of qualified electronic signatures
33 (2)	Standards for qualified validation services
34 (2)	Standards for qualified preservation services
38 (6)	Standards for qualified certificates for electronic seals
42 (2)	Standards for qualified electronic time stamps
44 (2)	Standards for qualified electronic delivery services
45 (2)	Standards for qualified certificates for website authentication

Table 1.3: Options for additional implementing acts

Article 50 - Repeal

Article 50 stipulates that the Signature Directive [1999/93/EC] is repealed with effect from *1 July 2016* and references to [1999/93/EC] shall be construed as references to the eIDAS-Regulation [(EU)910/2014].

Article 51 - Transitional measures

Article 51 defines details with respect to the transition from the previously applicable Signature Directive [1999/93/EC] to the eIDAS-Regulation [(EU)910/2014], whereas the following rules apply:

(1) Certified secure signature creation devices (SSCD) in the sense of [1999/93/EC] shall be considered as qualified electronic signature creation devices according to [(EU)910/2014].

(2) Qualified certificates issued to natural persons[32] under Directive [1999/93/EC] shall be considered as qualified certificates for electronic signatures under [(EU)910/2014] until they expire.

(3) Certification service providers issuing qualified certificates under [1999/93/EC] must send a conformity assessment report to its supervision body until *1 July 2017* and shall be considered as qualified trust service providers under [(EU)910/2014]

(4) or otherwise shall not be considered as qualified trust service providers after this date.

Article 52 - Entry into force

[32] It should be noted that there have been Member States in which it was possible to issue qualified certificates to legal persons before the advent of the eIDAS-Regulation. Such certificates are not addressed by the stipulation in *Article 51 (2)*.

Article 52 (1) stipulates that the eIDAS-Regulation enters into force on *17 September 2014*, which is twenty days after the publication in the Official Journal of the European Union on *28 August 2014*. According to *Article 52 (2)*, the eIDAS-Regulation is applicable from *1 July 2016* with the following three exceptions:

(a) Almost all[33] articles, which are related to the various implementing acts according to *Article 48*, which already have been created (see Table 1.1) or which may be created in the future (see Table 1.3), and the potential delegated acts according to *Article 30 (4)* and *Article 47* are applicable since *17 September 2014*,

(b) the general eID-related *Articles 7, 8 (1) and (2), 9, 10, 11 and 12 (1)* "apply from the date of application of the implementing acts referred to in *Articles 8(3)*[34] and *12(8)*[35]", which is *29 September 2015* and

(c) the mutual recognition of notified eID-schemes according to *Article 6* shall apply three years later, which is *29 September 2018*.

Article 52 (3) in addition states that, if a notified eID scheme is included in the list published by the Commission pursuant to *Article 9* before this date, the recognition of the eID means related to that scheme pursuant to *Article 6* shall take place no later than 12 months after the publication of the list, but not before *29 September 2018*.

Article 52 (4) finally states that after *29 September 2015*, eID means related to a notified and listed eID scheme may already be recognized by other Member States, whereas the involved Member States shall inform the European Commission, which in turn shall make this information public. At the time of writing (*2 September 2017*) the European Commission has not yet published a list of notified eID schemes, but Germany has initiated the notification process for its eID on *20 February 2017* (see [BSI-eIDAS-nPA]).

1.1.8 Annex I - Requirements for Qualified Certificates for Electronic Signatures

"Qualified certificates for electronic signatures shall contain:

(a) an indication, at least in a form suitable for automated processing, that the certificate has been issued as a qualified certificate for electronic signature;

(b) a set of data unambiguously representing the qualified trust service provider issuing the qualified certificates including at least, the Member State in which that provider is established and:

  - for a legal person: the name and, where applicable, registration number as stated in the official records,
  - for a natural person: the person's name;

(c) at least the name of the signatory, or a pseudonym; if a pseudonym is used, it shall be clearly indicated;

(d) electronic signature validation data that corresponds to the electronic signature creation data;

(e) details of the beginning and end of the certificate's period of validity;

(f) the certificate identity code, which must be unique for the qualified trust service provider;

[33] Article 39 (2), which forms the basis for the implementing act [(EU)2016/650], seems to constitute an exception to this rule.

[34] See [(EU)2015/1502].

[35] See [(EU)2015/1502].

(g) the advanced electronic signature or advanced electronic seal of the issuing qualified trust service provider;

(h) the location where the certificate supporting the advanced electronic signature or advanced electronic seal referred to in point (g) is available free of charge;

(i) the location of the services that can be used to enquire about the validity status of the qualified certificate;

(j) where the electronic signature creation data related to the electronic signature validation data is located in a qualified electronic signature creation device, an appropriate indication of this, at least in a form suitable for automated processing."

Certificate profiles have been standardised by ETSI in EN 319 412. For the qualified certificates for electronic signatures addressed in Annex I the parts [ETSI-319412-1(v1.4.4)], [ETSI-319412-2(v2.2.1)] and [ETSI-319412-5(v2.3.1)] are relevant. A mapping between the defined certificate profile and Annex I is provided in Annex A.1 of [ETSI-319412-5(v2.3.1)].

1.1.9 Annex II - Requirements for Qualified Electronic Signature Creation Devices

1. "Qualified electronic signature creation devices shall ensure, by appropriate technical and procedural means, that at least:
   (a) the confidentiality of the electronic signature creation data used for electronic signature creation is reasonably assured;
   (b) the electronic signature creation data used for electronic signature creation can practically occur only once;
   (c) the electronic signature creation data used for electronic signature creation cannot, with reasonable assurance, be derived and the electronic signature is reliably protected against forgery using currently available technology;
   (d) the electronic signature creation data used for electronic signature creation can be reliably protected by the legitimate signatory against use by others.
2. Qualified electronic signature creation devices shall not alter the data to be signed or prevent such data from being presented to the signatory prior to signing.
3. Generating or managing electronic signature creation data on behalf of the signatory may only be done by a qualified trust service provider.
4. Without prejudice to point (d) of point 1, qualified trust service providers managing electronic signature creation data on behalf of the signatory may duplicate the electronic signature creation data only for back-up purposes provided the following requirements are met:
   (a) the security of the duplicated datasets must be at the same level as for the original datasets;
   (b) the number of duplicated datasets shall not exceed the minimum needed to ensure continuity of the service."

1.1.10 Annex III - Requirements for Qualified Certificates for Electronic Seals

The differences between Annex I and Annex III are highlighted in **bold** font.

"Qualified certificates for electronic **seals** shall contain:

(a) an indication, at least in a form suitable for automated processing, that the certificate has been issued as a qualified certificate for electronic **seal**;

(b) a set of data unambiguously representing the qualified trust service provider issuing the qualified certificates including at least, the Member State in which that provider is established and:

  - for a legal person: the name and, where applicable, registration number as stated in the official records,
  - for a natural person: the person's name;

(c) at least the name of the **creator of the seal and, where applicable, registration number as stated in the official records;**

(d) electronic **seal** validation data that corresponds to the electronic **seal** creation data;

(e) details of the beginning and end of the certificate's period of validity;

(f) the certificate identity code, which must be unique for the qualified trust service provider;

(g) the advanced electronic signature or advanced electronic seal of the issuing qualified trust service provider;

(h) the location where the certificate supporting the advanced electronic signature or advanced electronic seal referred to in point (g) is available free of charge;

(i) the location of the services that can be used to enquire about the validity status of the qualified certificate;

(j) where the electronic **seal** creation data related to the electronic **seal** validation data is located in a qualified electronic **seal** creation device, an appropriate indication of this, at least in a form suitable for automated processing."

For the qualified certificates for electronic seals addressed in Annex III the parts [ETSI-319412-1(v1.4.4)], [ETSI-319412-3(v1.2.1)] and [ETSI-319412-5(v2.3.1)] of EN 319 412 are relevant. A mapping between the defined certificate profile and Annex III is provided in Annex A.2 of [ETSI-319412-5(v2.3.1)].

1.1.11 Annex IV - Requirements for Qualified Certificates for Website Authentication

The differences between Annex I and Annex IV are highlighted in **bold** font.

"Qualified certificates for **website authenticaton** shall contain:

(a) an indication, at least in a form suitable for automated processing, that the certificate has been issued as a qualified certificate for **website authenticaton**;

(b) a set of data unambiguously representing the qualified trust service provider issuing the qualified certificates including at least, the Member State in which that provider is established and:

  - for a legal person: the name and, where applicable, registration number as stated in the official records,
  - for a natural person: the person's name;

(c) for natural persons at least the name of the person to whom the certificate has been issued, or a pseudonym. If a pseudonym is used, it shall be clearly indicated; for legal persons: at least the name of the legal person to whom the certificate is issued and, where applicable, registration number as stated in the official records;

(d) elements of the address, including at least city and State, of the natural or legal person to whom the certificate is issued and, where applicable, as stated in the official records;

(e) **the domain name(s) operated by the natural or legal person to whom the certificate is issued;**

(f) details of the beginning and end of the certificate's period of validity;

(g) the certificate identity code, which must be unique for the qualified trust service provider;

(h) the advanced electronic signature or advanced electronic seal of the issuing qualified trust service provider;

(i) the location where the certificate supporting the advanced electronic signature or advanced electronic seal referred to in point (g) is available free of charge;

(j) the location of the **certificate validity status** services that can be used to enquire about the validity status of the qualified certificate."

For the qualified certificates for website authentication addressed in Annex IV all parts of EN 319 412[36] are relevant. A mapping between the defined certificate profile and Annex IV is provided in Annex A.3 of [ETSI-319412-5(v2.3.1)].

[36] [ETSI-319412-1(v1.4.4)],[ETSI-319412-2(v2.2.1)],[ETSI-319412-3(v1.2.1)],[ETSI-319412-4(v1.2.1)],[ETSI-319412-5(v2.3.1)]

2 Overview of eIDAS-Standards

2.1 Introduction

The overview of European " in the context of the eIDAS-regulation concentrates on creation and validation of electronic signatures, seals, timestamps and their preservation and trust services. The focus of this document is on the most relevant ETSI- and CEN-standardisations based on [M460] of European Commission to underpin the legal regulation of eIDAS with the aim to give an introduction and overview for users, technical experts as well as decision makers. "The scope of this mandate is to create the conditions for and achieve the interoperability of eSignature at intra-community level, by defining and providing a rationalised European eSignature standardisation framework, which must also include implementation guidelines." [M460]. For more information see [ENSIA]. This chapter should be taken as starting point for a deeper look on these subjects or a concrete project to implement electronic signatures, seals, timestamps or preservation in the business processes and infrastructure of public authorities or private companies. Beginning with a description of general standards regarding trust services on electronic signatures, seals, timestamps the different special ETSI-standards will be described in the following structure:

- Short description of trust services
- Overview on relevant standards with table. The order depends on the level of obligation of the mentioned standard. This means that, in order of importance, the formal European standards published by ETSI or the CEN have the highest level of obligation. These are followed by Technical Specification documents, Technical Reports, and finally Special Reports. Technical Specification are authoritative standards too but Technical- and Special Reports only give overview on state-of-the-art technology in the given subjects but formally, they are no obligatory standards like Specifications or EN.
- Relevant Publications of Federal Office for Information Security

2.2 Overview eIDAS ecosystem

Since September 2014, the eIDAS regulation was defined, which came fully into force in July 2016 as a European wide mandatory legal framework for trustworthy digital transactions between citizens, business and government. The eIDAS-regulation contains two parts which both affect trustworthy digital transactions in any business IT-systems: secure digital identities (identification systems) and trust services, in the context of this paper especially, of creation and validation of (qualified) electronic signatures, seals and timestamps as well as preservation services. Any notified electronic identification scheme has to be recognised and accepted by any public administration. Advanced electronic signature, seal or timestamp from any qualified trust service provider, of the European Member State has to be accepted and validated by any public administrations (of the European Member State). eIDAS is technically underpinned by corresponding European Standards of ETSI and CEN which have been developed within the scope of Mandate 460.[1].

[1] see also https://portal.etsi.org/TB-SiteMap/ESI/Trust-Service-Providers and https://portal.etsi.org/TB-SiteMap/esi/esi-activities

The standards are tied to special state-of-the-art-technologies, which achieve the technical and security requirements. Figure 2.1 illustrates the relationship between the eIDAS-regulation and related European ".

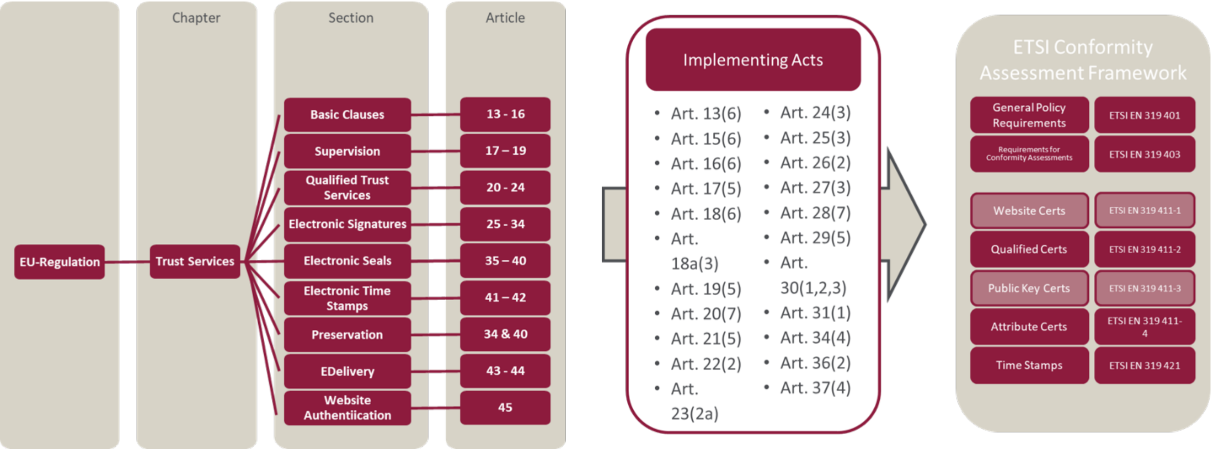


Figure 2.1: Relationship between eIDAS-regulation & ETSI-standards

eID-Services as well as trust service providers are interacting together in the eIDAS-ecosystem. Secure electronic identification is necessary to use trust services for creation of electronic signatures and seals. To build up trustworthy digital transactions the unambiguous identification of natural or legal entities interacting in the process as well as the utilisation of electronic signatures, seals and timestamp to make the transaction evident against third parties is mandatory. This means the eIDAS-ecosystem delivers a fundamental basement for a trustworthy digitisation between government, business and citizens.

Figure 2.2 shows the interaction of eID-Services and trust services in the eIDAS-ecosystem[2] in principle.

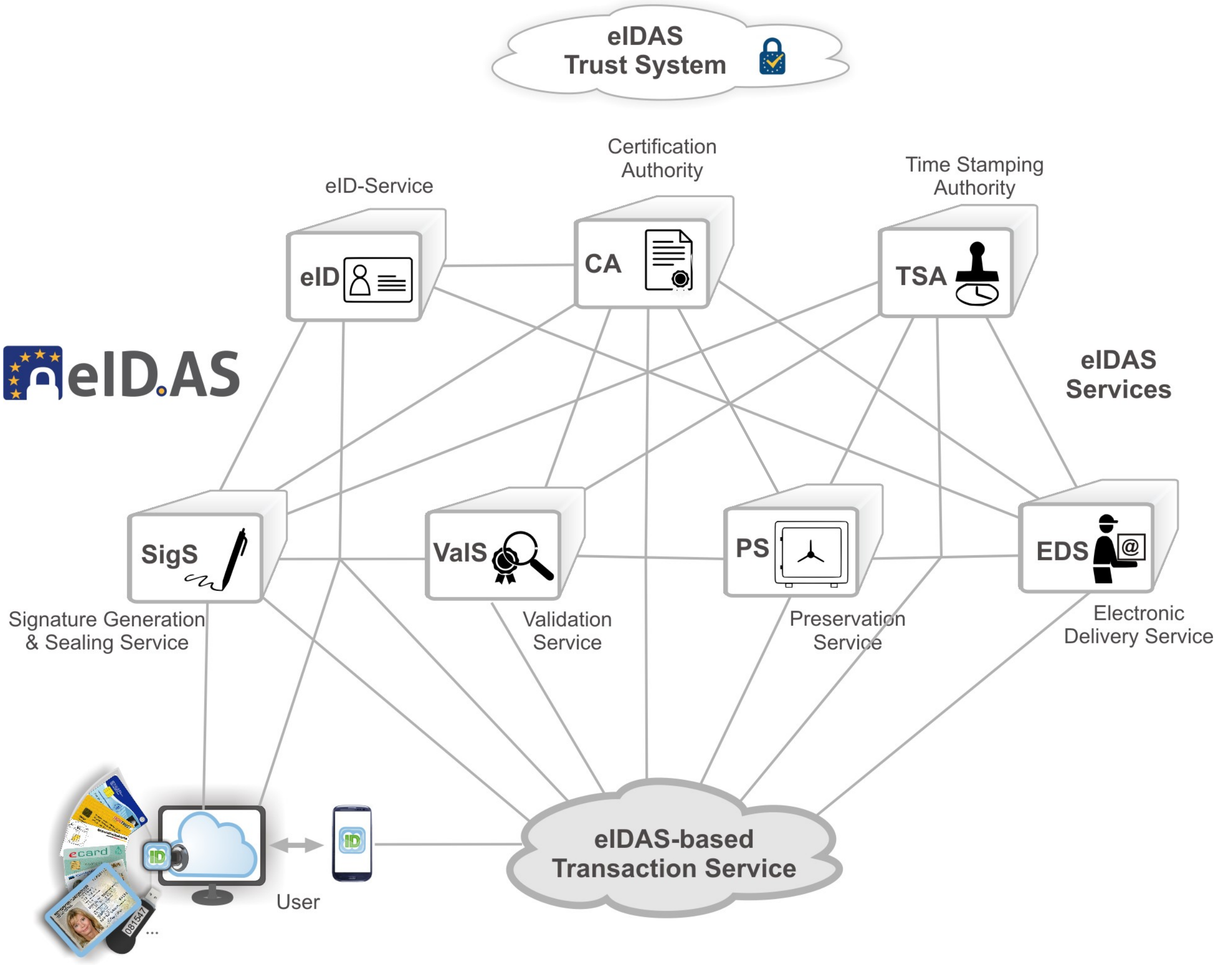


Figure 2.2: eIDAS-ecosystem

[2] see also https://blog.eid.as/de/eidas-oekosystem

2.2.1 eIDAS-Standards Framework

The legal framework of trust services according to Chapter III of the eIDAS regulation is underpinned by a framework of technical standards. Based on fundamentals e.g. definitions, guidelines and requirements on trust service status list etc. the standards define the basement for design, implementation and certification of trust services and trust service providers including the necessary technical devices such as qualified electronic signature creation devices and cryptographic suites as given in Figure 2.3.

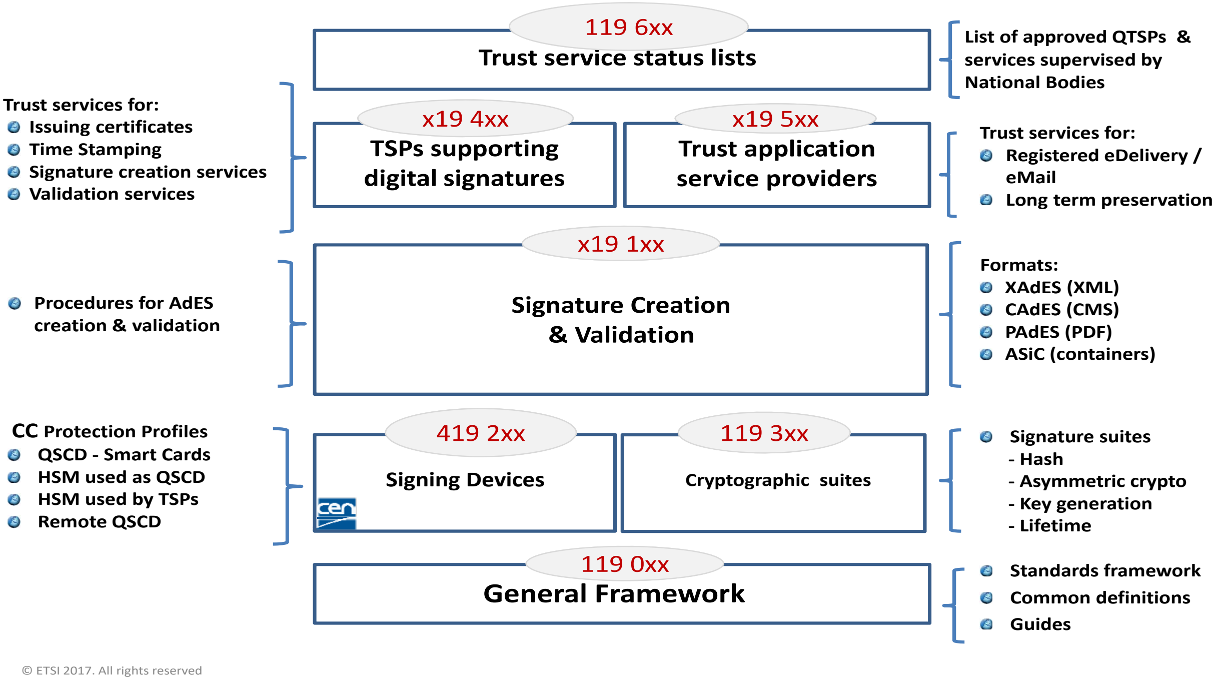
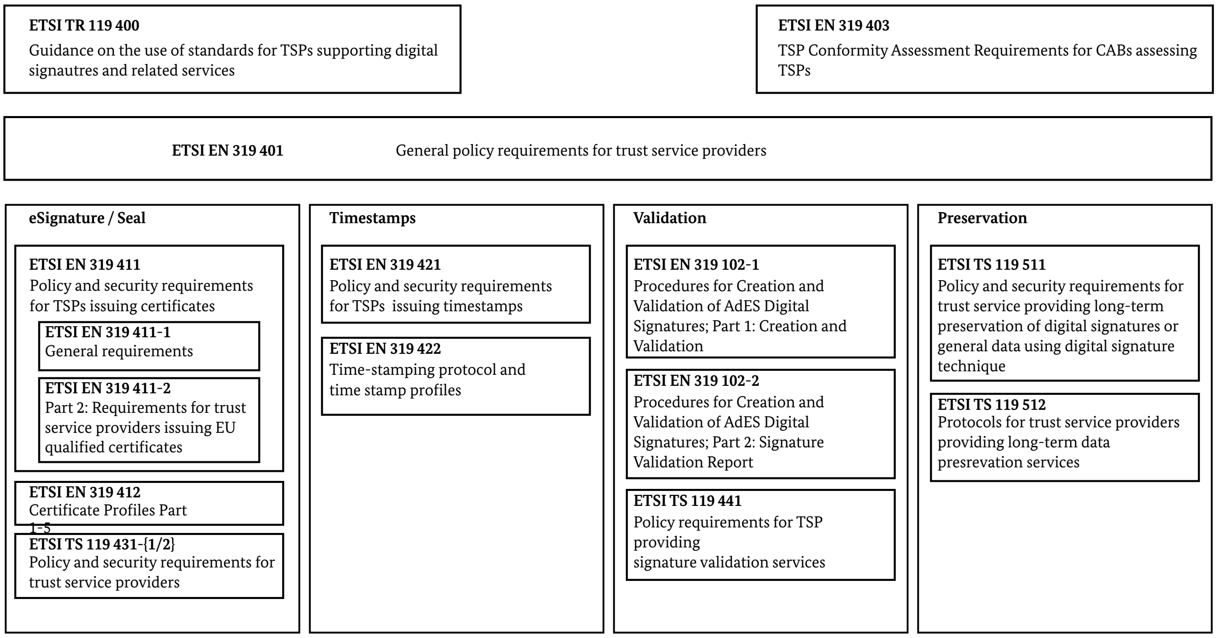


Figure 2.3: eIDAS-Standards framework

2.2.2 Relationship between eIDAS-Standards

Concerning digital signatures, seals, timestamps as well as their preservation first of all basic requirements on all (qualified) trust service providers have to be considered. These demands are trust service agnostic and relevant for all trust services. Based on this the specific standards for the different trust services are derived. Figure 2.4 illustrates this exemplary for preservation.



ETSI TR 119 400
Guidance on the use of standards for TSPs supporting digital signautres and related services

ETSI EN 319 403
TSP Conformity Assessment Requirements for CABs assessing TSPs

ETSI EN 319 401 General policy requirements for trust service providers

eSignature / Seal

ETSI EN 319 411
Policy and security requirements for TSPs issuing certificates

ETSI EN 319 411-1
General requirements

ETSI EN 319 411-2
Part 2: Requirements for trust service providers issuing EU qualified certificates

ETSI EN 319 412
Certificate Profiles Part 1-5

ETSI TS 119 431-{1/2}
Policy and security requirements for trust service providers

Timestamps

ETSI EN 319 421
Policy and security requirements for TSPs issuing timestamps

ETSI EN 319 422
Time-stamping protocol and time stamp profiles

Validation

ETSI EN 319 102-1
Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation

ETSI EN 319 102-2
Procedures for Creation and Validation of AdES Digital Signatures; Part 2: Signature Validation Report

ETSI TS 119 441
Policy requirements for TSP providing signature validation services

Preservation

ETSI TS 119 511
Policy and security requirements for trust service providing long-term preservation of digital signatures or general data using digital signature technique

ETSI TS 119 512
Protocols for trust service providers providing long-term data presrevation services

Figure 2.4: Fundamentals on relationship between eIDAS-standards

2.3 General Standards

2.3.1 Introduction

The third chapter of the eIDAS-Regulation addresses all trust services, defined in [(EU)910/2014, Article 3(16)]. This document focuses on the following trust services.

- Creation of (qualified) electronic signatures and seals
- Creation of (qualified) electronic timestamps
- Validation of (qualified) electronic signatures, seals and timestamps
- Preservation of (qualified) electronic signatures, seals and timestamps

First of all it is necessary to look on the general standards regarding these trust services. Those standards define fundamental requirements for trust service providers mainly regardless of the service they offer but also for all signature related trust services. This contains required policies, requirements on information security, business continuity of the offered service, compliance or documentation needs. Other important contents are the requirements for the Trust Service Status List and the structuring of existing standards and their relationships - especially necessary for Trust Service Providers planning certification to become Qualified Trust Service Providers, as well as for supervisory authorities (in Germany: Federal Network Agency) or Conformity Assessment Bodies. https://portal.etsi.org/TB-SiteMap/esi/esi-activities

2.3.2 Overview on essential general standards

Standard	minimal Version	Title	eIDAS (example)
ETSI EN 319 401	2.3.1	Electronic Signatures and Infrastructures (ESI); General Policy Requirements for Trust Service Providers	Art. 19, 24
ETSI EN 319 403	2.2.2	Electronic Signatures and Infrastructures (ESI); Trust Service Provider Conformity Assessment - Requirements for conformity assessment bodies assessing Trust Service Providers	Art. 20, 21, 24
ETSI TS 119 403-2	1.2.4	Electronic Signatures and Infrastructures (ESI); Trust Service Provider Conformity Assessment; Part 2: Additional requirements for Conformity Assessment Bodies auditing Trust Service Providers that issue Publicly-Trusted Certificates	Art. 20, 21, 24
ETSI TS 119 403-3	1.1.1	Electronic Signatures and Infrastructures (ESI); Trust Service Provider Conformity Assessment; Part 3: Additional requirements for conformity assessment bodies assessing EU qualified trust service providers	Art. 20, 21, 24
ETSI TS 119 600	1.2.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for trust service status lists providers	Art. 22
ETSI TS 119 612	2.2.1	Electronic Signatures and Infrastructures (ESI); Trusted Lists	Art. 22
ETSI TS 119 614-1	1.1.1	Electronic Signatures and Infrastructures (ESI); Testing Conformance and Interoperability of Trusted Lists; Part 1: Specifications for testing conformance of XML representation of Trusted Lists	Art. 22
ETSI TS 119 615	1.1.1	Electronic Signatures and Infrastructures (ESI); Trusted lists; Procedures for using and interpreting European Union Member States national trusted lists	Art. 22
ETSI TR 103 684	1.1.1	Electronic Signatures and Infrastructures (ESI); Global Acceptance of EU Trust Services	No relevant article
ETSI TR 119 400	1.1.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for trust service providers supporting digital signatures and related services	Art. 24, Section 4
ETSI TR 119 000	1.2.1	Electronic Signatures and Infrastructures (ESI); The framework for ” of signatures: overview	Section 4+5
ETSI TR 119 001	1.2.1	Electronic Signatures and Infrastructures (ESI); The framework for ” of signatures; Definitions and abbreviations	Section 4+5
...	...	...	...

Standard	minimal Version	Title	eIDAS (example)
ETSI SR 019 020	1.1.2	The framework for ” of signatures; Standards for AdES digital signatures in mobile and distributed environment	Section 4+5

Table 2.1: Essential general eIDAS-standards related to signatures, seals, timestamps and preservation

All published general standards regarding (qualified) trust service providers in the context of electronic signatures, seals and timestamps can be found here:
https://portal.etsi.org/TB-SiteMap/esi/esi-activities

2.3.3 Relevant Publication of Federal Office for Information Security concerning the ETSI EN 319401

The Federal Office for Information Security developed in collaboration with German National Supervisory Body the Federal Network Agency Criteria for Assessing Trust Service Providers against ETSI Policy Requirements so [ETSI-319401(v2.3.1)]. The catalogue is a recommendation for Conformity Assessment Bodies to facilitate the assessment of (qualified) trust service providers and to ensure a comparability of the audits. See [BSI-319401-AssP1]

2.4 Trust Services and Formats for (qualified) electronic signatures and seals

2.4.1 Introduction

The standards regarding trust services for issuance of certificates for electronic signatures and seals can be differentiated in those ones with fundamental requirements and those concerning special technical aspects e.g. signature-/seal formats as well as qualified electronic signature creation devices and cryptographic suites. Technically electronic signatures and seals are equal. The only difference is that certificates for (qualified) electronic signatures are issued for natural persons and seals for legal entities. This means also that the relevant standards are the same for both signatures and seals. Standardization also differentiates between the types of creation of electronic signatures and seals, so remote signatures or local creation typically based on smartcards or Hardware Security Module (HSM).

2.4.2 Overview on essential standards

2.4.2.1 Standards on certificates for electronic signatures and seals

ETSI	minimal Version	Title	eIDAS (examples)
EN 319 411-1	1.3.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 1: General requirements	Art. 24, 26, 28, 29, 38
EN 319 411-2	2.4.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 2: Requirements for trust service providers issuing EU qualified certificates	Art. 28, 29, 38, 39 Annex I+III
EN 319 412-1	1.4.4	Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 1: Overview and common data structures	Art. 28, 29, 38, 39
EN 319 412-2	2.2.1	Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 2: Certificate profile for certificates issued to natural persons	Art. 28, 29, 30
EN 319 412-3	1.2.1	Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 3: Certificate profile for certificates issued to legal persons	Art. 38, 39
EN 319 412-5	2.3.1	Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 5: QCStatements	Art. 14, 45
TS 119 172-1	1.1.1	Electronic Signatures and Infrastructures (ESI); Signature Policies; Part 1: Building blocks and table of contents for human readable signature policy documents	Art. 24
TR 119 411-4	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 4: Checklist supporting audit of TSP against ETSI EN 319 411-1 or ETSI EN 319 411-2	Art. 24, 26, 30, 38, 39, Annex I+III

Table 2.2: Essential eIDAS-standards on issuance certificates for electronics signatures and seals

2.4.2.2 Signature-/Seal Formats

Standard	Version	Title	Article eIDAS (examples)
ETSI EN 319 122-1	1.1.5	Electronic Signatures and Infrastructures (ESI); CAdES digital signatures; Part 1: Building blocks and CAdES baseline signatures	Art 27 & 2015/1506
ETSI TS 103 173	2.1.1	Electronic Signatures and Infrastructures (ESI); CAdES Baseline Profile	Art 27 & 2015/1506
ETSI EN 319 132-1	1.1.1	Electronic Signatures and Infrastructures (ESI); XAdES digital signatures; Part 1: Building blocks and XAdES baseline signatures	Art 27 & 2015/1506
ETSI TS 103 171	2.1.1	Electronic Signatures and Infrastructures (ESI); XAdES Baseline Profile	Art 27 & 2015/1506
ETSI EN 319 142-1	1.1.1	Electronic Signatures and Infrastructures (ESI); PAdES digital signatures; Part 1: Building blocks and PAdES baseline signatures	Art 27 & 2015/1506
ETSI TS 103 172	2.2.2	Electronic Signatures and Infrastructures (ESI); PAdES Baseline Profile	Art 27 & 2015/1506
ETSI EN 319 162-1	1.1.1	Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 1: Building blocks and ASiC baseline containers	Art 27 & 2015/1506
ETSI TS 103 174	2.2.1	Electronic Signatures and Infrastructures (ESI); ASiC Baseline Profile	Art 27 & 2015/1506
ETSI EN 319 162-2	1.1.1	Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 2: Additional ASiC containers	Art 27 & 2015/1506
ETSI TS 119 182-1	1.1.1	Electronic Signatures and Infrastructures (ESI); JAdES digital signatures; Part 1: Building blocks and JAdES baseline signatures	Art 27 & 2015/1506

Table 2.3: Essential eIDAS-standards on Signature-/Seal Formats

2.4.2.3 Digital Signature Creation and Validation

Standard	minimal Version	Title	eIDAS (example)
ETSI EN 319 102-1	1.2.3	Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation	Art. 29, 30, 32
CEN EN 419211-1	2014	Protection profiles for secure signature creation device — Part 1: Overview	Art. 29, 30, 39 Annex II
CEN EN 419211-2	2013	Protection profiles for secure signature creation device — Part 2: Device with key generation	Art. 29, 30, 39 Annex II
...	...	...	...

Standard	minimal Version	Title	eIDAS (example)
CEN EN 419211-3	2014	Protection profiles for secure signature creation device — Part 3: Device with key import	Art. 29, 30, 39 Annex II
CEN EN 419211-4	2014	Protection profiles for secure signature creation device — Part 3: Device with key import	Art. 29, 30, 39 Annex II
CEN EN 419211-5	2014	Protection profiles for secure signature creation device — Part 4: Extension for device with key generation and trusted channel to certificate generation application	Art. 29, 30, 39 Annex II
CEN EN 419211-6	2014	Protection profiles for secure signature creation device — Part 6: Extension for device with key import and trusted channel to signature creation application	Art. 29, 30, 39 Annex II
CEN EN 419221-5	2018	Protection Profiles for TSP Cryptographic Modules - Part 5 - Cryptographic Module for Trust Services	Art. 29, 30, 39, Annex II
CEN EN 419241-1	2018-09	Trustworthy Systems Supporting Server Signing Part 1: General System Security Requirements,	Art. 29, 30, 39, Annex II
CEN EN 419241-2	2019-05	Trustworthy Systems Supporting Server Signing Part 2: Protection Profile for QSCD for Server Signing	Art. 29, 30, 39, Annex II
ETSI TS 119 300	1.2.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for cryptographic suites	Art. 29, 30, 39, Annex II
ETSI TS 119 312	1.3.1	Electronic Signatures and Infrastructures (ESI); Cryptographic Suites	Art. 29, 30, 39, Annex II
ETSI TS 119 431-1	1.2.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for trust service providers; Part 1: TSP service components operating a remote QSCD / SCDev	Art. 29, 30 39, Annex II
ETSI TS 119 431-2	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for trust service providers; Part 2: TSP service components supporting AdES digital signature creation	Art. 29, 30, 39, Annex II
ETSI TS 119 432	1.2.1	Electronic Signatures and Infrastructures (ESI); Protocols for remote digital signature creation	Art. 29, 30, 39, Annex II
ETSI TR 119 100	1.1.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for signature creation and validation	Art. 29, 30 32, 39, Annex II
ETSI TS 119 101	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for applications for signature creation and signature validation	Art. 29, 30 32, 39, Annex II

Table 2.4: Essential eIDAS-standards on digital signature creation

All published standards regarding *electronic signatures, seals, signature-/seal creation and validation, trust service status list providers* can be found here: https://portal.etsi.org/TB-SiteMap/esi/esi-activities

2.4.2.4 Signing Devices and Cryptographic suites

Standard	minimal Version	Title	eIDAS (examples)
CEN EN 419211-1	2014	Protection profiles for secure signature creation device — Part 1: Overview	Art. 29, 30, 39, Annex II
CEN EN 419211-2	2013	Protection profiles for secure signature creation device — Part 2: Device with key generation	Art. 29, 30, 39, Annex II
CEN EN 419211-3	2014	Protection profiles for secure signature creation device — Part 3: Device with key import	Art. 29, 30, 39, Annex II
CEN EN 419211-4	2014	Protection profiles for secure signature creation device — Part 3: Device with key import	Art. 29, 30, 39, Annex II
CEN EN 419211-5	2014	Protection profiles for secure signature creation device — Part 4: Extension for device with key generation and trusted channel to certificate generation application	Art. 29, 30, 39, Annex II
CEN EN 419211-6	2014	Protection profiles for secure signature creation device — Part 6: Extension for device with key import and trusted channel to signature creation application	Art. 29, 30, 39, Annex II
CEN EN 419 221-5	2018	Protection Profiles for TSP Cryptographic Modules - Part 5 - Cryptographic Module for Trust Services	Art. 29, 30, 39, Annex II
CEN EN 419 241-1	2018-09	Trustworthy Systems Supporting Server Signing Part 1: General System Security Requirements,	Art. 29, 30, 39, Annex II
CEN EN 419 241-2	2019-05	Trustworthy Systems Supporting Server Signing Part 2: Protection Profile for QSCD for Server Signing	Art. 29, 30, 39, Annex II
ETSI TS 119 300	1.2.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for cryptographic suites	Art. 29, 30, 39, Annex II
ETSI TS 119 312	1.3.1	Electronic Signatures and Infrastructures (ESI); Cryptographic Suites	Art. 29, 30, 39, Annex II

Table 2.5: Essential eIDAS-standards on cryptographic suites

All published standards regarding signature creation and other related devices especially the relevant Protection Profiles can be found here: https://standards.cen.eu/dyn/www/f?p=204:32:0::::FSP_ORG_ID,FSP_LANG_ID:478566,25&cs=16448244FB7AC1BDE2E621EACB21E71E2

2.4.3 Relevant Publication of Federal Office for Information Security

The Federal Office for Information Security developed a guideline n coordination with the Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (BNetzA) and recommendations for formats of electronic signatures, seals, timestamps and evidence records [(EU)2015/1506, VDG17, (EU)910/2014, Article 33(1), 34 and 40]. The document also describes obligations concerning acceptance and validation of defined signature-/seal formats and timestamps for public administration and requirements on preservation. For federal administrations the guideline is mandatory for other institutions or private companies it's a recommendation [BSI-Sig-Leit].

2.5 Trust Services for Creation of qualified electronic timestamps

2.5.1 Introduction

Qualified electronic timestamps provide the unambiguous evidence that given data existed unaltered at the given time. They are created by qualified trust service providers for the creation of qualified electronic timestamps.

2.5.2 Overview on essential standards

Standard	minimal Version	Title	eIDAS (examples)
ETSI EN 319 421	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and Security Requirements for Trust Service Providers issuing Time-Stamps	Art. 42
ETSI EN 319 422	1.1.1	Electronic Signatures and Infrastructures (ESI); Time-stamping protocol and time-stamp token profiles	Art. 42

Table 2.6: Overview on essential standards for creation of qualified electronic timestamps

2.5.3 Relevant Publication of Federal Office for Information Security

The Federal Office for Information Security developed in coordination with the Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (BNetzA) a guideline and recommendations for formats of electronic signatures, seals, timestamps and evidence records [(EU)2015/1506, VDG17, (EU)910/2014, Article 33(1), 34 and 40]. The document also describes obligations concerning acceptance and validation of defined signature-/seal formats and timestamps for public administration as well as requirements on preservation. For federal administrations the guideline is mandatory for other institutions or private companies it's a recommendation [BSI-Sig-Leit].

2.6 Trust Services for Validation of qualified electronic signatures, seals and timestamps

2.6.1 Introduction

Qualified validation services are qualified trust services for the validation of (qualified) electronic signatures, seals and timestamps. It's not mandatory to use such services for the validation. If the validation is provided as a service for external clients it´s recommended resp. in some cases obligatory for the service provider to become qualified trust service provider for the validation of qualified electronic signatures, seals and timestamps (see section 2.6.3, [(EU)910/2014, Article 33(1), 34(1), 40]). The main advantage in the utilization of qualified validation service is that the validation result is done by a certified and so trustworthy 3rd party – the QTSP which may imply a high evidence value than a local verification based on internal software of the verifying institution.

2.6.2 Overview on essential standards

ETSI	minimal Version	Title	eIDAS (examples)
EN 319 102-1	1.2.3	Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation	Art. 29, 30, 32
TS 119 101	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for applications for signature creation and signature validation	Art. 29, 30, 32, 39, Annex II
TS 119 102-2	1.2.1	Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 2: Signature Validation Report	Art. 32,33
TS 119 441	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy requirements for TSP providing signature validation services	Art. 33
TS 119 442	1.1.1	Electronic Signatures and Infrastructures (ESI); Protocol profiles for trust service providers providing AdES digital signature validation services	Art. 32, 33
TR 119 100	1.1.1	Electronic Signatures and Infrastructures (ESI); Guidance on the use of standards for signature creation and validation	Art. 28, 29, 32

Table 2.7: Overview on essential standards for (qualified) validation services

2.6.3 Relevant Publication of Federal Office for Information Security

The Federal Office for Information Security developed a guideline [BSI-Sig-Leit] and recommendations for formats of electronic signatures, seals, timestamps and evidence records. The document also describes obligations concerning acceptance and validation of defined signature-/seal formats and timestamps for public administration as well as requirements on preservation. For federal administrations the guideline is mandatory for other institutions or private companies it’s a recommendation.

2.7 Trust Services for Preservation of data, documents and (qualified) electronic signatures, seals and timestamps

2.7.1 Introduction

If cryptographically signed data and so the (qualified) electronic signature, seal or timestamp have to be preserved for longer time than the underlying algorithm are still secure it's necessary to implement appropriate measures for the preservation. The relevant standards are given below.

2.7.2 Overview on essential standards

Standard	minimal Version	Title	eIDAS (examples)
Standard	Version	Title	as
ETSI TS 119 511	1.1.1	Electronic Signatures and Infrastructures (ESI); Policy and security requirements for trust service providers providing long-term preservation of digital signatures or general data using digital signature techniques	Art. 34, 40
ETSI TS 119 512	1.1.2	Electronic Signatures and Infrastructures (ESI); Protocols for trust service providers providing long-term data preservation services	Art. 34, 40
ETSI SR 019 510	1.1.1	Electronic Signatures and Infrastructures (ESI); Scoping study and framework for " of long-term data preservation services, including preservation of/with digital signatures	Art. 34, 40

Table 2.8: Essential eIDAS-standards for preservation of (qualified) electronic signatures, seals and timestamp

2.7.3 Relevant Publication of Federal Office for Information Security

The Federal Office for Information Security is responsible for German Technical Guideline TR-03125 TR-ESOR which defines technical requirements on concrete products for the preservation of cryptographically signed data and the (qualified) electronic signature, seal or timestamps. This Technical Guideline TR-ESOR describes a differentiated catalogue of obligatory (shall), recommended (should), and optional (can) requirements with regard to all elements and areas in which there is a need to develop effective, sustainable, and economical technical scenarios for the long-term preservation of signed and unsigned documents and data with preservation of evidence.

TR-ESOR v1.3 [BSI-TR-03125] includes:

- recommendations for a reference architecture including specifications of processes, modules and interfaces,
- recommended data and document formats,
- mandatory formats for archival information packages or archive data objects and evidence records

- mandatory formats for the upper interface of TR-ESOR and recommendations for the other interfaces
- functional and technical Conformity test specifications and test tools concerning the technical interoperability of the archive information packages, the evidence records and the upper interface of TR-ESOR for technically interoperable products.

The following links to the BSI web site shows the current version of TR-ESOR: `http://www.bsi.bund.de/EN/tr-esor` (English) and `https://www.bsi.bund.de/tr-esor` (German). A "Guideline for the preservation of evidence according to BSI TR-03125 TR-ESOR - An assistance for authorities and companies-" [BSI-TR-03125-lt]

3 Technical Background

3.1 Overview

Cryptography is widely used in our digital world. When visiting a secured website or an online shop for example. Generally speaking, it is necessary in any case of data exchange where the exchange is private and secured against external manipulation on its way. Algorithms are involved to make sure the informations are protected and can only be read properly by the intended receiver. This chapter focuses on the technical background of this process. The basic algorithms related to digital documents are explained in this chapter. Basically a method or algorithm is safe if it cannot be compromised within a huge calculation time[1] and all informations about the cryptosystem are public (except the private key), see Kerckhoffs' Principle. Since this is a topic of constant development and research, new methods breaking algorithms will be developed. Since the digitalisation takes a more important role mechanisms against changes are important to protect digital documents. This includes the document itself, the author and date of creation. These informations need to stay in direct touch with the document itself to verify its authenticity. The mechanisms to verify its authenticity at any time by any user should be given.

[1] even with modern computers millions of years are necessary

3.2 Cryptographic Foundations

The following explains the essential basics of cryptography relating to digital signatures. A systematic introduction to the field of cryptography can be found in [Buch10]. [MOV97] provides a comprehensive treatment of the material.

3.2.1 The principle of Digital Signatures

In this chapter electronic signatures, advanced electronic signatures, qualified electronic signatures, electronic seals, advanced electronic seals, and qualified electronic seals as per Regulation [(EU)910/2014, Article 3] are called digital signatures. One of the most important functions of electronic documents that need to be signed or sealed is to prove that the declaration embodied in the document comes from the signer (signatory) of the electronic signature or the creator of the electronic seal (authenticity function) and that this digital signature can be verified by the recipient of the document (verification function). In this case, only the signer (signatory) of the electronic signature or the creator of the electronic seal should be able to generate the digital signature, but it must be possible for anyone to verify the digital signature.

To construct the digital signature, it is possible to use one private key (private key, K_{priv}) and one public key (public key, K_{pub}), shown in Figure 3.1.

With the private key K_{priv} (which can only be accessed by the owner) a signature $\sigma = \mathrm{Sign}(m, K_{priv})$ is generated. The public key K_{pub} is available to everyone and is used to verify signatures with the $\mathrm{Verify}(m, \sigma, K_{pub})$ function. The keys are related to the extent that they can be used to execute the inverse of the creation and verification operations on the signatures. The security of the method is based on the fact that it is practically impossible to derive the private key from the public key, even with the help of high-performance computer systems and million years of processor time. The public key K_{pub} in this case is derived from the private key K_{priv} by applying a so-called one-way function f; $K_{pub} = f(K_{priv})$. For this reason, the public key can be stored in a publicly accessible directory without disclosing the private key.

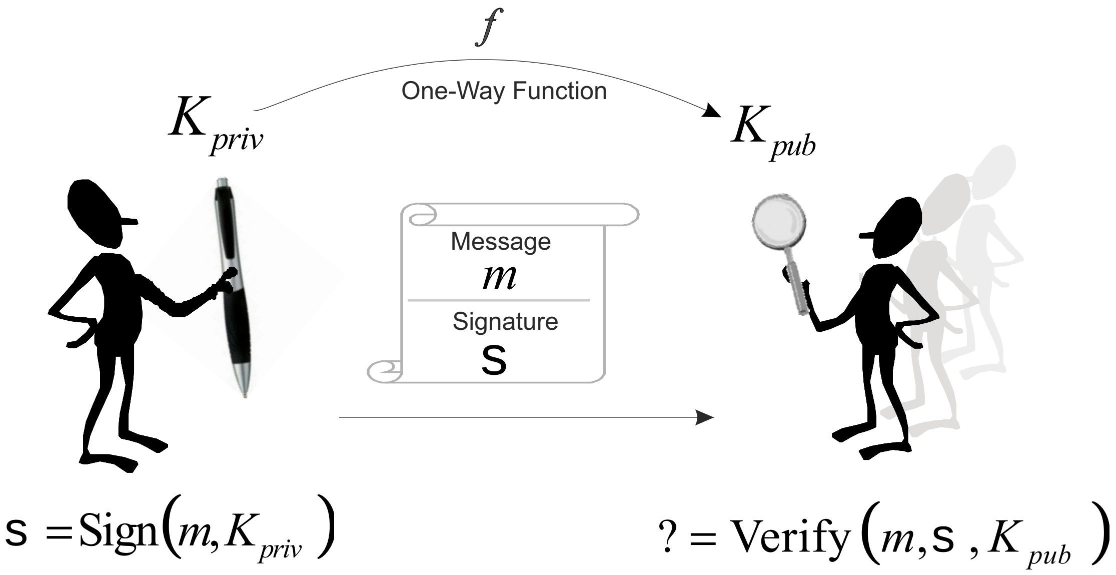


Figure 3.1: Principle of digital signatures

3.2.2 One-way functions and problems in number theory

A one-way function f is needed[2] for the construction of a digital signature method. Intuitively speaking, a one-way function $f : x \to y$ is a function that is easy to calculate, but very 'difficult' to invert. For example, while it is 'easy' to calculate the value $y = f(x)$ from x, it is difficult in practice to determine the original value $x = f^{-1}(y)$ from the value y.

3.2.2.1 Complexity of Algorithms

To define the terms "easy" and "difficult" more precisely, we examine the runtimes of the existing algorithms for solving the corresponding problems.

Two terms used in this context are bit operation and the O notation. Bit operation refers to the time required to operate on two bits. O notation permits the asymptotic estimate of the runtime. For two real-valued functions f, g, we write $f = O(g)$ if the inequality $|f(x)| \le c \cdot |g(x)|$ is true for a constant value $c > 0$ for sufficiently large values of x. This method of examination therefore concentrates on the difficulty of solving a problem for very large input values while abstracting it enough to disregard implementation-dependent details that would only affect the runtime of the algorithm by a constant factor.

If an algorithm is given a number n that is greater than Euler's number $e \approx 2.71828$ as input and needs $O(\log(n)^c)$ operations for a constant $c > 1$ to calculate the result, then the algorithm can be executed in polynomial time related to the input length of number n (number of binary characters required to represent the number n) of size $= O(\log(n))$. It should be noted that the base of the logarithm is irrelevant in O notation because of the fact that $\log_a(n) = c \cdot \log_b(n)$ with $c = \log_a(b)$ and that the O notation abstracts away from constant factors.

When analysing algorithms, it is possible to classify them based on their determinism. Algorithms that always produce the same output given the same input are called deterministic algorithm, while algorithms that produce different outputs given the same input due to the use of random values are called non-deterministic algorithms or probabilistic algorithm.

A problem that can be solved with a deterministic algorithm in polynomial time is called an "easy" problem. The set of all such problems is referred to as $\mathcal{P}$. The multiplication of integer numbers is in $\mathcal{P}$, for example.

The set of all probabilistic algorithms with polynomial runtime is referred to as $\mathcal{NP}$. A given solution to a problem in the $\mathcal{NP}$ class can be verified with an algorithm in $\mathcal{P}$. For example, the factorization problem analysed in Section 3.2.2.2 is in $\mathcal{NP}$.

A problem is called "difficult" if the probability of success for *every* algorithm from the $\mathcal{NP}$ class is "negligible"[3].

If a function f is "easy" and the inverse function f^{-1} is "difficult", then f is called a one-way function (cf. [Gold01, Definition 2.1]).

Since every problem that can be solved with a deterministic algorithm in polynomial time can also be solved with a probabilistic algorithm in polynomial time, $\mathcal{P} \subseteq \mathcal{NP}$ is true. It is not known, though, if the set $\mathcal{NP}$ is actually larger than $\mathcal{P}$. Whether or not $\mathcal{P} = \mathcal{NP}$ is true is probably the most important open problem in theoretical computer science. If one was able to prove the existence of a one-way function, then this would imply $\mathcal{P} \neq \mathcal{NP}$.

Since no one has been able yet to find a single function f from $\mathcal{P}$ for which it can be shown that f^{-1} is "difficult", it is necessary to use functions that *appear* to posses the one-way property at the current time for the construction of digital signature methods. Instead of requiring *every possible*

[2] In [Romp90] it was shown that one-way functions are not only necessary, but also sufficient for secure signature methods.

[3] A value is "negligible" if it is smaller than $\frac{1}{p(l)}$ for every polynomial p with sufficiently high input length l. For example a probability of success of $\frac{1}{2^l}$ is negligible.

algorithm from $\mathcal{NP}$ to have a negligible probability of success, it has become a standard in practice to use functions for which there is *no algorithm yet known* with a non-negligible probability of success.

To construct digital signature systems, problems are used, such as the factorization problem considered in Section 3.2.2.2 or the problem of the discrete logarithm discussed in Section 3.2.2.3, for which the only algorithms known have subexponential or exponential runtime. In particular, for appropriately selected instances of this problem there are no known polynomial time algorithms that can efficiently calculate the private key K_{priv} from the public key K_{pub}.

If an algorithm is given a number n as input and needs $O(c^{\log(n)})$ operations for a constant $c > 1$ to calculate the result, then it has exponential runtime related to the input length of the number n (number of binary characters required to produce the number n) of size $= O(\log(n))$.

Let e be Euler's number, $n > e$, $\log(n)$ the natural logarithm of the number n, u a real number with $0 \leq u \leq 1$, and v a positive real number. Then the description of the runtime of algorithms with subexponential runtime is usually based on the function defined in [BLP93]

$$L_n\left[u, v\right] = e^{v(\log(n))^u(\log(\log(n)))^{1-u}}. \tag{3.1}$$

Since $L_n[0, v]= e^{v(\log(n))^0 \log(\log(n))} = e^{v \log(\log(n))} = \log(n)^v = O(\log(n)^c)$ for a constant c, $L_n[0, v]$ corresponds to polynomial time. In a similar manner, $L_n[1, v]= e^{v \log(n)} =O(n^v)$ corresponds to exponential time, and for $0 < u < 1$, the time required lies between these two extremes, and the algorithm in this case has subexponential runtime.

For example, the best known algorithm for the factorization problem, the number field sieve [LLMP93], has an expected runtime of $L_n[1/3, (64/9)^{1/3} + o(1)]$, where $o(1)$ is a function that goes to 0 as n goes to infinity and can be viewed as a generalisation of the constant for the O notation. For appropriately selected DL problems in the point groups of elliptic curves, it is currently believed that their solution even requires exponential runtime. This explains why cryptographic systems based on elliptic curves can offer the same level of security with significantly shorter keys.

However, it must be emphasised that all statements made today regarding the security of asymmetric crypto algorithms are based on unproven conjectures. The one-way property of the underlying "one-way functions" is unfortunately itself a function of time. A chance discovery by a single mathematician can show that a supposedly difficult problem is easy to solve, and all signatures based on the corresponding method would suddenly be easy to forge. For this reason, the aspects relating to the augmentation of electronic signatures, seals, timestamps by digital signature techniques specified in [ETSI-119511(v1.1.1)], [ETSI-119512] and [BSI-TR-03125] always need to be taken into account in the long-term preservation of qualified electronic signatures ([ETSI-119511(v1.1.1)], [ETSI-119512] and [VDG17]). In addition, consideration should be given to using several separate crypto algorithms simultaneously for critical applications.

3.2.2.2 Factoring Problem

The factorization problem on which the RSA procedure presented in Section 3.2.3.1 is based, for example, consists of decomposing a large composite number into its prime factors. Since it is easy to multiply two prime numbers with each other but is substantially more difficult to factor a composite number into its prime factors, the factorization problem appears to be suitable for the construction of one-way functions.

As shown in a quote from the famous mathematician Carl Friedrich Gauss from 1801 [Gaus01, Article 329], the factorization problem is by no means a new problem:

> *"The task of distinguishing prime numbers from composite numbers and fractionise the latter into their prime factors is known to be one of the most important and useful tasks in arithmetic. It took the effort and the acumen of the ancient and contemporary geometers to such an extent that it would be superfluous to spend much time talking about it. So, the experienced calculator not infrequently has the chance to take great advantages from the*

> *prime factorization of great numbers, which compensates the high expenditure of time for doing so. Furthermore, it is likely to demand the dignity of science to perfect all means for the solution of such an elegant and famous problem.(translation)"*

As shown in [AKS04] (see also [Born02]), there is an algorithm in $\mathcal{P}$ for the first problem mentioned by Gauss in which it must be decided whether or not a given number is a prime number.

For the second problem mentioned, the factorization problem, there is no known algorithm that runs in polynomial time. An overview of the most popular factorization algorithms can be found in [Nebe00]. From the perspective of cryptography, these algorithms can be divided into three main categories:

- *Classic factorization algorithms*

  In addition to the obvious trial by division algorithm, there have been various factorization algorithms for a long time now due to the work of famous mathematicians like Fermat, Legendre, Euler, and Gauss (cf. [Ries94, Chapter 5]). These algorithms – as well as the ρ algorithm suggested by Pollard [Poll75] or the SQUFOF[4] algorithm from Shanks [Ries94, page 186 ff.] – are not used in practice because they have exponential runtimes and can only be used successfully for very short key lengths.

- *Factorization using smooth group orders*

  There is also a series of factorization algorithms available [Poll74, Will82, ScLe84, Lens87] that perform exponentiations in a finite abelian group G_n that is related to the number n to be factored and that are successful if and only if the group order $|G_n|$ is "smooth" – meaning it is only composed of small prime numbers. Such algorithms can efficiently determine small factors of a large composite number n. In order to make a digital signature method impossible to break using these algorithms, the numbers used for this purpose in cryptography are usually[5] numbers of the form $n = pq$ where p and q are about the same size.

- *Sieve-based factorization algorithms*

  The fastest factorization algorithms for large numbers of the form $n = pq$ use a so-called factor base for which relations are collected by applying sieve methods. If there are enough of these relations available, then a linear system of equations is solved whose solution can then be used to construct two integer numbers x and y for which $x^2 \equiv y^2 \pmod{n}$, but $x \not\equiv \pm y \pmod{n}$ is true. Once such numbers have been determined, it is possible to find a factor of n easily by calculating the greatest common factor $\gcd(x - y, n)$. The quadratic sieve[Pome85, Silv87] and the so-called number field sieve [LLMP93] differ in terms of how they find these relations. If the number to be factored contains more than about 130 decimal places, then the number field sieve is more efficient. A record, set by scientists from the University of Bonn in co-operation with the Federal Office for Information Security, had 200 decimal places [RSA200]. A slightly smaller number [RSA640] was also factored by this work group. On December 12, 2009, 232 decimal digits (768 bits) were factored [KAF10].

3.2.2.3 Discrete Logarithm Problem

The discrete logarithm problem (DLP) in a finite, abelian group G, which is the basis of the (EC)DSA method presented in Section 3.2.3.2, for example, consists of determining the exponent n from the given group elements $g, g^n \in G$.

[4] SQUare FOrms Factorization

[5] [RFC8017] specifies, in contrast to the previous versions of this standard, the use of several separate prime numbers for the RSA procedure. In addition, some crypto systems are based on the problem of factoring numbers of the form $n = pq^2$ [OkUc98, Pail99, Hueh04b] or $n = p^k q$ [Taka98].

In practical applications, the ”Square & Multiply” method or a variation of this method is used to calculate g^n (cf. [Gord98]).

EXAMPLE 3.2.2-1

The number g^5 is to be calculated. To do this, one examines the binary representation of the exponent $5_{10} = 101_2$. Now one starts with the value g and go through the binary representation of the exponent from left to right, whereby in each step to the next bit one first squares the number and then, if the bit to which one is going is equal to ”1”, multiply the current value by g. This calculates $g^5 = (g^2)^2 \cdot g$.

It is easy to see that one only needs $O(\log(n))$ group operations for the calculation of g^n using such a strategy.

In the groups used for cryptographic purposes, g^n can be calculated in polynomial time, while only subexponential or exponential algorithms exist for the solution of the discrete logarithm problem, which means it is possible to construct digital signature systems on this basis.

There are two basic types of algorithms available for solving the DLP.

On the one hand, there are so-called *generic algorithms* that can be used in any finite abelian groups. The most widely known generic algorithms for the calculation of discrete logarithms are the ”Baby-Step Giant-Step algorithm” from Shanks [Shan72] and the ”Rho method” from Pollard [Poll78]. Both have an (expected) runtime of $O(\sqrt{|G|})$ group operations, where $|G|$ is the order of the group G. These algorithms therefore run in exponential runtime on the order of $O(\log(|G|))$ in terms of their input length. In addition, the algorithm from Pohlig and Hellman can be used to reduce the calculation of discrete logarithms in a finite abelian group G for which the factorization of the group order is known to the calculation of discrete logarithms in the subgroups $U \subset G$. For this reason, no groups whose group order consists only of small prime numbers should be used for cryptographic purposes.

In addition, there are *sieve-based algorithms* available for certain groups, for example for multiplicative groups of finite fields, that allow the calculation of discrete logarithms in subexponential time. The difficulty of solving the DLP therefore depends primarily on which group G is used.

Cryptographic applications primarily use multiplicative groups of finite fields and point groups of elliptic curves. In addition, divisor class groups on hyperelliptic curves [Kobl89] or class groups of imaginary quadratic fields [BuWi88] have also been suggested for use in constructing difficult-to-solve DL problems.

Multiplicative groups of finite fields

The use of multiplicative groups of finite fields to construct difficult DL problems is as old as the DL problem itself as suggested by Diffie and Hellman [DiHe76].

To do this, simply select a prime number p , and whenever calculating another number with it, only consider the smallest remainder left over after subtracting an integer number multiple of p. This is referred to as calculating ”modulo” p and is written $a \equiv b \pmod{p}$ when a and b only differ by a multiple of p.

The residue classes modulo p $(0, 1, 2, \cdots, p-1)$ form a finite field – the ”prime field” $\mathbb{F}_p$ [6].

Basic arithmetic in $\mathbb{F}_p$ is very simple: one just adds and multiplies as usual and then reduces the result to a number modulo p.

EXAMPLE 3.2.2-2

We select $p = 7$ and calculate 2^5 in $\mathbb{F}_7$. As shown in Example 1, it is possible to exponentiate by 5

[6] in general, it is possible to construct a finite field with p^n elements for every prime power p^n. The basics of finite fields are explained in [Kobl94, Chapter II.1]. A comprehensive treatment of the material can be found in [LiNi86].

in two squaring steps and a subsequent multiplication operation. For this reason, $2^5 = (2^2)^2 \cdot 2 = 4^2 \cdot 2 = 16 \cdot 2 \equiv 2 \cdot 2 \equiv 4 \pmod{7}$.

There are two different approaches for inverting an element $x \in \mathbb{F}_p$, meaning for the calculation of x^{-1} so that $x^{-1} \cdot x \equiv 1 \pmod{p}$ is true:

1.) Exponentiation with $p - 2$

Using ”Fermat’s little theorem”, $x^{p-1} \equiv 1 \pmod{p}$ for all elements x that are relatively prime to p, and one therefore obtains the inverse element $x^{-1} \equiv x^{p-2} \pmod{p}$ by exponentiation with $p - 2$. In the example above, $2^5 \equiv 4 \pmod{7}$ is therefore the inverse element of 2 modulo 7.

2.) Extended Euclidian algorithm

With the extended Euclidean algorithm (EEA) (cf. [BaSh96, Chapter 4.3]), it is possible for two integer numbers a and b to calculate a linear combination of the greatest common divisor $\gcd(a, b) = s \cdot a + t \cdot b$ of the number input.

Line	r_i	=	$s_i \cdot a$	+	$t_i \cdot b$	q_i
1	a	=	$1 \cdot a$	+	$0 \cdot b$	
2	b	=	$0 \cdot a$	+	$1 \cdot b$	$q_2 = \lfloor a/b \rfloor$
			$\vdots$			
i	$r_{i-2} - q_{i-1} \cdot r_{i-1}$	=	$(s_{i-2} - q_{i-1} \cdot s_{i-1}) \cdot a$	+	$(t_{i-2} - q_{i-1} \cdot t_{i-1}) \cdot b$	$q_i = \lfloor r_{i-1}/r_i \rfloor$
			$\vdots$			
$n-1$	$\mathrm{ggT}(a, b)$	=	$s \cdot a$	+	$t \cdot b$	
n	0					

Table 3.1: Extended Euclidean algorithm (EEA)

To calculate this, one initializes the following table with the first two rows and calculate the value $q_2 = \lfloor a/b \rfloor$ using integer number division. A new row (i) is obtained by subtracting q_{i-1} times the last row $(i-1)$ from the second to last row $(i-2)$ and calculating the factor q_i used to determine the next row $(i+1)$. This is done until one finally obtains the remainder $r_n = 0$. The desired linear combination can then be found in row $n - 1$.

If the algorithm is executed on a number x that is relatively prime modulo p and results in the linear combination $1 = \gcd(p, x) = s \cdot p + t \cdot x$, then $t \equiv x^{-1} \pmod{p}$ is the desired inverse element.

EXAMPLE 3.2.2-3
Corresponding to the example given above, we will calculate the inverse element of $2 \pmod{7}$ again, but this time using the extended Euclidean algorithm.

Line	r_i	=	$s_i \cdot a$	+	$t_i \cdot b$	q_i
1	7	=	$1 \cdot 7$	+	$0 \cdot 2$	
2	2	=	$0 \cdot 7$	+	$1 \cdot 2$	$q_2 = \lfloor 7/2 \rfloor = 3$
3	$(7 - 3 \cdot 2)$	=	$(1 - 3 \cdot 0) \cdot 7$	+	$(0 - 3 \cdot 1) \cdot 2$	
	1	=	$1 \cdot 7$	+	$(-3) \cdot 2$	$q_3 = \lfloor 2/1 \rfloor = 2$
4	0					

Table 3.2: Inversion of $2 \pmod{7}$ using the EEA

> To do this, one initializes the first two rows in Table 3.2 and calculate $q_2 = \lfloor 7/2 \rfloor = 3$. Row 3 is now obtained by subtracting 3 times row 2 from row 1. Since row 4 already contains a remainder $r_4 = 0$, it can be seen in row 3 that $\gcd(7,2) = r_3 = 1$ and that the inverse of 2 is then $-3 \equiv 4 \pmod 7$.

While the inversion operation in $\mathbb{F}_p^*$ using exponentiation requires $O(\log(p)^3)$ bit operations, inversion using the extended Euclidean algorithm according to [BaSh96, Corollary 4.3.3] only takes $O(\log(p)^2)$ bit operations. For this reason, the extended Euclidean algorithm is usually used in practical applications for modular inversion.

There are also algorithms available for solving the DLP in the multiplicative group of a finite field that are as efficient as those for solving the factorization problem, in addition to the generic algorithms that work in every finite abelian group. In particular, it is also possible to use the number field sieve [Gord93, Webe96, JoLe03] to solve the DLP in these groups. The asymptotic runtime of these algorithms is identical to those used by the number field sieve for factorization. In [Joux13], an new index calculus algorithm for finite fields was published, which heuristically achieves an L(1/4+o(1)) time complexity.

Point groups of elliptic curves over finite fields

An elliptic curve (over a field K with characteristic $char(K) \neq 2, 3$[7]) is the set of all points $P = (x, y)$ on the "smooth"[8] curve

$$y^2 = x^3 + ax + b \tag{3.2}$$

together with the point $\mathcal{O}$ "at infinity".

It is possible to define an operation "+" on the set of points on this elliptic curve so that they form an abelian group with $\mathcal{O}$ as the neutral element. If the curve is defined over the finite field $\mathbb{F}_p$, then one obtains a finite abelian group. Such groups are used in Lenstra's "Elliptic Curve Method" [Lens87] for factorization or, as suggested by Miller [Mill85] and Koblitz [Kobl87], as a basis for difficult DL problems.

As illustrated in Figure 3.2, one obtains the point $P+Q$ on an elliptic curve over the real numbers if $P \neq \pm Q$ and $P \neq \mathcal{O}$ by placing a line through both points P and Q and reflecting the third point of intersection of this line with the elliptic curve through the x -axis. If $P = Q$, then one obtains the value $2P$ of the point P by placing a tangent on point P and reflecting the second point of intersection of the tangent with the curve across the x -axis. If $Q = -P$, then adding the points $P + Q = P - P = \mathcal{O}$ together results in a point infinitely far away – the neutral element of the point group.

To invert a point $P = (x, y)$, one simply reflects it across the x -axis $(-P = (x, -y))$.

[7] This includes, for example, the infinite field of rational numbers $\mathbb{Q}$, real numbers $\mathbb{R}$, and complex numbers $\mathbb{C}$, as well as the finite field $\mathbb{F}_p$, for a prime number $p \geq 5$. Corresponding analyses of fields with $char(K) = 2, 3$ can be found in [Mene93], for example.

[8] A curve is called "smooth" if it is possible to place a tangent on every point. In addition, it is also possible to verify this is the case when $4a^3 + 27b^2 \neq 0$.

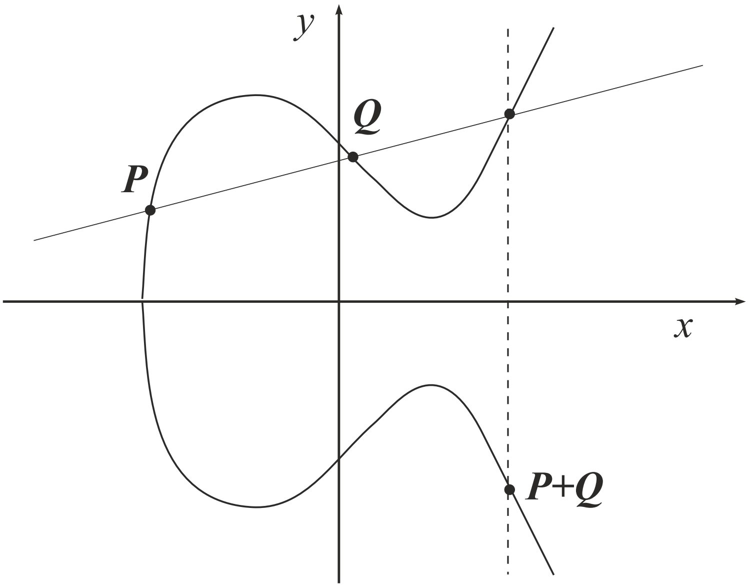


Figure 3.2: Group operation on an elliptic curve

Let $P = (x_1, y_1)$ and $Q = (x_2, y_2)$ be two points on the elliptic curve. It is then easy to calculate the sum $R = P + Q = (x_3, y_3)$ of these two points. The following is true:

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2, \\ y_3 &= -y_1 + \lambda(x_1 - x_3) \end{aligned} \tag{3.3}$$

with

$$\lambda = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1}, & \text{if } P \neq Q \text{ and } x_1 \neq x_2 \\ \\ \frac{3x_1^2 + a}{2y_1}, & \text{if } P = Q \text{ and } y_1 \neq 0. \end{cases}$$

If $P \neq Q$ and $x_1 = x_2$, then the line passes vertically through the two points P and Q and intersects the elliptic curve at infinity. In this case, $P + Q = \mathcal{O}$.

If $P = Q$ and $y_1 = 0$, then $P + Q = 2P = \mathcal{O}$ is also the point at infinity.

EXAMPLE 3.2.2-4
We examine the elliptic curve with the equation $E : y^2 = x^3 + 1$ over the real numbers $\mathbb{R}$.

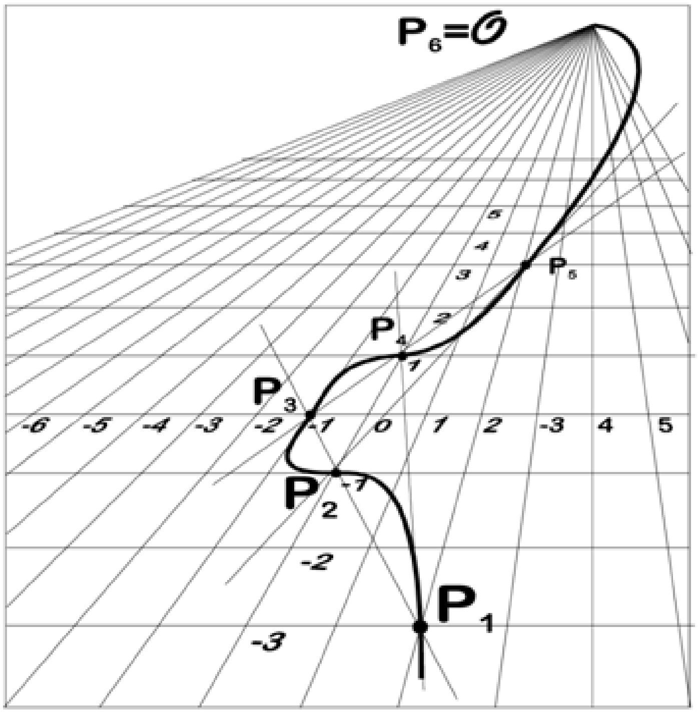


Figure 3.3: The elliptic curve $y^2 = x^3 + 1$

If one includes the point $\mathcal{O}$ at infinity in the solutions to the equation E, then one obtains the point group of the elliptic curve E over $\mathbb{R}$. In addition, one can see that there are five points on the curve with integer coordinates that, together with the point $\mathcal{O}$ at infinity, form a subgroup with group order 6. As can be seen in Figure 3.3, this subgroup is "generated" by the point P_1. The individual points can be calculated in the following manner:

$$\begin{aligned}
P_1 &= (2, -3) \\
P_2 &= 2 \cdot P_1 = P_1 + P_1 = (0, -1) \\
P_3 &= 3 \cdot P_1 = P_2 + P_1 = (-1, 0) \\
P_4 &= 4 \cdot P_1 = P_3 + P_1 = (0, 1) \\
P_5 &= 5 \cdot P_1 = P_4 + P_1 = (2, 3) \\
P_6 &= 6 \cdot P_1 = P_5 + P_1 = \mathcal{O}
\end{aligned}$$

One uses the formula for the group operation (3.3) to add points. If the elliptic curve is defined over a finite prime field $\mathbb{F}_p$, then one also needs to reduce the result of each calculation modulo p. The "multiplication" of a point with an integer number n, i.e. adding the point to itself n times, is analogous to exponentiation in a multiplicative group – the DL problem is the determination of the "factor" n for the given points P and $[n] \cdot P$.

For the DL problem in a elliptic curve over a finite field, there are generally[9] *no* subexponential algorithms in the style of the number field sieve like those used for factorization [LLMP93] or the

[9] The so-called "supersingular" and "anomalous" elliptic curves are exceptions because there are subexponential [MOV91, FMH99] and polynomial [SaAr98, Smar99] algorithms available for the solution of the DL problem in these curves. For this reason, such curves should not be used for the construction of digital signature systems. In addition, the use of certain finite fields should also be avoided [MTW04].

calculation of discrete logarithms in finite fields [Gord93]. In addition, the application of such methods on elliptic curves does not appear very promising in general [SiSu98]. Only generic algorithms are available for the solution of the DL problem (DLP) in the point group of an elliptic curve [Tesk98], for example the so-called ρ method from Pollard [Poll78]. For this reason, crypto systems based on elliptic curves offer a much higher level of security for a given key length. A further introduction to the mathematics behind the design of elliptic curves in cryptography and its application in practice is given in [Blake99].

3.2.2.4 Key Sizes

As explained above, there are subexponential algorithms available for solving the factorization problem and the DLP in the multiplicative group of a finite field. For the DLP in the point groups of elliptic curves, though, only algorithms with exponential complexity are available. This is why crypto systems based on elliptic curves can achieve the same level of security with much shorter keys.

3.2.3 Digital Signature Algorithms

In this section, we will use the factorization problem and the discrete logarithm problem to construct digital signature systems.

3.2.3.1 RSA

The RSA method, named after its inventors Rivest, Shamir, and Adleman [RSA78], was the first digital signature method, and today it is the most commonly used method in practical applications. It is based on[10] the factorization problem discussed in more detail in Section 3.2.2.2 and can be used in a similar form for the encryption of messages. In this case, the procedure followed for *en*cryption with the public exponent e corresponds approximately to the procedure for signature *verification*, and *de*cryption corresponds approximately to the procedure for signature *creation* with the private key d.

The RSA method

As shown in Figure 3.4, the system is set up by selecting two large random prime numbers p, q that are about the same size and a private key d. The two prime numbers p and q are selected in this case so that $n = pq$ cannot be factored based on current knowledge. To eliminate the possibility of attacks using specialised factorisation methods, p and q should be about the same size, but should not be too close together. In addition, the greatest common factor of the two numbers $p-1$ and $q-1$ should be small. For randomly selected large prime numbers, it is highly probable that the numbers will have these properties. As indicated in Section 3.2.2.4 the use of an RSA modulus $n = pq$ with at least 1900 bits will provide adequate security for about three years. To guarantee a higher level of security over long-term up to six years, it is generally recommended to increase the RSA modulus length up to at least 3000 bits (cf. [ETSI-119312(v1.4.2)] and [SOGISV1.2]).

The exponent d is selected randomly[11] so that it is relatively prime to the group order of $(\mathbb{Z}/n\mathbb{Z})^*$, the "prime residue class group" modulo n.

The group $(\mathbb{Z}/n\mathbb{Z})^*$ consists of all elements that are relatively prime to n. The group operation is multiplication followed by reduction modulo n, and the group order is calculated using the "Euler φ

[10] While the calculation of the private exponent d from the public exponent e using deterministic polynomial time reductions is equivalent to the factorization of $n = pq$ [May05], it is not yet clear if factorization is absolutely necessary to solve the so-called "RSA problem" (calculation of the e-th roots modulo n). It has not been proven theoretically yet that it is impossible to calculate the e-th roots without solving the factorization problem first.

[11] If the private key d is selected at random or is calculated as the inverse element of a specific public key e, then the probability of obtaining a small, and therefore insecure (cf. [BoDu99]), private key d is very low.

function". Since in our case $n = pq$ is composed of exactly two prime numbers, $\varphi(n) = (p-1)(q-1)$. Due to "Euler's theorem", which states that one obtains the neutral element in every finite abelian group by exponentiation with the group order, $a^{\varphi(n)} \equiv 1 \pmod{n}$ for every $a \in (\mathbb{Z}/n\mathbb{Z})^*$.

The public key consists of the modulus $n = pq$ and the exponent e for which $ed \equiv 1 \pmod{\varphi(n)}$ is true. A signature $s \equiv m^d \pmod{n}$ is generated by modular exponentiation of the message m with the private key d. To verify such a signature, the signature s is raised to the power of the public key e.

The signature is valid if and only if

$$s^e \equiv \left(m^d\right)^e \equiv m^{ed} \equiv m^{1 \bmod \varphi(n)} \equiv m \pmod{n}.$$

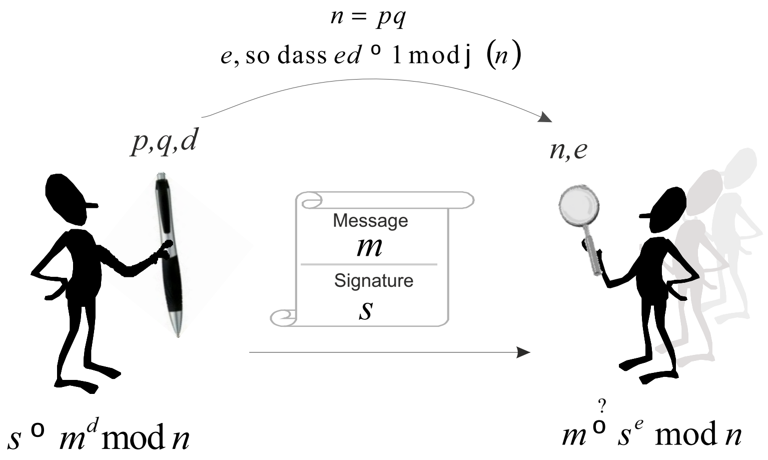

$n = pq$
e, so dass ed º 1 mod j (n)
p,q,d
n,e
Message
m
Signature
s
s º m^dmod n
$m \overset{?}{º} s^e$ mod n

Figure 3.4: The RSA signature method

EXAMPLE 3.2.3-1

The RSA signature method will be explained based on an example with small parameters: let $p = 11$ and $q = 17$. Then $n = 11 \cdot 17 = 187$ and $\varphi(n) = (p-1)(q-1) = 10 \cdot 16 = 160$. We use the public key $e = 3$ and determine the private key $d \equiv e^{-1} \pmod{\varphi(n)}$ using the extended Euclidean algorithm described in more detail in Section 3.2.2.3

Zeile	r_i	=	$s_i \cdot a$	+	$t_i \cdot b$	q_i
1	160	=	$1 \cdot 160$	+	$0 \cdot 3$	
2	3	=	$0 \cdot 160$	+	$1 \cdot 3$	$q_2 = \lfloor 160/3 \rfloor = 53$
3	$(160 - 53 \cdot 3)$	=	$(1 - 53 \cdot 0) \cdot 160$	+	$(0 - 53 \cdot 1) \cdot 3$	
	1	=	$1 \cdot 160$	+	$(-53) \cdot 3$	$q_3 = \lfloor 3/1 \rfloor = 3$
4	0					

Table 3.3: Inversion of $3 \pmod{160}$ using the EEA

As can be seen in Table 3.3, we obtain $d \equiv -53 \equiv 107 \pmod{160}$ and can easily verify that $e \cdot d \equiv 3 \cdot 107 \equiv 321 \equiv 1 \pmod{160}$. To sign a message $m = 10$, one calculates $s \equiv m^d \equiv 10^{107} \equiv 54 \pmod{187}$. To verify the message, one calculates $s^e \equiv 54^3 \equiv 10 \pmod{187}$ and determine that the signature is valid.

Using the RSA method naively for digital signatures as shown here poses a series of risks, though.

On the one hand, an attacker can select any signature s and determine the message $m \equiv s^e \pmod{n}$ belonging to this signature through exponentiation with the public exponent e. Although the ”message” m created in this manner probably will not contain any readable text, one should avoid such ”existential forgeries” in order to maintain non-repudiation.

In addition, the RSA method is ”multiplicative”. From two messages m_1, m_2 and the corresponding signatures $s_1 \equiv m_1^d \pmod n$ and $s_2 \equiv m_2^d \pmod n$, one can obtain the signature $s \equiv s_1 \cdot s_2 \equiv m_1^d \cdot m_2^d \equiv (m_1 \cdot m_2)^d \equiv m^d \pmod n$ for the message $m \equiv m_1 \cdot m_2 \pmod n$.

To exclude the possibility of both of these attacks, hash functions (cf. Section 3.2.4) are usually used in practice to prevent existential forgery and to destroy the multiplicative structure when using the RSA method. In particular, the widely used PKCS #1 format discussed in the following uses a hash function and more or less sophisticated padding mechanisms.

In addition, an attacker could attempt to foist his own public key under a false name on the person verifying the signature so that this person positively verifies the signatures forged by the attacker. Certificates bearing the public key and the name of the owner of the key and signed by a trusted party are used to prevent such attacks from being successful.

Signature formats for RSA

Various signature formats have been standardised for the RSA method. They differ primarily in how they fill in the hash value with padding characters before application of the RSA algorithm.

PKCS #1

The PKCS #1 standard specifies an RSA-based signature format for electronic signatures that is widely used in practical applications in which the signature is separated from the message.

In addition to the application of the RSA algorithm described above, this standard specifies in particular how the message m is mapped to the actual signed value $\overline{m} \in (\mathbb{Z}/n\mathbb{Z})^*$ through exponentiation with the private key d. Due to the security aspects mentioned above and to be able to sign messages of any length, it is not the message itself, but its cryptographic hash value that is used as the basis for encoding.

The current version of PKCS #1 is Version 2.1 [RFC8017] and contains two variants for encoding the hash value of the message:

- *PKCS #1 Version 2.2*

  is a simple encoding variant that is commonly used in practice and which was the subject of [PKCS1(v2.2), RFC8017].

- *Probabilistic signature scheme*

  is an encoding variant based on [BeRo96, BeRo98] available for the first time in PKCS #1 Version 2.1 for which a proof of its security (in the so-called Random Oracle Model) exists. According to the proof, the forging of a signature while executing an adaptive attack is equivalent to solving the RSA problem (i.e. calculating the e-th roots modulo n).

PKCS #1 Version 2.2

In the encoding variant already specified in [PKCS1(v2.2), RFC8017], one obtains the value $\overline{m} \in \mathbb{Z}$ from the octet string $\overline{M}$ of length $l_{\overline{M}} = \lceil (l_n - 1)/8 \rceil$ formed in the following manner, where l_n is the bit length of the RSA modulus n:

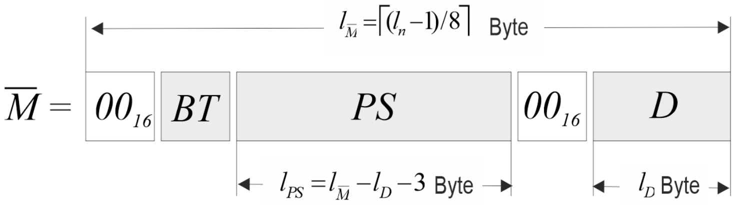

$l_{\overline{M}} = \lceil (l_n - 1)/8 \rceil$ Byte
$\overline{M} =$
00_{16}
BT
PS
00_{16}
D
$l_{PS} = l_{\overline{M}} - l_D - 3$ Byte
l_D Byte

Figure 3.5: PKCS #1 Version 1.5 encoding

In this case, the parts of $\overline{M}$ are given in the following manner:

- BT – Block Type

  This is set to the value $BT = 01_{16}$ for electronic signatures. In addition, PKCS #1 also defines the two block types $BT = 00_{16}$ and $BT = 02_{16}$. Block type 02_{16} is used for the purpose of encryption. Block type 00_{16} does not use any padding at all, which means the attacks mentioned in [DeOd85, Misa98, CNS99, CHJ99, BCCN01] are possible. For this reason, this block type should not be used.

- PS – Padding String

  This fills in the $l_{PS} = l_{\overline{M}} - l_D - 3$ bytes available with padding characters that depend on the block type . For block type 01_{16}, the value of the padding character is FF_{16}.

- D – Digest

  This is the DER[12] encoding of the `DigestInfo` – a sequence consisting of the hash algorithm and hash value.

  The length l_D of the DER-encoded `DigestInfo` depends on the hash function used (cf. [RFC8017] and Table 3.4, page 82).

Probabilistic Signature Scheme

In this encoding variant, which was specified for the first time in [RFC8017], the padding characters used are not defined in advance like in [PKCS1(v2.2)], and the padding is calculated based on the message and a random value instead.

[12] While PKCS #1 Version 2.2 still required BER encoding, DER encoding needs to be used in Version 2.0, which means that – at least theoretically – interoperability problems could arise between the various versions for certain hash functions.

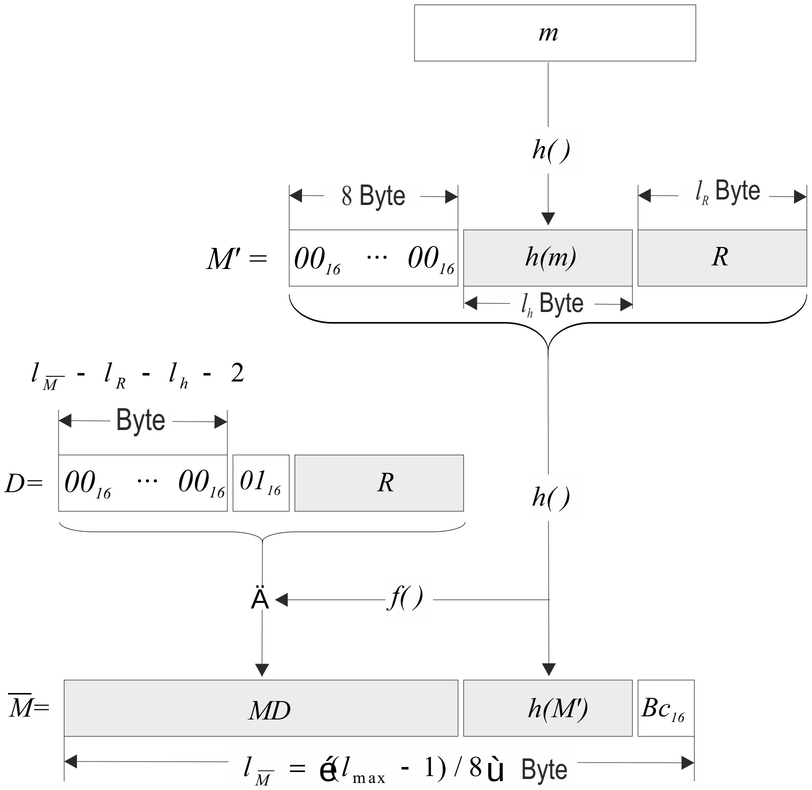


Figure 3.6: PSS encoding (PKCS #1 Version 2.2)

As can be seen in Figure 3.6, the padding characters MD are calculated by exclusively OR-ing (XOR) the value D , which basically consists of the random value R, with a mask $f(h(M'))$ derived from the message itself and the random value R.

The function $f()$ used for the calculation of this mask is easy to construct from the hash function $h()$ as explained in [RFC8017, appenidx B.2.1]. In this case, a sequential index i is appended to the input value S, and this string $S|i$ is then passed to the hash function $h()$.

The octet string $\overline{M}$, which is subsequently used to generate the actual signature (through exponentiation with the private key d), consists of the following three parts:

- MD

  This is the masked value of D obtained by XORing it with $f(h(M'))$. In this case, D is composed, as can be seen in Figure 3.6, of a sequence of $l_{\overline{M}} - l_R - l_h - 2$ null bytes, a 01_{16} byte, and the random value R of length l_R.

- $h(M')$

  This is the hash value of M', where M' consists of 8 null bytes, the hash value $h(m)$ of the message m, and the l_R random bytes R.

- BC_{16}

  This is a constant value that terminates the signature that was introduced for reasons of compatibility to the Rabin-Williams variants in [IEEE-P1363] and [ISO9796-2].

This method has the advantage that even a particular powerful[13] attacker will only actually be able to forge signatures if he is also able to solve the RSA problem [BeRo96, BeRo98].

[13] When referring to the power of an attacker against a signature system, one differentiates between two basic types of attackers: passive and active attackers. An active attacker is able to calculate signatures for the selected messages. In this case, there is also a difference between an attacker who needs to select all the messages he wants to have signed

Additional signature formats for RSA

There are other standards for RSA in addition to the signature formats specified in the PKCS #1 standard, but these are not widely used in practical applications.

The [ISO9796-2] standard allows part of the message, or in the case of correspondingly short messages, the entire message, to be reconstructed from the signature. RSA signatures according to [ANSI-X9.31] are structured similar to [PKCS1(v2.2)] signatures, although different padding characters are used and the hash function used for signature generation encodes differently.

3.2.3.2 Signatures based on the Discrete Logarithm Problem

The first signature method that was based on the discrete logarithm problem (DLP) was described in 1984 by Taher ElGamal [ElGa85]. After that, Claus-Peter Schnorr [Schn89, Schn91] suggested that essential parts of the signature calculation should not be performed in the complete group $\mathbb{F}_p^*$, but only in a subgroup of the order q with $q|(p-1)$ instead, which results in significantly faster calculations and smaller signatures. The Digital Signature Algorithm (DSA) based on this approach [FIPS186-4] was standardised in 1994 by NIST. The Elliptic Curve Digital Signature Algorithm (ECDSA) [ANSI-X9.62] is the variant in the point group of an elliptic curve that was standardized by ANSI.

before starting his attack and an attacker who can adapt the selection of messages he wants to have signed in the course of the attack. The latter case is referred to as an ”adaptive active attack” (adaptively chosen message attack). Ideally, a signature method will also be immune to this type of attacks.

ElGamal signature

The signature algorithm from ElGamal [ElGa85] has the same system design as the Diffie-Hellman procedure for key exchange [DiHe76]. In this case, one selects a large prime number p and a generator g of the multiplicative group $\mathbb{F}_p^*$ of the finite field $\mathbb{F}_p$ so that it is impossible in practice to calculate discrete logarithms in $\mathbb{F}_p^*$.

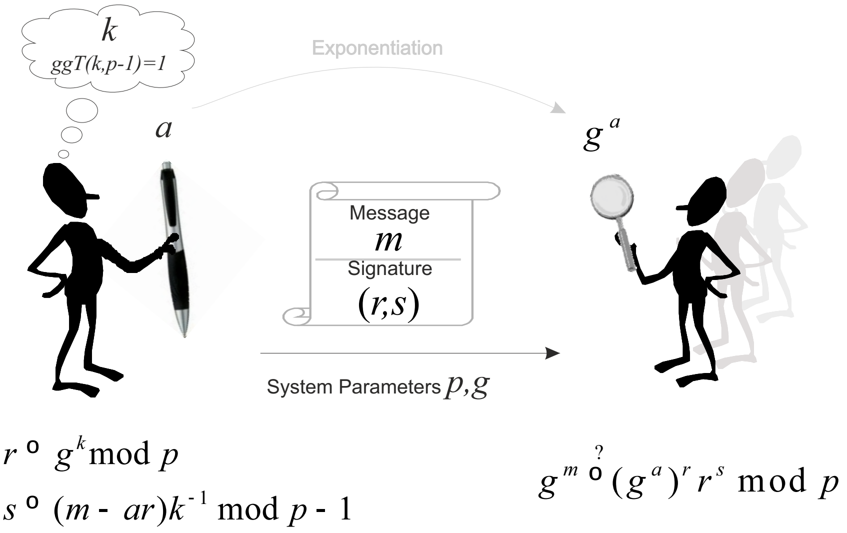


Figure 3.7: The ElGamal signature method

As can be seen in Figure 3.7, the private key consists of a random number a with $1 \leq a \leq p-2$. The corresponding public key g^a is calculated by exponentiation.

To generate a signature, one selects a random number k that is invertible modulo $p-1$ and generate the signature (r, s) for a message m through one exponentiation modulo p and one calculation modulo $p-1$. In this case, $r \equiv g^k \pmod{p}$ and $s \equiv (m - ar)k^{-1} \pmod{p-1}$. To verify the signature calculated, one calculates $(g^a)^r\, r^s \pmod{p}$ and compare this value to $g^m \pmod{p}$. This signature method is correct because

$$(g^a)^r\, r^s \equiv g^{ar} g^{k(m-ar)k^{-1}} \equiv g^m \pmod{p}. \tag{3.4}$$

To avoid modular inversion in the calculation of $k^{-1} \pmod{p-1}$ when generating the signature, it is also possible to use a variant of this method [AMV90] in which the private key a is invertible modulo $p-1$, but not the random number k. Similar variations of this method are described in [HMP94, HMP95].

EXAMPLE 3.2.3-2

We demonstrate the ElGamal signature method using a brief example: We select $p = 11$ and the generator $g = 2$ of the multiplicative group $\mathbb{F}_{11}^*$. We also select $a = 8$ as the private key and obtain $g^a \equiv 2^8 \equiv 3 \pmod{11}$ as the public key. To sign the message $m = 5$, we select a random $k = 9$ that fulfils the requirement $\gcd(k, p-1) = \gcd(9, 10) = 1$. We can now perform the calculation of the signature (r, s). We obtain $r \equiv g^k \equiv 2^9 \equiv 6 \pmod{11}$ and must calculate $s \equiv (m - ar)k^{-1} \equiv (5 - 8 \cdot 6) \cdot 9^{-1} \pmod{p-1}$. The element $9 \pmod{10}$ is its own inverse; $9 \cdot 9 \equiv 1 \pmod{10}$ and we obtain $s \equiv (5 - 8 \cdot 6) \cdot 9 \equiv 3 \pmod{10}$. The signature for the message $m = 5$ is therefore the pair $(6, 3)$. To verify the signature, it is necessary then to calculate $(g^a)^r\, r^s \equiv 3^6 \cdot 6^3 \equiv 10 \pmod{11}$ and compare the result to the value $2^5 \equiv 10 \pmod{11}$.

Since the methods would be completely broken if an attacker were able to calculate the private key from the public, the parameters must be selected so that the solution of the discrete logartihm problem is impossible in practice. According to (cf. [ETSI-119312(v1.4.2)] and [SOGISV1.2]) referring to the DSA method explained in more detail below, the selection of a prime number p with 2048 bits and g with 224 or 256 bits will provide adequate security from 2019 until 2022 for three years - from 2022 until 2025 it is necessary to use a 3072 bit moduli and g with 256 bits.

The calculation of $r \equiv g^k \pmod p$ for the ElGamal signature is independent from the message and can be performed in a preliminary calculation phase. However, it must be ensured that that the random number k does not become public because otherwise it would be easy to calculate the private key $a \equiv (m - sk)r^{-1} \pmod{p-1}$ from a single signature (r, s). It must be noted here that even only partial knowledge of the random numbers used can cause problems [NgSh02]. Such attacks are especially effective when short private keys and random numbers are used that are also used in actual applications in naive implementations (cf. [Nguy04]).

Using the random number more than once would also be problematic because then it would be highly likely that an attacker will be able to determine the random number k from the two messages m_1 and m_2 and the corresponding signatures (r, s_1) and (r, s_2).

To do this, consider the system of equations

$$\begin{array}{rcll} s_1 & \equiv & (m_1 - ar)k^{-1} & \pmod{p-1} \\ s_2 & \equiv & (m_2 - ar)k^{-1} & \pmod{p-1} \end{array}$$

from which we obtain

$$s_1 - s_2 \equiv (m_1 - ar)k^{-1} - (m_2 - ar)k^{-1} \equiv (m_1 - m_2)k^{-1} \pmod{p-1}.$$

If the value $s_1 - s_2$ modulo $p - 1$ is invertible, then it is possible to calculate

$$k \equiv (m_1 - m_2)(s_1 - s_2)^{-1} \pmod{p-1}$$

and determine the private key a as described above.

Like with the RSA method discussed in Section 3.2.3.1, the existential forgery of a signature is also possible when using the ElGamal signature method as it is described here (cf. [ElGa85] and [Buch10, chapter 11.4.6]). For this reason, this method should only be used to sign the hash values of messages and not the messages themselves. In addition, it is important to check when verifying the signature if $1 \leq r < p$ is true. If this is not checked, then an attacker could generate a ”valid” signature (r', s') for any message m' from a valid signature (r, s) for a message m as shown in [Blei96]. It was also shown in [Blei96] that it is possible to forge signatures for any message when the generator g only consists of small prime numbers and is a factor of $p - 1$. For this reason, $g = 2$ in particular should never be used.

Schnorr signature

The Schnorr method for identification and signatures was presented for the first time at the rump session of EUROCRYPT 1989. The detailed paper appeared a few weeks later in the Proceedings of the CRYPTO [Schn89]. It combines the ideas of [ElGa85], [FiSh86] and [CEG87] to construct an efficient signature system based on the discrete logarithm.

The essential idea is not only to use the structure of the multiplicative group $\mathbb{F}_p^*$, but also to exploit the fact that subgroups exist in this group. Which subgroups exist in the group $\mathbb{F}_p^*$ depends on the factorisation of $p - 1$. Since p is a large prime number, $p - 1 = 2 \cdot r$ has at least two factors and therefore at least two subgroups: one subgroup of order 2 and another subgroup of order r.

In the Schnorr signature method, one now selects the prime number p so that the number $p - 1$ has a prime factor q with 160 bits, for example, and g generates a subgroup of $\mathbb{F}_p^*$ of order q. While some of the calculations still need to be performed modulo p, one is still moving within the subgroup created by g so that all exponents can be reduced modulo q. The length of the signature (s, t) now only depends on the bit length of the hash value and the size of the subgroup q. This generates relatively compact signatures.

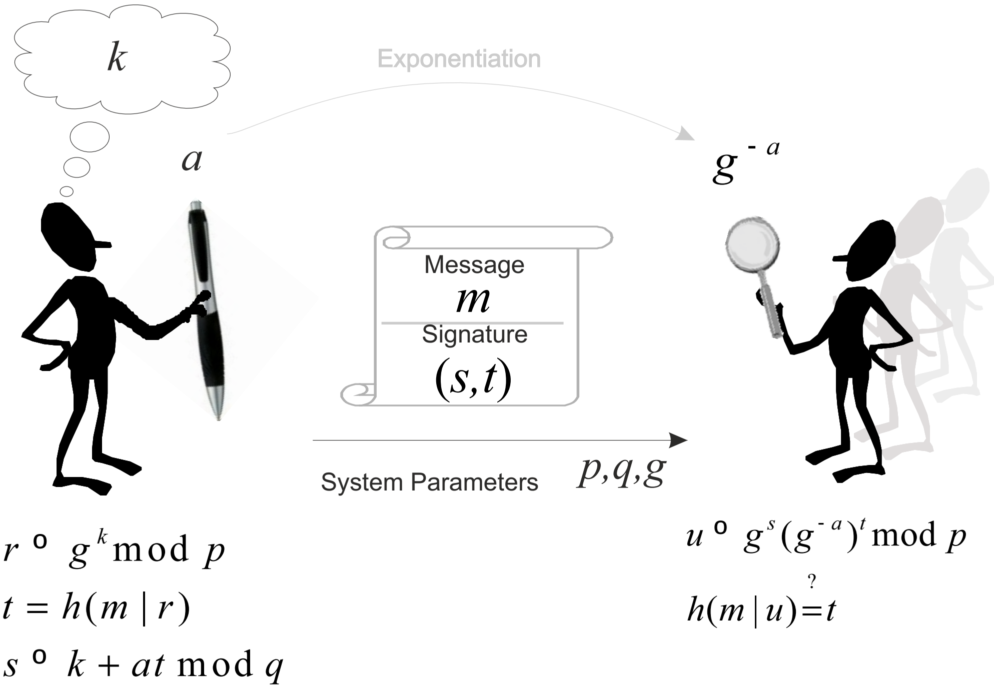

k
a
Exponentiation
g^{-a}
Message
m
Signature
(s,t)
System Parameters
p,q,g
$r \,^{\circ}\, g^{k}\mathrm{mod}\ p$
$t = h(m\,|\,r)$
$s \,^{\circ}\, k + at\ \mathrm{mod}\ q$
$u \,^{\circ}\, g^{s}(g^{-a})^{t}\mathrm{mod}\ p$
$h(m\,|\,u)\overset{?}{=}t$

Figure 3.8: The Schnorr signature method

Figure 3.8 shows that the public key is $g^{-a} \pmod p$ just like in the ElGamal variant in [AMV90].

To generate a signature, one selects – just like in the ElGamal method discussed above – a random number k and calculate $r \equiv g^k \pmod p$. While the value r is part of the signature in the ElGamal method, it is converted to an octet string in the Schnorr method that is then appended to the message m and mapped using a hash function h (cf. Section 3.2.4) to the value $t = h(m|r)$. From the value t, the random number k, and the private key a one finally calculates $s \equiv k + at \pmod q$. The signature consists of the two values (s, t).

To verify the signature, one calculates the value $u \equiv g^s \left(g^{-a}\right)^t \pmod p$ and check if $h(m|u) = h(m|t)$ is true. This required check of the signatures is based on the fact that

$$u \equiv g^s \left(g^{-a}\right)^t \equiv g^{k+at}g^{-at} \equiv g^k \equiv r \pmod p.$$

EXAMPLE 3.2.3-3

To illustrate the Schnorr signature method, we select the prime numbers $q = 11$ and $p = 23, p-1 = 2 \cdot q$ as well as a generator $g = 2$ of the subgroup with order q. We use $a = 5$ as the private key and then obtain the public key $g^{-a} \equiv 2^{-5} \equiv 2^6 \equiv 18 \pmod{23}$. To generate a signature for a message m, we select the random number $k = 7$ and calculate $r \equiv g^k \equiv 2^7 \equiv 13 \pmod p$. We now calculate the hash value $h(m|r)$ and obtain, for example, the value $t = h(m|r) = h(m|13) = 4$, and therefore $s \equiv k + at \equiv 7 + 5 \cdot 4 \equiv 5 \pmod q$. The signature consists of the two values $(s, t) = (5, 4)$. To verify the signature, we calculate $u \equiv g^s \left(g^{-a}\right)^t \equiv 2^5 \cdot 18^4 \equiv 9 \cdot 4 \equiv 13 \pmod p$ and we finally see that $t = h(m|13) = 4$.

The security of the method is based on two different, but closely related DL problems: on the one hand, the DL problem modulo p that can be solved using the subexponential number field sieve [Gord93, Webe96, JoLe03], and on the other hand the DL problem in the subgroup of order q that can be solved using the generic ρ algorithm [Poll78].

The Schnorr signature method also possesses the interesting property that – in contrast to the Digital Signature Algorithm (DSA) – the order of the subgroup q is not needed for the purpose of verification and can therefore be part of the private key. This allows to construct variants of the method that are also based on the factorization problem [HuMe00, Hueh01] and allow particularly efficient signature generation. Schnorr signatures are not commonly used because of former patent issues (the U.S. Patent 4,995,082 expired in February 2008).

In [ETSI-119312(v1.4.2)] and [SOGISV1.2]) only the (EC-SDSA-opt) version from the EC-XDSA Schnorr variants, defined in [ISO14888-3], is selected, because the optimized version only needs minimal data transfer for smart cards.

Digital Signature Algorithm (DSA)

The Digital Signature Algorithm [FIPS186-4] is a variant of the Schnorr method that was standardised in 1994 by NIST. In addition to the precise specification of how to select p, q, and g and how to use the "Secure Hash Algorithm" [FIPS180-4] as a hash function, there are also other minor differences in the design of the system as well as in the generation and verification of the signatures.

For the order of the subgroup, one selects a prime number q with $2^{159} < q < 2^{160}$ and a prime number p with (up to) 1024 bits so that $q|(p-1)$. To obtain a generator g of the subgroup of order q, one selects a random helper number h with $1 < h < p-1$ and calculate $g = h^{(p-1)/q} \pmod p$ until one obtains a $g \not\equiv 1 \pmod p$. This ensures that g generates the subgroup of order q. The private key is a random value a with $0 < a < q$. As in the original ElGamal method, the public key is $g^a \pmod p$. However, the private key a is not inverted during the calculation of the public key when setting up the system, and it is necessary instead to invert the random number k every time a signature is generated.

As indicated in Figure 3.9, one selects a random number k with $0 < k < q$ and calculate the value $r \equiv \left(g^k \pmod p\right) \pmod q$ to generate the signature. To calculate the second value s, one inverts the random number k modulo q and obtain

$$s \equiv k^{-1}(h(m) + ar) \pmod q, \tag{3.5}$$

where $h(m)$ is the hash value of the message m. To verify the signature, one calculates $w \equiv s^{-1} \pmod q$, $u_1 \equiv h(m) \cdot w \pmod q$, $u_2 \equiv r \cdot w \pmod q$ and then $r' \equiv (g^{u_1} (g^a)^{u_2} \pmod p) \pmod q$. The signature is accepted as valid if and only if $r' \equiv r \pmod q$.

This verification requirement is due to the fact that

$$g^{u_1} (g^a)^{u_2} \equiv g^{h(m)s^{-1}} g^{ars^{-1}} \equiv g^{(h(m)+ar)s^{-1}} \equiv g^k \equiv r \pmod p. \tag{3.6}$$

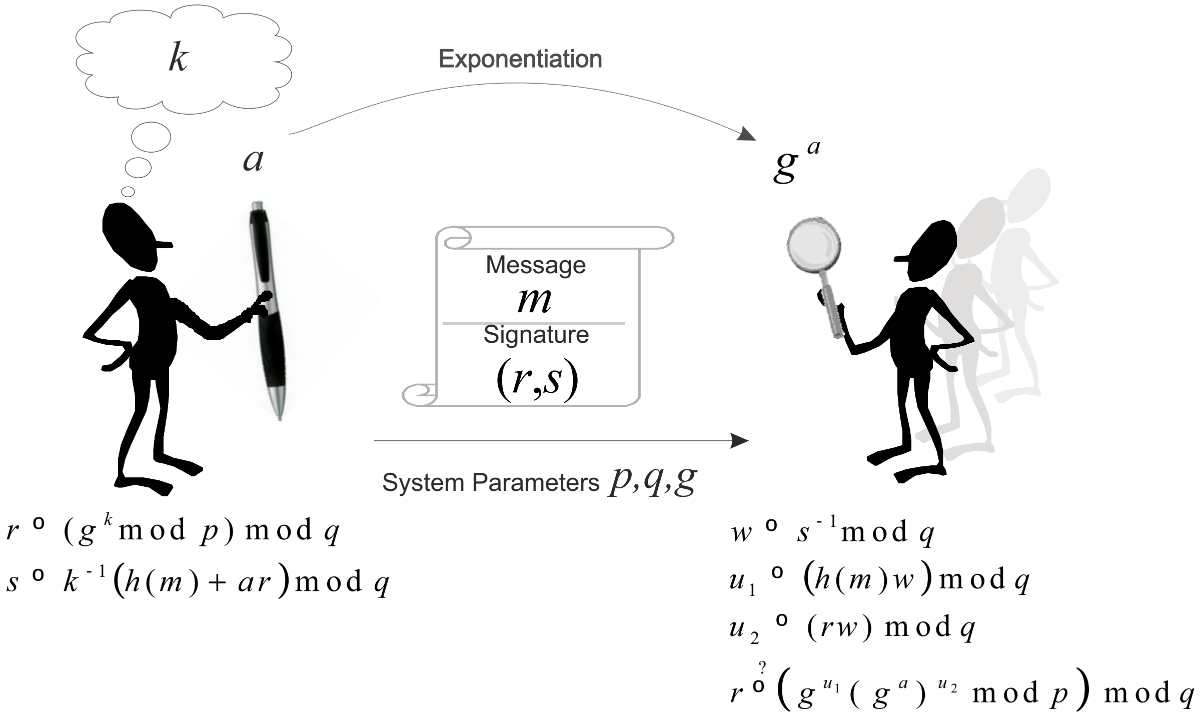

k
Exponentiation
a
g^a
Message
m
Signature
(r,s)
System Parameters p,q,g
$r \equiv (g^k \bmod p) \bmod q$
$s \equiv k^{-1}(h(m) + ar) \bmod q$
$w \equiv s^{-1} \bmod q$
$u_1 \equiv (h(m)w) \bmod q$
$u_2 \equiv (rw) \bmod q$
$r \overset{?}{\equiv} \left(g^{u_1}(g^a)^{u_2} \bmod p\right) \bmod q$

Figure 3.9: The Digital Signature Algorithm

EXAMPLE 3.2.3-4
As in Example 3, we select $p = 23$, $q = 11$, $g = 2$, and $a = 5$. In this case, we obtain the public key $g^a \equiv 2^5 \equiv 9 \pmod{23}$. To generate a signature, we select the random number $k = 7$ and calculate $r \equiv \left(g^k \equiv 13 \pmod{p}\right) \equiv 2 \pmod{q}$. The inverse of the random number k modulo q is $k^{-1} \equiv 8 \pmod{q}$, and using a hash value $h(m) = 4$ as an example, we then obtain the value $s \equiv k^{-1}(h(m)+ar) \equiv 8 \cdot (4+5 \cdot 2) \equiv 2 \pmod{q}$. The signature consists of the values $(r, s) = (2, 2)$. To verify the signature, one calculates $w \equiv s^{-1} \equiv 2^{-1} \equiv 6 \pmod{q}$, $u_1 \equiv h(m)w \equiv 4 \cdot 6 \equiv 2 \pmod{q}$, $u_2 \equiv rw \equiv 2 \cdot 6 \equiv 1 \pmod{q}$ and $r' \equiv g^{u_1}(g^a)^{u_2} \equiv 2^2 \cdot 9^1 \pmod{p} \equiv 36 \pmod{p} \equiv 13 \pmod{p} \equiv 2 \pmod{q}$. Since $r' \equiv r \pmod{q}$, the signature is valid.

The security aspects mentioned above for ElGamal and Schnorr signatures also need to be considered when using the digital signature algorithm. In particular, no part of the random number k should be made public [NgSh02]. In addition, it was shown in [Vaud96] that an attacker can select parameters p, q, and g when setting up the system specifically for the purpose of forging signatures. For this reason, it is important to check if the procedures specified in [FIPS186-4] are being followed when generating the system parameters.

There are also some other variants of the procedure in addition to the digital signature algorithm standardised by NIST:

- Nyberg and Rueppel [NyRu94] suggested a variation of DSA in which part of the message can be recovered from the signature. This procedure is part of the [IEEE-P1363] standard.

- The ”Korean Certificate-based Digital Signature Algorithm (KCDSA)” [KCDSA99, LiLe98] is a variation of DSA in which the operation applied to the hash value of the message, the private key a, and the random number k in (3.5) is not multiplication modulo q but the exclusive OR operation (XOR) instead. In addition, the hash value of the certificate for the public key g^a is also used in the generation of the signature.

- In the Russian analogue to DSA, GOST 34.10 [MNP96], the signature equation $s \equiv ra + kh(m) \pmod{q}$ is used instead of (3.5), and the subgroups used must have a size of 256 bits.

According to (cf. [ETSI-119312(v1.4.2)] and [SOGISV1.2]), the selection of a prime number p with 2048 bits and g with 224 or 256 bits will provide adequate security from 2019 until 2022 for three years - from 2022 until 2025 it is necessary to use a 3072 bit moduli and g with 256 bits.

Elliptic Curve Digital Signature Algorithm (ECDSA)

The ”Elliptic Curve Digital Signature Algorithm” (ECDSA) [JoMe99] is a DSA-analogue in which the group $\mathbb{F}_p^*$ is replaced by the point group of an elliptic curve. ECDSA was standardised in 1998 by ISO [ISO14888-3] and ANSI [ANSI-X9.62], added in the year 2000 to the [IEEE-P1363] standard, and has been accepted in the meantime by NIST [FIPS186-4].

As can be seen when comparing Figure 3.9 and Figure 3.10, the ECDSA method is a DSA-analogue, in which only difference is that modular multiplication in $\mathbb{F}_p^*$ is replaced by point addition on an elliptic curve.

The system parameters a, b, p define the elliptic curve $EC : y^2 \equiv x^3 + ax + b \pmod p$, in which the point G generates a cyclic subgroup of order q. The private key is a with $1 \leq a \leq q$, and the public key is the point $A = [a]G$.

To generate a signature, one selects a random number k and calculate the point
$(x, y) = [k]G$. The x-coordinate of this point is reduced modulo q to obtain the first part of the signature $r \equiv x \mod q$. The calculation of the second signature value s is then performed exactly like in the DSA method (3.5). Even the calculations of w, u_1, and u_2 for signature verification are identical to those in DSA. Only the calculation of the point $(x, y) = [u_1]G + [u_2]A$ is performed again in the point group of the elliptic curve.

The verification of the signature is correct because the following is true just like for (3.6):

$$[u_1]G + [u_2]A = [h(m)s^{-1}]G + [ars^{-1}]G = [(h(m) + ar)s^{-1}]G = [k]G. \tag{3.7}$$

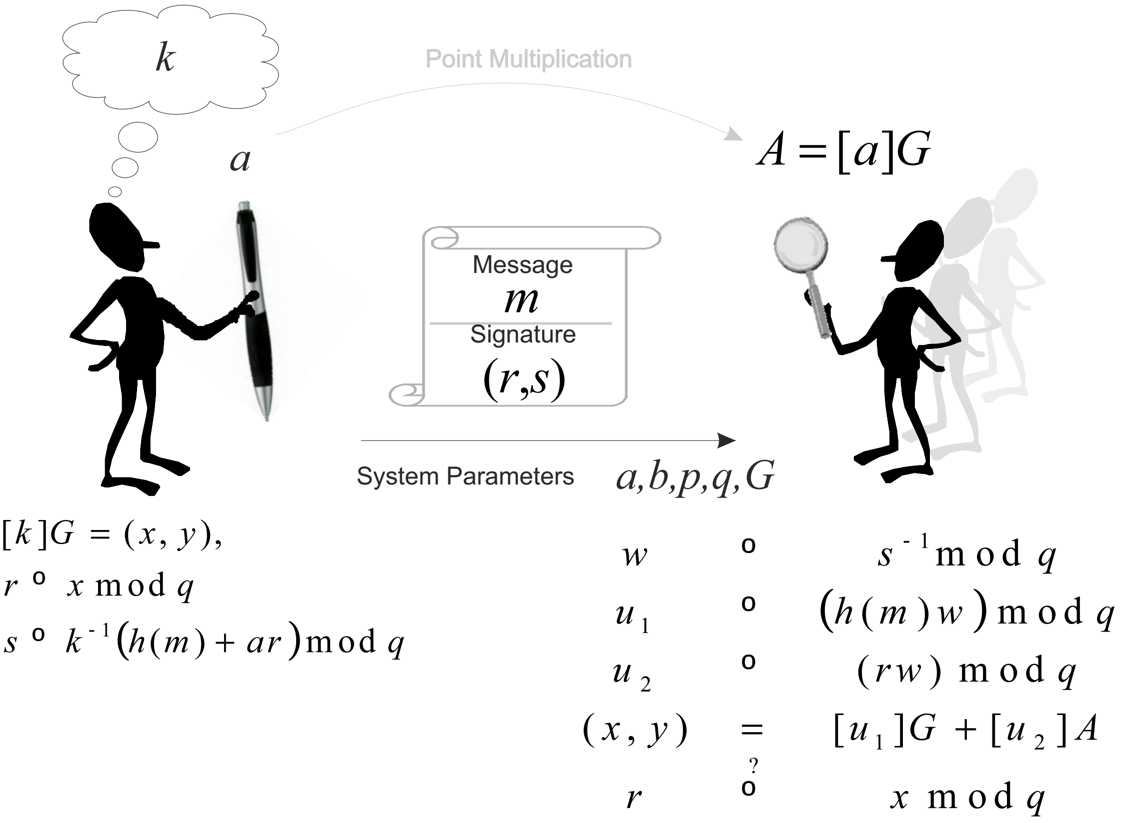


Figure 3.10: The Elliptic Curve Digital Signature Algorithm

In terms of the security of the ECDSA method, it is necessary to consider the same attacks as those targeting DSA. In particular, it is also necessary to avoid partial disclosure of the random number k [NgSh03].

Since the security of the method does not depend any more on the problem of the DLP in $\mathbb{F}_p^*$ but on the problem of the DLP in the point group of the elliptic curve instead, it is possible to use significantly shorter parameters (cf. Section 3.2.2.4). In the signature method in [BSL04], the so-called Weil pairings on an elliptic curve are used to permit especially short signatures.

According to (cf. [ETSI-119312(v1.4.2)] and [SOGISV1.2]), the following parameters p and g for EC-DSA and EC-SDSA-opt with the selection of p = g with 256 or 384 or 512 bits will provide adequate security up to six years from 2019 to 2025.

3.2.4 Hash Functions

A hash function $h : \{0,1\}^* \to \{0,1\}^{l_h}$ converts a message $m = \{0,1\}^*$ of any length to a hash value $h(m) = \{0,1\}^{l_h}$ with a fixed bit length l_h.

A hash function h suitable for use in cryptography must ensure that it is impossible in practice to calculate the message m from its hash value $h(m)$ (*one-way property*, cf. Section 3.2.2) and that it is impossible to find two messages m_1 and m_2 having the same hash value $h(m_1) = h(m_2)$ (*collision resistance*). For this reason, the hash value $h(m)$ of a message m is also referred to as the ”digital fingerprint” of the message.

In actual practice, the design principle of Merkle and Damgård [Damg89, Merk89] is usually used to construct hash functions. In this case, it is generated by iteratively applying a so-called *compression function* $f : \{0,1\}^{l_b} \times \{0,1\}^{l_h} \to \{0,1\}^{l_h}$ to the message m block-by-block.

As indicated in Figure 3.11, the message m of bit length l_m is divided during the *initialisation phase* into blocks of length l_b, and the last block m_k is filled in using a padding mechanism that also takes the length of the message into account. In addition, the value h_0 is specified during initialisation. The value $h_i = f(m_i, h_{i-1})$ is calculated during the *iteration phase*. The hash value $h(m)$ of the message m is then equal to the value h_k obtained once the last block m_k of the message has been processed.

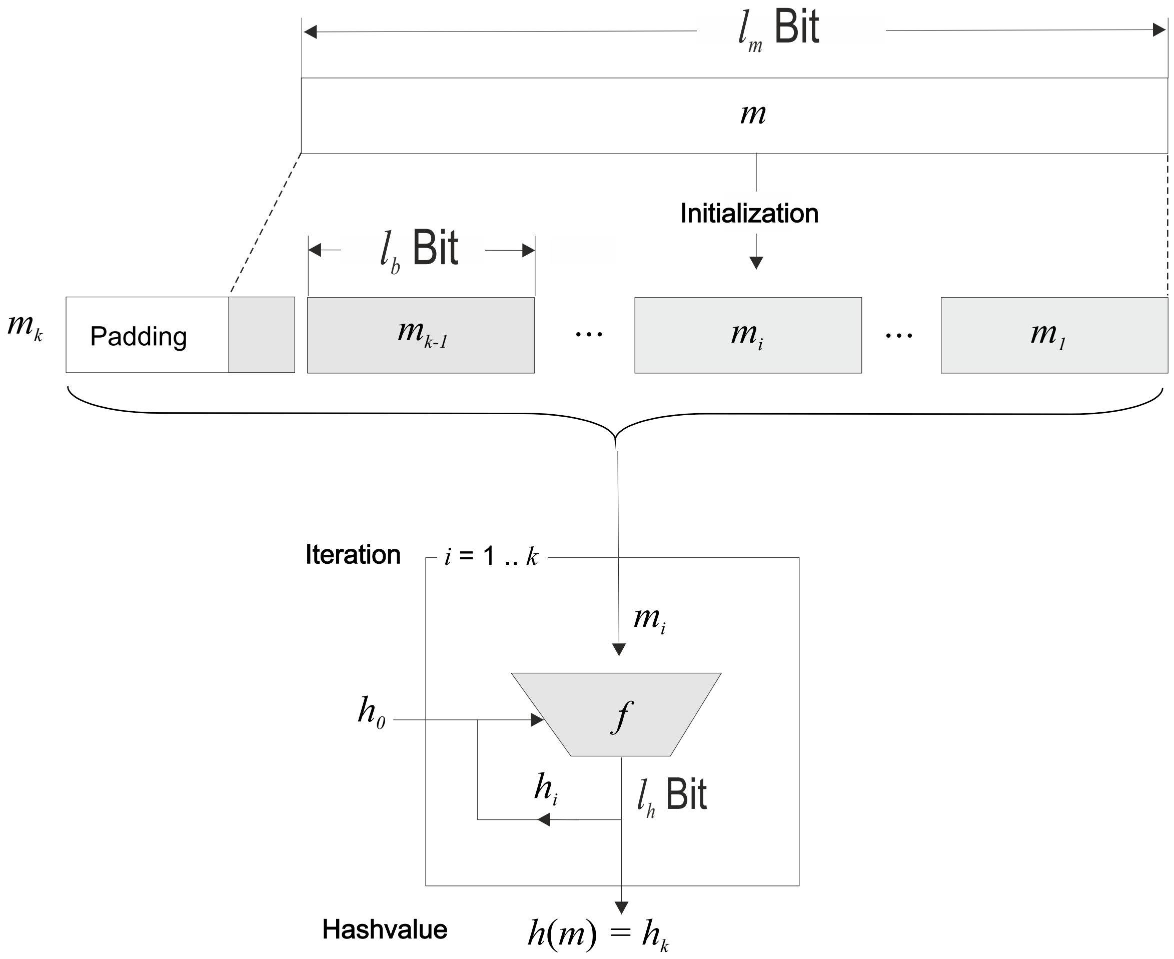


Figure 3.11: Structure principle of iterative hash functions

All hash functions in Table 3.4 follow this general structure.

Hash function	Length of the hash value (l_h bit)	Block length (l_b bit)	Length of the DER-encoded `DigestInfo` (l_D bytes)	Reference
SHA-256	256	512	51	[FIPS180-4]
SHA-384	384	1024	67	[FIPS180-4]
SHA-512	512	1024	83	[FIPS180-4]
SHA3-256	256	1152		[FIPS202]
SHA3-384	384	832		[FIPS202]
SHA3-512	512	574		[FIPS202]

Table 3.4: Popular hash functions

An obvious approach for generating collisions is to randomly select messages m and store them together with the corresponding hash value $h(m)$ until a collision is found. Due to the so-called ”birthday paradox” (cf. [Buch10, chapter 4.3]), the probability of finding a collision in this manner is greater than $\frac{1}{2}$ when one selects slightly more than $2^{l_h/2}$ random messages.

Since this attack requires the calculation and storage of 2^{80} hash values, only cryptographic hash values that are at least 256 bits long are used today for security reasons. For this reason alone, the use of MD5 ([RFC1321]) and SHA-1 ([FIPS180-4]) in the context of digital signatures presents problems because it only generates hash values with 128 or 160 bits. The attacks [WaYu05, Klim05, LeWe05] have demonstrated that MD5 is completely broken. There has already been an attack on the SHA-1 hash function, which is possibly the most commonly used hash function in practical applications, that enabled the attacker to generate random collisions with 2^{69} [WYY05a] operations. During the rump session of the CRYPTO 2005, one person even sketched out an attack that should only require 2^{63} operations [14]. The first SHA1 collision was announced on February 23, 2017, published in [F-Col17]. In the article [CP-Col17], a new technique to turn collision attacks into chosen-prefix collision attacks is presented. This chosen-prefix collision attack against SHA-1 has a complexity between $2^{66.9}$ and $2^{69.4}$.

As shown in [DaLu05, GIS05], it is possible for hash functions constructed according to the design principles of Merkle and Damgård (cf. Figure 3.11) to construct various useful messages in popular document formats (e.g. Postscript, TIFF, PDF, and Word 97) from a single random collision. For this reason, broken hash functions such as MD5 and SHA-1, for example, should not be used for the purpose of generating electronic signatures. In addition, strong consideration should be given to using one of its successors SHA-256, SHA-384, SHA-512 or SHA3-256, SHA3-384, SHA3-512.

3.2.5 Post-Quantum Cryptography

Post-quantum cryptographyis a relatively new field in cryptography which aims to develop algorithms that are resistant to attacks using large-scale quantum computers as well as classical hardware. The security of the algorithms currently used for digital signatures is essentially based on the assumed hardness of two mathematical problems: the integer factorization problem and the discrete logarithm problem. The relationship between quantum computers and digital signatures mainly stems from the fact that there exists an algorithm, called Shor algorithm [Shor97], that, given a large-scale quantum computer, can solve these two problems asymptotically much faster than currently known algorithms on classical hardware can do. This means that, in case quantum computers of sufficient size could be built, it would become feasible to break state of the art cryptographic algorithms used for digital signatures. Therefore, the need for long-term preservation of digital signatures is acknowledged among others in the eIDAS-Regulation [(EU)910/2014]. [ETSI-119511(v1.1.1)] and [ETSI-119512] describe the requirements and techniques of long-term preservation of digital signatures or general data using digital signature techniques (digital signature, electronic time stamps or Evidence Records) with suitable cryptographic algorithms [ETSI-119312(v1.4.2), SOGISV1.2].

Both theoretical and practical research on building quantum computers has been done for quite some time. In 2018, the BSI commissioned the study ”Development status quantum computer” for assessing the state of the art of the quantum computer technology [BSI-Q20]. The study concludes that currently short-term development leaps towards cryptographically relevant quantum computers are unlikely. The American National Security Agency (NSA) started work to facilitate the development of a cryptographically relevant quantum computer ([Schn15]).

Parallel to the progress in the development of quantum technologies a new field of work in cryptographic research evolved: post-quantum cryptography (also known as Quantum Safe Cryptography). Post quantum cryptography deals with the development and investigation of cryptographic mechanisms that cannot be broken even with quantum computers. These mechanisms are based on mathematical problems, for the solution of which neither efficient classical algorithms nor efficient quantum algorithms are known today. In 2016, the American National Institute for Standards and Technology (NIST launched a competition for standardising algorithms from post-quantum cryptography for future use. Besides algorithms for public-key encryption and key establishment, the competition also includes signature algorithms. As of 2022, the process is in its fourth round of standardisation.

Post-quantum signature algorithms are based on mathematical problems which are different from integer factorisation and the computation of discrete logarithms and for which no efficient algorithm using either classical hardware or quantum computers is known to date. In particular, the following concepts are the subject of current research:

[14] See `http://www.schneier.com/blog/archives/2005/08/new_cryptanalyt.html`.

Hash-based signature schemes are based on the security properties of cryptographic hash functions (cf. Section 2.4). As of today, these security properties would not be significantly affected, even if large-scale quantum computers could be built. Hash-based signature schemes are divided into two classes. Stateful hash-based signature schemes, first introduced by [Merk79], have the drawback that they require the signer to keep track of exactly which one-time signature keys have already been used. For this reason, for preventing misuse a proper implementation is crucial. In addition, the number of possible signatures needs to be fixed during the creation of the private key. Stateful hash-based signature schemes are not treated in the NIST process, since their underlying technology is well understood and they are already considered to be mature quantum-computer-resistant signature schemes. The stateful hash-based signature schemes [RFC8554] and [RFC8391] have already been standardized by the IETF. In 2019, (NIST) published a draft for a special publication which plans to adopt these standards. In contrast, stateless hash-based signature schemes do not require the signer to keep track of the signature keys already used, but are not as well understood yet.

Lattice-based signature schemes are based on the hardness of different problems in lattices, which are discrete structures in high-dimensional spaces. For instance, a lattice in R^n may be a subset of all points with integer coordinates. The lattice problems that the signature schemes are based on are considered to be computationally intractable both using classical hardware and large-scale quantum computers. One important lattice problem is the shortest vector problem (SVP), which consists in finding the shortest non-zero vector in a given lattice. The learning with errors (LWE) problem in turn is another important problem whose hardness is related to that of SVP. The search version of the LWE problem is defined as follows: For a vector $s \in (Z_q^n)$ and a distribution χ over Z_q, given m independent samples (a, b), where $a \in (Z_q^n)$ is chosen uniformly at random and $b \leq a$, $s > +e \pmod{\varphi(q)}$ with e chosen according to χ , find s. The decision LWE problem consists in deciding whether m given samples (a, b) were sampled using the procedure just mentioned or whether a and b were chosen uniformly at random from $Z_q^n \times (Z_q)$.

A third category considered in the NIST process are multivariate signature schemes. The security of these schemes is based on the hardness of solving large systems of multivariate polynomial equations, i. e. equations of polynomials in more than one variable. Whereas such systems of equations can be easily solved when the polynomials are linear, finding a solution is much harder for polynomials of degree greater than one. No efficient algorithm using either quantum computers or classical hardware is currently known which can solve this problem in general. Briefly put, multivariate signature schemes use the system of equations as the public key and a signature for a message is a solution to the equation system yielding the message. The second round of the NIST process contained four multivariate signature schemes (GeMSS, LUOV, MQDSS, Rainbow).

More detailed information and mathematical descriptions of several candidates are contained e.g. in the book [Bern09].

According to [ETSI-119312(v1.4.2)] and [SOGISV1.2], an RSA signature with a bitlength of the modulus n with at least 3000 bits will offer adequate security for three years. For a resistance of three years 3 years, at least 3000 bits must be used, while only 256 or 384 or 512 bits are adequate when using elliptic curves. ”The public exponent e shall be an odd positive integer such that $2^{16} < e < 2^{256}$.” pursuant to [ETSI-119312(v1.4.2)].

The third round of the NIST process [NIST-8413] selected the signature algorithms CRYSTALS-Dilithium [Dili18], Falcon [Falcon20] and SPHINCS+ [SPHINCS18] for standardisation and the encryption algorithm CRYSTALS-KYBER [kyber].

4 Formats of Signature, Seal, Timestamp and Evidence Record

4.1 Overview of Services and Standards for Digital Signatures, Timestamps and Evidence Records

In the area of services and standards related to digital signatures one may consider, on the one hand (horizontally, from left to right), the *life-cycle phases* of a digital signature, which comprise services for creation, validation and preservation of signatures, and, on the other hand (vertically, top down) the different formats for digital signatures (see Section 4.3), the API-standards for accessing the different services and finally the standards with respect to policy and security requirements for the different services.

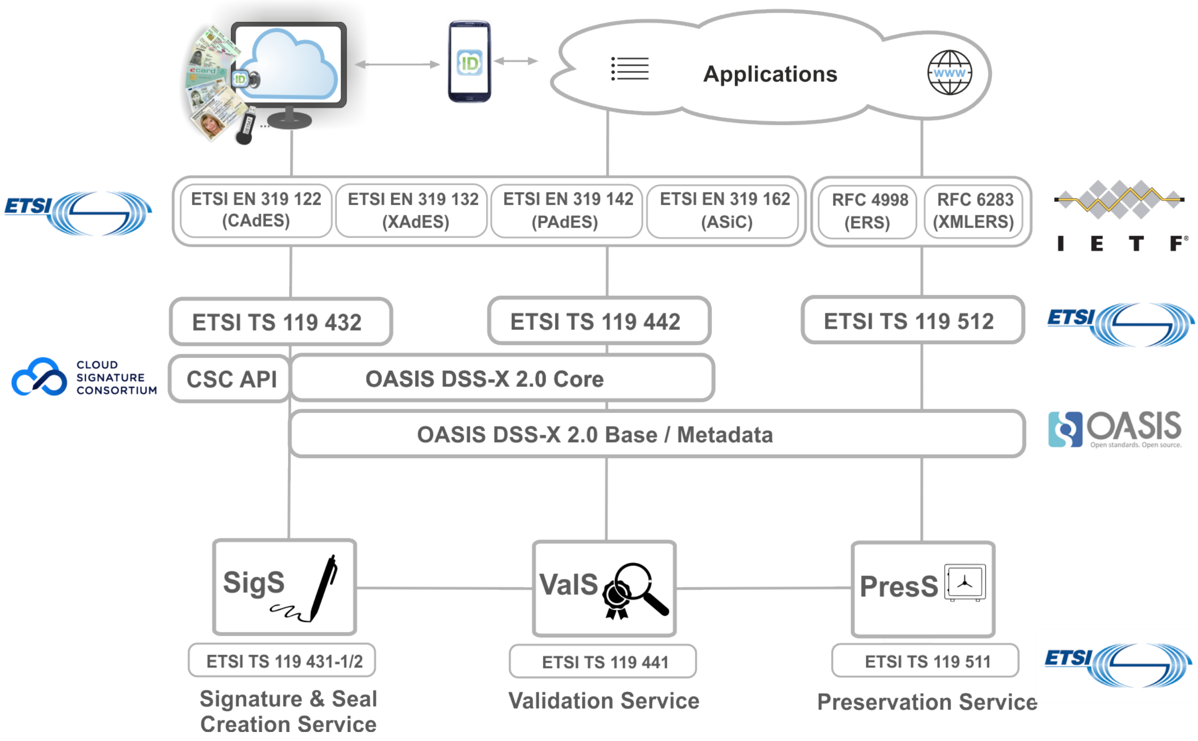


Figure 4.1: Overview of signature-related standards

4.2 Definition of Digital Signature

In the following text the term ”digital signature” covers:

- ”advanced electronic signatures” pursuant to [(EU)910/2014, Article 3(11)],
- ”qualified electronic signatures” pursuant to [(EU)910/2014, Article 3(12)],
- ”advanced electronic seals” pursuant to [(EU)910/2014, Article 3(26)],
- ”qualified electronic seals” pursuant to [(EU)910/2014, Article 3(27)]

Electronic seals means data in electronic form, which is attached to or logically associated with other data in electronic form to ensure the latter’s origin and integrity [(EU)910/2014, Article 3 (25)].

More details in

- electronic signatures (see Section 1.1.4.4),
- electronic seals (see Section 1.1.4.5),
- electronic time stamps (legal aspects in Section 1.1.4.6, technical aspects in section 4.4),
- Evidence Records (see Section 4.5)

4.3 Digital Signature Formats

When it comes to digital signature formats, one may distinguish between basic ”low-level signature formats”, which specify the details with respect to encoding and padding of the digital signature value produced with a particular digital signature algorithm (see [ETSI-119312(v1.4.2)] and Section 3.2), and ”high-level signature formats”, which specify the format of an advanced electronic signature (AdES) in the sense of [(EU)910/2014, Article 3 (11)], for example[1] such as:

- **CAdES** digital signatures [ETSI-319122-1(v1.2.1)] (see Section 4.3.1),
- **XAdES** digital signatures [ETSI-319132-1(v1.2.1)] (see Section 4.3.2),
- **PAdES** digital signatures [ETSI-319142-1(v1.1.1)] (see Section 4.3.3), and
- **ASiC** (Associated Signature Containers) [ETSI-319162-1(v1.1.1)] (see Section 4.3.4).

While low-level signature formats only allow checking the mathematical correctness of the digital signature value, high-level formats support the full validation process according to [ETSI-319102-1(v1.3.1)] and support advanced features, such as sequential, parallel or counter-signatures and timestamps.

The main signature formats, which are also required by [(EU)2015/1506] to be accepted by public sector bodies across Europe, are also described in the following subsections.

[1] In addition to the AdES-formats defined in the aforementioned European Norms developed within ETSI ESI, the ”baseline formats” [ETSI-103171(v2.1.1), ETSI-103173(v2.2.1), ETSI-103172(v2.2.2), ETSI-103174(v2.2.1)], which are based on the preceding versions of the corresponding specifications, are worth to be mentioned here, as they are referenced in [(EU)2015/1506]. Furthermore, there is ongoing standardisation work which aims at creating ”JAdES digital signatures” [ETSI-119182-1(v1.1.1)], which will be a digital signature format which is based on JSON Web Signatures according to [RFC7515] and hence JSON structures according to [RFC7159, RFC8259].

4.3.1 Cryptographic Message Syntax (CMS) and CAdES Digital Signatures

4.3.1.1 Cryptographic Message Syntax (CMS)

The CAdES-Format according to [ETSI-319122-1(v1.2.1), ETSI-319122-2(v1.1.1), ETSI-119122-3(v1.1.1), ETSI-101733(v2.2.1)] is based on the Cryptographic Message Syntax (CMS) according to [RFC5652]. The CMS-specification [RFC5652] aims at supporting different cryptographic use cases and hence defines different container formats[2], whereas the `SignedData` format is used for digitally signed data.

The heart of the CMS signature format consists of the ASN.1 structures `SignedData` and `SignerInfo`, as depicted in Figure 4.2.

The `SignedData` container primarily contains the data to be signed and one or more signatures in the form of the `SignerInfo` structure.

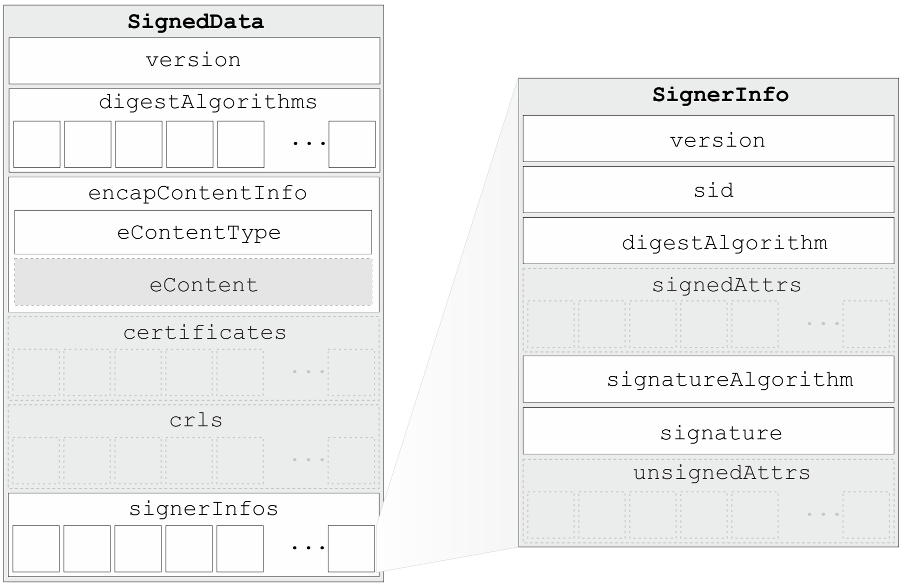


Figure 4.2: Structure of the `SignedData`-Container of CMS/PKCS #7

As shown in Figure 4.2, the `SignedData` container consists of the following parts:

- `version`
  is an integer value that ensures compatibility between the various versions of the CMS specifications [PKCS7(v1.5), RFC2315, RFC5652].

- `digestAlgorithms`
  lists the hash algorithms which appear later on in the set of `SignerInfo` structures within `signerInfos`. The indication of the hash algorithms used before the contents to be hashed makes it possible to process them within one pass.

- `encapContentInfo`
  consists of the following fields: `eContentType`, which indicates the type of data, and the actual

[2] Examples include `Data` for arbitrary data, `EncryptedData` for encrypted data, `EnvelopedData` for encrypted data with additional recipient-specific encrypted message encryption keys and `AuthenticatedData` for data that was saved with a Message Authentication Code (MAC) and the encryption of the symmetrical key used for creation of the MAC for one or more recipients, for example.

content (eContent). Possible data types include, for example, the CMS container types mentioned above. Because the eContent field does not have to be present, so-called ”detached signatures”[3], which refer to externally stored contents, can also be created.

- certificates

  is an optional field that can contain a collection of certificates, which may allow building a certification path to some trust anchor, which will be inspected while validating the signature.

- crls

  is an optional field which may contain revocation status information for information on the certificate status. Pursuant to the current CMS specification [RFC5652], revocation lists and other revocation information such as that of the OCSPResponse type can be contained here.

- signerinfos

  includes a number of signatures that are composed of the following information:

  - version
    is an integer value by means of which one can differentiate between the different SignerInfo structures. The only differences here are the sid of the SignerIdentifier type.
  - sid makes it possible to identify the (certificate of the) signing party.
  - digestAlgorithm
    indicates which hash algorithm was used for creation of the signature.
  - signedAttrs
    can contain a number of attributes that are included in the signature.
  - signatureAlgorithm
    contains the signature algorithm. Please refer to [ETSI-119312(v1.4.2)] for suitable algorithms and required parameter sizes.
  - signature
    includes the digital signature value in the indicated low-level signature format.
  - unsignedAttrs
    can contain a number of attributes that, unlike in the signedAttrs above, are not included in the signature.

Note that the CMS signature format allows constructing *parallel CMS signatures* for a single data object by having multiple SignerInfo elements in a single SignedData container.

On the other hand, one may construct *sequential CMS signatures* by treating a SignedData container as eContent and sign it with a second signature. It should be noted here that for CAdES signatures the admissible content-type for the eContent is limited to id-data (see [ETSI-319122-1(v1.2.1), Clause 6.3, Table 1, Note f)]) and hence there is no ”full nesting” of CAdES signatures.

If it is necessary to ”counter-sign” a CAdES signature, there is the countersignature attribute according to [ETSI-319122-1(v1.2.1), Clause 5.2.7], which applies to a single SignerInfo element and parallel signature.

4.3.1.2 Attributes for CMS and CAdES

As explained above, the SignerInfo structure can contain signed attributes in the signedAttrs field and unsigned attributes in the unsignedAttrs field.

[3] These digital signatures have been called ”external signatures” in [RFC2315, Section 7, Note 3] and subsequent CMS specifications.

The attributes used in [ETSI-319122-1(v1.2.1)] are discussed in the following, whereas one may distinguish the following types of attributes:

- Mandatory signed attributes for basic CAdES signatures
- Optional attributes for basic CAdES signatures
- Additional attributes for higher level CAdES signatures

Mandatory signed attributes for CAdES signatures
The following attributes are defined in [RFC5652] and are mandatory for any CAdES signature:

- `content-type`
  The `content-type` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.1.1], goes back to [RFC5652, Clause 11.1] and is used as a signed attribute which indicates the type of signed data in `eContent`. According to [ETSI-319122-1(v1.2.1), Clause 6.3, Table 1, Note f)] the `content-type` shall be `id-data` for a CAdES signature.

- `message-digest`
  The `message-digest` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.1.2], goes back to [RFC5652, Clause 11.2] and is used as a signed attribute which contains the message digest which is to be signed. The details of the message digest calculation process are specified in [RFC5652, Clause 5.4].

- `signing-time`
  The `signing-time` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.1], goes back to [RFC5652, Clause 11.3] and is used as signed attribute which specifies the time at which the signer claims to have performed the signing process.

- `signing-certificate` and `signing-certificate-v2`
  Because the certificates contained in the `SignedData` container are not signed during the creation of the signature, it would be possible to exchange the certificates at a later point in time. To counter such an attack, the `signing-certificate` and `signing-certificate-v2` attributes are defined in [ETSI-319122-1(v1.2.1), Clause 5.2.2] and are both used as signed attributes, which essentially allows binding the (hash value of the) signing certificate to the produced CAdES signature.

  The `signing-certificate` is defined in [RFC2634, Clause 5.4] and statically uses SHA-1, while `signing-certificate-2` is defined in [RFC5035, Clause 4] and allows using other hash algorithms. Against the background of recent research results such as [SBK+17, GaPe20], for example, it is more than advisable to use `signing-certificate-2` with a suitable hash algorithm for the generation of new signatures.

Optional attributes for basic CAdES signatures

- `commitment-type-indication`
  The `commitment-type-indication` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.3] and is used as a signed attribute which qualifies the signed data object and indicates a commitment made by the signer when signing the data object. The commitment type is essentially an identifier with an optional qualifier and can either be defined as part of the applicable signature policy[4] or be a registered type.

  [ETSI-119172-1(v1.1.1), Annex B] defines the following generic commitment types:

[4] See [RFC3125, ETSI-119172-1(v1.1.1), ETSI-119172-2(v1.1.1), ETSI-119172-3(v1.1.1), ETSI-119172-4(v1.1.1)].

  - proofOfOrigin (1) - indicates that the signatory created, approved and sent the message. This corresponds to the conjunction of the three OID proofOfCreation, proofOfApproval and proofOfSender.
  - proofOfReceipt (2) - indicates that the signature creator acknowledges having received the message.
  - proofOfDelivery (3) - indicates that the signature creator has delivered the message to a local memory in the recipient's access.
  - proofOfSender (4) - indicates that the signature creator sent the message but did not necessarily generate it.
  - proofOfApproval (5) - indicates that the signature creator has approved the message.
  - proofOfCreation (6) - indicates that the signature creator created the message but did not necessarily approve or send it.

- content-hints
  The content-hints attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.4.1], goes back to [RFC2634, Clause 2.9] and is used as a signed attribute which allows providing information on the innermost signed content of a multi-layer message, where one layer of content is encapsulated in another one. The content-hints attribute uses the ContentHints ASN.1 structure, which contains an optional UTF8 encoded string (contentDescription) in addition to the mandatory OID which indicates the type of the content (contentType), which necessarily is id-data, because of [ETSI-319122-1(v1.2.1), Clause 6.3, Table 1, Note f)], for a CAdES signature.

- mime-type
  The mime-type attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.4.2], is used as a signed attribute and allows specifying the MIME type of the signed content in line with [RFC2045].

- signer-location
  The signer-location attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.5], is used as a signed attribute and allows specifying an address associated with the signer at a particular geographic location.

- signer-attributes-v2
  The signer-attributes-v2 attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.6.1], is used as a signed attribute and allows integrating further attributes of the signer into the CAdES signature. The corresponding SignerAttributeV2 ASN.1 structure allows inserting the following data structures:

  - ClaimedAttributes – for attributes claimed by the signer which have not been certified or asserted by a third party,
  - CertifiedAttributesV2 – for attribute certificates according to [RFC5755, X.509], for example, and
  - SignedAssertions – for one or more signed assertions according to [SAML(v2.0)].

- countersignature
  The countersignature attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.7], goes back to [RFC5652, Clause 11.4], is used as an unsigned attribute and allows including a counter signature on the CAdES signature where this attribute is included.

- content-time-stamp
  The content-time-stamp attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.8], is used

as a signed attribute and encapsulates one time-stamp token[5] of the signed data content, which is either the `eContent` or in case of a detached signature the external content, before it is signed.

- `signature-policy-identifier`
  The `signature-policy-identifier` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.9.1], is used as a signed attribute and allows explicitly specifying the applicable signature policy[6] and optionally providing further information within `SigPolicyQualifierInfo` structures according to [ETSI-319122-1(v1.2.1), Clause 5.2.9.2].

- `signature-policy-store`
  The `signature-policy-store` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.10], is used as an unsigned attribute and allows including the applicable signature policy document (`sigPolicyEncoded`) or a link to a local signature policy repository (`sigPolicyLocalURI`) in the CAdES signature.

- `content-reference`
  The `content-reference` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.11], goes back to [RFC2634, Clause 2.11] and is used as a signed attribute which allows linking one `SignedData` element to another one. This attribute may be used in conjunction with the `content-identifier` attribute explained below for the construction of signed receipts.

- `content-identifier`
  The `content-identifier` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.2.12], goes back to [RFC2634, Clause 2.7] and is used as a signed attribute which provides an identifier of the signed content to which a second `SignedData` structure can refer by using the `content-reference` attribute explained above.

[5] See [RFC3161, RFC5816, ETSI-319422(v1.1.1)].

[6] See [RFC3125, ETSI-119172-1(v1.1.1), ETSI-119172-2(v1.1.1), ETSI-119172-3(v1.1.1), ETSI-119172-4(v1.1.1)].

Additional attributes for higher level CAdES signatures

In addition to the basic `CAdES-B-B` signatures, one may construct higher level CAdES signatures as depicted in Figure 4.3, which utilise different unsigned attributes as explained below.

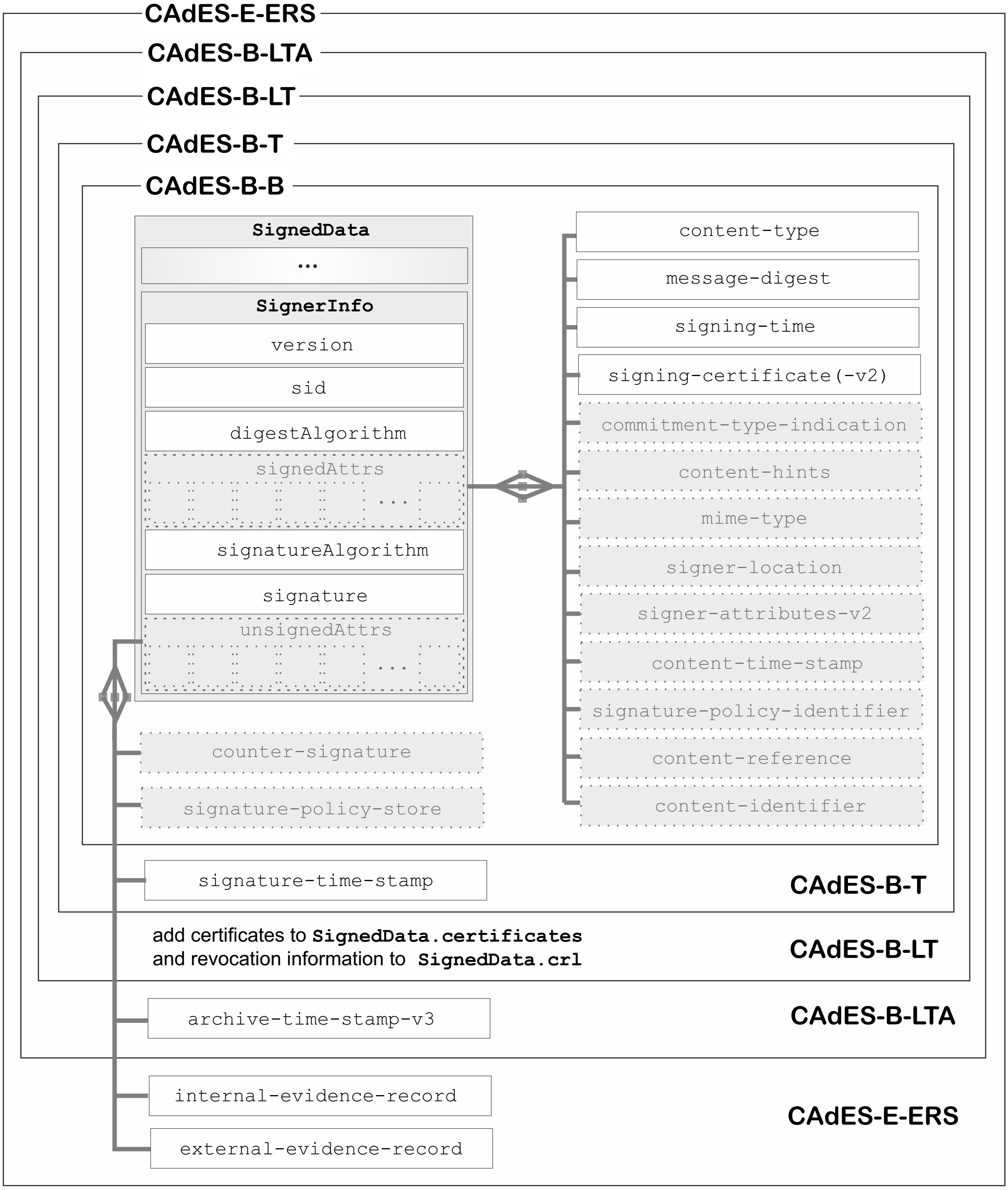


Figure 4.3: CAdES Signature Levels

The CAdES Signature Levels depicted in Figure 4.3 are as follows:

- `CAdES-B-B` is the basic form of a CAdES signature, which contains the mandatory signed attributes and one or more of the optional attributes explained above. This signature level is appropriate if the validation and/or preservation of the signature by the intended recipient(s) is expected to take place shortly after the generation of the signature and hence there is no need for adding a `signature-time-stamp` attribute and creating a `CAdES-B-T`.

- `CAdES-B-T` adds a `signature-time-stamp` attribute as explained below to the basic signature and provides a proof of existence of the CAdES signature. This signature level is appropriate if it cannot be guaranteed that the validation and/or preservation of the signature takes place shortly after the generation of the signature and within the validity period of the signing certificate, as omitting the time-stamp would yield invalid signatures in the ”chain model”[7].

- `CAdES-B-LT` requires that the necessary validation material is added to the CAdES signature. While [ETSI-119102-1(v1.2.1), Annex A] contains a variety of attributes, which would allow adding certificates, revocation information (i.e. OCSP responses and CRLs), references to such data objects and time-stamps on the aforementioned data objects, [ETSI-119102-1(v1.2.1), Clause 6.3, Table 1] effectively deprecates the use of these attributes and requires for `CAdES-B-LT` signatures, as for the baseline signatures according to [ETSI-101733(v2.2.1)], the storage of the certificates required for validation in `SignedData.certificates` and the storage of revocation information in `SignedData.crl`.

- `CAdES-B-LTA` adds additional `archive-time-stamp-v3` and `ats-hash-index-v3` attributes explained below to `CAdES-B-LT` signatures in order to protect the integrity of the CAdES signature in the long term.

- `CAdES-E-ERS` provides an alternative solution for the long-term protection of CAdES signatures based on evidence records as specified in [ETSI-119122-3(v1.1.1)] and [RFC4998, Annex A]. As explained below, this involves the `internal-evidence-record` (see figure 4.5) attribute for CAdES signatures with integrated `eContent` and the `external-evidence-record` (see figure 4.6) attribute for detached CAdES signatures. The difference between the two approaches is that the utilisation of evidence records is more efficient with respect to the number of required time-stamps, because the ”Merkle Hash Tree” construction according to [RFC4998] allows protecting many independent signatures and data objects with a single time-stamp.

[7] See [ETSI-119102-1(v1.2.1)].

Note that the previous CAdES specification [ETSI-101733(v2.2.1)], which is referenced in [ETSI-103171(v2.1.1)] and hence indirectly in [(EU)2015/1506], defines slightly different signature levels, which more or less map to the levels presented here, as outlined in the following table:

[ETSI-319122-1(v1.2.1)]	[ETSI-101733(v2.2.1)]	**Note**
CAdES-B-B	CAdES-BES CAdES-EPES	without `signature-policy-identifier` with `signature-policy-identifier`
CAdES-B-T	CAdES-T	While [ETSI-319122-1(v1.2.1)] only specifies the `signature-time-stamp` attribute, [ETSI-101733(v2.2.1)] also mentions an alternative ”Time Mark” for reaching the CAdES-T level.
CAdES-B-LT	CAdES-C, CAdES-X, CAdES-LT	As [ETSI-103171(v2.1.1), Clause 8] requires that the attributes constituting CAdES-C, CAdES-X and CAdES-LT shall *not* be used, there is effectively no difference for baseline signatures.
CAdES-B-LTA	CAdES-LTA	In addition to the `archive-time-stamp-v3` attribute, [ETSI-101733(v2.2.1), Clause 6.4.1] also specifies the `archive-time-stamp` attribute, which however has been deprecated in [ETSI-319122-1(v1.2.1), Annex A.2.4]. Furthermore, it should be noted that [(EU)2015/1506] explicitly excludes LTA level signatures.

Table 4.1: Comparing the CAdES Signature Levels

- `signature-time-stamp`
  The `signature-time-stamp` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.3] and allows encapsulating one time-stamp token[8] computed on the digital signature value for a specific signer (attribute `SignerInfo`). This attribute provides proof of existence for the signature. It is especially required for the construction of signatures of type `CAdES-B-T`.

[8] See [RFC3161, RFC5816, ETSI-319422(v1.1.1)].

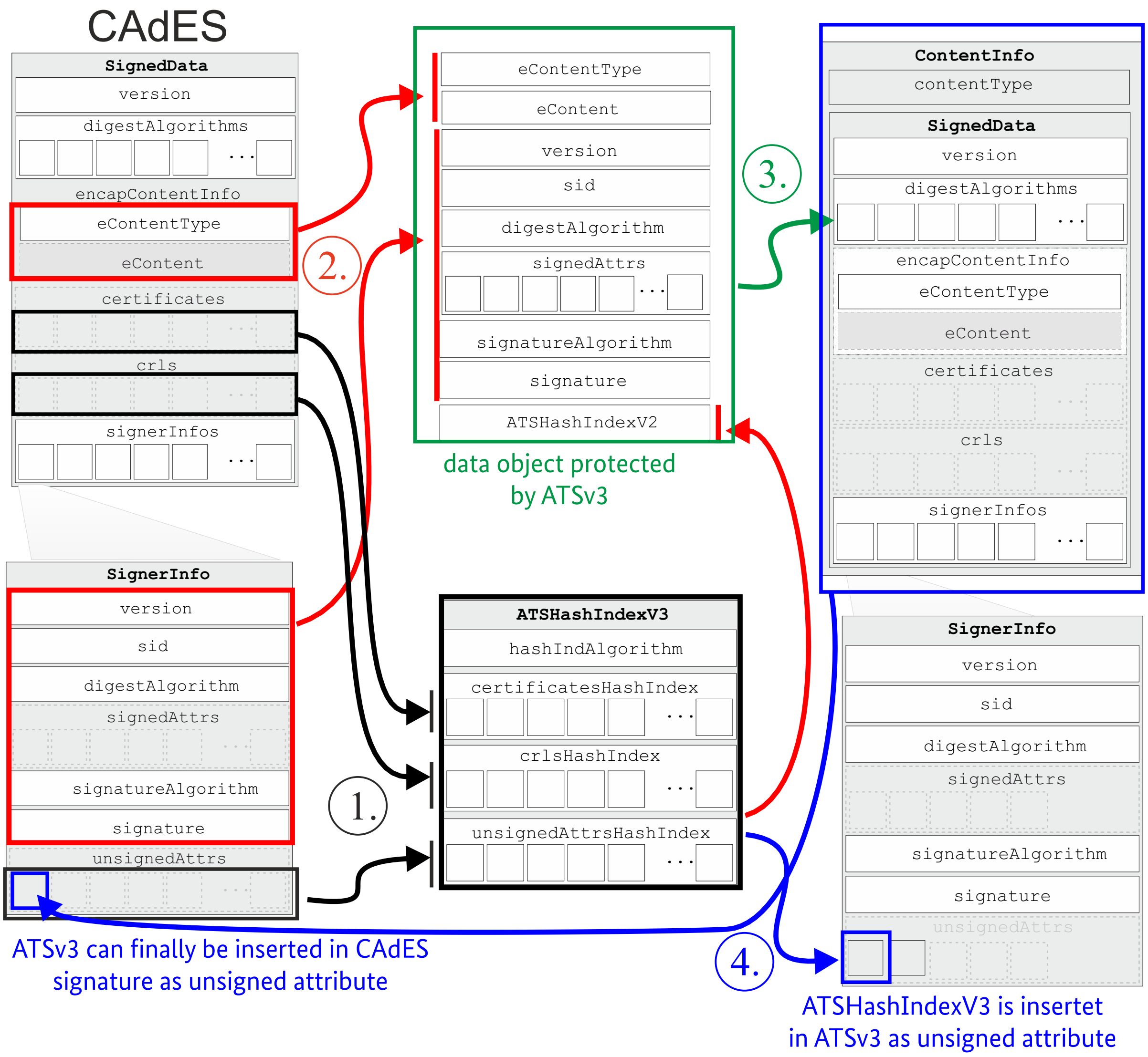


Figure 4.4: Construction of `archive-time-stamp-v3` using `ats-hash-index-v3`

- `archive-time-stamp-v3` and `ats-hash-index-v3`
  The `archive-time-stamp-v3` attribute is defined in [ETSI-319122-1(v1.2.1), Clause 5.5.3] and is applied together with the `ats-hash-index-v3` attribute according to [ETSI-319122-1(v1.2.1), Clause 5.5.2] to protect the integrity of the necessary validation material in the long term.

  As outlined in Figure 4.4, the generation of an `archive-time-stamp-v3` attribute consists of the following four steps:

  1.) In the first step, the computation of the `ATSHashIndexV3` structure takes place, which involves the certificates from `SignedData.certificates`, the revocation information from `SignedData.crls` and the unsigned attributes of the CAdES signature.

  2.) Next, the remaining part of the effectively protected (green) data object is compiled, which involves the `SignedData.eContentType`, `SignedData.eContent` (or the external data object in case of a detached signature) and the essential signed part of the `SignerInfo` structure, which comprises

  - `SignerInfo.version`
  - `SignerInfo.sid`

- `SignerInfo.digestAlgorithm`
- `SignerInfo.signedAttrs`
- `SignerInfo.signatureAlgorithm`
- `SignerInfo.signature`

3.) Now the `ATSHashIndexV3` structure created in the first step is appended to the structure compiled in the second step and the result undergoes the calculation of the `messageImprint` of the `ArchiveTimeStampToken` of the ATSv3.

4.) In the last step, the `ATSHashIndexV3` structure is inserted as an unsigned attribute into the `SignedData.SignerInfo` of the ATSv3, and the ATSv3 is inserted as an unsigned attribute into the `SignedData.SignerInfo` of the CAdES signature.

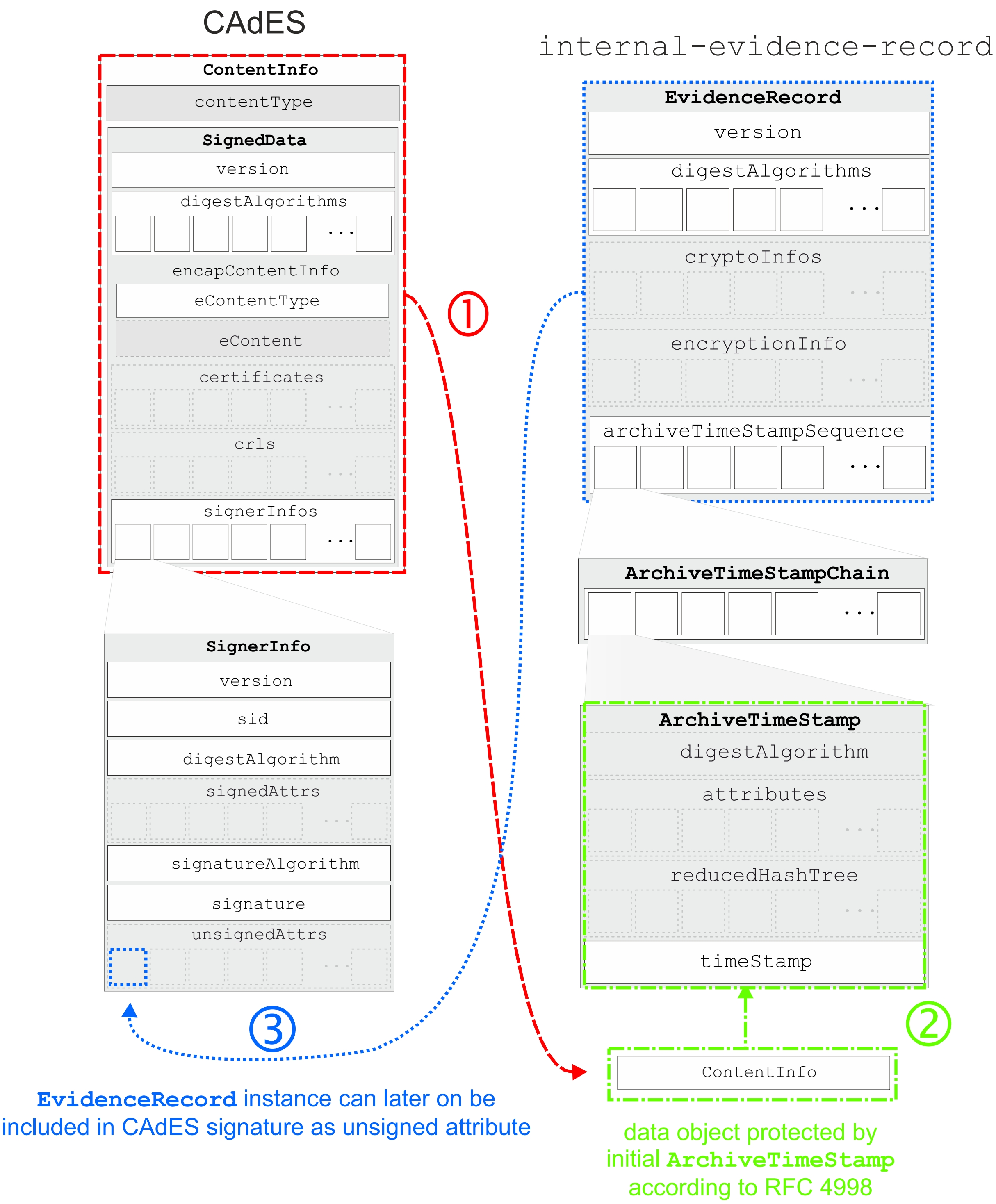


Figure 4.5: The `internal-evidence-record` attribute

- `internal-evidence-record`
  The `internal-evidence-record` attribute is defined in [ETSI-119122-3(v1.1.1)] and [RFC4998, Annex A] and is used as an unsigned attribute which contains an evidence record according to [RFC4998], which applies to the CAdES signature with included `eContent`.

  As outlined in Figure 4.5, the generation of an `internal-evidence-record` attribute consists of the following three steps:

  1.) The complete `ContentInfo` instance serves as input for the `messageImprint` calculation of the initial `ArchiveTimeStamp` instance.

2.) The `EvidenceRecord` instance according to [RFC4998] is built based on the initial `ArchiveTimeStamp` instance. If necessary, the integrity of the `EvidenceRecord` is maintained over a long period of time, which may involve time stamp renewals and/or hash tree renewals as specified in [RFC4998].

3.) Finally, the `EvidenceRecord` is inserted as an unsigned `internal-evidence-record` attribute into the CAdES signature.

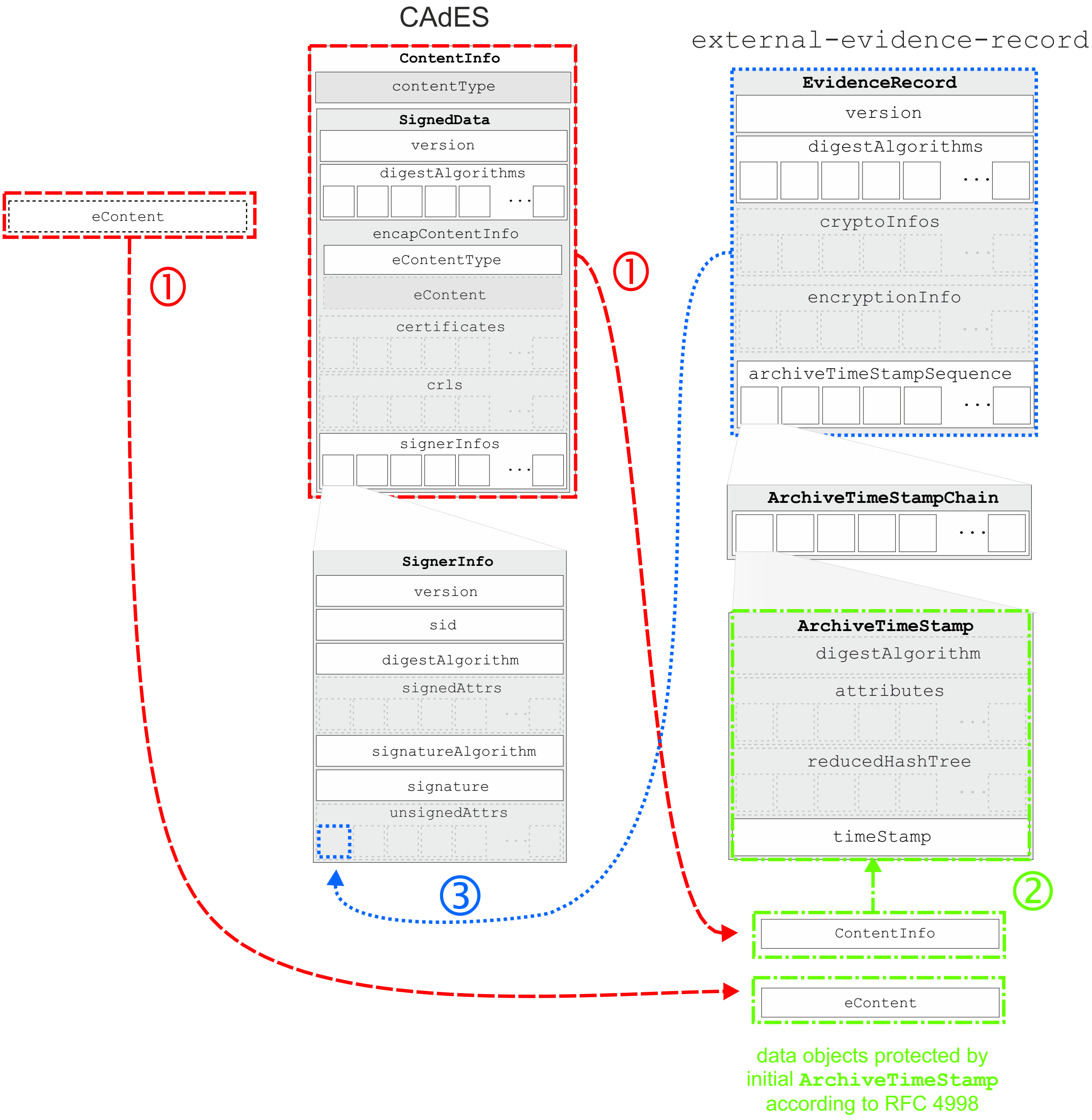


Figure 4.6: The `external-evidence-record` attribute

- `external-evidence-record`
  The `external-evidence-record` attribute is defined in [ETSI-119122-3(v1.1.1)] and [RFC4998, Annex A] and is used as an unsigned attribute which contains an evidence record according to [RFC4998], which applies to the CAdES signature with external `eContent`.

As outlined in Figure 4.6, the generation of an `external-evidence-record` attribute consists of the following three steps:

1.) Both the complete `ContentInfo` instance and the external `eContent` serve as input for the `messageImprint` calculation of the initial `ArchiveTimeStamp` instance.

2.) The `EvidenceRecord` instance according to [RFC4998] is built based on the initial `ArchiveTimeStamp` instance. If necessary, the integrity of the `EvidenceRecord` is maintained over a long period of time, which may involve time stamp renewals and/or hash tree renewals as specified in [RFC4998].

3.) Finally, the `EvidenceRecord` is inserted as an unsigned `external-evidence-record` attribute into the CAdES signature.

4.3.2 XML-Digital Signature and XAdES

4.3.2.1 XML-Digital Signature

For the creation of digital signatures of data in XML format, a specific signature format was developed by a working group of W3C [XML-DSig, RFC3275]. In comparison to the Cryptographic Message Syntax signature format explained in Section 4.3.1, the XML signature offers a higher level of flexibility that is needed in order to exploit the full potential of XML with regard to digital signatures.

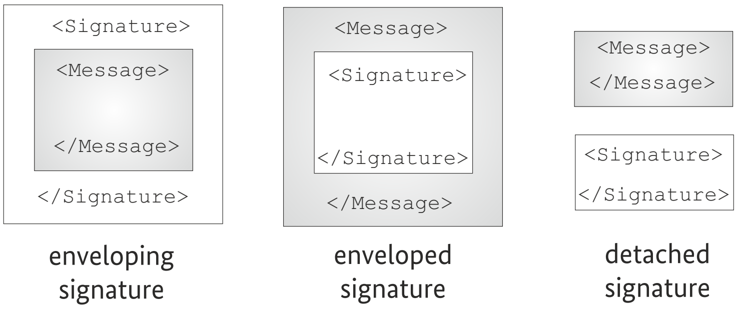


Figure 4.7: XML Signature Types

Whereas the CMS format only supports the creation of enveloping signatures and detached signatures, an XML signature pursuant to [XML-DSig, RFC3275] can also be embedded in the message to be signed (enveloped signature) (see Figure 4.7). A similar construction is possible in a standardised form for CMS-based signatures in conjunction with certain document types, such as PDF (see Section 4.3.3). When using an XML signature, one can include complete files of any type or particular parts of an XML document or transform the data to be signed before the creation of the signature in a certain manner. For example, an XPath or XSL transformation can be carried out. It is possible with the XPath Transformation [XPath] to omit certain parts of an XML document when the signature is created so that these data fields can be changed later without invalidating the signature. This can be used, for example, when enveloped signatures are created. The data in XML format can be linked to a certain layout [XSL] before the creation and verification of a signature with an XSL transformation [XSLT].

```
<Signature ID>
        <SignedInfo>
                <CanonicalizationMethod/>
                <SignatureMethod/>
                <Reference URI>
                        <Transforms>
                        <DigestMethod>
                        <DigestValue>
                </Reference>
        </SignedInfo>
        <SignatureValue>
        <KeyInfo>
        <Object ID>
</Signature>
```

As shown above, an XML signature begins with the `<Signature>` tag, ends with the `</Signature>` tag, and mainly includes the following elements:

- `SignedInfo` – contains information about which data are to be signed. In particular, it includes the following parts:

  - CanonicalizationMethod – indicates which canonicalization method is used.
  - SignatureMethod – specifies the signature algorithm.
  - Reference – is a field that can be present one or more times and contains a reference to the data to be signed that are addressed with a Uniform Resource Identifier (URI) and information for preparing them. It includes the following child elements:
    - Transforms – is optional and indicates how the data is transformed before the hash function is applied.
    - DigestMethod – specifies the hash function to be used.
    - DigestValue – contains the hash value that was calculated with the indicated hash function from the referenced and possibly transformed data.

- SignatureValue – contains the digital signature
- KeyInfo – is an optional field in which, for example, certificates can be stored.
- Object – can be used as often as desired (also never) and may contain any desired data object. For example, some messages to be signed, SignatureProperties (additional characteristics of the signature, e.g. time of signing), QualifyingProperties (denoting an XML element that qualifies the signature, the signed data object or the signer), in case of a XAdES signature, an additional Signature or a Manifest element.

As outlined on page 101, XML digital signatures are encoded in ”human readable” XML structures, whereas CMS-based signatures are encoded in ASN.1.

Analogous to the CAdES extensions for CMS [RFC5652] presented in Section 4.3.1, there are also corresponding extensions for [XML-DSig] to XML advanced digital signatures (XAdES). On the basis of the initial XAdES specification [ETSI-101903(v1.4.2)], which also appeared as W3C note [XAdES], the development of the current XAdES specification [ETSI-319132-1(v1.2.1), ETSI-319132-2(v1.1.1), ETSI-119132-3(v1.1.1)] took place with several intermediate steps.

4.3.2.2 QualifyingProperties

In a similar manner as for signed and unsigned attributes in CAdES signatures, there are corresponding data objects (<QualifyingProperties>) for XAdES digital signatures, which give rise to an ordered sequence of XAdES signature levels (cf. Figure 4.8).

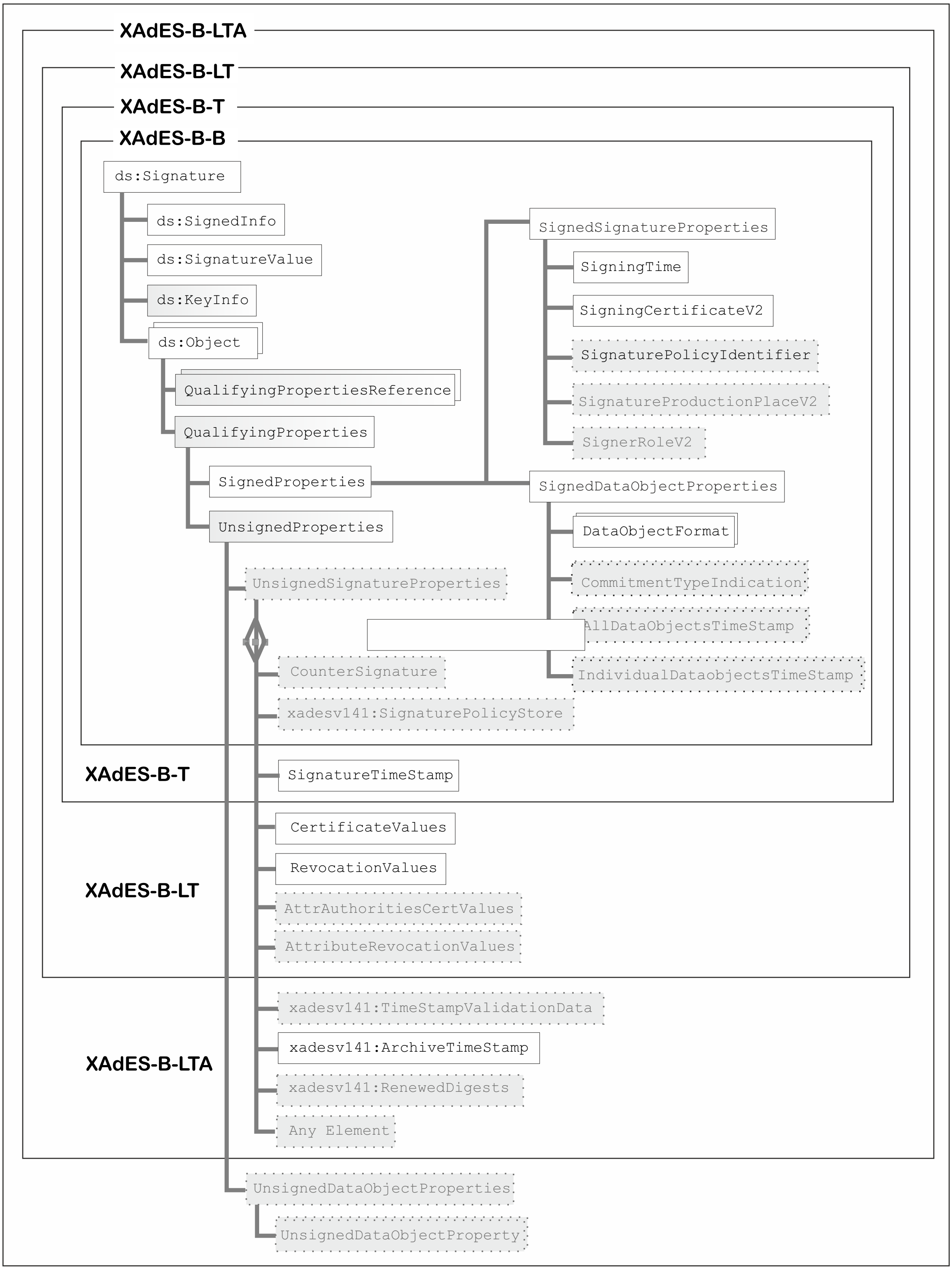


Figure 4.8: XAdES Signature Levels

A XAdES digital signature is an XML signature according to [XML-DSig], which contains a `<QualifyingProperties>` element (cf. [ETSI-319132-1(v1.2.1), Clause 4.3] or one or more `<QualifyingPropertiesReference>` elements (cf. [ETSI-319132-1(v1.2.1), Clause 4.4.3] in an `<ds:Object>` (cf. Section 4.3.2.1) element. The following two child elements exist in the

<QualifyingProperties> element:

- <SignedProperties>
- <UnsignedProperties>

SignedProperties
This element is included in the signature via the <ds:Reference> element and contains the following two child elements:

- <SignedSignatureProperties>
- <SignedDataObjectProperties>

SignedSignatureProperties
The <SignedSignatureProperties> container is defined in [ETSI-319132-1(v1.2.1), Clause 4.3.4] and contains signed properties that are assigned to the signature.

The following elements are defined in [ETSI-319132-1(v1.2.1)]:

- <SigningTime>
  is defined in [ETSI-319132-1(v1.2.1), Clause 5.2.1] and can be used in a similar manner as the signing-time attribute for the CAdES signature to document the time at which the signatory claimed that the signature was produced.
- <SigningCertificateV2>
  is defined in [ETSI-319132-1(v1.2.1), Clause 5.2.2] and can be used in a similar way as the signing-certificate-v2 attribute of the CAdES signature in order to uniquely identify the certificate on which the signature is based and to include it in the signature. The difference to the deprecated[9] SigningCertificate property is that hash functions beyond SHA-1 may be used.
- <SignaturePolicyIdentifier>
  is defined in [ETSI-319132-1(v1.2.1), Clause 5.2.9] and can be used in a similar way as the signature-policy-identifier attribute of the CAdES signature to determine the signature policy on which the signature is based[10].
- <SignatureProductionPlaceV2>
  is defined in [ETSI-319132-1(v1.2.1), Clause 5.2.5] and can be used in a similar way as the signer-location attribute for the CAdES signature to specify the location of the signature generation. The difference to the deprecated[11] <SignatureProductionPlace> property is that the <SignatureProductionPlaceV2> may also contain a <StreetAddress> element.
- <SignerRoleV2>
  is defined in [ETSI-319132-1(v1.2.1), Clause 5.2.6] and can be used in a similar manner as the signer-attributes-v2 attribute in the CAdES signature. It allows specifying the role of the signer in more detail. As with the CAdES signature, there may by <ClaimedRoles>, <CertifiedRolesV2>, which may contain attribute certificates, and <SignedAssertions>. The difference to the deprecated[12] <SignerRole> property is that <SignedAssertions> may be included in <SignerRoleV2>.

[9] See [ETSI-319132-1(v1.2.1), Annex D]. In contrast to this, the signing-certificate-v2 attribute of the CAdES signature is still in use.

[10] See [RFC3125, ETSI-119172-1(v1.1.1), ETSI-119172-2(v1.1.1), ETSI-119172-3(v1.1.1), ETSI-119172-4(v1.1.1)].

[11] See [ETSI-319132-1(v1.2.1), Annex D].

[12] See [ETSI-319132-1(v1.2.1), Annex D].

`SignedDataObjectProperties`
The `<SignedDataObjectProperties>` element contains signed properties that are associated with the signed data. The following properties are specified in [ETSI-319132-1(v1.2.1)]:

- `<DataObjectFormat>`
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.2.4] and can be used in a similar manner as the `content-hints` and `mime-type` attributes for CAdES signatures, which allow specifying the format of the signed data.

- `<CommitmentTypeIndication>`
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.2.3] and can be used in a similar manner as the `commitment-type-indication` attribute of the CAdES signature to assign a specific meaning to a signature.

  The following URI (each with the prefix `http://uri.etsi.org/01903/v1.3.2#`) can be used for this purpose:

  - `proofOfOrigin` – indicates that the signatory created, approved and sent the message. This corresponds approximately to the three URIs `ProofOfCreation`, `ProofOfApproval` and `ProofOfSender`.
  - `ProofOfReceipt` – indicates that the signatory acknowledges to have received the message.
  - `ProofOfDelivery` – indicates that the signatory has delivered the message to a local store, which is accessible to the recipient of the signed data.
  - `ProofOfSender` – indicates that the signatory sent the message but did not necessarily generate it.
  - `ProofOfApproval` – indicates that the signatory has approved the message.
  - `ProofOfCreation` – indicates that the signatory created the message but did not necessarily approve or send it.

- `<AllDataObjectsTimeStamp>`
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.2.8.1] and contains a time stamp which refers to *all* data objects embedded in the signature via the set of `ds:Reference` elements.

- The `<IndividualDataObjectsTimeStamp>`
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.2.8.2] and contains one (or more) time stamp(s), which refers to some data objects embedded in the signature via some of the `ds:Reference` elements. *”The qualifying property shall encapsulate one or more electronic timestamps, generated before the signature production, whose message imprint computation input is the concatenation of the objects obtained after processing as specified in XMLDSIG [1], clause 4.4.3.2; some of the `ds:Reference` elements within the `ds:SignedInfo` or also signed ds:Manifest.”*[ETSI-319132-1(v1.2.1), Clause 5.2.8.2]

`UnsignedProperties`
This element is not included in the signature and can therefore also be created or supplemented after signature creation – for example, during the validation of a signature. It contains the following two child elements:

- `<UnsignedSignatureProperties>`

- `<UnsignedDataObjectProperties>`

UnsignedSignatureProperties

The UnsignedSignatureProperties element can contain any sequence of the following elements:

- <CounterSignature>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.2.7] and can be used in a similar manner as the countersignature attribute of the CAdES signature to countersign an existing signature. As explained in [ETSI-319132-1(v1.2.1), Clause 5.2.7], the Type attribute in the <Reference> element, which specifies the signed value can be set to the value http://uri.etsi.org/01903#CountersignedSignature to indicate that the signature under consideration is a countersignature. As explained in [ETSI-319132-1(v1.2.1), Clause 5.5.2.1], a signature with a time stamp can in principle also be countersigned. In this case, however, care must be taken to ensure that the original signature must not have any additional unsigned properties, since otherwise the time stamp would lose its validity.

- <SignatureTimeStamp>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.3] and can be used in a similar manner as the signature-time-stamp attribute of a CAdES signature in order to prove the existence of a signature at a certain point in time. This element is especially needed for the construction of signatures of the type XAdES-B-T.

- <CertificateValues>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.4.1] and can be used to store certificate values (e.g. of the signatory, the involved CA certificates, the involved authorities which sign revocation information, the trust anchor), which are not yet present in the ds:KeyInfo element. The <CertificateValues> property is used for the construction of signatures of the type XAdES-B-LT.

  This property is similar to the certificate-values attribute according to [ETSI-319122-1(v1.2.1), Annex A.1.1.2], which however is deprecated for CAdES Baseline Signatures.

- <RevocationValues>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.4.2] and can be used to store revocation values, such as CRLs or OCSP responses, which are not yet present in the ds:KeyInfo element. The <RevocationValues> property is used for the construction of signatures of the type XAdES-B-LT.

  This property is similar to the revocation-values attribute according to [ETSI-319122-1(v1.2.1), Annex A.1.2.2], which however is deprecated for CAdES Baseline Signatures.

- <AttrAuthoritiesCertValues>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.4.3] and is used to store certificate values of authorities, which issue attribute certificates or signed SAML assertions. The <AttrAuthoritiesCertValues> property is used for the construction of signatures of the type XAdES-B-LT.

  Note that there is no corresponding attribute for CAdES signatures because the corresponding certificates are treated as any other certificate and stored within SignedData.certificates.

- <AttributeRevocationValues>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.4.4] and is used to store revocation information in the form of CRLs or OCSP responses related to certificates of authorities which issue attribute certificates or signed SAML assertions.

  The <AttributeRevocationValues> property is used for the construction of signatures of the type XAdES-B-LT.

Note that there is no corresponding attribute for CAdES signatures because the corresponding revocation values are stored within SignedData.crl.

- <TimeStampValidationData>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.5.1] and is a container, which allows storing any missing validation data, such as CertificateValues and RevocationValues, required for the full validation of a time stamp which is present within the existing XAdES signature.[13]

- <ArchiveTimeStamp>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.5.2] and allows adding an archive time stamp to an existing XAdES signature in order to protect its integrity and authenticity in the long term.[14]

  The <ArchiveTimeStamp> property is used for the construction of signatures of the type XAdES-B-LTA.

  Note that there was another <ArchiveTimeStamp> property defined in the namespace whose URI is http://uri.etsi.org/01903/v1.3.2#, which differs in subtle details with respect to the generation and validation of the archive time stamp property. This property had already been deprecated in [ETSI-101903(v1.4.2)].

  From an abstract perspective, this property is similar [15] to the archive-time-stamp-v3 attribute for the CAdES signature and also requires a separate time stamp for each data object or signature which is to be preserved. If the number of signed data objects which need to be preserved is high, it is recommended to use hash tree-based archive time stamps and evidence records instead. See also [ETSI-119132-3(v1.1.1)].

- <RenewedDigestsV2>
  is specified in [ETSI-319132-1(v1.2.1), Clause 5.5.3] and allows performing a hash renewal for data which are protected by signed <Manifest> elements.

4.3.2.3 XML Advanced Electronic Signature (XAdES)

As can be seen in Figure 4.8, using the various <QualifyingProperties> child elements, there is a sequence of XAdES Signature Levels (see [ETSI-319132-1(v1.2.1), Clause 6.1] and Figure 4.8):

- Basic XAdES Signature (XAdES-B-B)
- XAdES Signature with Time (XAdES-B-T)
- XAdES Signature with Validation Data (XAdES-B-LT)
- XAdES Signature with Archive Time Stamp (XAdES-B-LTA)

Basic XAdES Signature (XAdES-B-B)
The XAdES-B-B is the simplest form of the XAdES Baseline Signature, which may contain the following signed and unsigned properties:

- <QualifyingProperties>
  - <SignedProperties>
    - <SignedSignatureProperties>

[13] http://uri.etsi.org/01903/v1.4.1

[14] http://uri.etsi.org/01903/v1.4.1

[15] A conceptual difference is that for XAdES signatures there is no property which would correspond to the ats-hash-index-v3 attribute of CAdES, see clause 4.3.1.2.

        - <SigningTime>
        - <SigningCertificateV2>
        - <SignatureProductionPlaceV2>
        - <SignerRoleV2>
      - <SignedDataObjectProperties>
        - <DataObjectFormat>
        - <CommitmentTypeIndication>
        - <AllDataObjectsTimeStamp>
        - <IndividualDataObjectsTimeStamp>
    - <UnsignedProperties>
      - <UnsignedSignatureProperties>
        - <CounterSignature>
        - <SignaturePolicyStore>

XAdES Signature with Time (XAdES-B-T)
Analogously to the CAdES-B-T signature, there is the XAdES-B-T signature, which contains one or more <SignatureTimeStamp> properties, which give rise to a proof of existence for the signature.

XAdES Signature with Validation Data (XAdES-B-LT)

The XAdES-B-LT signature additionally contains validation data, where the following unsigned properties may be added:

- <CertificateValues>
- <RevocationValues>
- <AttrAuthoritiesCertValues>
- <AttributeRevocationValues>
- <TimeStampValidationData>

XAdES Signature with Archive Time Stamp (XAdES-B-LTA)

Finally, there is the XAdES-B-LTA signature level, which may contain the following unsigned properties:

- <ArchiveTimeStamp> (shall be present, according to http://uri.etsi.org/01903/v1.4.1#)
- <RenewedDigestsV2>

4.3.3 PDF Signature and PAdES

While [(EU)2015/1506, Article 1] stipulates that EU Member states shall recognise PDF advanced electronic signatures which comply with the ETSI technical specification [ETSI-103172(v2.2.2)], which is based on [ETSI-101733(v2.2.1)], [ETSI-102778-3(v1.2.1)] and [ETSI-102778-4(v1.1.2)], it should be mentioned that the European Standard [ETSI-319142-1(v1.1.1)] provides a more up-to-date version of the PAdES specification, which is based on [ETSI-319122-1(v1.2.1)] and only lists [ETSI-103172(v2.2.2)] as an informative reference. [ETSI-319142-1(v1.1.1)] as well as the set of previous technical specifications define PDF Advanced Electronic Signatures (PAdES) based on a profile of the PDF 1.7 standard [ISO32000-1], for which by now updated PDF 2.0 specifications [ISO32000-2:2020] are available. [ETSI-319142-2(v1.1.1)] provides additional PAdES signature profiles.

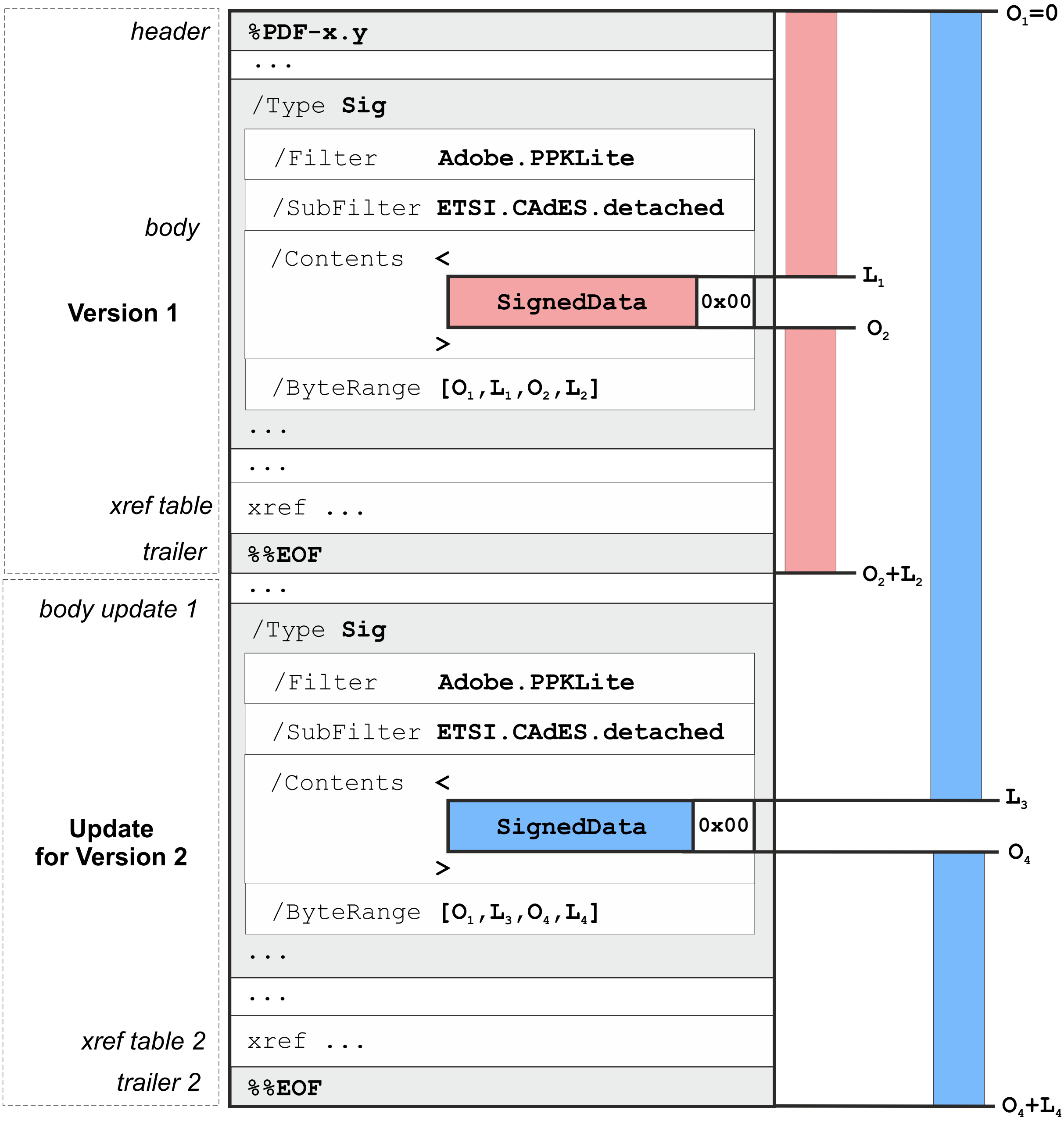


Figure 4.9: General structure of PDF documents and PAdES signatures

4.3.3.1 General structure of PDF documents

As outlined in Figure 4.9 and specified in [ISO32000-1, Section 7.5] a PDF document roughly consists of the following parts:

- a *header* (%PDF-x.y), which indicates the version number of the utilised PDF specification,
- a *body*, which comprises the objects (cf. [ISO32000-1, Section 7.3]) contained in the PDF document,
- a *cross-reference (xref) table* that permits random access to indirect objects and
- a *trailer* (%%EOF), which allows a quick search for find certain objects in the PDF document.

The PDF specification defines different types of objects among which is the *dictionary object* (cf. [ISO32000-1, Section 7.3.7]). A dictionary object is an associative table composed of *entries* that consist of a *key* and its corresponding *value*, which can be an object.

4.3.3.2 Construction of PAdES signatures

The *signature directory* defined in [ISO32000-1, Section 12.8.1]) is especially important for PAdES, as it is used for embedding digital signatures according to CMS and CAdES into PDF documents, as outlined in Figure 4.9 and explained in the following. The possible entries of a signature directory are listed in [ISO32000-1, Table 252]) and [ISO32000-2:2020, Table 255]), where the following keys are relevant for the construction of PAdES signatures:

- `Type` (Optional): Specifies the type of the present dictionary object and has the value `Sig` in case of a signature dictionary.
- `Filter` (Required)[16]: contains the name of the suggested signature handler to be used when validating the signature, such as `Adobe.PPKLite`.
- `SubFilter` (Optional): Describes the encoding of the signature value and key information in the signature directory. The admissible values depend on the specific standard and profile under consideration[17], whereas the PAdES specifications require `ETSI.CAdES.detached` here (cf. [ETSI-102778-3(v1.2.1), Section 4.2, e)] and [ETSI-319142-1(v1.1.1), Section 6.3, l)]). Other values may be used, but would need to be properly defined and registered by developers as described in [ISO32000-2:2020, Annex E].
- `Contents` (Required): The signature value. In case of a PAdES signature according to [ETSI-319142-1(v1.1.1)], this is the DER-encoded `SignedData` object of a CAdES signature according to [ETSI-319122-1(v1.2.1)]. As specified in [ETSI-319142-1(v1.1.1), Clause 4.1], there shall only be a single signer (i.e. one single component of `SignerInfo` type within the `signerInfos` element) in any PDF signature and there shall be no data encapsulated in the `SignedData` field. ”Since the length of CMS objects is not entirely predictable, the value of `Contents` shall be padded with zeros at the end of the string (before the ’>’ delimiter) before writing the CMS to the allocated space in the file.”[18]

[16] As explained in [ETSI-102778-2(v1.2.1), Section 5.2] or [ETSI-319142-1(v1.1.1), Section 6.3, j)], a PDF reader may use an alternative signature handler for validating the signature as long as it supports the specified `SubFilter` format.

[17] [ISO32000-1, Section 12.8.1] lists the values `adbe.x509.rsa_sha1`, `adbe.pkcs7.detached` and `adbe.pkcs7.sha1`, while the ETSI profile of ISO 32000-1 [ETSI-102778-2(v1.2.1), Section 5.2, b)] only allows `adbe.pkcs7.detached` and `adbe.pkcs7.sha1`. [ISO32000-2:2020, Section 12.8.1] lists all mentioned values, but remarks that `adbe.x509.rsa_sha1`, `adbe.pkcs7.sha1` is deprecated in PDF 2.0.

[18] See [ISO32000-2:2020, Section 12.8.3.3.1].

- ByteRange (Required): Specifies which parts of the PDF document are protected by the signature. While for general PDF signatures it would be possible to specify an arbitrary number of pairs for the byte offset and the byte length, for PAdES signatures where the SubFilter is ETSI.CAdES.detached, it is required that the ByteRange shall cover the entire file, including the signature directory but excluding the Contents value. From a security perspective this stipulation is crucial as it prevents attacks presented in [MMM+19] which manipulate the ByteRange entry of a PDF signature.

- Name (Optional)[19]: The name of the person or authority signing the document.

- M (Optional)[20]: Specifies the time the document was signed.

- Location (Optional): The CPU host name or physical location of the signing.

- Reason (Optional)[21]: The reason for the signing, such as ”I agree ...”.

- ContactInfo (Optional): Information provided by the signer to enable a recipient to contact the signer.

In addition to the PDF digital signatures outlined above, which are called ”approval signatures” or ”recipient signatures” in [ISO32000-2:2020, Section 12.8], the PDF specification also mentions ”certification signatures” or ”author signatures”, which allow the specification of a set of admissible changes of a PDF document[22].

Furthermore, it should be mentioned that the PDF specifications[23] contain specific stipulations for *signature fields*, which are interactive form fields that allow an interactive form to be locked and hence prevent the modification of the form fields after creating a signature. Furthermore, a signature field may have a visual signature representation, which is realised by an appropriate associated widget annotation[24].

An interesting aspect of the PDF specification[25] is that it treats revocation information as *signed* attribute and hence it is required to collect the revocation information for the involved certificates *before* the generation of the digital signature takes place. For this purpose there is the adbe-revocationInfoArchival attribute, which may contain CRLs or OCSP responses, for example, and which shall be integrated as signed attribute in a non-PAdES signature, for which the SubFilter is different from ETSI.CAdES.detached. If this is not feasible, one may add validation data later on using the structures introduced in [ETSI-319142-1(v1.1.1), Clause 5.4] and explained in the paragraph Validation data and archive validation data attributes below.

An important feature of PDF is that it supports incremental updates (cf. [ISO32000-1, Section 7.5.6]), which immediately gives rise to sequential signatures as outlined in Figure 4.9.

A practical advantage of PAdES signatures is that they can be validated with freely available software, such as the Adobe Reader, which is already installed on many devices. So in most cases no installation of additional software is needed. This facilitates the process compared to the validation of digital signatures for other types of documents.

[19] [ISO32000-1, Table 252] and [ISO32000-2:2020, Table 255] recommend to use this entry only if it is not possible to extract the name from the signature.

[20] While this element is considered to be optional in the PDF specifications (cf. [ISO32000-1, Table 252] and [ISO32000-2:2020, Table 255]), it is required for PAdES baseline signatures according to [ETSI-319142-1(v1.1.1), Section 6.3, g)].

[21] [ETSI-319142-1(v1.1.1), Section 6.3, m)] stipulates that the Reason entry shall not be used if the commitment-type-indication attribute or a signature-policyidentifier attribute is present in the CMS signature.

[22] See Reference entry in [ISO32000-2:2020, Table 255] and [ISO32000-2:2020, Table 256] as well as the transform methods ”DocMDP” and ”FieldMDP” in [ISO32000-2:2020, Section 12.8.2] for more information.

[23] See [ISO32000-1, Section 12.7.4.5] and [ISO32000-2:2020, Section 12.7.5.5].

[24] See [ISO32000-2:2020, Sections 12.5.6.19 and 12.5.2] and [ETSI-102778-6(v1.1.1), Section 5], for example.

[25] See [ISO32000-1, Section 12.7.4.5] and [ISO32000-2:2020, Section 12.8.3.3].

4.3.3.3 Attributes of PAdES signatures

PAdES signatures according to [ETSI-319142-1(v1.1.1)] utilise attributes defined in [ISO32000-1] and [ETSI-319122-1(v1.2.1)] and also define new attributes that are specific for PAdES signatures, which have in the meantime been included in [ISO32000-2:2020].

CMS and CAdES defined attributes

As specified in [ETSI-319142-1(v1.1.1), Section 5.2], the following CMS / CAdES attributes may be used to generate the DER-encoded `SignedData` object, which is inserted as value of the `Contents` key as explained above (cf. Section 4.3.3.2 (Construction of PAdES signatures)). Their syntax and semantics are as defined in [ETSI-319122-1(v1.2.1)] (cf. Section 4.3.1):

- `content-type`
- `message-digest`
- `signing certificate reference attributes`
  - `ESS signing-certificate`
  - `ESS signing-certificate-v2`
- `commitment-type-indication`
- `signer-attributes-v2`
- `content-time-stamp`
- `signature-policy-identifier`
- `signature-time-stamp`

ISO 32000-1 defined attributes

According to [ETSI-319142-1(v1.1.1), Section 5.3], the entries of a signature directory are as defined in [ISO32000-1, clause 12.8.1] and explained in Section 4.3.3.2 (Construction of PAdES signatures) above.

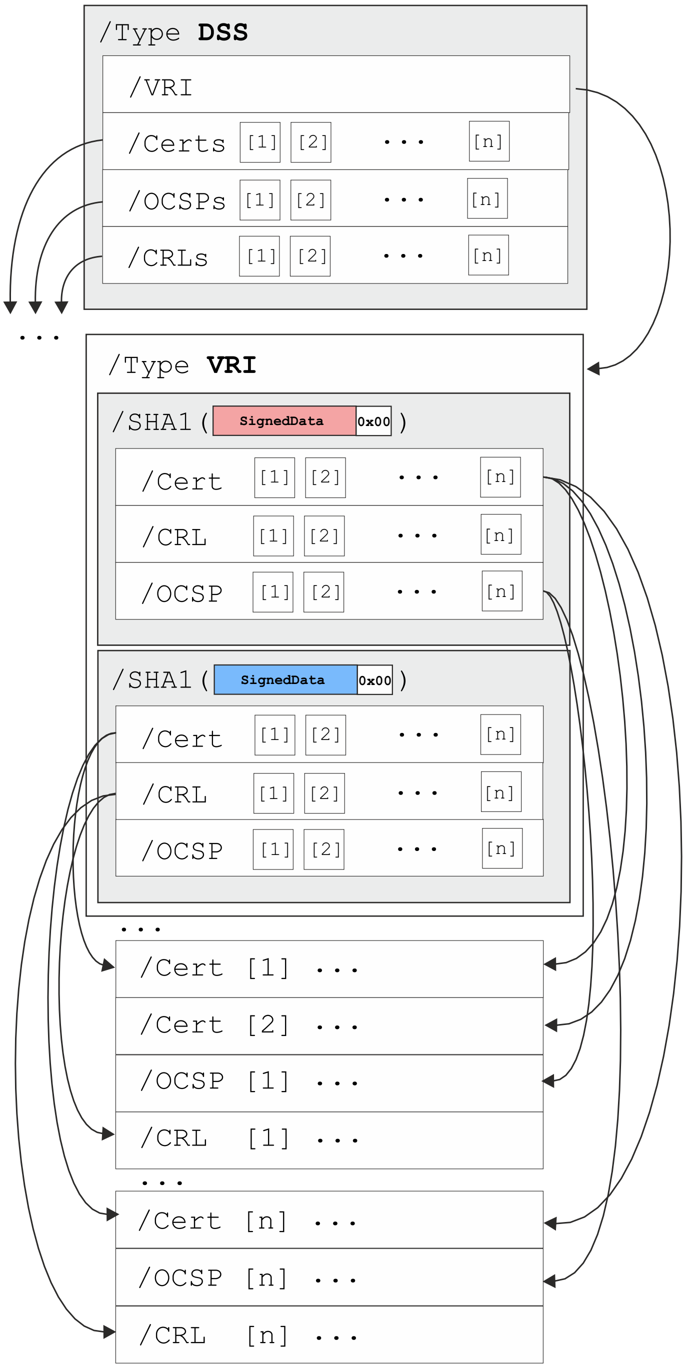


Figure 4.10: DSS and VRI dictionaries

Validation data and archive validation data attributes

As mentioned above, validation-related information has traditionally been treated as signed attribute of a PDF signature, and until the advent of PAdES there was no viable concept for long-term validation of PDF signatures. Against this background the PAdES specifications[26] introduced the following extensions to [ISO32000-1], which have been integrated in [ISO32000-2:2020] in the meantime:

- Document Security Store (DSS) Dictionary
- Validation Related Information (VRI) Dictionary
- Document Time-Stamp (DTS) Dictionary

Document Security Store (DSS) Dictionary
The Document Security Store (DSS) Dictionary[27] is an optional entry in the document catalog dictionary[28] of a PDF document and contains validation-related information for signatures and document time stamps.

As outlined in Figure 4.10 and specified in [ETSI-319142-1(v1.1.1), Clause 5.4.2.2] and [ISO32000-2:2020, Table 261], a DSS dictionary contains the following entries:

- `Type` (Optional): This key of type `Name` always has the value `DSS` for a DSS dictionary.
- `VRI` (Optional): This directory contains `VRI` dictionary entries, which refer to the validation-related information of a specific signature, whereas a `VRI` dictionary may refer to certificates, CRLs and OCSP responses. The key of each entry is the base-16-encoded SHA-1 digest of the value of the `Contents` entry of the corresponding signature (cf. Figure 4.9) or document time stamp (cf. Figure 4.11). For each successfully validated signature or document time stamp in the document, there is a corresponding signature `VRI` dictionary, which contains references to the corresponding validation related information (see paragraph Validation Related Information (VRI) Dictionary below).
- `Certs` (Optional): An array of indirect references to certificates. Each entry consists of one DER-encoded X.509 certificate (cf. [RFC5280]), which can be used in the process of validating a signature or document time stamp in the document.
- `OCSPs`(Optional): An array of indirect references to OSCP responses. Each entry consists of one DER-encoded OSCP response (cf. [RFC6960]), which can be used in the process of validating a signature or document time stamp in the document.
- `CRLs` (Optional): An array of indirect references to certificate revocation lists (CRLs). Each entry consists of one DER-encoded CRL (cf. [RFC5280]), which can be used in the process of validating a signature or document time stamp in the document.

Validation-Related Information (VRI) Dictionary
As outlined in Figure 4.10 and specified in [ETSI-319142-1(v1.1.1), Clause 5.4.2.3] and [ISO32000-2:2020, Table 262], a VRI dictionary contains the following entries:

- `Type` (Optional): This key of type `Name` always has the value `VRI` for a VRI dictionary.
- `Cert` (Optional): An array of indirect references to certificates. Each entry consists of one DER-encoded X.509 certificate (cf. [RFC5280]), which can be used in the process of validating a signature or document time stamp in the document.

[26] See [ETSI-102778-4(v1.1.2), Annex A] and [ETSI-319142-1(v1.1.1), Clause 5.4].
[27] See [ETSI-102778-4(v1.1.2), Annex A.1], [ETSI-319142-1(v1.1.1), Clause 5.4.2.1] and [ISO32000-2:2020, Clause 12.8.4.3].
[28] See [ISO32000-2:2020, Clause 7.7.2 and Table 29].

- CRL (Optional): An array of indirect references to CRLs. Each entry consists of one DER-encoded CRL (cf. [RFC5280]), which can be used in the process of validating a signature or document time stamp in the document.
- OCSP (Optional): An array of indirect references to OSCP responses. Each entry consists of one DER-encoded OSCP response (cf. [RFC6960]), which can be used in the process of validating a signature or document time stamp in the document.

Additionally, [ISO32000-2:2020, Table 262] specifies the following entries of a VRI dictionary. These, however, are *not recommended* to be used in PAdES signatures[29], especially if there is a subsequent document time stamp, which proves that the VRI directory was generated before the time indicated in the document time stamp.

- TU (Optional): The date/time at which this signature VRI dictionary was created. TU shall not be used, if TS is present. The date is formatted according to [ISO32000-1, Clause 7.9.4].
- TS (Optional): An [RFC5816] compliant DER-encoded time stamp token. Similar as with TU above, TS shall not be used if TU is present. As "the hash value to be contained in the timestamp token is left undefined", the potential advantage of using TS instead of TU may be questionable.

As all required validation-related information is already contained in the DSS dictionary, the use of VRI is optional and [ETSI-319142-1(v1.1.1), Clause 6.3, v)] recommends not use the VRI dictionary for PAdES-B-LT and PAdES-B-LTA signatures.

Document Time Stamp (DTS) Dictionary

A Document Time Stamp (DTS) dictionary is a signature dictionary as defined in [ISO32000-1, Clause 12.8.1] with the following changes as specified in [ETSI-319142-1(v1.1.1), Clause 5.4.3]:

- Type (Required): This key of type Name is always of value DocTimeStamp for a DTS dictionary.
- SubFilter (Required): The value should be ETSI.RFC3161.
- Contents (Required): If the value of SubFilter is ETSI.RFC3161, Contents is a TimeStampToken as defined in [RFC3161] and updated by [RFC5816] as a hexadecimal string (cf. [ISO32000-1][Clause 7.3.4.3]). The messageImprint entry of the TimeStampToken is a hash of the parts of the document that are specified by the ByteRange, which in case of PAdES is the entire document including the DTS dictionary but without the value of the Contents entry itself.
- V (Optional): Default value 0. Specifies the version of the Signature Dictionary format, which is 0 for DTS dictionaries.

The Document Time Stamp dictionary does not specify additional keys but only specific values, which shall be used instead of the values in a regular signature directory. The keys Name, M, Location, Reason, and ContactInfo should not be present in case of a Document Time Stamp dictionary.

[29] See requirements a) and b) in [ETSI-319142-1(v1.1.1), Clause 5.4.2.3].

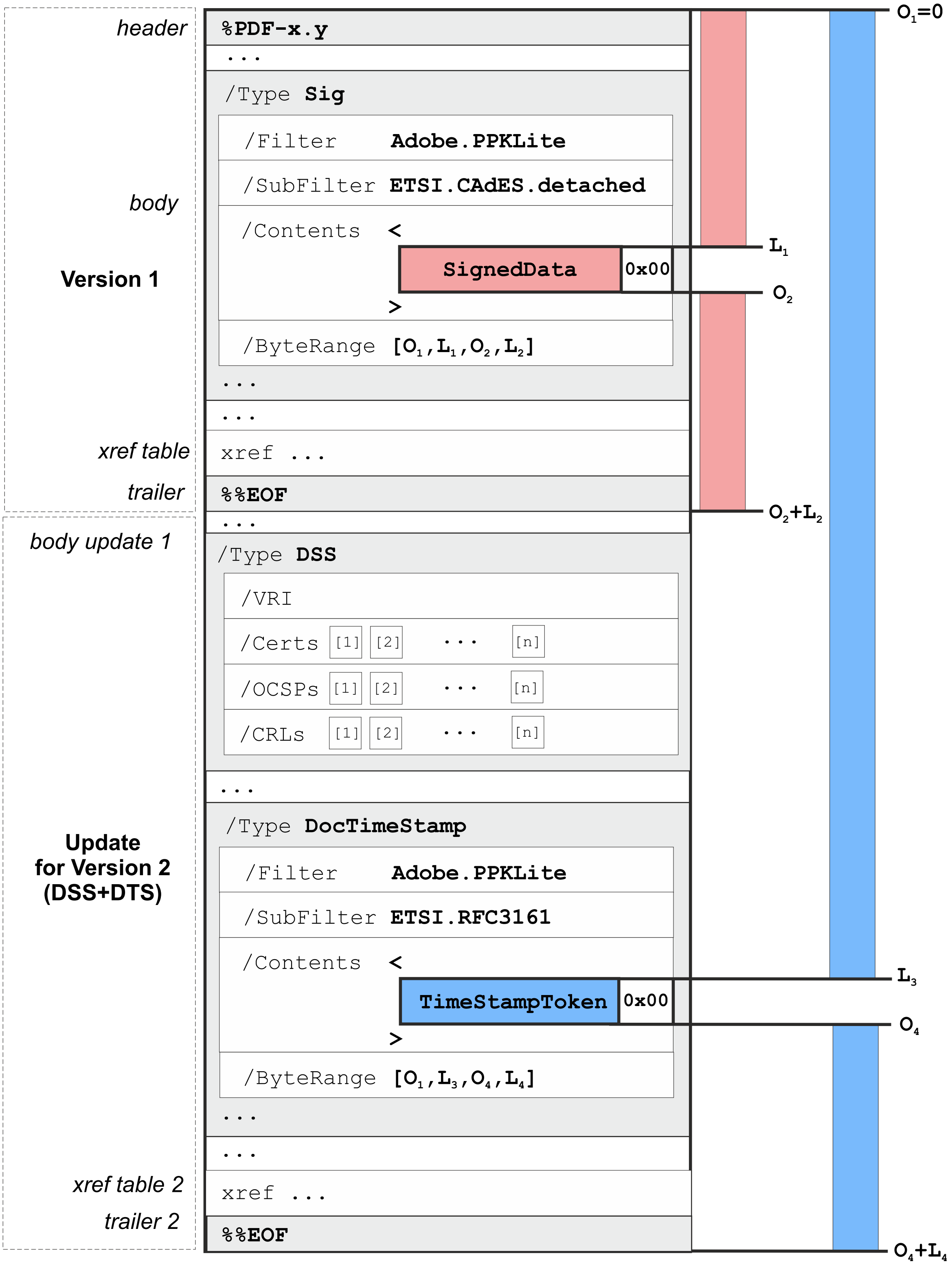


Figure 4.11: PDF Signature with Long-Term Validation (LTV)

Adding a DSS dictionary for the validation-related information and subsequently applying a DTS will yield a PDF signature with Long-Term Validation (LTV), as outlined in Figure 4.10. Applying this strategy over and over again with appropriate cryptographic algorithms allows the conclusiveness of a PDF signature to be preserved over very long periods of time, as explained in [ETSI-119512].

Encryption Requirements

If a document has to be encrypted and signed at the same time, the document has to be encrypted before a signature is computed for the document (cf. [ETSI-319142-1(v1.1.1), Clause 5.5]). Encryption includes all strings of a PDF document except: the ID entry in the trailer, strings in the `Encrypt` dictionary (cf. [ISO32000-2:2020, Clause 7.6]), already encrypted content or compressed object streams, hexadecimal strings, that contain the value of the `Contents` key in the signature dictionary.

PAdES Baseline Signatures

[ETSI-319142-1(v1.1.1)] provides four levels of baseline signatures:

- `B-B`: Requirements for inclusion of signed and unsigned attributes in the computed signature.
- `B-T`: Requirements for inclusion and computation of a trusted token for an existing signature. The trusted token shall prove that the signature in question actually existed at a certain date and time. As specified in [ETSI-319142-1(v1.1.1), Clause 6.3, n)] the trusted token may either be a `signature-time-stamp` attribute within the CAdES signature or a document time stamp as explained in paragraph Document Time-Stamp (DTS) Dictionary above.
- `B-LT`: Requirements for the inclusion of all relevant aspects in a signature document that are needed to validate a signature. The scope of this level is to provide a solution for the long term availability of the validation material.
- `B-LTA`: Requirements for including electronic time stamps in the document which enable the signature to be validated a long time after its creation. The scope of this level provides a solution for the long term availability as well as the long term integrity of the validation material.

The levels `B-LT` and `B-LTA` should be used if it has to be possible to validate the signature even if the certificate has expired, has been revoked or if the originally used algorithms have become obsolete. See [ETSI-319142-1(v1.1.1), Clause 6.3] for a complete table of the required attributes for each signature level.

Additional PAdES signature profiles

In addition to the baseline signatures specified in [ETSI-319142-1(v1.1.1)], the European Standard [ETSI-319142-2(v1.1.1)] defines additional profiles for digital signatures according to PAdES. These profiles use the `signature-policy-identifier` attribute of CAdES (cf. [ETSI-319142-2(v1.1.1), Clause 5.4]) or XAdES signatures to sign XML content within PDF containers (cf. [ETSI-319142-2(v1.1.1), Clause 6]).

4.3.4 Associated Signature Container (ASiC)

ASiC is defined in [ETSI-319162-1(v1.1.1)] as a ZIP-based container that includes a given collection of objects and the respective digital signatures or time assertions. Part two of the ETSI European Norm concerning ASiC (cf. [ETSI-319162-2(v1.1.1)]) defines additional containers for reasons of interoperability with existing mechanisms. An ASiC container has the following general structure:

- `root` folder, containing the `META-INF` folder and other folders with information about the content structure.
- `META-INF` folder in the root folder, containing files related to metadata of the signed content, including the respective signature or time assertion files.

Container types
In general, ASiC containers can include multiple types of signature or time assertion files.

An ASiC compliant signature consists either of a CAdES object (cf. Section 4.3.1) or one or more XAdES signatures (cf. Section 4.3.2) [ETSI-319162-1(v1.1.1), 4.1.2]. CAdES objects and XAdES signatures are mutually exclusive. The same principle applies to time assertion files, which can either be an [RFC3161] compliant time stamp token (TST) (cf. [ETSI-319422(v1.1.1)]) or an ASN.1 or XML-based Evidence Record as defined in [RFC4998] and [RFC6283], respectively.

ASiC containers are strictly compliant with the ZIP specification[30] and add additional restrictions to the specification. The following restrictions have to apply in order to generate an ASiC compliant container.

ASiC containers do not:

- use the multiple volumes split feature,
- encode file names in any other format than [ISO10646-1] compliant UTF-8,
- use any compression method except the methods ”stored/no compression” (value 0) or ”deflated” (value 8) (cf. [RFC1951]) from the ZIP specification.

[ETSI-319162-1(v1.1.1)] defines the two containers ASiC Simple (`ASiC-S`) and ASiC Extended (`ASiC-E`), which can be used for different signing mechanisms. Any one of these two container types has to be applied to generate an ASiC compliant signature container.

4.3.4.1 ASiC-S

ASiC-S containers associate one data file with either a signature file with one or more detached digital signatures or with a time assertion file. ASiC-S container types are defined for ASiC-S with XAdES signatures (cf. Section 4.3.2), CAdES signatures (cf. Section 4.3.1) and with time assertions.

ASiC-S media types
An ASiC-S compliant container has the file extension `.asics`. Alternatively, the file extension can be shortened to `.scs` if the operating system or file system does not allow file extensions longer than 3 characters. The file extension `.zip` is used if manual handling of the the content of the ASiC-S container is the goal.

[30] cf. https://pkware.cachefly.net/webdocs/APPNOTE/APPNOTE-6.3.3.TXT

ASiC-S container content

Besides a required single data file at the root level of the container, the META-INF folder contains one of the following required files that are providing information with regards to the file they are applied to:

- `timestamp.tst`: a TST compliant with [RFC3161] and its update [RFC5816]
- `signature.p7s`: CAdES signature (cf. Section 4.3.1)
- `signatures.xml`: XAdES signature (cf. Section 4.3.2)
- `evidencerecord.ers`: an Evidence Record according to [RFC4998]
- `evidencerecord.xml`: an Evidence Record according to [RFC6283]

Furthermore, the META-INF folder can contain various optional files (cf. [ETSI-319162-1(v1.1.1), Section 4.3.3.2]). Figure 4.12 shows a basic example of an ASiC-S container. [ETSI-319162-1(v1.1.1), Section 4.3.3] provides an additional example for nested containers.

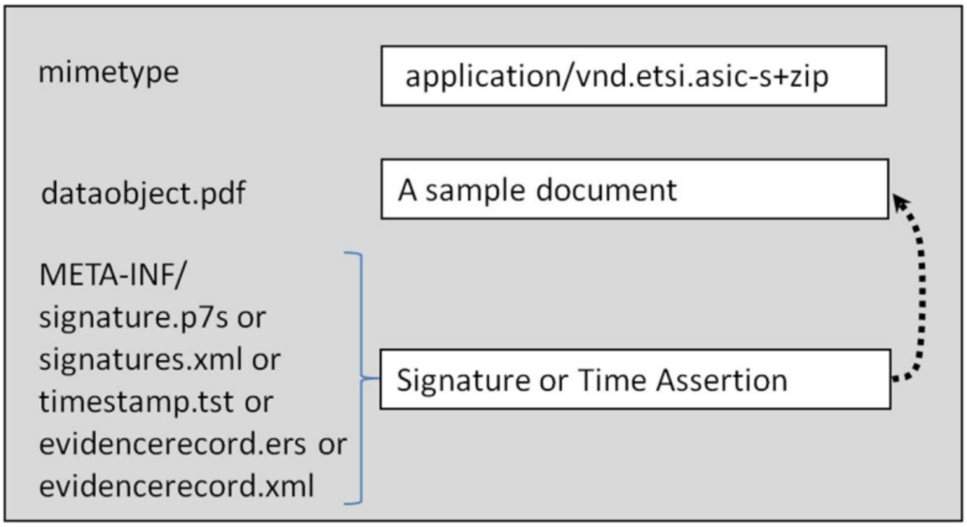


Figure 4.12: ASiC-S structure with a plain file object, cf. [ETSI-319162-1(v1.1.1)]

ASiC-S containers may also contain a `mimetype` file. If this file is present, the media type is `application/vnd.etsi.asic-s+zip` if the file extension is `.zip` (cf. Paragraph ASiC-S media types) or the signed file object has no media type specified. Otherwise, the media type is equivalent to the media type of the signed file.

ASiC-S long-term validation

”Long term availability and integrity of ASiC-S shall be achieved for the different container types as follows: 1) For ASiC-S containers with XAdES signatures and ASiC-S containers with CAdES signatures, the attributes specified in ETSI EN 319 122-1, ETSI EN 319 122-2, ETSI EN 319 132-1 and ETSI EN 319 132-2 shall be used for achieving long term availability and integrity. This shall apply to all the signatures present in the containers. 2) For ASiC-S containers with time stamp token one or more ASiCArchiveManifest files and one timestamp token for each ASiCArchiveManifest file applied to its content shall be added to the container following the rules specified in clause A.7. 3) For ASiC-S containers with ER, the internal mechanism of IETF RFC 4998 [8] and IETF RFC 6283 [9] shall be used.”[ETSI-319162-1(v1.1.1)]

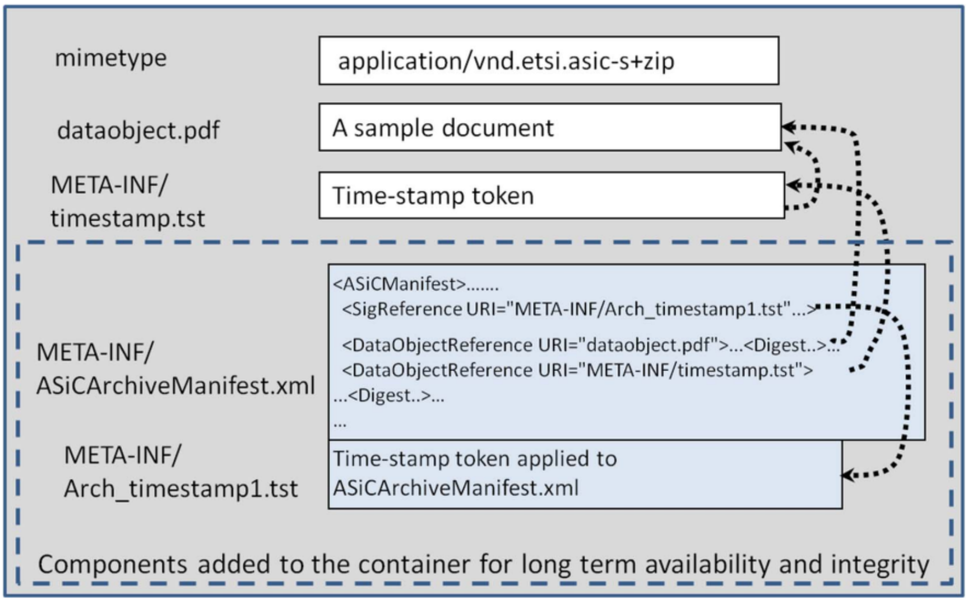

mimetype
application/vnd.etsi.asic-s+zip
dataobject.pdf
A sample document
META-INF/
timestamp.tst
Time-stamp token
META-INF/
ASiCArchiveManifest.xml
<ASiCManifest>.......
<SigReference URI="META-INF/Arch_timestamp1.tst"...>
<DataObjectReference URI="dataobject.pdf">...<Digest..>...
<DataObjectReference URI="META-INF/timestamp.tst">
...<Digest..>...
...
META-INF/
Arch_timestamp1.tst
Time-stamp token applied to
ASiCArchiveManifest.xml
Components added to the container for long term availability and integrity

Figure 4.13: ASiC-S with TST and long-term components, cf. [ETSI-319162-1(v1.1.1)]

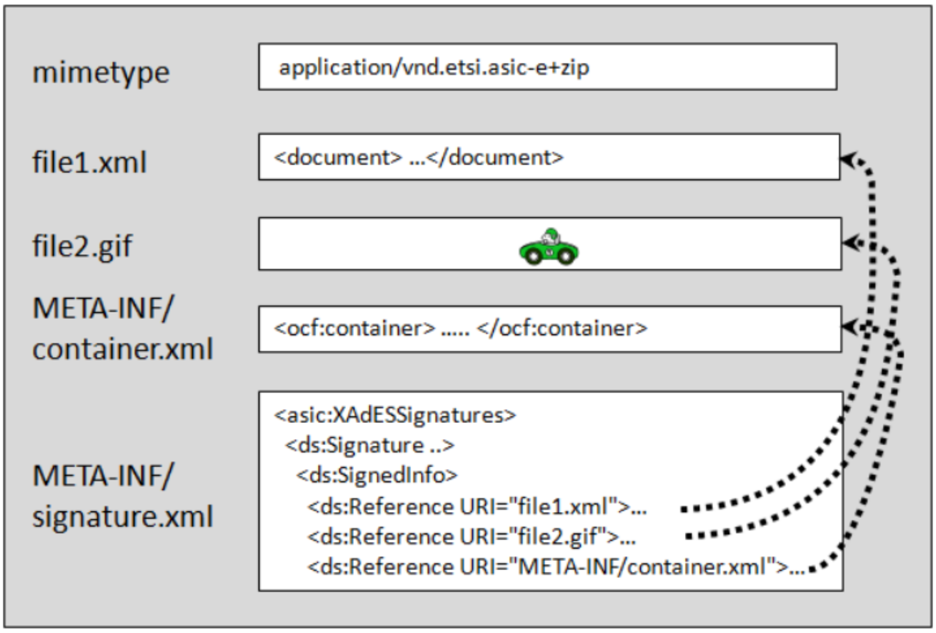
4.3.4.2 ASiC-E

While ASiC-S only supports the association of one data file with either a signature or a time assertion file, ASiC-E containers enable the correlation between one or more signature and time assertion files and one or more file objects. File objects included in the container can be associated with additional (metadata) information, which themselves can be associated with a signature in the container. Furthermore, ASiC-E containers can either be extended by additional objects after the first creation, or can be defined as not modifiable, preventing any future changes to the container. ASiC-E containers are defined for XAdES signatures (cf. Section 4.3.2) and for CAdES signatures (cf. Section 4.3.1) or time assertions (cf. Section 4.4 or 4.5).

General Requirements

ASiC-E containers with XAdES or CAdES have the file extensions .asice. Alternatively, the file extension can be shortened to .sce if the operating system or file system does not allow file extensions longer than 3 characters. The mimetype of an ASiC-E container is identical with the original media type of the container or is application/vnd.etsi.asic-e+zip if no media type has been defined. The .ZIP file comment[31] may also be of type application/vnd.etsi.asic-e+zip.

ASiC-E with XAdES



mimetype
application/vnd.etsi.asic-e+zip
file1.xml
<document> ...</document>
file2.gif
META-INF/
container.xml
<ocf:container> ..... </ocf:container>
META-INF/
signature.xml
<asic:XAdESSignatures>
<ds:Signature ..>
<ds:SignedInfo>
<ds:Reference URI="file1.xml">...
<ds:Reference URI="file2.gif">...
<ds:Reference URI="META-INF/container.xml">...

Figure 4.14: ASiC-E with XAdES and direct ds:reference usage, cf. [ETSI-319162-1(v1.1.1)]

Besides the optional mimetype file, the following internal structure is defined for the contents of the META-INF of ASiC-E containers with XAdES signatures:

- a signatures.xml with one of the following possibilities as root element:
  - asic:XAdESSignatures
  - document-signatures according to ODF[32]
  - signatures according to OCF[33]
  - ds:Signature as defined in [XML-DSig]
  - an element from another namespace only if it is a sibling of ds:Signature
- If any number of ASiCEvidenceRecordManifest files are present, they are represented by one ASiCManifest element that has to be compliant with [ETSI-319162-1(v1.1.1), Annex A.4] and also includes an Evidence Record with the same digest algorithm as was used for the initial Archive Time-stamp for the first ReducedHashTree according to [RFC4998] or [RFC6283].

[31] cf. https://pkware.cachefly.net/webdocs/APPNOTE/APPNOTE-6.3.3.TXT
[32] http://docs.oasis-open.org/office/v1.2/OpenDocument-v1.2-part3.pdf
[33] http://www.idpf.org/epub/30/spec/epub30-ocf.html

- The Evidence Record applies to all the container files referenced by the `ASiCManifest` and shall be named `evidencerecord.ers` (cf. [RFC4998]) or `evidencerecord.xml` (cf. [RFC6283]).
- The `META-INF` folder can also contain other application specific files (cf. [ETSI-319162-1(v1.1.1), Section 4.4.3.2]).

Figure 4.14 depicts the typical structure of an ASiC-E with XAdES container using [XML-DSig].

ASiC-E with CAdES - time assertions

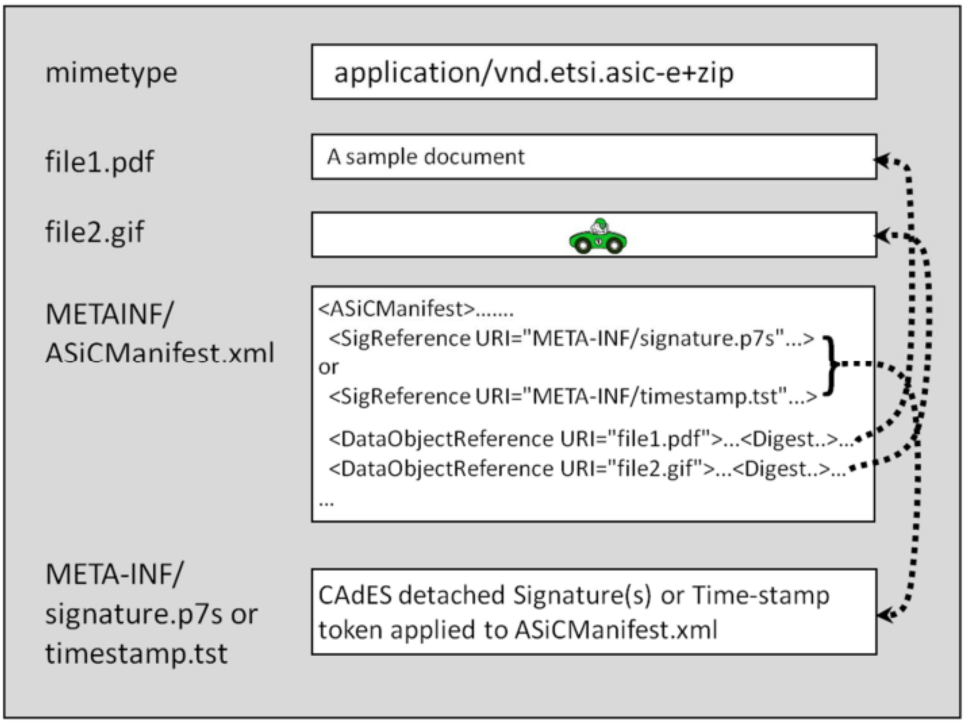

mimetype
application/vnd.etsi.asic-e+zip
file1.pdf
A sample document
file2.gif
METAINF/
ASiCManifest.xml
<ASiCManifest>.......
<SigReference URI="META-INF/signature.p7s"...>
or
<SigReference URI="META-INF/timestamp.tst"...>
<DataObjectReference URI="file1.pdf">...<Digest..>...
<DataObjectReference URI="file2.gif">...<Digest..>...
...
META-INF/
signature.p7s or
timestamp.tst
CAdES detached Signature(s) or Time-stamp
token applied to ASiCManifest.xml

Figure 4.15: ASiC-E with CAdES signature or TST, cf. [ETSI-319162-1(v1.1.1)]

Besides the optional `mimetype` file, the following internal structure is defined for the contents of the `META-INF` folder of ASiC-E containers with CAdES signature or ”time assertions” in [ETSI-319162-1(v1.1.1)]. An ASiC-E container also contains one or more `ASiCManifest` or `ASiCEvidenceRecordManifest` files. Both file types can be included in one ASiC-E container at the same time.

- Each `ASiCManifest` file correlates to one TST or signature in the `META-INF` folder:
  - `*signature*.p7s` is a signature file that contains one CAdES object that includes at least one detached CAdES signature according to [ETSI-319122-1(v1.2.1)] or [ETSI-319122-2(v1.1.1)], applied to the `ASiCManifest` file,
  - `*timestamp*.tst` is a time stamp token according to [RFC3161] and updated by [RFC5816], applied to the `ASiCManifest` file.
- For each `ASiCEvidenceRecordManifest` file one ER file shall be present in the `META-INF` folder:
  - `*evidencerecord*.ers` according to [RFC4998] that applies to the file object specified in the `ASiCManifest` file,

  - *evidencerecord*.xml according to [RFC6283] that applies to the file object specified in the ASiCManifest file.

- Validation applications have to verify for each ASiCManifest*.xml file in the META-INF folder that the content is compliant with [ETSI-319162-1(v1.1.1), Annex A.4] and identify the URI attribute of the SigReference element as the correlating signature.
- If the signature reference is of the type *signature*.p7s (CAdES signature) or of type *timestamp*.tst (time stamp token), the referenced object is validated against the ASiCManifest file.
- Validation applications have to verify for each ASiCEvidenceRecordManifest*.xml file in the META-INF folder that the content is compliant with [ETSI-319162-1(v1.1.1), Annex A.4] and identify the URI attribute of the SigReference element that points to *evidencerecord*.ers or *evidencerecord.xml.
- Afterwards, the referenced ER is validated against all the ds:DigestValue objects in DataObjectReference in the ASiCManifest file.
- An error has to be raised whenever a digest value mismatch is detected within any ds:DigestValue in DataObjectReference and the digest computed over the referenced file object [ETSI-319162-1(v1.1.1)].

Figure 4.15 depicts the typical structure of an ASiC-E with a CAdES signature or time stamp token. Figure 4.16 from [ETSI-319162-1(v1.1.1), Figure 8] shows an example of an ASiC-E with Evidence Record.

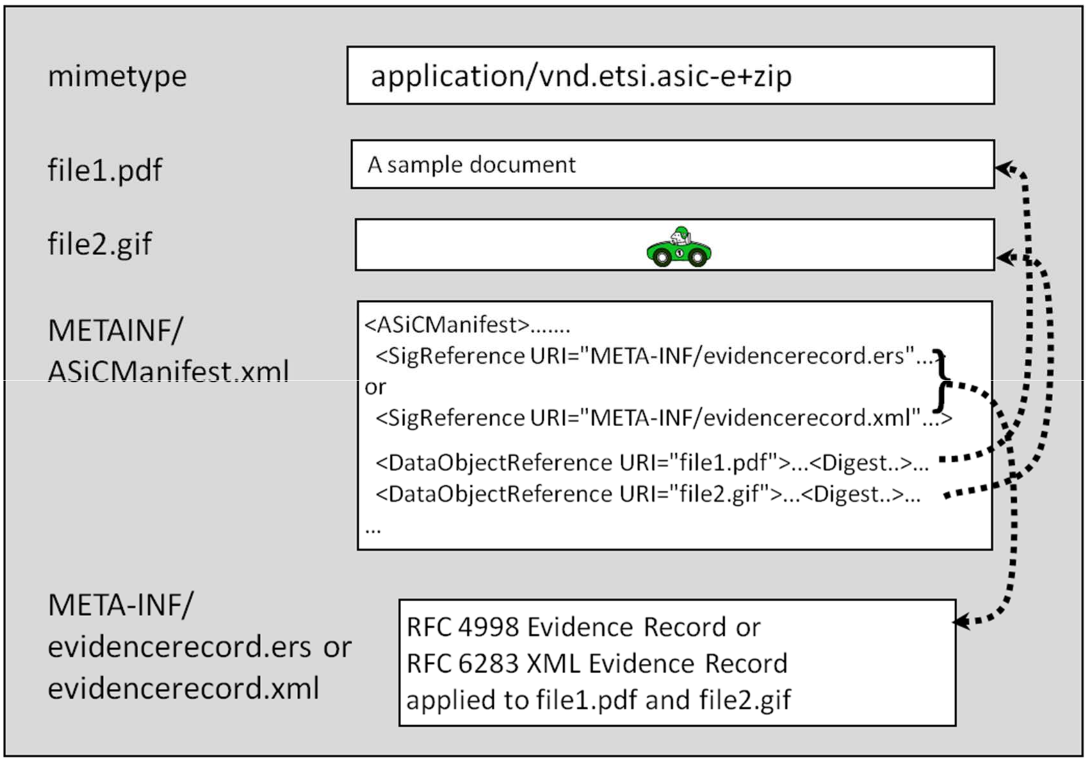

mimetype
application/vnd.etsi.asic-e+zip
file1.pdf
A sample document
file2.gif
METAINF/
ASiCManifest.xml
<ASiCManifest>.......
<SigReference URI="META-INF/evidencerecord.ers"...
or
<SigReference URI="META-INF/evidencerecord.xml"...>
<DataObjectReference URI="file1.pdf">...<Digest..>...
<DataObjectReference URI="file2.gif">...<Digest..>...
...
META-INF/
evidencerecord.ers or
evidencerecord.xml
RFC 4998 Evidence Record or
RFC 6283 XML Evidence Record
applied to file1.pdf and file2.gif

Figure 4.16: ASIC-E with Evidence Record, cf. [ETSI-319162-1(v1.1.1)]

Long-term availability and integrity of ASiC-E

Long-term availability and integrity of ASiC-E containers are achieved for the different container types as follows:

1.) For ASiC-E containers with XAdES signatures, the mechanisms specified in [ETSI-319132-1(v1.2.1)] and [ETSI-319132-2(v1.1.1)] or in the evidence record specifications [RFC4998] and [RFC6283] apply.

2.) For ASiC-E containers with CAdES time assertion, one may either add at least one `ASiCArchiveManifest` and correlating time stamp token to the container according to [ETSI-319162-1(v1.1.1), Annex A.7], or one may add at least one `ASiCEvidenceRecordManifest` file which applies to all signed data, time-asserted data, signatures and/or TST files.

3.) For ASiC-E containers with ER, the internal mechanisms of [RFC4998] and [RFC6283] shall be used.

ASiC baseline containers

ASiC containers have to be compliant with the algorithm and key length requirements in [ETSI-119312(v1.4.2)]. Further general and ASiC baseline container specific requirements are specified in [ETSI-319162-1(v1.1.1), Sections 5.2 and 5.3]. The normative Annex A of [ETSI-319162-1(v1.1.1)] provides detailed information about the different components of ASiC containers.

ASiC levels

[ETSI-319162-1(v1.1.1)] defines the four levels of baseline containers `B-B`, `B-T`, `B-LT` and `B-LTA`, which have to be compliant with the respective levels of baseline signatures from either CAdES (cf. Section 4.3.1) or XAdES (cf. Section 4.3.2).

The ASiC levels apply to ASiC-S in combination with CAdES and XAdES.

Additional ASiC containers

In addition to the baseline containers specified in [ETSI-319162-1(v1.1.1)], the European standard [ETSI-319162-2(v1.1.1)] defines various other ASiC compliant containers. In particular [ETSI-319162-2(v1.1.1), Annex B.3] outlines how ASiC may be used to create containers for evidence record based archival systems.

4.4 Time Stamp

”’Electronic time stamp’ means data in electronic form which binds other data in electronic form to a particular time establishing evidence that the latter data existed at that time” [(EU)910/2014, Article 3 (33)]. A qualified electronic time stamp pursuant to [(EU)910/2014, Article 3], ”shall enjoy the presumption of the accuracy of the date and the time it indicates and the integrity of the data to which the date and time are bound” [(EU)910/2014, Article 41 (2)]. Qualified time stamps are signed using an advanced electronic signature or sealed with an advanced electronic seal of the qualified trust service provider. Therefore, qualified electronic time stamps establish the evidence of integrity and proof of existence of the data at a specific date/time.

Principally, time stamps can be created by the user himself or be obtained – for example by means of a Time Stamp Protocol (see 5.1.4.1) – from a trustworthy Trust Service Provider acting as a Time Stamp Authority (TSA) . The two cases differ particularly with regard to the conclusiveness of the time stamp. A time stamp created by the user himself has, under certain circumstances less conclusiveness. However, if the time stamp is generated by a Time Stamp Authority as a qualified time stamp pursuant to [(EU)910/2014, Article 42], then this qualified time stamp has very high conclusiveness in court.

TimeStampToken ”is defined as a ContentInfo ([CMS]) and SHALL encapsulate a signed data content type.

```
TimeStampToken ::= ContentInfo
-- [RFC3161]
-- contentType is id-signedData ([RFC5652])
-- content is SignedData ([RFC5652])
```

The fields of type EncapsulatedContentInfo of the SignedData construct have the following meanings:

`eContentType` is an object identifier that uniquely specifies the content type. For a time stamp token it is defined as:

```
id-ct-TSTInfo OBJECT IDENTIFIER ::= { iso(1) member-body(2)
us(840) rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) ct(1) 4}
```

`eContent` is the content itself, carried as an octet string. The eContent SHALL be the DER-encoded value of TSTInfo.

The time stamp token MUST NOT contain any signatures other than the signature of the Time Stamp Authority. The certificate identifier (ESSCertID) of the TSA certificate MUST be included as a signerInfo attribute inside a SigningCertificate attribute.” [RFC3161]

[RFC3161] is updated by[RFC5816] in order also to allow `ESSCertIDv2` to be used to include an identifier of the signing certificate as defined in [RFC5035], in case that another hash algorithm than “SHA-1” is used.

More details concerning the format of a time stamp can be found in [RFC3161], updated by [RFC5816] or [ISO14533-1].

4.5 Evidence Record Syntax

Whilst the power of evidence of qualified electronic signatures basically can be maintained by using archive time stamps in accordance with CAdES (Section 4.3.1 Cryptographic Message Syntax (CMS) and CAdES digital signatures) (see also [ETSI-319122-1(v1.2.1)]) or XAdES (Section 4.3.2 XML-Digital Signature and XAdES Signatures) (see [ETSI-319132-1(v1.2.1)]), unfortunately such systems would be less scalable because an additional time stamp will be required for every single archive data object in case of renewal of the time stamp. That is why an alternative method using Merkle hash trees [Merk80] was proposed within the scope of the ArchiSig project [RoSc05], requiring, irrespective of the number of documents to be protected, only a single time stamp in case of renewal of a time stamp. These approaches have been standardised as ASN.1-based ”Evidence Record Syntax” (ERS) in [RFC4998] and as XML-based ”Evidence Record Syntax” (ERS) in [RFC6283]. Figure 4.18 shows such a hash tree.

The Evidence Record as a reduced Merkle Hashtree is shown in 4.19:

Pursuant to the Evidence Record Syntax (ERS) Standard of the IETF [RFC4998] or [RFC6283] , an Evidence Record is a unit of data with which the existence of stored data and documents at a defined point in time may be technically proven. It includes cryptographic Evidence Records with which the integrity and authenticity of electronically saved data and documents may be verified at all times. Technically, the ERS standard is based on the approach that cryptographic checksums (hash values) of the archive data objects are organised in a hash tree (pursuant to Merkle [Merk80]) when stored in the archive or preservation system as cryptographically unique representatives of the data to be stored and that the roots of the hash tree are secured (”sealed”) with a qualified time stamp for proving the integrity (see also Annex [BSI-TR-03125-M3]). This first initial time stamp is also referred to as initial archive time stamp pursuant to the ERS standard [RFC4998] or [RFC6283].

The source of trust for the archive time stamp and thus for legally compliant re-signing or re-sealing or time stamp renewal pursuant to § 15 German Trust Service Law (Vertrauensdienstegesetz [VDG]) is the qualified time stamp. Its data structure should fulfil the requirements of the ”TimeStamp Protocol (TSP)” [RFC3161], updated by [RFC5816], and the ”Cryptographic Message Syntax (CMS)” pursuant to [RFC5652] and [ETSI-319422(v1.1.1)] and [ISO14533-1].

In the case that re-signing or re-sealing or a time stamp renewal is necessary that is sufficient if only the digital signature procedure threatens to lose its suitability as a security measure, but the hash algorithm remains suitable, a new archive time stamp includes the hash value of the original time stamp in the hash tree that is to be generated with a new final qualified time stamp so that a secure and verifiable, chronological chain of evidence made of cryptographically linked archive time stamps arises. The Evidence Record resulting from this contains an additional ArchiveTimeStamp element in the ArchiveTimeStampChain element pursuant to [RFC4998] or [RFC6283] that has already existed beforehand.

If the suitability of the used hash algorithm as a security measure is (also) threatened, the hash tree shall be renewed. In doing so, the archive data object is hashed with a suitable algorithm and a new ArchiveTimestampChain element with a corresponding ArchiveTimeStamp element is inserted into the ArchiveTimeStampSequence element. Further information in this respect may also be found in [RFC4998] or [RFC6283].

The technical proof of the maintenance of the integrity and therefore, if necessary, the authenticity of the data stored in the electronic long-term storage, then occurs, along with the presentation of the actual archival data and the associated, valid certificates of existing digital signatures, in particular with the proof of the integrity of the cryptographic representatives of the archive data objects, i.e., the hash values and archive time stamps.

For these purposes, the ERS standard specifies a so-called Evidence Record. This Evidence Record contains in particular a sequence of ArchiveTimestamps with which the integrity and authenticity of the archive data objects may be proven. An ArchiveTimestamp, in turn, contains all necessary data from the hash tree (reducedHashTree) needed for verifying that the archive data object belongs to the hash tree. The root of the hash tree is sealed with a qualified timestamp (see also Annex M.3 [BSI-TR-03125-M3]) ([BSI-TR-03125-F], clause 5.5).

4.5.1 ASN.1-based Evidence Record pursuant to RFC4998

Evidence Records pursuant to RFC4998 have the following ASN.1 Syntax.

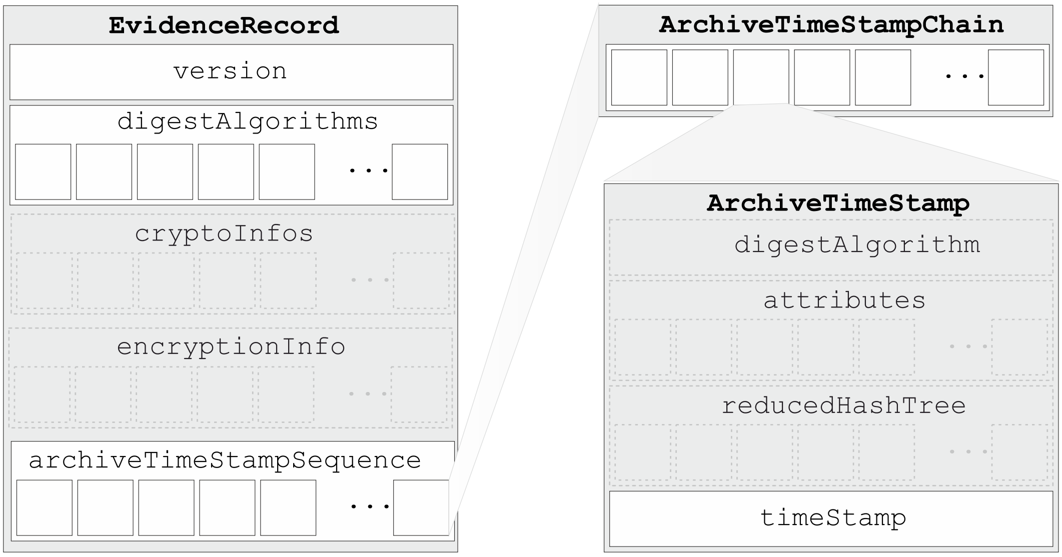


Figure 4.17: The Evidence Record

```
EvidenceRecord ::= SEQUENCE {
        version                     INTEGER { v1(1) },
        digestAlgorithms            SEQUENCE OF AlgorithmIdentifier,
        cryptoInfos                 [0] CryptoInfos OPTIONAL,
        encryptionInfo              [1] EncryptionInfo OPTIONAL,
        archiveTimeStampSequence    ArchiveTimeStampSequence
}
CryptoInfos ::= SEQUENCE SIZE (1..MAX) OF Attribute
```

- `version`
  contains the version of the structure and equals '1' in accordance with [RFC4998, Section 3],

- `digestAlgorithms`
  contains a list of all hash algorithms to generate the hash values of the data objects during the retention period.

- `cryptoInfos` (optional)
  may contain a sequence of information for the verification of the `archiveTimeStampSequence`. This may include e.g. anchors of confidence, certificates, revocation information or information on the suitability of cryptographic algorithms in accordance with [RFC5698],

- `encryptionInfo` (optional)
  may include necessary information on the correct handling of encrypted contents.

- `archiveTimeStampSequence`
  contains a sequence of `ArchiveTimeStampChain` elements that, in turn, consist of a sequence of `ArchiveTimeStamp` elements.
  ```
  ArchiveTimeStampSequence ::= SEQUENCE OF ArchiveTimeStampChain
  ```

ArchiveTimestampChain
An ArchiveTimestampChain element contains a sequence of at least one or several elements of the ArchiveTimeStamp type that are sorted in ascending order according to the time of the time stamp contained therein.

```
ArchiveTimeStampChain ::= SEQUENCE OF ArchiveTimeStamp
```

ArchiveTimeStampChain and ArchiveTimeStampSequence are chronologically ordered ascending by time of timestamp. Within an archive time stamp chain, all reduced hash trees are based on the same hash algorithm. (see see [RFC4998, clause 5.1.])

ArchiveTimeStamp
Pursuant to the IETF's ERS standard, an archive time stamp is defined as follows:

```
ArchiveTimeStamp ::= SEQUENCE {
       digestAlgorithm    [0] AlgorithmIdentifier OPTIONAL,
       attributes         [1] Attributes OPTIONAL,
       reducedHashtree    [2] SEQUENCE OF PartialHashtree OPTIONAL,
       timeStamp          ContentInfo -- TimeStampToken pursuant to [RFC3161]
}
```

- digestAlgorithm (optional)
  identifies the hash algorithm used. If the field is not available, the ERS standard assumes that the hash algorithm of the time stamp was used for the generation of the hash value.

- attributes(optional)
  may contain additional information on the rules applied to re-signing or re-sealing or the application of the archive time stamp.

- reducedHashtree(optional)
  contains all hash values that are needed for the mathematical verification of the hash value nodes into which the original hash value of the archive data object and the final qualified time stamp has been incorporated.

- timeStamp
  includes a qualified time stamp that was generated as cryptographic confirmation of the existence of the data returned with the Evidence Record, using an advanced electronic signature or seal. The format of the TimeStampToken is described in Section 4.4. (see [BSI-TR-03125-F])

4.5.1.1 Generation of Evidence Record

To generate an EvidenceRecord element, three steps are required.

1.) Selection of a data object or data object group to be preserved

2.) Creation of the initial ArchiveTimeStamp (see below)

3.) Refresh the ArchiveTimeStamp when necessary (Can be achieved by a renewal of the hash tree or the Timestamp)

4.5.1.2 Verification of Evidence Record

To verify an Evidence Record the data object or data object group needs to be selected. If the encryption field is used, the data object or data object group needs to be re-encrypted. In the last step the ArchiveTimestampSequence is verified.

4.5.1.3 Generation of an ArchiveTimestamp

The generation is based on Merkle [Merk80] hash trees. To build such a hash tree for the data objects to be timestamped, a secure hash algorithm has to be chosen. With this algorithm the leaves of the hash tree are generated, which represent the data by a hash value. If there are multiple documents in one group the hashes (of each document) are sorted in binary ascending order. If there is more than one hash value these values are placed in groups (binary sorted ascending) and then concatenated. New hash values are now generated and build the inner nods of the hash tree. This step can be repeated until there is only one hash value left. This value is called the root node of the hash tree. For this root hash value a timestamp is obtained. Here the hash algorithm in the timestamp request must be the same as the hash algorithm used in the hash tree. Otherwise the `digestAlgorithm` in the `ArchiveTimeStamp` must be present to define the used algorithm.

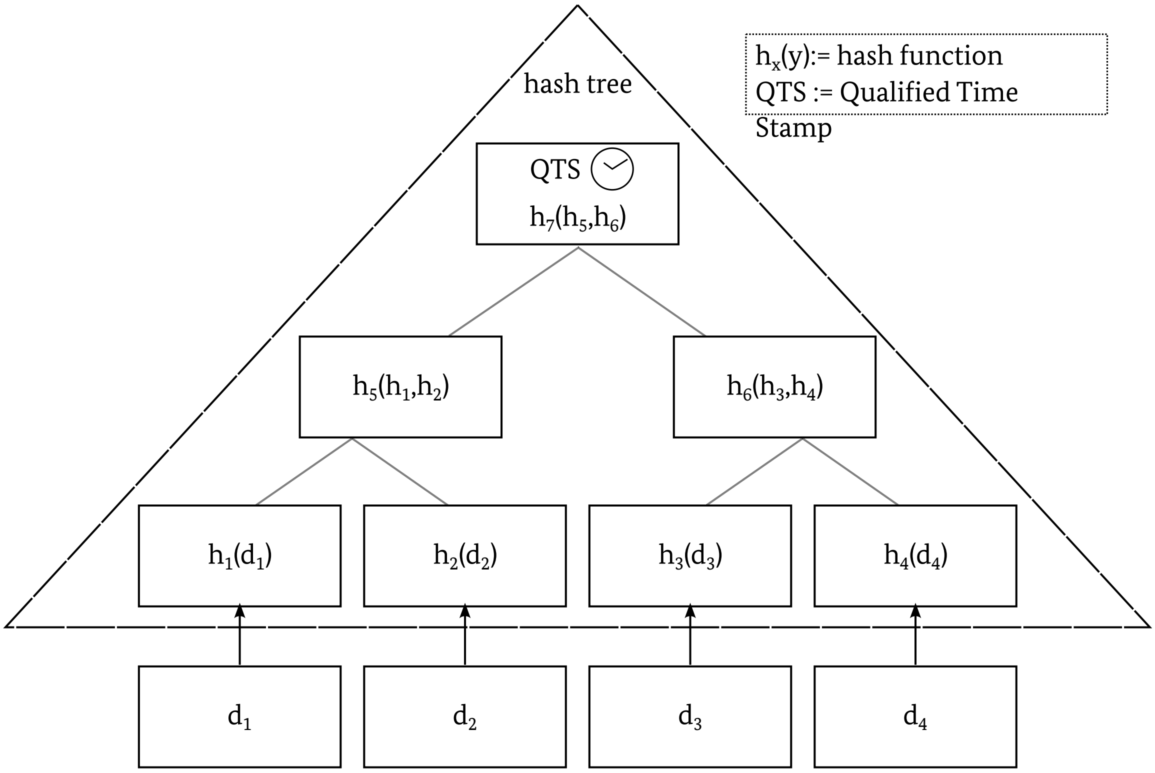


Figure 4.18: Merkle-Hashtree for Preservation of data, documents and (qualified) electronic signatures, seals and timestamps

Proof of existence for a single data object by creating a reduced hash tree
In order to get the proof of existence (of an specific single data object) the generation of the reduced hash tree (see figure 4.19) is necessary. The following steps describe this process. To prove the existence for a specific single data object the hash tree can be reduced exclusory to the hash values, necessary to produce a proof of existence for a single data object.

1.) In the first step, generate the hash value h of this specific data object using the hash algorithm of the hash tree. “

2.) In the next step, all hash values are selected, which have the same father node as h. “Generate the first list of hash values by arranging these hashes in binary ascending order. This will be stored in the structure of the PartialHashtree. Repeat this step for the father node of all hashes until the root hash is reached. The father nodes themselves are not saved in the hash lists – they are computable.” ([RFC4998], clause 4.2)

3.) The reducedHashtree consists of the list of all PartialHashtrees.

4.) Create an ArchiveTimestamp (ATS) by adding the timestamp and the info concerning the hash algorithm

As example, see the Evidence Record in figure 36 with the PartialHashtrees

$$\begin{aligned} \text{Phts1} &= seq(h_1 = \text{Hash}(d_1), h_2 = \text{Hash}(d_2)) \\ \text{Phts2} &= seq(h_6 = \text{Hash}(h_3||h_4)) \end{aligned}$$

[RFC4998]

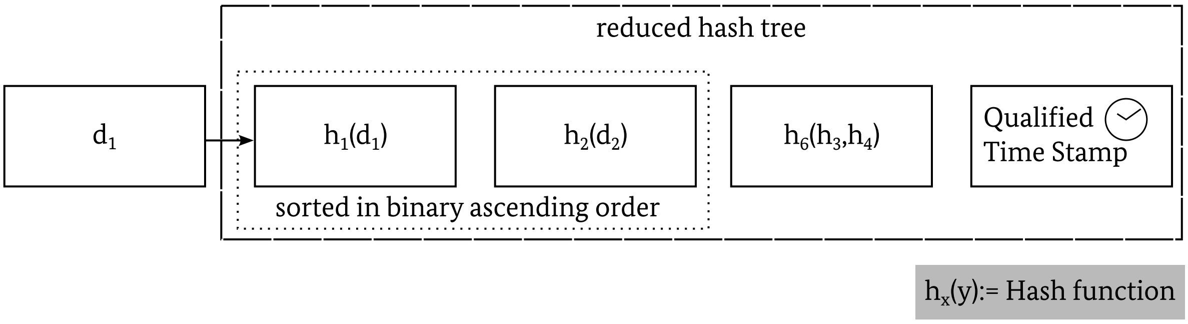


Figure 4.19: Evidence Record with reduced Merkle-Hashtree

4.5.1.4 Verification of an ArchiveTimestamp

”An Archive Timestamp shall prove that a data object existed at a certain time, given by timestamp. This can be verified as follows:

1.) Calculate hash value h of the data object with hash algorithm H given in field digestAlgorithm of the Archive Timestamp.

2.) Search for hash value h in the first list (partialHashtree) of reducedHashtree. If not present, terminate verification process with negative result.

3.) Concatenate the hash values of the actual list (partialHashtree) of hash values in binary ascending order and calculate the hash value h’ with algorithm H. This hash value h’ MUST become a member of the next higher list of hash values (from the next partialHashtree). Continue step 3 until a root hash value is calculated.

4.) Check timestamp. In case of a timestamp according to [RFC3161], the root hash value must correspond to `hashedMessage`, and `digestAlgorithm` must correspond to `hashAlgorithm` field, both in `messageImprint` field of `timeStampToken`. In case of other timestamp formats, the hash value and digestAlgorithm must also correspond to their equivalent fields if they exist.” [RFC4998, clause 4.3]

4.5.1.5 Generation of ArchiveTimestampChain and ArchiveTimespampSequence

The (first) `ArchiveTimestamp` in the (first) `ArchiveTimeStampChain` becomes invalid,

- if hash algorithms or public key algorithms used in its hash tree or the timestamp become weak or
- if the validity period of the timestamp authority certificate expires or is revoked.

Prior to such an event, the ArchiveTimestamp or hash tree shall be renewed. Depending on whether the

- timestamp becomes invalid (renewal of the time stamp) or
- the hash algorithm of the hash tree becomes weak (renewal of the hash tree),

Renewal of the time stamp
If only the signature algorithm used for creating the time stamp is threatened with expiration but the employed hash algorithm can be maintained, it is sufficient to renew the time stamp. In this case, a new time stamp is created by using last existing `ArchiveTimeStamp` and is added into the same `ArchiveTimeStampChain`.

Renewal of the hash tree
If the hash algorithm used for creation of the hash tree is threatened with expiration, a renewal of the hash tree shall be performed. Here, the old `ArchiveTimestamps` and all data objects are subject to calculation of new hash values and determination of the corresponding root of the hash tree, and are then provided with a `ArchiveTimestamp`. The newly created `ArchiveTimestamp` is stored in a new `ArchiveTimestampChain`. One or more `ArchiveTimestampChains` for a data object or data object group form an `ArchiveTimestampsSequence`. When creating an Evidence Record, the reduced hash tree for the corresponding data object and/or a group of linked data objects will be derived from the whole hash tree and then inserted into a fresh `ArchiveTimeStampChain`.

4.5.1.6 Verification of ArchiveTimestampChain und ArchiveTimestampSequence

In order to complete the non-repudiation proof for an archive object, the last ArchiveTime-Stamp (ATS) has to be valid and `ArchiveTimeStampChains` (ATSCs) and their relations to each other have to be proved. Therefore the following steps are necessary:

1.) ”Verify that the first `ArchiveTimestamp` of the first `ArchiveTimestampChain` (the initial `ArchiveTimeStamp`) of the Evidence Record contains the hash value of the preserved data object or data object group.

2.) Verify that concerning each `ArchiveTimestampChain` the first hash value list of each `ArchiveTimestamp` (except the initial `ArchiveTimeStamp`) shall contain the hash value of the Timestamp of the previous `ArchiveTimestamp`. Each `ArchiveTimestamp` shall be valid relative to the time of the following `ArchiveTimestamp`. All `ArchiveTimestamps` within an `ArchiveTimeStampChain` shall use the same hash algorithm and this algorithm shall be secure at the time of the first Archive Time-stamp of the following `ArchiveTimestampChain`.

3.) “Verify that the first hash value list (partialHashtree) of the first `ArchiveTimestamp` of all other `ArchiveTimestampChains` of the Evidence Record contains a hash value of the concatenation of the data object hash and the hash value of all older `ArchiveTimestamp` Chain. Verify that this `ArchiveTimestamp` was generated before the last `ArchiveTimestamp` of the Archive Time-stamp Chain became invalid.

In order to complete the non-repudiation proof for the data objects, the last `ArchiveTimestamp` has to be valid at the time the verification is performed.

If the proof is necessary for more than one data object, steps 1 and 3 have to be done for all these data objects.

To prove that the ArchiveTimestampSequence relates to a data object group, verify that each first Archive Time-stamp of the first ArchiveTimestampChain of the ArchiveTimestampSequence of each data object does not contain other hash values in its first hash value list (than the hash values of the other data objects)." [RFC4998, clause 5.3].

4.5.2 XML-based Evidence Record pursuant to RFC6283

The XML-based Evidence Record has an XSD Schema defined in [RFC6283]. Also defined in this Syntax are required and optional elements. A Hash tree is optional. To spend less amounts of expensive time stamps multiple objects can also be hashed and protected by a single time stamp. An Evidence Record pursuant to [RFC6283] is defined using the Extensible Markup Language (XML). Below, the structure of an Evidence Record is represented using the extracts from the [RFC6283] and [BSI-TR-03125-ERS].

The following definitions were represented using a pseudo XML dialect. The following assumptions regarding the cardinality of the elements apply:

- "?" - means 0 or 1 (0..1),
- "+" - means 1 or more (1..n),

Structure of the <**EvidenceRecord**> element:

```
<EvidenceRecord Version>
      <EncryptionInformation>
         <EncryptionInformationType>
         <EncryptionInformationValue>
      </EncryptionInformation> ?
      <SupportingInformationList>
         <SupportingInformation Type /> +
      </SupportingInformationList> ?
      <ArchiveTimeStampSequence>
         <ArchiveTimeStampChain Order>
            <DigestMethod Algorithm />
            <CanonicalizationMethod Algorithm />
            <ArchiveTimeStamp Order>
               <HashTree /> ?
               <TimeStamp>
                  <TimeStampToken Type />
                  <CryptographicInformationList>
                     <CryptographicInformation Order Type /> +
                  </CryptographicInformationList> ?
               </TimeStamp>
               <Attributes>
                  <Attribute Order Type /> +
               </Attributes> ?
            </ArchiveTimeStamp> +
         </ArchiveTimeStampChain> +
      </ArchiveTimeStampSequence>
```

```
          </EvidenceRecord>
```

- <EncryptionInformation> [optional]
  If necessary, the <EncryptionInfo> element may include information on how to deal with encrypted data. Further details on the EncryptionInfoType may be found in [RFC6283].
- <SupportingInfoList> [optional]
  The <SupportingInfoList> element may include further supporting information, such as information on the suitability of cryptographic algorithms as security measures pursuant to [RFC5698]. Further details on the SupportingInformationType can be found in [RFC6283].
- <ArchiveTimeStampSequence>
  Similarly to the ASN.1-based Evidence Record pursuant to [RFC4998], the <ArchiveTimeStampSequence> element contains a sequence of <ArchiveTimeStampChain> elements, that, in turn, includes a sequence of <ArchiveTimeStamp> elements. Further details on the ArchiveTimeStampSequenceType can be found in [RFC6283].
- <ArchiveTimeStamp>
  The <ArchiveTimeStamp>-element in particular contains a <TimeStamp>-element and may optionally contain a reduced <HashTree>.
- <HashTree> [optiona]
  The <HashTree> element must correspond to the following data structure

```
<HashTree>
       <Sequence Order>
           <DigestValue>base64 encoded hash value</DigestValue>
       </Sequence>
</HashTree>
```

- <TimeStamp>
  Depending on the value of the Type attribute, the <TimeStamp> element contains either a timestamp token created pursuant to [RFC3161] or an alternative form, such as [TS-ENTRUST]
- <CryptographicInformationList> [optional]
  This optional element may store additional validation information (e.g., certificates or certificate revocation lists or OCSP responses) if it cannot be stored within the TimeStampToken itself.

4.5.2.1 Life cycle of an Evidence Record pursuant to [RFC6283]

Concerning [RFC6283] there are the following processes, similar to [RFC4998] to be done:

- ArchiveTimestamp
  - Generation
  - Verification
  - Resigning
  - Rehashing
- ArchiveTimestampSequence and ArchiveTimestampChain
  - Generation
  - Verification.

5 Trust Services of the eIDAS Regulation in Practice

5.1 Trust Services

5.1.1 Overview

According to the legal definition in Art. 3 No. 16 of [(EU)910/2014] a trust service "means an electronic service normally provided for remuneration which consists of:

(a) the creation, verification, and validation of electronic signatures, electronic seals or electronic time stamps, electronic registered delivery services and certificates related to those services, or

(b) the creation, verification and validation of certificates for website authentication; or

(c) the preservation of electronic signatures, seals or certificates related to those services."

This legal definition is summarised in [ETSI-119001(v1.2.1)] with the short definition of a trust service as an "electronic service which enhances trust and confidence in electronic transactions".

A trust service provider is a legal or natural person, who provides one or more of these trust services.

Trust service providers are an essential element to establish trust between parties to enable trustworthy and verifiable digital transaction between these parties. They have to be confident about the security of trust services. To ensure the security requirements the TSP has to establish a set of procedures, processes and security measures to avoid or minimize the corresponding risks. As stated in [ETSI-319401(v2.2.1)] the TSP stays fully responsible for the information security of the provided services. Nevertheless, it is legitimate to outsource parts of the operation of the trust service to another organization.

In Germany, the Federal Network Agency is responsible for supervising the trust services for the generation, validation and preservation of (qualified) electronic signatures, seals and timestamps. The responsibilities of the Federal Network Agency are defined in [VDG17] (translated by author), §2: (1) The tasks of the supervisory authority pursuant to Article 17 of Regulation (EU) No. 910/2014 and pursuant to this Act as well as pursuant to the statutory instrument pursuant to § 20 shall be incumbent upon

1. The Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (Bundesnetzagentur) for the areas of

   [a)] the creation, verification and validation of electronic signatures, electronic seals or electronic time stamps and services for the delivery of electronic registered mail as well as certificates relating to these services in accordance with Article 3 number 16 letter a of Regulation (EU) No. 910/2014 and

   [b)] the preservation of electronic signatures, seals or certificates relating to those services, as referred to in Article 3(16)(c) of Regulation (EU) No 910/2014; and

2. the Federal Office for Information Security for the area of creation, verification and validation of certificates for website authentication in accordance with Article 3(16)(b) of Regulation (EU) No 910/2014.

Detailed information on trust services[1] and their certification can be found on the website of the Federal Network Agency. A list of qualified trust service providers and the qualified trust services they provide can be found in the trusted lists of the national supervisory bodies, those via the trusted list of the European Commission[2].

[1] https://www.elektronische-vertrauensdienste.de/cln_122/EVD/DE/Home/start.html

[2] https://eidas.ec.europa.eu/efda/tl-browser

5.1.2 Electronic Signature Generation

The generation of an electronic signature and electronic seal is based on three calculation steps:

1.) **Hashing**
The document to be signed is brought to a hash value of fixed length by a cryptographic hash function.

2.) **Padding**
The bit string with the hash value is suitably padded to the length required for the signature process and the signature key.

3.) **Signature**
Depending on the signature algorithm, the filled-in bit string is combined with the private signature key to form a signature.

Secret information is only processed in the third step: the private key and possibly also secret random numbers that are included in the signature. These calculations should therefore be made in an environment that is protected against eavesdropping by third parties. Ideally, the private signature key is stored and used exclusively in special hardware, the signature creation unit, which effectively prevents reading. In practice, chip cards with an integrated microprocessor (smart cards) or specialized hardware security modules are used for this. The specific form depends on the application, i.e. whether the signatureis generated locally (SmartCard or HSM) or as a remote signature (HSM only).

Modern smart cards and Hardware Security Modules (HSM) can also generate random signature key pairs so that the private key never leaves the device. Qualified electronic signature creation units, i.e. for generating qualified electronic signatures, must comply with the requirements of Annex II [(EU)910/2014]. This is proven by product certifications.

In practice, the hash value is usually calculated outside of the signature creation unit, so that only the short hash value and not a large message has to be transferred to it. To ensure that the correct data is actually signed, the entire signature creation process must be secure against manipulation (e.g. by viruses or Trojans). This not only affects the calculations for generating the digital signature, but also the transfer of the data to be signed and intermediate results (e.g. the hash value) between the components involved.

In cases in which an electronic signature is to be understood as a declaration of intent by a person, the data to be signed should be displayed beforehand. Particularly in the case of qualified electronic signatures, which in most cases have been given the same status as a handwritten signature by the legislature, the creator of the signature must be able to be sure that he is only signing what he sees. File formats that can contain hidden information (e.g. comments, meta data, text with white font color, etc.) open the door to scammers and are therefore rather unsuitable.
To avoid disadvantages, he should ensure that he:

- keeps his signature creation unit and the associated 2-factor/multi-factor authentication safe (under own control),
- signs Documents only after acknowledgment and examination, creating

  either advanced electronic signatures

  or qualified electronic signatures with a qualified electronic signature creation device, and which is based on a qualified certificate for electronic signatures [(EU)910/2014, Article 3(12)]
- immediately has his certificate revoked if his private key or signature creation unit is compromised.

There is a multitude of (software) solutions and trust service providers with which (qualified) electronic signatures can be generated. These products differ in their application. The specific practical process in the application, based on the products used in the specific case, is to be requested from the respective provider (trust service provider, product manufacturer or the person who enables the use of a specific product for their applications). Depending on the trust service provider, users can choose between local qualified electronic signature creation devices (QSCD) in form of smart cards for example at a workstation or server and an Hardware Security Module (HSM) based QSCDs, which can also be operated as a remote signature service, using remote signature protocol (see [SchHue19, p.38],[HHS19]).

5.1.3 Electronic Seal Generation

The requirements for (qualified) electronic seal creation devices are the same as for signature devices (with the necessary modifications) and are described in [(EU)910/2014, annexII]. Depending on the trust service provider, users can choose between a seal card at the workstation or server and an HSM, which can also be operated as a remote seal service. [SchHue19, p.38]. From the technical point of view, the Electronic Seal Generation is based on the Electronic Signature Generation (see Section 5.1.2).

5.1.4 Time Stamp Generation

In connection with electronic signatures or seals (see Section 4.2), [(EU)910/2014, art.42] describes the legal effects of electronic timestamps and the requirements for qualified time stamps.

Qualified trust service providers are available for this purpose after their qualified status has been indicated in the trusted lists [eIDAS-TL] and Section 5.2.

5.1.4.1 Time Stamp Protocol (TSP)

As shown in Figure 5.1, TSP is a client/server protocol in which the client maintains a connection to the server pursuant to [RFC3161]. The TSP client sends a time stamp request (`TimeStampReq`) that contains, in particular, the hash value of the data to be time stamped to the TSP responder directly over this socket connection or using higher-level protocols such as HTTP, for example. Basically, the TSP then adds the current time to the hash value received, signs this data, and sends this time stamp back to the client (`TimeStampResp`). According to [(EU)910/2014, Article 42], the time stamp Authority must use the time which "is based on an accurate time source linked to Coordinated Universal Time" to produce the time stamp when issuing qualified electronic time stamps.

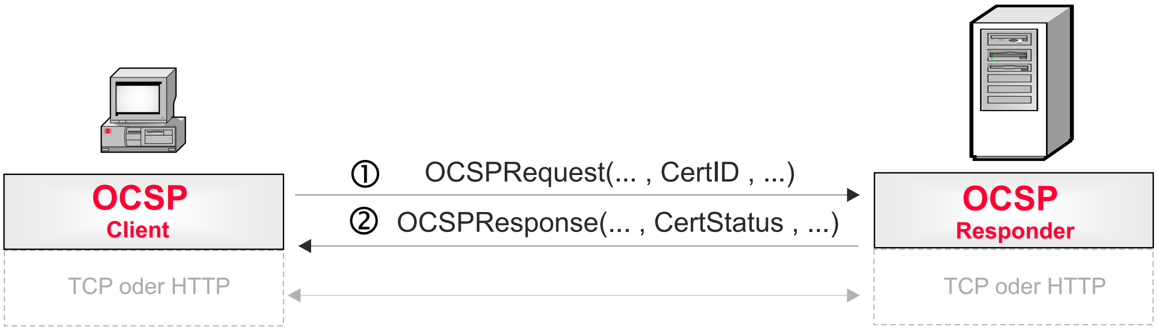

OCSP
Client
TCP oder HTTP
① OCSPRequest(... , CertID , ...)
② OCSPResponse(... , CertStatus , ...)
OCSP
Responder
TCP oder HTTP

Figure 5.1: TSP protocol structure

As indicated in Figure 5.2, a time stamp request in the TSP protocol (`TimeStampReq`) consists of the following data:

- `version`
  Specifies the version of the `TimeStampReq` syntax.

- `MessageImprint`
  Contains the hash value of the data to be time stamped (`hashedMessage`) and information specifying which hash function was used for this purpose (`hashAlgorithm`).

- `reqPolicy` (optional)
  Specifies which policy should be applied by the time stamp authority when generating the time stamp.

- `nonce` (optional)
  "Allows the client to verify the timeliness of the response when no local clock is available" [RFC3161]. "The nonce, if included, allows the client to verify the timeliness of the response when no local clock is available. The nonce is a large random number with a high probability that the client generates it only once (e.g., a 64 bit integer). In such a case the same nonce value MUST be included in the response, otherwise the response shall be rejected." [RFC3161, clause 2.4.1]

- `certReq` (default false)
  Specifies whether or not the time stamp authority should include a reference to the signature certificate used as the basis for creating the time stamp in a `SigningCertificate` attribute according to [RFC2634] in the time stamp it returns.

- `extensions` (optional)
  Provides a generic capability for including extensions. However, [RFC3161] does not specify any specific extensions and only refers to the general definition of extensions found in [RFC5280].

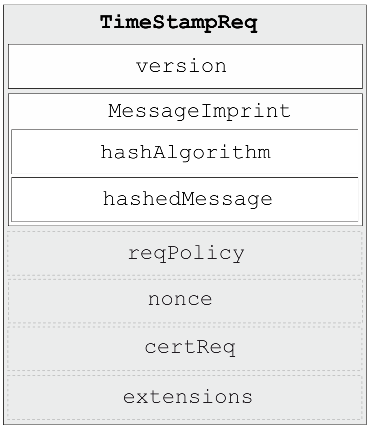


Figure 5.2: TSP request

As indicated in Figure 5.3, the response from the TSP responders consists of a specification of the status at a minimum. If successful, the desired time stamp (`timeStampToken`) is also sent back to the client. This time stamp is a Cryptographic Message Syntax (CMS) structure according to [RFC5652] of type `SignedData` in which the signed content (`eContent`) consists of the actual time stamp data. The `TSTInfo` structure consists of the following information:

- `version`

  Specifies the version of the `TSTInfo` syntax.

- `policy`

  Specifies the policy under which the time stamp was created.

- `MessageImprint`

  Contains the hash value of the data to be time stamped (`hashedMessage`) and information on which hash function was used for this purpose (`hashAlgorithm`) and passes as a response to the time stamp request.

- `serialNumber`

  A whole number consisting of up to 160 bits. How this number is formed depends on the policy of the time stamp authority. A sequential serial number is usually selected for the time stamps issued.

- `genTime`

  The time at which the time stamp was applied.

- `accuracy`

  Optionally specifies the accuracy of the time source used to for time stamp in seconds, milliseconds, and microseconds.

- `nonce`

  When needed, the random number passed in the time stamp request.

- `tsa`

  The name of the time stamp authority as specified in the `subject` field of the corresponding certificate. The time stamp certificate is identified exactly using the `SigningCertificate` attribute according to [RFC2634] provided that this was desired in the request (cf. `certReq` for the TSP request).

- `extensions`
  Provides a generic capability for including extensions. However, [RFC3161] does not specify any specific extensions in this case either.

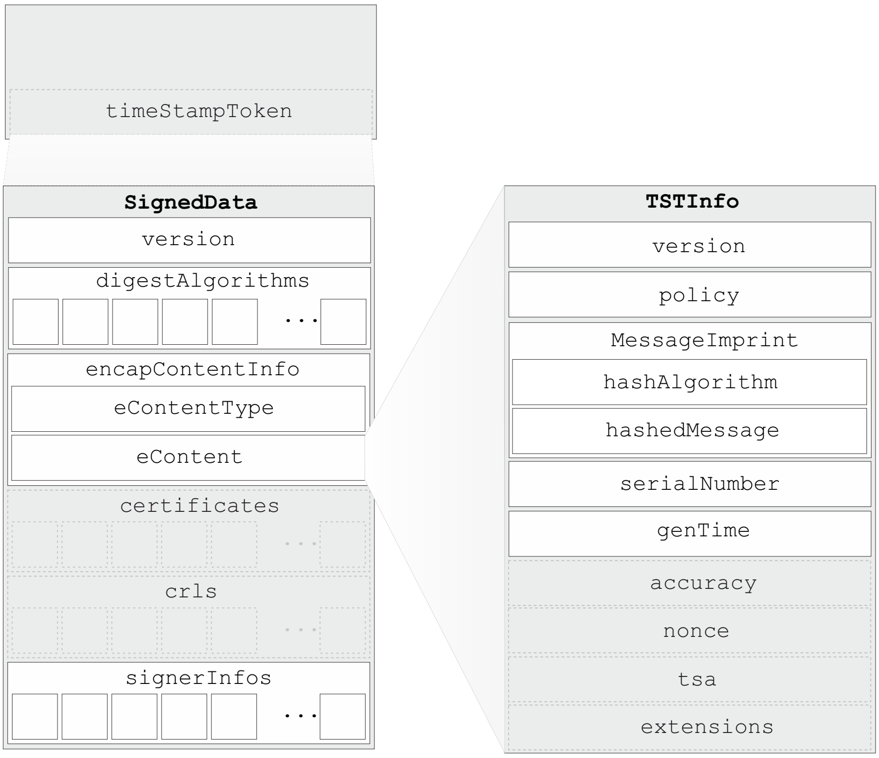


Figure 5.3: TSP response

5.1.4.2 Trust Service Providers issuing Timestamps

According to [ISO18014-1], time stamps are used to verify that certain data already existed before a certain time. In actual practice, the Time Stamp Protocol (TSP) specified in [RFC3161] and updated by [RFC5816] is usually used for this purpose. For example, all certification authorities existing today use this protocol to issue qualified time stamps. Further details are to be found in[ETSI-319421(v1.1.1)]

5.1.5 Validation Service

5.1.5.1 Validation of electronic signatures, seals or time stamps

The validation services make it possible to verify the trustworthiness of (qualified) electronic signatures, (qualified) electronic seals. Principally, signatures and seals can be validated by the user himself or by a trustworthy non-qualified or qualified Validation Provider. A qualified Trust Service Provider fulfils the security requirements set forth in the eIDAS Regulation, is granted the qualified status by the supervisory body and is officially included in the European eIDAS Trusted List (TL) [(EU)910/2014, Article 22]. Depending on the application, the validation of electronic signatures, seals or time stamps can also include other aspects, e.g. whether the certificate of the creator of the electronic signature is a qualified one, or whether the certificates were issued under a specific certification policy. If the electronic signature- , seal- or timestamp validation is provided as an IT service to other external institutions, the validation should be carried out by a qualified validation service. See also `https://www.bundesnetzagentur.de/EVD/SharedDocuments/Downloads/QES/BSI_TR_03125.pdf?__blob=publicationFile&v=1`.

5.1.5.2 Certificates and Certificate Path

In practice, a validation service for electronic signatures or seals verifies the mathematical correctness of the digital signatures or time stamps and the validity and applicability of the certificate(s) (standards for certificats in Section 2.4.2.1) assigned to them. A certification path to a trustworthy root certificate or trust anchor (e.g. pursuant to „Trusted List“ [ETSI-119612(v2.2.1)]) that is trustworthy from the point of view of the verifying party must be created and verified. In the case of most signature or seal application components, certificates from (qualified) trust services are already integrated and present in the digital signature. The user can import additional certificates from certification authorities that he considers trustworthy. Hereby, it is important that he verifies the authenticity of the certificates in a suitable manner. For the electronic seal, the owner of the certificate is a legal person, for the electronic signature it is a natural person. The requirements and the content of (qualified) certificates for electronic signatures (see [(EU)910/2014, Annex I]) differs from the requirements and content of certificates for electronic seals [(EU)910/2014, Annex III].

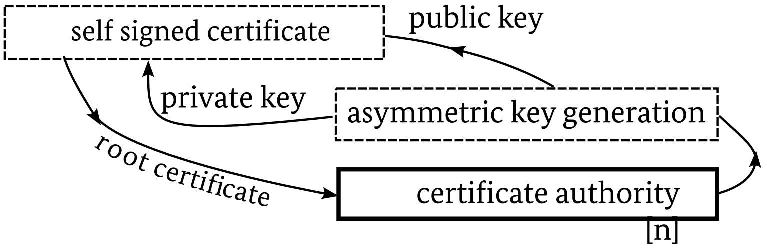


Figure 5.4: generation of root certificate/ trust anchor

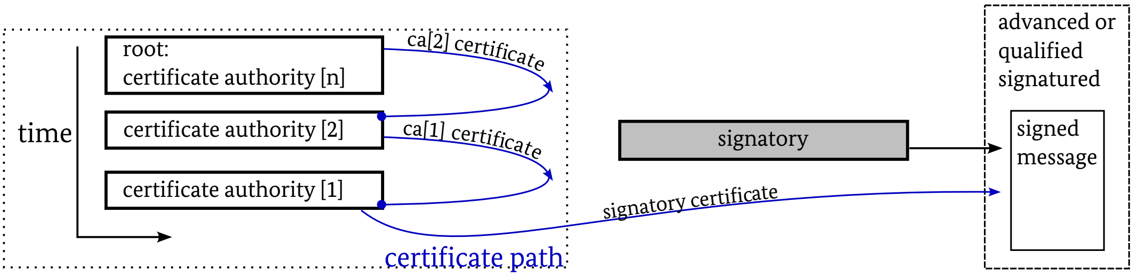


Figure 5.5: certificate path

Figure 5.5 shows a certification path with 3 levels. Level 3 describes the root certificate in this case.

- *signatory certificate* is the certificate of the signature or seal key owner (for the public signature or seal key with which one can validate the signature or seal to be verified),

- Figure 5.4 shows the creation of the trustworthy root certificate or trustworthy trust anchor , and

- for all i from 1 to (n-1), the owner of *ca[i+1] certificate* signed and issued the certificate *ca[i] certificate* so that the signature or seal from *ca[i] certificate* can be verified with the public key certified using the *ca[i+1] certificate*.

The verification is then done along the certification path. In doing so, the following items are verified at a minimum:

- the mathematical validity of the signatures or seals,
- the validity of the certificates in the certification path up to a trustworthy root certificate or trust anchor pursuant to the validity model (chain model or shell model),
- the correctness of the use of the certificates.

According to the application, the verification of the signature or seal can also include additional aspects, for example whether the *signator certificate* is qualified, or whether the certificates were issued based on a certain Certificate Policy (CP).

The electronic signature or seal for *ca[i] certificate* must be verified with the public key contained in *ca[i+1] certificate* also called signature validation data. Naturally, this step is not done for the trusted certificate *ca[n]=root certificate*. Insofar as it is the root certificate or trust anchor, *ca[n]* is self-signed; i.e. it can be verified with the public key contained therein. This is superfluous, though, because the certificate is considered trustworthy. Usually, the fingerprint of the root certificate or trust anchor, i.e. the hash value of the certificate, which includes in particular the public key, is published (see [eIDAS-TL]). This can be used to verify the integrity of the public key.

5.1.5.3 Validity of the Certificates Pursuant to the Validity Model

The certificates must have been valid at the decisive point in time (i.e. neither revoked nor expired). In doing so, the decisive point in time depends on which validity model is used (shell model or chain model) (see [COMMON-PKI-BNetzA]). The verification of the validity of the certificates includes the verification of the validity period and the assessing of the status information, in each case with regard to the points in time stipulated by the validity model. The status information can be retrieved from revocation lists or Online Certificate Status Protocol (OCSP) retrieved by a Certificate Status Authority. In the case of the shell model and when revocation lists are used, it is important that the verifying party has a reliable system time so that it is impossible for expired certificates or revocation lists to be presented to him without him noticing. ”A certificate MUST NOT appear more than once in a prospective certification path” [RFC5280].

Chain Model

The chain model requires that all certificates at the time of the use of the private key assigned (to the certified public key also called signature validation data) are valid. Concretely, this means that in the case of $i \geq 1$ the certificate *ca[i]* must have been valid up to the issuance of *ca[i+1]* and the *signatory certificate* at the time of the creation of the signature to be verified. This requirement is significantly weaker than that of the shell model, because, for example, if *ca[1]* were expired or revoked during creation of the signature to be verified, then the signature would only be invalid in the shell model. In order to ensure that *signatory certificate* is not a counterfeit certificate that a swindler created with the key of an CA that was taken out of service or compromised, one needs a service such as OCSP that answers ”unknown” (i.e. not from this CA) to counterfeit certificates. Furthermore, one must be able to determine the time at which the signature to be verified was created [COMMON-PKI-BNetzA]. The uri:oid:1.3.6.1.4.1.8301.3.5.1 [TUDarmstadt] may be used to indicate the chain modell as the used validity model.

See [BSI-TR-03125-PEPT], clause 7.7.5.2, To-do-7.7.5-3).

One of the URLs

- `http://www.bsi.bund.de/DE/tr-esor/sigpolicy/verify-xaip` (both shell or chain) or
- `http://www.bsi.bund.de/DE/tr-esor/sigpolicy/verify-xaip/shell` (only in case of TR-ESOR V1.3 and higher) (default) or

- http://www.bsi.bund.de/DE/tr-esor/sigpolicy/verify-xaip/chain (only in case of TR-ESOR V1.3 and higher)

may also be included in the TR-S.4- or TR-S.512-interface using DefaultPolicy/SignaturePolicyIdentifier element within the dss:OptionalInputs-Element of a Verify-request (see also [BSI-TR-03125-E, clause 3]) or ValidateEvidence-request (see also [BSI-TR-03125-E, clause 4] and [ETSI-119512] in the profiling of [BSI-TR-03125-TRANS])"

Shell Model

In the case of the shell model, one requires that all certificates in the certification path are valid at the time of the verification, i.e. the decisive point in time is "now", if there is no trustworthy timestamp included in the signature. The shell model has the consequence that signatures that were valid when they were created can become invalid if a certificate in the certification path expires or is revoked. However, it also has the consequence that the validity of the certificates cannot go beyond the validity of the certificate authority. For this reason, the certificates of a CA that are in a Public Key Infrastructure (PKI) that uses the shell model must be valid much longer that those of the end user, and they also lose their validity in the event of the revocation of the associated CA certificate. The shell model is used for the X.509 standard that was especially created for the purpose of authentication. The uri:oid:1.3.6.1.4.1.8301.3.5.2 [TUDarmstadt] may be used to indicate the shell modell as the used validity model.

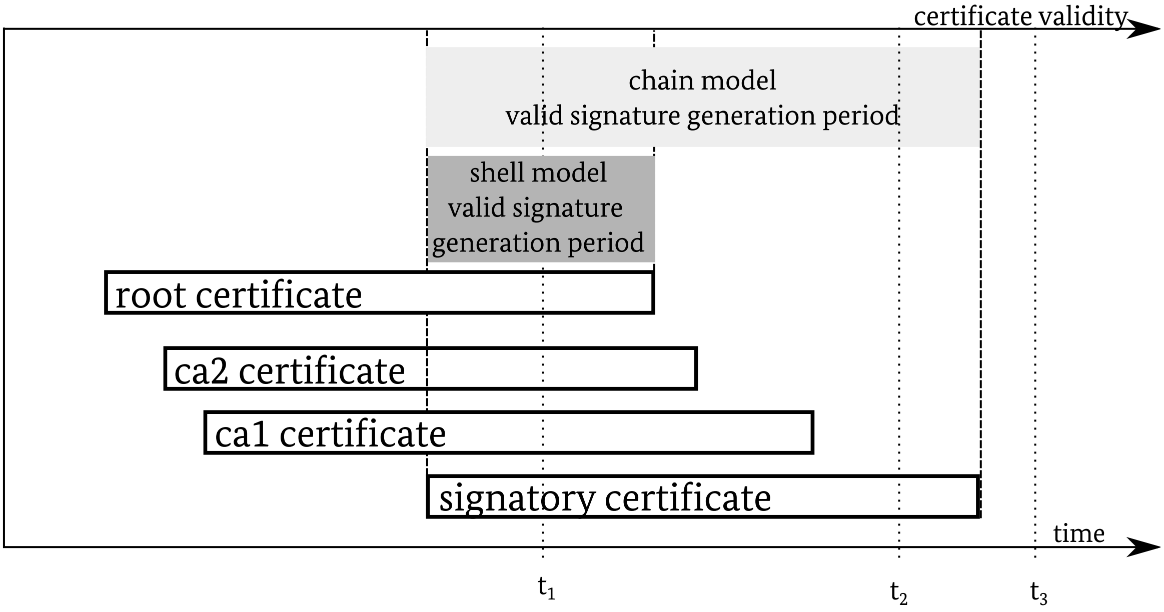


Figure 5.6: generation of signatures / seals

Figure 5.6 shows the validity of signatures/seals depending on the used model. Table 5.1 shows the different points in time and the validity depending on the used model.

validation time	chain model	shell model
t_1	valid	valid
t_2	valid	invalid
t_3	valid	invalid

Table 5.1: validity result depending on used model, created in the time period given in figure 5.6

5.1.5.4 Time Stamp Validation

More details concerning time stamp validation are to be found in [ETSI-319102-1(v1.3.1)]. Validation process needs the ”timestamp token” as a mandatory input requirement. Additional inputs like the ”trust anchor list” are optional and stated in the table 5.2 below. As a result the output of the validation is the status which indicates the validity of the timestamp [ETSI-319102-1(v1.3.1), p.58].

Input	Requirement
Timestamp token	mandatory
Trust anchor list	optional
Signature validation policies	optional
Local configuration	optional
Timestamp certificate	optional

Table 5.2: Inputs for timestamp validation

5.1.6 Preservation Service

5.1.6.1 General Aspects of Long-Term Data Preservation

A preservation service is a “service capable of extending the validity status of a digital signature over long periods of time and/or of providing proofs of existence of data over long periods of time” (see [ETSI-119511(v1.1.1), section 3]) using digital signature techniques”, even if later the signing key becomes compromised, the certificate expires, or cryptographic attacks become feasible on the signature algorithm or the hash algorithm used in the submitted signature” (see [ETSI-119512], Section 1).

In principle, federal and private institutions or organisations may have their own preservation service or preservation product or may use a (qualified) preservation service. As the usage of valid qualified electronic signatures, seals or time-stamps has legal advantages , the Regulation (EU) [(EU)910/2014] requires in Article 34: “A qualified preservation service for qualified electronic signatures may only be provided by a qualified trust service provider that uses procedures and technologies capable of extending the trustworthiness of the qualified electronic signature beyond the technological validity period”.

Article 34 shall also apply mutatis mutandis to the preservation of qualified electronic seals [(EU)910/2014, Article 40].

Qualified preservation trust service providers are available for this purpose after their qualified status has been indicated in the trusted lists (see [eIDAS-TL] and Section5.2).

In many areas of public administration, justice, business and science there is the ambition to replace paper based workflow and data handling by electronic business processes. One hurdle that has to be overcome in this context is caused by the nature of electronic documents and data. They are not embodied in themselves like a paper document and so they can also be changed or deleted without leaving any recognisable traces. Therefore, it is very important to preserve the integrity, authenticity and the ability to get a proof of existence of electronic data in order “to maintain the conclusiveness of the documents supporting legal claims of the issuer or third parties and the proof of their correctness in electronic legal and business transactions” [SKH14]. This can be achieved without restricting the elementary marketability by (qualified) digital signature techniques based on electronic signatures or seals (see Section 4.3) or electronic time-stamps (see Section 4.4) or Evidence Records (see Section 4.5).

5.1.6.2 Self-contained Archival Information Packages

To decrease the effort to preserve the signed or sealed or time-stamped data as well as the signatures, seals, time-stamps themselves, further supplemental evidence data (certificates, certificate revocation lists, OCSP responses, signature- or seal- or time-stamp verification reports, etc.) and Evidence Records, these objects are typically preserved in so called self-contained Archival Information Packages AIPs (see [ISO14721]), which contain all information needed to fulfil the ability of extending the validity status of a digital signature over long periods of time and/or of providing proofs of existence of data over long periods of time".

Example 1: XAIP or LXAIP packages pursuant to [BSI-TR-03125] TR-ESOR (see [ETSI-119512, Annex A] and [BSI-TR-03125-F, Section 3.1 or Section 3.2], Section 3.1 or Section 3.2).
Example 2: ASiC-E pursuant to [ETSI-319162-1(v1.1.1)] and (see [ETSI-119512, Annex A.1.3])
Example 3: ASiC-ERS pursuant to TR-ESOR (see [BSI-TR-03125-F, Annex A.3.1.3],
and [ETSI-119512, Annex A.1.3]).

5.1.6.3 Preservation Evidence Augmentation

The cryptographic algorithms, parameters and processes used by the digital signature techniques can be subject to "technical decay" over the course of time. Therefore, concerning the cryptographic algorithms and parameters, used in the context of (qualified) electronic signatures or seals, or time-stamps, the trust service providers should comply with the algorithm catalog in accordance with [ETSI-119312(v1.4.2)] and [SOGISV1.2] in the current version. During the preservation period, preservation services shall monitor the cryptographic algorithms, they have used, in order to make sure that the preservation evidences are still suitable to achieve their corresponding preservation goals. In practice that means: If the signature algorithm or hash algorithm will foreseeably lose its suitability as a security mechanism or a certificate will lose its validity, the preservation service has to augment the preservation evidences (see [ETSI-119511(v1.1.1), clause 7.15]) by "addition of data to an existing preservation evidence to extend the validity period of that evidence.

EXAMPLE 5.1.6-1
Adding a new electronic time-stamp, protecting the previous signature or time-stamp and additional validation data "which can be used to validate a previous signature and/or time-stamp, and/or the hash of the protected data using a stronger hash algorithm." [ETSI-119511(v1.1.1), Section 3]

For this purpose, see also new `archive-time-stamp-v3` in Section 4.3.1.2. In this case, if a used cryptographic algorithm is in danger to become weak and e.g. a new qualified time-stamp is required for each archive data object, then this solution would not be cost efficient and scalable.

EXAMPLE 5.1.6-2
In order to minimize the number of new qualified time-stamps required, the Archival Information Packages (AIPs) are to be protected by Merkle-Hashtrees (see Section 4.5) as given below. The tree secures all to-be-preserved data objects stored in the AIP with one qualified time-stamp as shown in Figure 4.18. With the reduced hashtree (see Figure 4.19), the so-called Evidence Record, the authenticity and integrity and proof of existence of the data can unambiguously be made evident by keeping the transferability of the objects in the AIP.

In case of Evidence Record (see Section 4.5 and especially Section 4.5.1.1 and 4.5.1.5), augmentation can be done by time stamp renewal or hash tree renewal pursuant to [RFC4998, RFC6283]. Because the operation of a hash tree renewal can take a signification amount of time depending on the number of archive data objects stored, it is recommended that redundant hash tree is generated as specified in the ERS standard ("Section 7: Security Considerations" Paragraph "Redundancy") " (see [BSI-TR-03125-M3, clause (A4.8-7)]). The use of Merkle hash trees in accordance with

[RFC4998, RFC6283] enables an economic preservation of evidence, since a large number of data objects can be protected by a hash tree with one single time-stamp.

5.1.6.4 [BSI-TR 03125] TR-ESOR on base of (EU)910/2014) and the ETSI Preservation Standards

The [BSI-TR-03125] TR-ESOR specifies a preservation product according to [ETSI-019510(v1.1.1), ETSI-119511(v1.1.1), ETSI-119512, (EU)910/2014] for preserving signed and un-signed data objects with their preservation evidences according to the state of the art. TR-ESOR describes the necessary processes, modules and interfaces in a reference architecture. The preservation techniques used in this TR-ESOR are also described in [ETSI-019510(v1.1.1), Sections 4.7.3 and 5.2 and B3.2]. In this way, the integrity, authenticity and the ability to get a proof of existence of electronically archived data and documents in particular can be preserved until the end of the legally prescribed retention period while maintaining the legally effective Evidence Records (see Section 4.5).

Specifically, the BSI TR 03125 TR-ESOR includes:

- Requirements, recommendations and descriptions for a TR-ESOR reference architecture, its processes, modules and interfaces as the concept of a middleware and data and document formats,
- Functional Conformity Test Specifications,
- Technical Conformity Test Specifications and Test Tools for technically interoperable products concerning the technical interoperability of
- the Archival Information Packages (AIP) (L)XAIP and ASiC-AIP (see Section 5.1.6.2) and [BSI-TR-03125-F, clause 3],
- the Evidence Records (see Section 4.5) and [RFC4998, RFC6283, BSI-TR-03125-ERS],
- the upper interfaces TR-S.4 (of TR-ESOR) and TR-S.512 (as a profiling of the [ETSI-119512] [BSI-TR-03125-TRANS]) for technically interoperable products.
- the signature verification reports pursuant to [[OASIS-VR, ETSI-119102-2(v1.3.1), BSI-TR-03125-VR].

The following figures show a schematic depiction of the IT Reference Architecture with the upper interface TR-S.4 (see Figure 5.7) and with the upper interface TR-S.512 (see Figure 5.8).

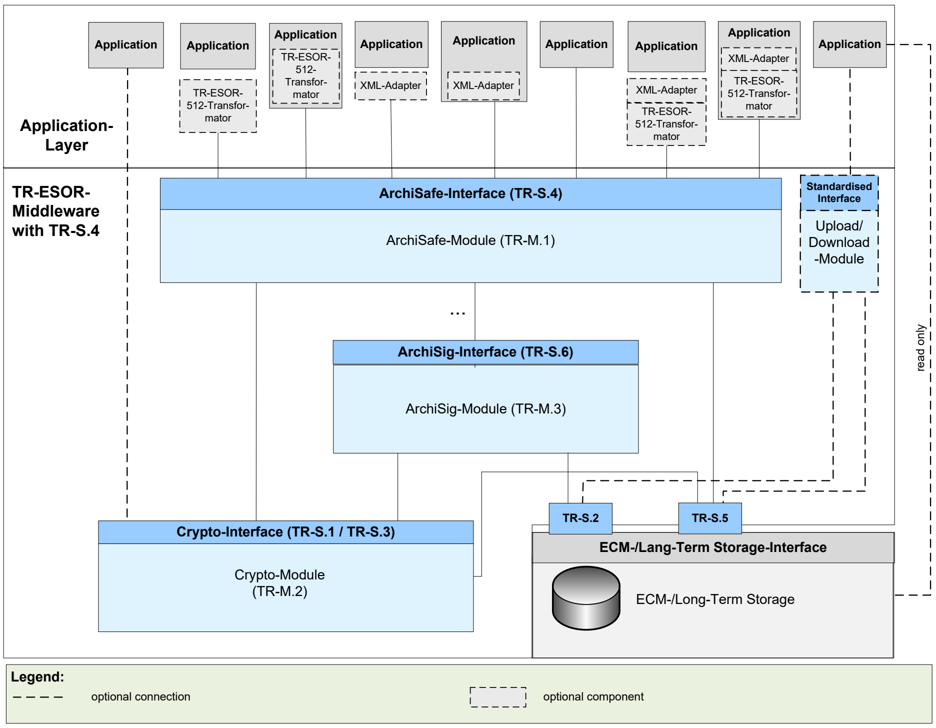


Figure 5.7: Schematic depiction of the IT Reference Architecture

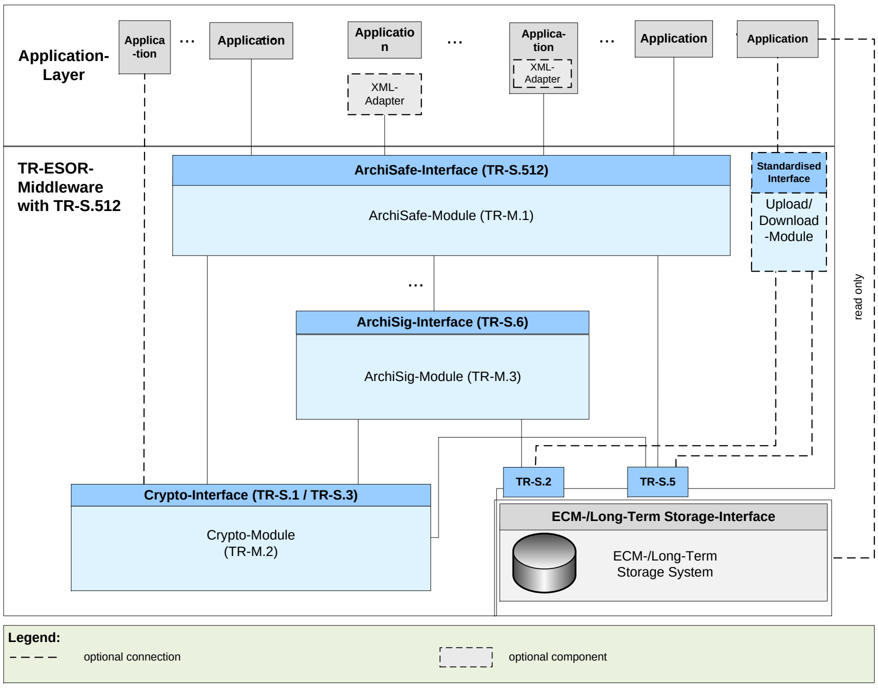


Figure 5.8: Schematic representation of the IT Reference Architecture with TR-S.512

Especially, BSI-TR-03125 TR-ESOR Version 1.3 specify a preservation product standard establishing a technical interoperability with (qualified) preservation services according to Art. 34 and 40 of the eIDAS Regulation [(EU)910/2014] and the ETSI Preservation Standards [ETSI-019510(v1.1.1), ETSI-119511(v1.1.1), ETSI-119512].

BSI-TR-03125 TR-ESOR Version 1.2.2 and Version 1.3 also establishes interoperability with the technical specifications for (qualified) preservation services according to Art. 34 and 40 of the eIDAS Regulation [x] and the ETSI Preservation Standards [ETSI-119511(v1.1.1)] and [ETSI-119512]. Therefore, the following preservation techniques used in [BSI-TR-03125] are included as normative elements in ETSI119512 and TR-ESOR:

- the preservation protocol ”TS119512” in ETSI TS 119 512, chapter 5.3 and [[BSI-TR-03125-E, section 4],
- the preservation object format ”ASiC-E” and “ASiC-ERS ” in [ETSI-119512, chapter A.1.4 and A.3.1], according to [BSI-TR-03125-F, clause 2 Notice 5 and chapter 3.3],
- the preservation object format ”XAIP and LXAIP” in [ETSI-119512, chapter A.1.5 and A.3.2] according to [BSI-TR-03125-F, chapter 3.1,3.2], ,
- the ”preservation evidence format” ”Evidence Record” in [ETSI-119512, Chapters A.2.2 and A.2.3] and [BSI-TR-03125-F], chapter 5.5] as well as [BSI-TR-03125-ERS].

Furthermore, the interface TR-S.512 as a profiling of the [ETSI-119512] and the verification report pursuant to ETSI and the following preservation object formats:

- ASiC-ERS (in TR-ESOR v1.3 called ASiC-AIP) pursuant to [ETSI-119512, Annex A.3.1 and A.3.1.3] (http://uri.etsi.org/ades/ASiC/type/ASiC-ERS) and pursuant to [TR-03125-F] clause 3.3;
- CAdES pursuant to [ETSI-119512, Annex A.1.1] (http://uri.etsi.org/ades/CAdES). If there is no MIME type filled, then the default application/cms is used;
- XAdES pursuant to [ETSI-119512, Annex A.1.2] (http://uri.etsi.org/ades/XAdES). If there is no MIME type filled, then the default application/xml is used
- PAdES pursuant to [ETSI-119512, Annex A.1.3] (http://uri.etsi.org/ades/PAdES). If there is no MIME Type filled, then the default application/pdf is used;
- ASiC-E pursuant to [ETSI-119512, Annex A.1.4] (http://uri.etsi.org/ades/ASiC/type/ASiC-E). If there is no MIME type filled, then the default application/vnd.etsi.asic-e+zip is used;
- ASiC-S pursuant to [ETSI-319162-1(v1.1.1), ETSI-319162-2(v1.1.1)] (http://uri.etsi.org/ades/ASiC/type/ASiC-S). If there is no MIME type filled, then the default application/vnd.etsi.asic-s+zip is used;
- DigestList pursuant to [ETSI-119512, Annex A.1.6] (http://uri.etsi.org/19512/format/DigestList). If there is no MIME Type filled, then the default application/xml is used; are included in [BSI-TR-03125] TR-ESOR Version 1.3.

Preservation Products Products that want to be certified pursuant to the Technical Guideline [BSI-TR-03125] TR-ESOR shall prove their conformity pursuant to the corresponding both test specifications [BSI-TR-03125-C1](Functional Conformity) and [BSI-TR-03125-C2] (Technical Conformity). “For conformance level 2 - technical interoperability testing - comprehensible technical proof shall be provided that the tested components or modules are implemented correctly and interoperably. The prerequisite for performing the level 2 test is the prior successful completion of the functional conformity test in level 1. The successful completion of level 2 is mandatory for certification. The BSI-approved test center therefore tests the following test items in accordance with [BSI-TR-03125-C2] on site at the product manufacturer with regard to technical interoperability

a) by means of the BSI test tools:

- the upper input interface TR-S.4 or the interface according to [ETSI-119512] in the profiling according to [BSI-TR-03125-TRANS],
- the Evidence Record data format and the timestamps contained therein,
- the XAIP or LXAIP archive information package format and the electronic signatures, seals and time stamps contained therein,
- the archive data format ASiC-AIP;

b) by means of further manual visual inspections

- the VerificationReport format.

The primary goal of this additional verification is to demonstrate proof that a level of technical interoperability can be reached on the basis of a well-defined standard. This is relevant in particular if using open, interoperable and standardised data formats and manufacturer-independent interfaces pursuant to national and European regulations and standards (eIDAS, ETSI) as well as international standards is generally desired or if only individual modules are verified that are sold as stand-alone products and thus have to work with other modules/systems. The test of technical interoperability is essentially based on the European ETSI standard TS 119 512." (see [BSI-TR-03125, Section 9.1.2])

The following links to the BSI web site shows the current version of TR-ESOR: `http://www.bsi.bund.de/EN/tr-esor` (English) and `https://www.bsi.bund.de/tr-esor` (German). Additional literature regarding this section can be found in [ETSI-019510(v1.1.1)] ,[ETSI-119511(v1.1.1), ETSI-119512, BSI-319401-AssP1, BSI-319511-AssP2] and in the Section "Additional Literature".

5.1.6.5 Assessing Preservation Trust Service Provider (PSP) with test facilitation through the use of a certified TR-ESOR Product

Trust services, as specified in the Regulation [(EU)910/2014], shall give participants of electronic commerce confidence in the security of these trust services. This confidence is expected to result from the fulfillment of the requirements of a set of procedures, processes and security measures, the Trust Service Provider (TSP) has established in order to minimize the operational and financial threats and risks associated. More specifically, [ETSI-119511(v1.1.1)] extends the general requirements of [ETSI-319401(v2.3.1)] for a Trust Service Provider (TSP), which provides long-term preservation of digital signatures or general data using digital signature techniques. If the [ETSI-319401(v2.3.1)] defines general requirements on the TSP's public documentation, including the TSP's management and operation, the [ETSI-119511(v1.1.1)] defines specific requirements to documentation and policies relating to the preservation service (e.g. Preservation Service Practice Statement, Preservation Evidence Policy, etc.) including technical and operational requirements relating to the preservation service (e.g. Preservation (e.g. Preservation Criteria for Assessing Trust Service Providers against ETSI Policy Requirements Profiles, Preservation Protocol, Notification Protocol, etc.). In summary, neither the TSP specific standards (ETSI EN 319 4x1) nor the CAB specific standard [ETSI EN 319 403] provide dedicated assessment criteria for an application for the conformity assessment of TSP. In case of preservation the document [BSI-319401-AssP1] has the goal to bridge this gap with respect to [ETSI-319401(v2.3.1)], and [BSI-319511-AssP2] bridges the gap with respect to [ETSI-119511(v1.1.1)]. It specifies assessment criteria to be used by accredited conformity assessment bodies (CAB) to assess the conformity of (qualified) trust service providers ((Q)TSPs) against the standard [ETSI-119511(v1.1.1)]. (see [BSI-319511-AssP2, section 1.1])

According to [BSI-319511-AssP2, Section 3.4.1], holds: "If the PSP claims to use a certified TR-ESOR product [BSI-TR-03125] of version V1.2.1 or higher and the claimed TR-ESOR certified product is in fact deployed for providing this service, proved e.g. by comparing the digital fingerprint of relevant executables, then the assessment result of the equivalent [ETSI-119511(v1.1.1)] - test case is substituted by the TR-ESOR-certification result and this [ETSI-119511(v1.1.1)] - assessment test step shall be omitted." That means, that certification for preservation services is facilitated if a product certified in accordance with [BSI-TR-03125] TR-ESOR is used by the trust service provider. Details on certification can be found on the website of the Federal Network Agency: https://www.bundesnetzagentur.de/cln_111/EVD/DE/Anbieter/Infothek/Grundlagen/start.html and the Federal Office for Information Security (BSI): http://www.bsi.bund.de/EN/tr-esor (English) and https://www.bsi.bund.de/tr-esor (German).

For details, see [BSI-319401-AssP1] and [BSI-319511-AssP2].

5.1.7 Other Trust Services

5.1.7.1 Electronic Registered Mail and Delivery Service

There are serveral ETSI Standards concerning Electronic Registered Mail and Delivery Service:

- EN 319 521: Policy and Security Requirements for Electronic Registered Delivery Service Providers
- EN 319 531: Policy and Security Requirements for Electronic Registered Electronic Mail Service Providers
- EN 319 522 Electronic Registered Delivery Services (Part 1-4)
- EN 319 532 Registered Electronic Mail (REM) Services

There are several requirements and policies which must meet for a potential "Electronic Registered Delivery Service Provider" [ETSI-319521(v1.1.1)]. In case of a qualified trusted service provider for delivery services, the protection goals for information security like authenticity, integrity and confidentiality were fulfilled and verified by an independent third party [(EU)910/2014, Article 43 and 44]).

5.1.7.2 Qualified Website Authentication Certificate (QWAC)

[(EU)910/2014, Article 3 no.16] introduced a new trust service: "the creation, verification and validation of certificates for website authentication." A certificate profile for web site certificates is to be found in [ETSI-319412-4(v1.2.1)].

5.1.8 Trust Service Provider

There are several requirements and organizational aspects for a potential trust service provider [ETSI-319401(v2.3.1), Section 7.1.1]. The figure 2.4 illustrates the relationship between the different eIDAS standards. For a potential applicant first there will be a conformity assessment for CABs regarding [ETSI-319403(v2.2.2)]. Requirements to become a "(Qualified) Trust Service Provider (QTSP)" are to be found in:

- eSignature/Seal [ETSI-319411-2(v2.4.1)] [ETSI-319412-1(v1.4.4)]
- Timestamps [ETSI-319421(v1.1.1)] [ETSI-319422(v1.1.1)]
- Validation [ETSI-319102-1(v1.3.1)] [ETSI-319102-2(v1.3.1)]
- Preservation [ETSI-119511(v1.1.1)] [ETSI-119512]

See also Figure 2.4.

5.2 Trusted List

The status of a Trust Service Provider and its trust service(s) are published via a trusted list (TL). [ETSI-119612(v2.2.1)] aims to offer a common template for trust lists as well as a guidelines for Trusted List Scheme Operators (TLSO). TLs do not only provide information about the current status of a trust service, but also a status history to enable relying parties to check the status of a particular trust service at a particular point in time. The basis for the intended use of [ETSI-119612(v2.2.1)] is not only the regulation of [(EU)910/2014], but also the legal context of other, non-EU countries, to enable the recognition of trust services and to validate digital signatures, and potentially to allow relying parties to verify the status of non-EU trust services. [ETSI-119612(v2.2.1)] describes the format and structure of a TL as well as mechanisms, that shall be established, to support relevant parties locating, assessing and authenticating TLs.

In Germany, the Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (BNetzA) is the competent German authority for establishing, maintaining and publishing the national trusted list and the corresponding trusted list is published on the website of the BNetzA. The corresponding trusted list of the European Commissions, which links to all national trusted lists, is available under the following Link
https://esignature.ec.europa.eu/efda/tl-browser/#/screen/home

5.2.1 Structure and Format of the TL

The logical component parts of a TL are listed and explained in the following. The number of recorded TSPs and trust services per TL is not limited. The logical components shall be recorded as a TL in XML format. These XML formats are described in detail in Annex B and C of [ETSI-119612(v2.2.1)]. Furthermore uniform resource identifiers (URI) shall be used within the definition of TL fields to display the meaning of the particular field.

5.2.1.1 Trusted List Tag

The trusted list tag facilitates the identification of the TL for electronic retrievals.

Scheme information

The scheme information provides basic information about the list itself and the kind of scheme, that is used for structuring. The scheme information includes:

- format version identifier,
- TL sequence (or release) number,
- TL type information,
- TL scheme operator information (information about the entity responsible for establishing, publishing and maintaining the TL, such as name, address),
- information about underlying approval scheme(s) with association to the TL (pointers to other TLs),
- TL policy and/or legal notice, liabilities and responsibilities,
- TL issue date and time plus next planned update.

TSP Information
The TSP information contains unambiguous information about every TSP, that is recorded within the scheme. The TSP information includes:

- TSP (trade) name,
- TSP addresse,
- TSP Services (List of trust services of the TSP and their status).

Service Information
The service information contains unambiguous information about every specific trust service, that is recorded within the scheme. These include:

- identifier of the type of service.

The trust services according to [(EU)910/2014] are listed and each of them is assigned with a URI, that may be used within TLs, to identify the recorded trust services.

- (trade) name of service,
- identifier of service,
- identifier of current status of services,
- current status including starting date and time stamp,
- additional information (e.g. access information, service definitions offered by the TSP).

The formats of possible information extensions are listed and described. Service approval history The service approval history provides information about the status history of each recorded trust service. This includes:

- service type identifiers,
- service name,
- service digital identity,
- service previous status (including starting date and time of status).

Digital signature
The TL is digitally signed itself by the entity, that is listed as scheme operator name, to prove authenticity and integrity.

Operations concerning the TL
As already stated, TLs have to be available to relevant parties. To ensure this, the TL shall be published through Hypertext Transfer Protocol (HTTP) and referred to by URI. The TL shall be available 24 hours a day at 7 days a week. To fulfill this requirement the TLSO shall implement appropriate measures, practices and policies to guarantee, that the information provided via the TL is timely, accurate, comprehensive and authentic.

Glossary

Abelian Group
An Abelian group is a group $(G, \cdot)$ in which the sequence of the elements may be switched so that the commutativity law $a \cdot b = b \cdot a$ applies for all $a, b \in G$.

Abstract Syntax Notation One (ASN.1)
Abstract Syntax Notation One (ASN.1) makes it possible to specify the syntax of data precisely. It was developed as part of the [X.408]-standard and then as a X.208 recommendation and in the meantime standardized in [X.680]. ASN.1 allows the abstract specification of data independent of their actual coding that is determined by specific coding rules. The Basic Encoding Rules (BER), the Canonical Encoding Rules (CER) and the Distinguished Encoding Rules (DER) are defined in [X.690]. The Packed Encoding Rules (PER) [X.691] and the XML Encoding Rules (XER) [X.693] also exist. ASN.1 is used, for example, in the X.509 standard and in the standards of the PKCS-series. This material is treated in detail in [Dubu00].

ArchiveTimestamp
"An ArchiveTimestamp is a timestamp and a set of lists of hash values. The lists of hash values are generated by reduction of an ordered Merkle hash tree [Merk80]. The leaves of this hash tree are the hash values of the data objects to be timestamped. Every inner node of the tree contains one hash value, which is generated by hashing the concatenation of the children nodes. The root hash value, which represents unambiguously all data objects, is timestamped." [RFC4998]

AdES (digital) signature
Pursuant to [ETSI-119001(v1.2.1)] , an "AdES (digital) signature" means a "digital signature that is either a CAdES signature, or a PAdES signature or a XAdES signature".

Advanced Electronic Seal
See electronic seal, advanced.

Advanced Electronic Signature
See electronic signature, advanced.

American National Standards Institute (ANSI)

The ANSI (`http://www.ansi.org/`) is a private commercial standardisation organ of the United States of American and is the representative of the USA in ISO.

Anonymity
Anonymity means, that the identification of an entity in a set of possible entities, the so-called "anonymity set", cannot be performed (see [ModTerm, Section 4.2] and [PfHa07]).

Application Programming Interface (API)
A application programming interface is a documented software interface with the help of which a software system can use certain functions from another software system.

ASiCManifest Container
The ASiC is a data container holding a set of file objects and associated digital signatures and/or time assertions using the ZIP format (see [ETSI-319162-1(v1.1.1)]).

ASiCManifest file
It is the file whose name matches "*ASiCManifest*.xml" containing one ASiCManifest element instance conformant to clause A.4 of (see [ETSI-319162-1(v1.1.1)]).

Assertion
An English word for claim.

Asymmetric Cryptographic Algorithms
For asymmetric cryptographic algorithms, there is a complementary pair of keys, (private key and public key) that can be used to realize electronic signatures, electronic seals, electronic time stamps, for agreements on secret keys or for asymmetric encryption. The concept of asymmetric cryptographic algorithms can be traced back to Whitfield Diffie and Martin Hellman [DiHe76]. The most common asymmetric cryptographic algorithm today is the RSA-Algorithm.

Attribute
In general, an attribute is a special characteristic of an entity provided with a name. In the field of electronic signatures, electronic seals and electronic time stamps attributes can be components of certificates – public key certificates or specialized attribute certificates – or of high-level-signatures, e.g. PKCS#7/CMS-format. For example, a certificate can include an attribute from which one can see that the owner of the certificate is a physician. Pursuant to [ETSI-119001(v1.2.1)], an "AdES (digital) signature" means a "digital signature that is either a CAdES signature, or a PAdES signature or a XAdES signature".

Attribute Certificate
An attribute certificate is a certificate that does not contain a public key itself, but rather merely refers in an unambiguous manner to a public key certificate. Pursuant to [ETSI-119001(v1.2.1)], it is "digitally signed by an attribute authority, that binds some attribute values with identification information about its holder". It is used in order to assign additional attributes to the referenced public key certificate.

Attribute Authority
An attribute authority is a part of a PKI and certifies that the applicant for a certificate has a certain property and privileges which can be included as an attribute in the attribute certificate applied for.

AUTACK
AUTACK is a EDIFACT type of message specified in [ISO9735-6] for transmitting integrity and authenticity information regarding sent payload data. There are also application rules for the use of AUTACK in [DIN16560-15].

Authentication
Authentication describes the procedure of verifying a claim of a partial identity of a person or a computer system by means of a certain characteristic (see authenticity).

Authenticity
Electronic data is authentic if it corresponds to the original data and the identity of an issuer (author, creator and/or sender) can be assigned. In other words the authenticity describes the trustworthiness and reliability of electronic data.

Proof of Authenticity
Proof of authenticity of electronic data represents the proof of the integrity of data (see integrity) and the unambiguous assignment to author, generator and/or sender.

Authorisation
Authorization is the function of granting access to an object, if the corresponding required permission exists for this purpose.

Bitwise
A bitwise operation designates the effort for an algorithm to operate on one or more bits or to compare them. For example, for the addition of numbers of the bit length l $O(l)$ bit operations are needed.

CAdES (digital) signature
Pursuant to [ETSI-119001(v1.2.1)] , a "CAdES (digital) signature" means a "digital signature that satisfies the requirements specified within [ETSI-319122-1(v1.2.1)] or [ETSI-319122-2(v1.1.1)]" or [ETSI-119122-3(v1.1.1)]".

CEN
See Comité Européen de Normalisation.

Certificate
Certificates for electronic signatures and electronic seals are electronic certifications that are issued (signed) by a Trust Service Provider with which certain information, especially a public key, can be assigned to the certificate owner and that confirm "at least the name or the pseudonym of that person" (see *Article 3 (14)* in the eIDAS-Regulation [(EU)910/2014]). The most common format for certificates is X.509. In addition to the public key, the certificate particularly includes personal information that was verified by the issuing entity at the time the certificate was issued and information regarding the period of validity.

Certification Authority
Pursuant to [ETSI-119001(v1.2.1)], a certification authority (CA) is a "authority trusted by one or more users to create and assign public-key certificates" and in that to furnish the contents of the certificate with a digital signature. Usually, the certificate authority also issues revocation lists that is signed in a similar manner.

Certification Path
Pursuant to [ETSI-119001(v1.2.1)], a "certification path" is an "ordered list of one or more public-key certificates, starting with a public-key certificate signed by the trust anchor, and ending with the public-key certificate to be validated".

A certification path consists of a chain of certificates $Z_1 - Z_2 - \cdots - Z_n$, in which case for all i from 1 to $n-1$ the owner of Z_{i+1} issued the certificate Z_i and Z_n is the certificate of a trust anchor.

Certificate Policy (CP)
A certificate policy consists of a number of regulations that are taken into account upon the issuance of a certificate. It can be decided on the basis of a certificate policy whether a certificate offers sufficient security for a certain use. There is a framework for the development of Certificate Policies in [RFC3647].

Certification Practice Statement

Pursuant to [ETSI-119001(v1.2.1)], a certification practice statement is ”a statement of the practices which a certification authority employs in issuing managing, revoking, and renewing or re-keying certificates.” (see also [RFC3647])

Certificate Revocation List (CRL)

See revocation list.

Chain Model

The chain model is a validity model for digital signatures for which it is required that each signature (the signature of the user to be verified and all signatures on the certificate in the certificate path) be based on a valid certificate at the time of creation. The later revocation of a certificates would not change anything about the validity of the signature.

Characteristic

The characteristic $char(K)$ of a finite field K indicates how often one must add the neutral element of the multiplicative group (1) in order to get the neutral element of the additive group (0). Because all finite fields are finite extensions of primary fields, the characteristic of a finite field is always a prime number p. Infinite fields, such as the rational numbers $\mathbb{Q}$ or the real numbers $\mathbb{R}$ are given the characteristic $char(K) = 0$.

Chip Card

A chip card is a card that is usually made of plastic that contains one or more semiconductor chips. One differentiates between storage cards on which one can merely store data and microprocessor cards in which a processor has been integrated with which data can also be processed. Microprocessor cards are also called smart cards. Comprehensive information on chip cards can be found in [RaEf02].

Chip Card Terminal

A chip card terminal is a device that makes power supply and data exchange possible with a chip card. Furthermore, this device can be equipped with a keyboard and a display in order to make secure PIN-entry possible.

Civil Identity

For natural persons or legal entities the civil identity is a partial identity, which contains at least the real name of the person and which is stored in State Register involved (e.g. central register of natural persons and Commercial and Companies Register or Register of Associations for legal persons. (see [PfHa07, Fn. 63]).

Claim

An English word for assertion. A claim (engl. also assertion) consists of at least one attribute and ”is a statement made by an entity about a client, service or other resource (e.g. name, identity, key, group, privilege, capability, etc.)” as a submitted declaration (see [WS-Security(v1.1)]) or statement made (see [WS-Trust(v1.3)]).

In particular, a partial identity may be object of a claim. In this case, the generation of the claim is called authenticity and the verification of such a claim is called authentication.

Comité Consultatif International Téléphonique et Télégraphique (CCITT)
The CCITT was the advisory committee for the telegraph and telephone service, out of which in 1993 the committee of the ITU-T (Telecommunication standardisation Sector of the International Telecommunications Union) of the International Telecommunication Union (ITU) arises, responsible for technical regulations, standards and recommendations for fostering cooperative standards for telecommunications equipment and systems. For example, the first version of the X.509-Standards [X.509] was published by CCITT.

Comité Européen de Normalisation (CEN)
The CEN is the European committee for standardisation located in Brussels that supports the goals of the European Community with the development of voluntary technical standards and regulations for compliance testing. These standards are developed and published as ”Technical Reports (TR)”, ”Technical Specifications (TS)”, ”CEN Workshop Agreements (CWA)”, ”Draft European Standards (prEN)” and finally ”European Standards (EN)”.

Common Criteria (CC)
With the Common Criteria for Information Technology Security Evaluation (Common Criteria [CC] for short), an international standard (ISO 15408) for the assessment and certification of the security of computer systems was created so that components or systems do not have to be certified multiple times in different countries. The Common Criteria, into which the European IT Security Criteria (ITSEC) among other things have been included, stipulates different evaluation assurance levels. The levels range from ”EAL1” (functionally tested) to ”EAL7”(formally verified design and tested) by which certain requirements in the light of the following aspects are defined:

- configuration management,
- delivery and operation,
- development,
- handbooks,
- lifecycle support,
- tests,
- vulnerability assessment.

The requirements of trustworthiness are staggered in such a way that in level EAL(n), the requirements of EAL$(n-1)$ are required as a minimum.

According to the Commission Implementing Decision [(EU)2016/650] of 25 April 2016, for products for qualified electronic signatures or qualified electronic seals an approval pursuant to Common Criteria - [ISO15408-1, ISO15408-2, ISO15408-3] with [ISO18045] is necessary by using corresponding protection profiles pursuant to [EN419211-1, EN419211-2, EN419211-3, EN419211-4, EN419211-5, EN419211-6] and by a subsequent acknowledgement by the national supervisory body.

Complexity Class $\mathcal{P}$
The complexity class $\mathcal{P}$ contains the set of the problems that can be solved with a deterministic algorithm in polynomial run- time.

Complexity Class $\mathcal{NP}$
The complexity class $\mathcal{NP}$ contains the set of the problems that can be solved with a probabilistic algorithm in polynomial run time. A given solution to a problem in the $\mathcal{NP}$ class can be checked with a deterministic algorithm in polynomial run time. $\mathcal{P} \subseteq \mathcal{NP}$ applies. A survey of the status of this problem can be found Aaronson, Scott. ”$\mathcal{P} \overset{?}{=} \mathcal{NP}$” (PDF) 2017, see `https://www.scottaaronson.com/papers/pnp.pdf`

Confidentiality
Confidentiality has the goal of preventing unauthorized access to and disclosure of information.

Conformity Assessment Body (CAB)
Pursuant to [ETSI-119001(v1.2.1)], a conformity assessment body is a ”body that performs conformity assessment services which is accredited as competent to carry out conformity assessment of a qualified trust service provider and the qualified trust services it provides”.

Creator of an Electronic Seal
See creator of an electronic seal.

Credential
A credential for digital identity cards is a proof of identity and a proof of permission, with which an entity can demonstrate, that he is allowed to access specific information and ressources or to carry out particular actions.

Cryptographic Message Syntax (CMS)
The cryptographic message syntax [RFC5652], is a further development of the PKCS #7 standard supported by the IETF. In this standard which is already supported by many standard software components, a very common High-Level-Signaturformat is specified, among other things. Furthermore, it is the basis for the S/MIME format for encrypting and signing email messages and for specific messages for certificate management [RFC5272].

Data
The generic term for all information that is read by electronic media, electronically processed or saved on electronic media. In information technology, data is often differentiated from document.

Declaration of Intent
A declaration of intent is the expression of the intent to induce a legal effect (see [Pala04, Einführung vor § 116] und [Bert01, chapter 2.2]). It can be manifested as an explicit declaration, by means of conclusive action, or even by means of silence. Whereas the form can be freely chosen under German law, certain legal transactions principally require a certain form for validity, such as the written form, because they would otherwise be null and void pursuant to § 125 [BGB].

Decryption
Decryption is a procedure using mathematical algorithms and private or secret keys to make electronic data readable and processable again. In encrypted form, the data (as unencrypted plain text) cannot be read by unauthorized third parties. The data can only be returned to the original form by the owner of the corresponding private or secret key.

Delta-CRL
A delta CRL is a revocation list that does not include all revocations, but rather merely updates on a basis CRL that it references.

Deterministic Algorithm
A deterministic algorithm is an algorithm that always executes the same sequence of operations for a certain input. The following processing step of the algorithm is stipulated in an unambiguous manner at every point in time and is not dependent on coincidence. The opposite of a deterministic algorithm is a probabilistic algorithm.

Differental Identity
A differential identity is a partial identity, which is used in a technical component (e.g. a chip card) for authenticity, authentication or for other cryptographical operations (see [ISO24727-3]). For example, a differential identity can be a password, a PIN, a private key, one or more secret keys, a public key, a biometric template or a digital identity card.

Digital Identity
Synonymous with Electronic Identity.

Digital Identity Card
A digital identity cards is a differential identity, by which the integrity of a claim is secured (vgl. [ModTerm, Section 4.11]). Commonly used kinds of digital identity cards are certificates, SAML-assertions or other integrity-protected security features.

Digital Signature
A digital signature is an electronic signature or an electronic seal based on asymmetric cryptographic algorithms. In doing so, a digital signature can only be created with the private key but verified by anyone using the public key.

Also see advanced electronic signature and qualified electronic signature or advanced electronic seal and qualified electronic seal.

Digital Signature Algorithm (DSA)

The Digital Signature Algorithm (DSA) [FIPS186-4] is a signature algorithm on the basis of the discrete logarithm in the multiplicative group of a finite field.

DIN 31647
The German DIN-standard 31647 specifies functional and technical requirements of a system, which perserves the evidence of cryptographically signed documents.

Directory (DIR)
See directory service and X.500.

Directory Service
A directory service is a component of a PKI and is used for the publication of certificates and certificate status information in the form of revocation lists or OCSP answers.

Discrete Logarithm

In addition to asymmetric cryptographic algorithms that are based on factoring problems, cryptographic systems that are based on Discrete Logarithm Problems (DLP) are also increasingly being used in practice. The problem of the discrete logarithm in a finite abelian group G is the calculations of the exponent n from a given group element $g^n \in G$. The difficulty for the solution of the problem depends largely on the used group G. For example, for cryptographic purposes, multiplicative groups of finite fields or point groups on elliptic curves are used. Signature procedures such as DSA or ECDSA can be constructed using these groups.

Distinguished Name (DN)

Pursuant to [X.501], a Distinguished Name (DN) is a sequence of ”Relative Distinguished Names (RDN)” that in turn consist of one or more values. The DN describes the path from a directory entry to the root node so that all entries in a X.500 directory can be addressed in a unique manner by the DN. For example, a DN can consist of entries for the country (C), the organisation (O), the organisational unit (OU), and the common name (CN) of the object.

DLP

See discrete logarithm.

Documentary Evidence - Documentary Proof

In civil procedures, there is a difference in the evidentiary value of private and public documents. Private documents (§ 416 [ZPO]) only provide the evidence that the issuer delivered the declaration included therein. On the other hand, a public document (§ 415 [ZPO]) also proves the procure certified therein. The proof is reported by presenting the document (§ 420 [ZPO]). In the case of a private document, the rebutting party must deliver a declaration on the authenticity of the document and, as the case may be, on the authenticity of the signature on the document (§ 439 [ZPO]). If the authenticity of the document is recognised, then the collection of evidence with this regard is concluded. If the authenticity of the document is disputed, then the rebutting party must prove the authenticity - possibly by means of a handwriting analysis (§ 441 [ZPO]) (§ 440 [ZPO]). In the case of public documents, the collection of evidence is, as a rule, even easier, because they are presumed to be authentic (§ 437 [ZPO]).

Electronic Business XML (ebXML)

ebXML (`http://www.ebxml.org`) is a joint initiative from UN/CEFACT and OASIS started in 1999 by means of which a range of specifications for the use of XML in electronic business processes was developed.

EC-SDSA-opt

EC-SDSA-opt was defined in ISO/IEC 14888-3:2006/Amd 2:2012 ”Optimizing hash inputs” and integrated in in ISO/IEC 14888-3:2016, now replaced by [ISO14888-3], 2018: IT Security techniques — Digital signatures with appendix — Part 3: Discrete logarithm based mechanisms.

Electronic Form

For the electronic form defined in § 126a [BGB] that can, as a rule, replace written form pursuant to § 126 [BGB], the issuer of the declaration shall add his name to it and furnish the electronic document with a qualified elektronic signature pursuant to the eIDAS-Regulation [(EU)910/2014]. Pursuit to *Article 3(10)* in the eIDAS-Regulation [(EU)910/2014] elektronic signatures mean data in electronic form which is attached to or logically associated with other data in electronic form and which is used by the signatory to sign.

Electronic Identity
An electronic identity is an electronic representation of a partial identity (see [ModTerm, Section 4.2]).

Electronic Registered Delivery Service
Pursuant to pursuant to *Article 3(36)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic registered delivery service means a service that makes it possible to transmit data between third parties by electronic means and provides evidence relating to the handling of the transmitted data, including proof of sending and receiving the data, and that protects transmitted data against the risk of loss, theft, damage or any unauthorised alterations”.

Electronic Seal
Pursuant to pursuant to *Article 3(25)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic seal means data in electronic form, which is attached to or logically associated with other data in electronic form to ensure the latter’s origin and integrity”. In this case the creator of an electronic seal is a legal person.

Electronic Seal, Advanced
Pursuant to *Article 3(26)* in the eIDAS-Regulation [(EU)910/2014], an ”advanced electronic seal means an electronic seal, which meets the requirements set out in *Article 36* in the eIDAS-Regulation [(EU)910/2014]”. Also see creator of electronic seal and electronic seal.

Electronic Seal, Qualified
Pursuant to *Article 3(27)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified electronic seal” ”means an advanced electronic seal that is created by a qualified electronic seal creation device, and which is based on a qualified certificate for electronic seals”.

Qualified Certificate for Electronic Seal
Pursuant to *Article 3(30)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified certificate for an electronic seal” ”means a certificate for an electronic seal, that is issued by a qualified trust service provider and meets the requirements laid down in *Annex III*” in the eIDAS-Regulation [(EU)910/2014].

Qualified Electronic Seal Creation Device
Pursuant to *Article 3(32)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified electronic seal creation device” ”means a electronic seal creation device that meets mutatis mutandis the requirements laid down in *Annex III*” in the eIDAS-Regulation [(EU)910/2014].

Electronic Seal Creator
Pursuant to *Article 3(24)* in the eIDAS-Regulation [(EU)910/2014], a ”creator of a seal means a legal entity which creates anelectronic seal”. Also see electronic seals and advanced electronic seals.

Electronic Seal Creation Data
Pursuant to *Article 3(28)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic seal creation data” ”means unique data, which is used by the creator of the electronic seal to create an electronic seal”. See also creator of an electronic seal.

Electronic Seal Creation Device
Pursuant to *Article 3(31)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic seal creation device” ”means configured software or hardware used to create an electronic seal”. See also qualified electronic seal creaton device.

Electronic Seal Validation Data
Pursuant to *Article 3(40)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic seal validation data” ”means data that is used to validate an electronic seal..”.

Electronic Signature
Pursuant to *Article 3 (10)* in the eIDAS-Regulation [(EU)910/2014]) electronic signature ”means data in electronic form which is attached to or logically associated with other data in electronic form and which is used by the signatory to sign” with the aim of authenticity, integrity and non-repudiation. Beside simple signatures, e.g. easily forgeable bitmaps of hand written signatures, the spectrum of possible forms of electronic signatures ranges from advanced electronic signatures pursuant to *Article 3 (11)* in the eIDAS-Regulation [(EU)910/2014] to qualified elektronic signatures pursuant to *Article 3 (12)* in the eIDAS-Regulation [(EU)910/2014] as a very secure form of a digital signature.

Electronic Signature, Advanced
Pursuant to pursuant to *Article 26* in the eIDAS-Regulation [(EU)910/2014], an advanced electronic signature ”shall meet the following requirements:

(a) it is uniquely linked to the signatory;

(b) it is capable of identifying the signatory;

(c) it is created using electronic signature creation data that the signatory can, with a high level of confidence, use under his sole control; and

(d) it is linked to the data signed therewith in such a way that any subsequent change in the data is detectable.”

An advanced electronic signature is an electronic signature with special characteristics by means of which at least a basic amount of authenticity and integrity can be ensured. However, unlike the qualified electronic signature, a merely advanced electronic signature cannot replace the written form pursuant to § 126 [BGB] and has less power as evidence before a court (see § 371a [ZPO]). Usually, one uses digital signatures and certificates in order to get advanced electronic signatures.

Electronic Signature, Qualified
Pursuant to *Article 3(12)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified electronic signature” ”means an advanced electronic signature that is created by a qualified electronic signature creaton device, and which is based on a qualified certificate”. A qualified electronic signature has the same legal effect as a hand written signature.

Qualified Certificate for Electronic Signature
Pursuant to *Article 3(15)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified certificate for an electronic signature” ”means a certificate for an electronic signature, that is issued by a qualified trust service provider and meets the requirements laid down in *Annex I*” in the eIDAS-Regulation [(EU)910/2014].

Electronic Signature Creation Data
Pursuant to *Article 3(13)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic signature creation data means unique data which is used by the signatory to create an electronic signature”.

Electronic Signature Creation Device
Pursuant to *Article 3(22)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic signature creation device” ”means configured software or hardware used to create an electronic signature”. See also qualified electronic signature creaton device.

Qualified Electronic Signature Creaton Device
Pursuant to *Article 3(23)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified electronic signature creation device” ”means a electronic signature creation device that meets the requirements laid down in *Annex II*” in the eIDAS-Regulation [(EU)910/2014].

Electronic Signature Validation Data
Pursuant to *Article 3(40)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic signature validation data” ”means data that is used to validate an electronic signature.” Also called public key in this document.

Electronic Time-Stamp
Pursuant to *Article 3(33)* in the eIDAS-Regulation [(EU)910/2014], an ”electronic time stamp” ”means data in electronic form which binds other data in electronic form to a particular time establishing evidence that the latter data existed at that time”. Often, as in the case of a time stamp protocol from [RFC3161], time stamps are generated with the use of digital signatures. Thus, time stamps are a form of electronic certification that the data signed by the time stamp were presented at the time of the signature in the signed form.

With regard to the legal effect and admissibility as evidence in legal proceedings of time stamps, one differentiates between simple (self-generated) time stamps and qualified time stamps that were issued by qualified trust services providers pursuant to the eIDAS-Regulation [(EU)910/2014].

Electronic Time-Stamp, Qualified
Pursuant to *Article 3(34)* in the eIDAS-Regulation [(EU)910/2014], a ”qualified electronic time stamp” ”means an electronic time stamp which meets the requirements laid down in *Article 42*” in the eIDAS-Regulation [(EU)910/2014].

Elliptic Curve
An elliptic curve (over a field K with characteristic $char(K) \neq 2, 3$) is a set of all points $P = (x, y)$ of the ”smooth”[3] curve $y^2 = x^3 + ax + b$ together with the point $\mathcal{O}$ in ”infinity”. For the cryptographic application of such elliptic curves it is important that the set of points defines a finite abelian group with $\mathcal{O}$ as the neutral element, in which it is possible to calculate efficiently, but the problem of discrete logarithm is extremely hard. Therefore, one needs cryptographic systems based on elliptic curves, as e.g. ECDSA, which use significantly less amount of storage or computer capacity for the same security level than conventional solutions, as e.g. RSA und DSA.

Elliptic Curve Digital Signature Algorithm (ECDSA)
The ECDSA [ANSI-X9.62] is a signature algorithm based on the discrete logarithm in the group of points of an elliptic curve over a finit field.

[3] A curve is called ”smooth”, if a tangent may be drawn in each point. It is possible to verify that this is exactly the case, if the following formula is valid: $4a^3 + 27b^2 \neq 0$.

Encryption
During encryption, a plain text will be turned into a cipher text with the use of a symmetric or asymmetric cryptographic algorithm and secret or public keys so that the original message is protected against unauthorised access. The recipient of the message can decrypt it to make is readable again.

Entity
An entity is a (natural or legal) person or an object (e.g. a technical component, a service, data, etc.), which is characterised by a technical component (see [ModTerm, Section 4.15])

European Telecommunications Standards Institute (ETSI)
ETSI, based in Sophia Antipolis (France), is officially responsible for standardisation of Information and Communication Technologies within Europe and is officially recognized by the European Commission.

Target of Evaluation (ToE)
In the case of an evaluation pursuant to ITSEC or Common Criteria, one calls the product or system to be assessed the ”target of evaluation”(ToE). A ToE can consist of several components. The *security specific* and *security relevant* components are of particular importance to the evaluation. In doing so, a component is *security specific* if it contributes directly to the achievement of the security goals. A component, that is not specifically security enforcing but still has to work correctly for the security of the ToE to be guaranteed, is called *security relevant*.

Exponential Run-time
In the complexity theory, one designates a problem as solvable in exponential time if there is an algorithm that needs for the input length l and a constant $c > 1$ $O(c^l)$ operations. One says that the algorithm has exponential run-time. Each problem from the complexity class $\mathcal{NP}$ can also be solved with *exponential run-time* (see [Wagn94, clause 5.10]).

Extended Euclidean Algorithm
With the extended Euclidean algorithm for two integer numbers a and b one calculates a linear combination of the largest common divider $d = \mathrm{ggT}(a, b) = s \cdot a + t \cdot b$ of the entered numbers. If one calls up the algorithm for a x which is relatively prime to the modulus n and receives the linear combination $1 = \mathrm{ggT}(n, x) = s \cdot n + t \cdot x$, then $t \equiv x^{-1} \pmod{n}$ is the inverse element of x modulo n.

Extensible Markup Language (XML)
XML is a flexible standard [XML(v1.0), XML(v1.1)] developed by the work group of W3C for the creation of structured, machine and human readable files.

Factoring Problem
The security of all asymmetric cryptographic algorithms known today is based on the supposed perceived difficulty of certain mathematical problems. A very popular problem in cryptography is the factoring problem that consists of breaking down a large compound number $n = pq$ into its prime factors p and q. If one chooses the prime numbers p and q randomly so that n has approximately 300 decimal places, the factoring problem is not solvable in practice based on the current status of science and technology, and a cryptographic procedure based on it, such as the RSA-Algorithmus, is secure. ”The latest factoring (and discrete logarithm) records are depicted in Figure 2, culminating with the factorization of RSA-250 (a 829-bit composite number) in February 2020, for a computation cost of about 2,700 core-years.” [BOU-2022]

Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (BNetzA)
In Germany, the Federal Network Agency for Electricity, Gas, Telecommunications, Post and Railway (BNetzA) is the supervisory body for the electronic trust services concerning electronic signatures, electronic seals, electronic time stamps and electronic registered delivery services pursuant to the eIDAS-Regulation [(EU)910/2014]) since June 1st, 2016.

Federated Identity
A federated identity is a digital identity, which connects a partial identity of an entity in a special context with another partial identity of this entity in another context (vgl. [ModTerm, Section 4.16]).

Field
In mathematics, a field is a set of elements K for which the two operations $+$ and $\cdot$ are defined, so that K with regard to $+$ builds an (additive) group with neutral element 0 and the elements from K without the element 0 with regard to $\cdot$ build a (multiplicative) group, so that one may ”multiply out” $a \cdot (b + c) = a \cdot b + a \cdot c$, thus the ”distributive property” applies. Furthermore, known bodies are the rational numbers $\mathbb{Q}$ (fractions) or the real numbers $\mathbb{R}$ (fractions and roots). A field, that only has a finite number of elements, is called a finite field. These finite fields play an important role in cryptography, because the calculation of discrete logarithms in their multiplicative group is a difficult problem according to the current state of knowledge.

Finite Field
A finite field is a field that only has a finite number of elements. In the case of such a field K, the characteristic of the field $char(K)$ indicates how often one must add the 1-element in order to get the 0-element. If p is a prime number, then one gets a finite field if one calculates modulo of this prime number p, also called arithmetic with residue classes, thus only takes the remainder in account during addition and multiplication (after deduction of multiples of p). Such a field has the characteristic $char(K) = p$. One gets all finite fields if one takes finite field extensions of prime fields into account. These finite fields have $q = p^k$ elements; in doing so k is the degree of the field extensions. Finite fields play an important role in cryptography because the calculation of discrete logarithms in their multiplicative groups is a difficult program according to the level of knowledge today.

Group
In mathematics, a group G is a set of elements, for which an operation of two elements is defined (e.g. $a \cdot b$) that fulfils the following characteristics:

- Closure:
  If a and b are elements of the group G, for short $a, b \in G$, then $a \cdot b$ is also an element in G.
- Associativity:
  The order in which the operations are performed does not matter as long as the sequence of the operands is not changed. That means for every element a, b and c in G the following equation holds: $a \cdot (b \cdot c) = (a \cdot b) \cdot c$.
- Identity Element:
  There is an element 1 in the group, so for every element a in G, the following equation holds: $a \cdot 1 = 1 \cdot a = a$.
- Inverse element:
  For each element a in G, there is an inverse element in G. The inverse of a ist $1/a$. which has the characteristic, that the linking with a produces the identity element 1. That means: $a \cdot 1/a = 1/a \cdot a = 1$.

Furthermore, if one can switch the sequence of the elements so that $a \cdot b = b \cdot a$ (commutativity of multiplication), then one speaks of an abelian group. A group that has a finite number of elements is called a *finite* group. Finite Abelian groups play a large role in cryptography because they allow the construction of cryptographic systems on the basis of the discrete logarithm.

Order of a Group

The cardinal number of the Group $|G|$, that means the number of elements in G, is called the order of the group.

Hardware Security Module (HSM)

An HSM is specialised hardware that makes it possible to preserve cryptographic keys in a particularly secure form and apply them in an efficient manner.

Hash Function

A hash function h is a cryptographic algorithm with which messages m of any length can be mapped on a hash value with a fixed length (e.g. 256 bit). In the case of cryptographically suitable hash functions, it is practically impossible to find two messages with the same hash value (collision resistance) and to reconstruct an input message m from a given hash value $h(m)$ (one-way characteristic).

A list of hash algorithms that has been assessed as suitable as security measures is published by [ETSI-119312(v1.4.2)] and [SOGISV1.2].

Hash Value

A hash value is a mathematical checksum that is created from an electronic message by applying a hash function. Because it is practically impossible to find two messages with an identical hash value in the case of a cryptographically suitable hash function, the hash value is also called the ”digital fingerprint” of a message. Because one finds a collision in the case of a l-bit hash function with great probability on account of the so-called birthday paradox if one chooses some $2^{l/2}$ random messages, a hash function that produces 240 bit hash values at a minimum should be chosen for electronic signatures (see [ETSI-119312(v1.4.2)] and [SOGISV1.2]).

High-Level-Signature Format

A high level signature format specifies how a raw signature in a low level signature format can be supplemented with additional information relevant for the verification of the signature, such as the time of the creation of the signature, the certificate needed for the verification of the signature, or the corresponding certificate status information and how the signature can be linked to or embedded into the payload data. Typical high level signature formats include PKCS #7 / CMS, S/MIME, PDF (embedded PKCS #7 and PKCS #1), XML-DSig, or the format standardised in ISO9735-5/6 for signing of EDIFACT data.

The Commission Implementing Decision [(EU)2015/1506] of 8 September 2015 lays ”down specifications relating to formats of advanced electronic signatures and advanced electronic seals to be recognised by public sector bodies pursuant to Articles 27(5) and 37(5) of Regulation (EU) No 910/2014.” (see [(EU)910/2014] and [(EU)2015/1506]). Pursuant to this Commission Implementing Decision [(EU)2015/1506], four binding high-level-signature formats are defined: CAdES (see [ETSI-103173(v2.2.1)]), XAdES (see [ETSI-103171(v2.1.1)]), PAdES (see [ETSI-103172(v2.2.2)]) and ASiC (see [ETSI-103174(v2.2.1)]).

Hypertext Transport Protocol (HTTP)

The HTTP protocol specified in [RFC7230] is a protocol for the transmission of data, which is in particular used in the World Wide Web and which is usually based on the connection oriented

TCP. HTTP can also be used, for example, to transport OCSP messages or XML-based web service-messages.

Identification
Identification means a process using claims or observed attributes in order to identify which entity it is (see [ModTerm, Section 4.18]).

Identification Data
Pursuant to *Article 3(3)* in the eIDAS-Regulation [(EU)910/2014], person ”identification data” ”means a set of data enabling the identity of a natural or legal person, or a natural person representing a legal person to be established”.

Identifier
An identifier consists of at least one attribute and clearly and unambiguously identified an attribute in a specific context (see [ModTerm, Section 4.20]).

Identity
The identity of an entity is determined by the set of attributes, where an entity possesses exactly one identity (see [ModTerm, Section 4.21] and partial identity).

Identity Management
Identity Management is defined as the management of partial identities. This includes the definition, allocation and administration of attributes and the creation, selection and usage of partial identities (see [ModTerm, Section 4.22]).

Information Technology Security Evaluation Criteria (ITSEC)
[BSI-ITSEC] is a European standard for the testing and certification of products and systems with regard to their trustworthiness. In doing so, one considers the effectiveness and correctness of the used security mechanisms. For the effectiveness, the minimum strength of the critical security mechanism, that is divided into the classes ”low”, ”medium” or ”high”, plays a particularly important role. With regard to the correctness, one differentiates between the assurance levels from ”E1” to ”E6” with increasing trustworthiness. For example, from evaluation level E3 on the source code is also verified. The European ITSEC criteria were also included in the development of the internationally harmonised Common Criteria (CC).

Integrity
Under ”proof of the integrity of electronic data” one understands the proof that they are complete and unchanged.

Signed Security Token
A signed security token is a security feature, whose integrity is protected by a cryptographic mechanism (see [WS-Security(v1.1)]).

International Organization for standardisation (ISO)
ISO (`http://www.iso.org`) is an international organisation of the standardisation committees of 151 countries. It adopts international standards in all technical areas. In ISO, Germany is represented by the ”Deutsche Institut für Normung (DIN)” (`http://www.din.de`) and the USA is represented by ANSI.

International Telecommunication Union (ITU)

ITU is a worldwide organisation that concerns itself with the technical aspects of telecommunications. In its Telecommunications standardisation Bureau (ITU-T) – formerly ”Comité Consultatif International Téléfonique et Télégraphique (CCITT)”– technical standards are worked out and published as recommendations. The recommendation X.509, in which, among other things, a widespread format for certificates is specified, is of particular importance for electronic signatures.

Internet Engineering Task Force (IETF)

The Internet Engineering Task Force (IETF) is a large, open, international collective that applies itself to the problem-free operation and further development of the Internet architecture. The standards and recommendations developed in the IETF are published with a certain sequential number as Requests for Comments (RFC) at `http://www.ietf.org`.

Internet Protocol Security (IPSec)

IPSec is a security architecture developed by IETF for guaranteeing the authenticity, integrity and confidentiality in IP networks.

Interval-Qualified Time-Stamp - SigG-specific

Interval qualified time stamps are self-created time stamps that are linked to two qualified time stamps in a manner so that it can be proven mathematically (see [Hueh04a, Theorem 1]) that they were created after the first qualified time stamp but before the second qualified time stamp.

ISO 9735

This standard was developed by a joint work group of ISO and UN/CEFACT and specifies a data exchange format for the electronic exchange of structured data (EDIFACT). ISO 9735 consists of ten parts [ISO9735-1, ISO9735-2, ISO9735-3, ISO9735-4, ISO9735-5, ISO9735-6, ISO9735-7, ISO9735-8, ISO9735-9, ISO9735-10]. Of particular relevance to electronic signatures are parts 5 and part 6 ([ISO9735-5, ISO9735-6]) because they determine how signatures can be integrated into EDIFACT messages and how the specialised AUTACK messages are structured that serve to protect the proof of authenticity and integrity and confirm the receipt of the EDIFACT messages.

ISO 14533

The standards ISO14533 ”Processes, data elements and documents in commerce, industry and administration - Long term signature profiles” consists of three published standards:

- [ISO14533-1]:
  Long term signature profiles for CMS Advanced Electronic Signatures (CAdES signature)
- [ISO14533-2]:
  Long term signature profiles for XML Advanced Electronic Signatures (XAdES signature)
- [ISO14533-3]:
  Long term signature profiles for PDF Advanced Electronic Signatures (PAdES signature)
- [ISO14533-4]:
  Attributes pointing to (external) proof of existence objects used in long term signature formats (PoEAttributes).

”The purpose of this part of ISO 14533 is to ensure the interoperability of implementations with respect to long term signatures that make electronic signatures verifiable for a long term.”

Kerckhoff's Principle

The principle of Kerckhoff describes the security of a system, even if all the used algorithms and the system itself is publically known. Even with all these informations the system is safe against attacks. In any case the private key needs still stay private. Attacks against compromising the private keys are not included in this principle.

Legal Inspection and Evidence by Inspection

Legal inspection is the direct perception through the senses of a person acting for a court or an authority with the goal of finding facts that are relevant to evidence (e.g. by means of seeing, hearing, smelling). The evidence obtained by means of legal inspection (evidence by inspection), (see also prima facie evidence) that is regulated in the German Code of Civil Procedure §§ 371 ff [ZPO] et. seq., includes all evidence that is not regulated separately as evidence from witnesses, documents or experts. It consists of the court using direct perception through the senses in order to get an impression of the characteristics of an item, the location of objects, or the existence and conduct of a person.

Legal Transaction

A legal transaction consists of at least one declaration of intent. A legal transaction can be unilateral (e.g. termination) or multilateral (e.g. contract).

$L_n[u,v]$**-Function**

In order to to describe the run time of subexponential algorithmen, one uses the $L_n[u,v]$-function, defined in the following manner:

$$L_n\left[u,v\right] = e^{v(\log(n))^u(\log(\log(n)))^{1-u}}.$$

In doing so, $\log(n)$ designates the natural logarithm of a number n that is larger as the Euler's number e and $0 \leq u \leq 1$ and $v > 0$.

Because $L_n[0,v] = e^{v(\log(n))^0\log(\log(n))} = e^{v\log(\log(n))} = \log(n)^v = O(\log(n)^c)$ for a constant $c > 0$, $L_n[0,v]$ corresponds to polynomial run time. In a similar way, $L_n[1,v] = e^{v\log(n)} = O(n^v)$ corresponds to the exponential run time, and for $0 < u < 1$ the effort lies between these two extremes.

Liabilities - Binding Character

Binding character means that a legal transaction develops its legal effect. The prerequisite for this is, in part, the compliance with formal requirements (e.g. written form). Furthermore, in order to achieve effectiveness, the existence of evidence is also necessary.

Lightweight Directory Access Protocol (LDAP)

With the Lightweight Directory Access Protocol (LDAP) [RFC4510], information saved in a directory service can be retrieved or modified. More details on the implementation of LDAP can also be found in [RFC4510].

Legal person

A company or institution is described as a legal person.

Low-Level-Signature Format

In the case of low-level-signature formats, in order to assure bit-level accuracy, it is specified how the data to be signed or a hash value thereof shall be prepared before the actual application of the asymmetric cryptographic algorithm, for example with padding. The format itself is marked significantly by the used cryptographic algorithm. For example, the signature formats PKCS #1 and ISO9796-2 are commonly used for the RSA procedure based on the factoring problem. Widespread signature formats on the basis of the discrete logarithm include DSA and ECDSA.

Message Authentication Code (MAC)
A Message Authentication Code (MAC) serves to secure the integrity and authenticity of a message. Unlike in the case of a digitalen signature, no asymmetric cryptographic algorithms are used here, rather symmetrical algorithms and secret keys for the creation and verification of the MACs are used. Because the same secret key will be used to create and verify, the MAC cannot reach the security goal of non-repudiation.

National Institute for Standards and Technology (NIST)
NIST is a governmental standardizing institution in the USA. The standards published by NIST include for example DSA.

Natural person
A natural person is a real person.

Non-Repudiation
Non-repudiation means that the origin, sending, or receipt of data and information cannot be denied. Non-repudiation is a prerequisite for the binding character.

Number Field Sieve
The number field sieve is currently the most powerful known algorithm for factoring large numbers and calculating discrete logarithms in multiplicative finite fields. It has an expected run time of $L_n[1/3, (64/9)^{1/3} + o(1)]$.

Object Identifier (OID)
An Object Identifier (OID) is a globally unique identifier for an information object. An OID represents a node in a hierarchically assigned namespace defined by the ASN.1 standard. (see. [X.660]).

Official Document - Certificate
An official document is a physical declaration of a thought that is suitable and intended as evidence in legal transactions and allows an issuer to be identified. If the issuer of the declaration as seen in the document does not concur with the actual creator of the document, the corpus delicti of forgery pursuant to § 267 [StGB] is fulfilled. The German Code of Civil Procedure stipulates certain regulations for evidence with documents, the so-called documentary evidence, which allows an accelerated procedure.

One-Way Function
A function f is called a one-way function (see [Gold01, Definition 2.1]) if it is ”easy” to calculate but ”difficult”to invert. In this case, ”easy” means that there is a deterministic algorithm solvable in polynomial time that calculates the functional value $f(x)$ for a given x.

”Difficult” means that for sufficient long input lengths each probabilistic algorithm solvable in polynomial time has a ”negligible” change of success for the calculation of the inverse function $f^{-1}(f(x))$. The value is also called ”negligible”if it is smaller than $\frac{1}{p(l)}$ for sufficiently long input length l for each polynomial p.

It is currently unknown whether such one-way functions actually exist. However, there is a number of functions that *seem* to have the one-way characteristic, because there has not yet been found a probabilistic algorithm solvable in polynomial time for the calculation of $f^{-1}(f(x))$ that has a non-negligible chance of success. For example, the security of many popular signature procedures is based on the unproven assumption that the multiplication of large, randomly selected prime numbers (see factorisation problem) or the exponents of certain finite abelian groups (see discrete logarithm) is a one-way function.

O-Notation
The O-Notation is used for the asymptotic estimation of the run time of algorithms. For two functions f, g $f = O(g)$ is written, if for sufficiently large values of x the inequality $f(x) \leq c \cdot g(x)$ applies for a constant value c.

Online Certificate Status Protocol (OCSP)
OCSP is a client-server protocol standardised by IETF in [RFC6960] to request of the status of certificates. For example, this online request can be used to verify whether a certificate has been revoked by the user.

Organization for the Advancement of Structured Information Standards (OASIS)
OASIS (`http://www.oasis-open.org/`) is a non-commercial, global consortium for the development and implementation of standards for eBusiness and XML.

PAdES (digital) signature
Pursuant to [ETSI-119001(v1.2.1)] , a ”PAdES (digital) signature” means a ”digital signature that satisfies the requirements specified within [ETSI-319142-1(v1.1.1)] or [ETSI-319142-2(v1.1.1)]”.

Padding
Padding is generally understood to be the augmentation of a sequence of characters with additional characters in order to reach a certain overall length. For example, the hash value of a message is augmented by certain padding characters in the RSA procedure for security reasons before the signature is created by exponentiation with the private key. The padding can occur in a deterministic (see [PKCS1(v2.2), ANSI-X9.31]) or probalistic manner (see [BeRo96, BeRo98, RFC8017]).

Partial Identity
A partial identity is a certain subset of attributes of an entity (see [ModTerm, Section 4.28], differential identity, federated Identity and identity).

Permission
A permission is the right of an entity to access to specific objects in a particular manner. A permission is represented by a set of access control rules.

Personal Security Environment (PSE))
A PSE is a storage medium for private keys and trustworthy certificates. A PSE can be realised either as a software solution, e.g. a password-protected file in PKCS #12-format, or as a hardware solution, e.g. in the form of a smart card. In this case, the PSE can simultaneously serve as the electronic signature creation device.

Personal Identification Number (PIN), PAuswVwV
A electronic signature creation device can be protected against unauthorized access by means of a PIN, usually a personal secret code of four to eight character secret digits. PIN of six digits offers sufficient security in order to fulfil the requirements of identification data defined in [PAuswVwV] ”Allgemeinen Verwaltungsvorschrift der Bundesregierung, Allgemeine Verwaltungsvorschrift zur Durchführung des Personalausweisgesetzes und der Personalausweisverordnung”.

Polynomial Time
In complexity theory, one describes a problem as solvable in polynomial time if the calculation time grows maximally like a polynomial function with increasing problem size. One says that the algorithm has polynominal run time. For the input length l the algorithm only needs $O(l^c)$ operations for a constant c.

Portable Document Format (PDF)
The Portable Document Format developed by Adobe Inc. ([PDF(v1.3), PDF(v1.4), PDF(v1.5), PDF(v1.6)],[ISO32000-2:2020, PDF2.0]) is a document format that is used in particular for the exchange of documents in the Internet. In doing so, signatures in PKCS #1 and PKCS #7 formats can be embedded in PDF documents so that they can be verified with the free Acrobat Reader. The Commission Implementing Decision [(EU)2015/1506] of 8 September 2015 references the PDF-format PAdES pursuant to [ETSI-103172(v2.2.2)]). ETSI published [ETSI-319142-1(v1.1.1)] and [ETSI-319142-2(v1.1.1)].

Post Quantum Cryptography
Post quantum cryptography deals with the development and investigation of cryptographic mechanisms that cannot be broken even with quantum computers. These mechanisms are based on mathematical problems, for the solution of which neither efficient classical algorithms nor efficient quantum algorithms are known today. Post-quantum signature algorithms are based on mathematical problems which are different from integer factorisation and the computation of discrete logarithms and for which no efficient algorithm using either classical hardware or quantum computers is known to date.

Pretty Good Privacy (PGP)
PGP is a program developed by Phil Zimmermann for encryption and creation of digital signatures of data by using asymmetric cryptographic algorithms. The authenticity of the public key is ensured as a rule by means of a so-called ”web of trust”; alternatively the use of X.509-certificates is also possible. The PGP message format is specified in [RFC4880].

Prima Facie Evidence
There is prima facie evidence if a circumstance that is taken for granted as a fact of life suggests a certain (typical) course of events for a circumstance related to it, and thus appears to prove it indirectly. Prima facie evidence is possible in the case of typical event sequences. If there is a circumstance which suggests a certain course of events based on all experience of daily life, then this course of events can be considered proven. The prima facie evidence is not legally regulated. However, the law sometimes refers to prima facie evidence in individual cases (such as § 371a of the German Code of Civil Procedure [ZPO] for qualified electronic signatures).

Prime Field
There are two definitions: a) If K is a finite field, the smallest subfield of K is called a prime field. b) In case of finite fields a finite field with p elements, where p is a prime number, is also called a prime field.

Prime Number

A prime number is a number that can only be divided by the number 1 or itself. In cryptography, prime numbers play an important role in particular in the design of cryptographic systems on the basis of the factoring problem or the discrete logarithm in the multiplicative group of finite fields.

Private Key

A private key is part of a cryptographic pair of keys to which only the owner of the pair of keys has access. It is kept in a personal security environment and used in order to create digital signatures or electronic time stamps or to decrypt data. [(EU)910/2014] defines the private key as signature creation data.

Probabilistic Algorithm

In the case of a probabilistic algorithm the sequence of steps during the execution is not just dependent on the input, but also on chance. If an algorithm does not depend on chance, then one speaks of a deterministic algorithm.

Provisioning Service Point

A provisioning service point is a component of a service oriented architecture, by which attributes of users can be administrated, using messages, defined in the service provisioning markup language.

Pseudonym

A pseudonym is an identifier of entities, which is unequal to the real name (see [PfHa07] and [ModTerm, section 4.38]).

Public Key

A public key is the part of a pair of cryptographic keys that is publicly known and freely accessible. It is usually a part of a certificate and is used, beside the verification of digital signatures or electronic time stamps, to encrypt data for a certain person. Only this person can then decrypt the data again with the associated private key to which only this person has access. [(EU)910/2014] defines the public key as signature validation data.

Public Key Certificate

A Public Key Certificate is a certificate that contains in particular the name of the certificate owner and the public key.

Public Key Cryptography Standards (PKCS) - RFC8017

PKCS is a set of standards for technology developed in the laboratories of the American company RSA Security Inc. on the basis of asymmetric cryptographic algorithms. The most important standards in this set in practice include:

- PKCS #1: RSA Cryptography Standard

  A frequently used low-level-signature format on the basis of the RSA algorithmus is specified in version 2.2 of this standard [PKCS1(v2.2), RFC8017]. The current version of this standard is [RFC8017].

- PKCS #7: Cryptographic Message Syntax Standard

  A widespread high-level-signature format is specified in this standard [PKCS7(v1.5), RFC2315]. Many standard software components already support it today. The CMS specifications of IETF [RFC5652] are based on PKCS #7.

- PKCS #11: Cryptographic Token Interface Standard

  The [PKCS11] standard defines a programming interface for standardised access to chip card functions.

- PKCS #12:Personal Information Exchange Syntax Standard Standard

  [PKCS12] standardises a data format for the exchange of private keys encrypted by means of a password.

Public Key Cryptographic Systems
Public Key Cryptographic Systems use asymmetric cryptographic algorithms.

Public Key Infrastructure (PKI)
A public key infrastructure (PKI) is a technical and organisational infrastructure that makes it possible to roll out and administer cryptographic pairs of keys (private keys in the form of PSEs and public keys in the form of certificates) in order to to support authentication, encryption, integrity or non-repudiation services.

Qualified Certificate
see qualified certificate for electronic seal or qualified certificate for electronic signature.

Qualified Electronic Seal
See qualified electronic seal.

Qualified electronic Signature
See electronic signature, qualified.

Qualified electronic Time-Stamp
See electronic time stamp, qualified.

Qualified Trust Service Provider
A qualified trust service provider means ”a trust service provider who provides one or more qualified trust services and is granted the qualified status by the supervisory body” (see *Article 3 (20)* in the eIDAS-Regulation [(EU)910/2014]).

Random Oracle Model
In the case of the Random Oracle Model [BeRo93] one makes the idealised assumption that a hash function acts like a ”random oracle”. For each question directed to the oracle, one receives an answer that appears to be random. Evidence for the security of popular cryptographic systems can be provided based on this idealistic assumption.

Real Name
The real name is the true name of a natural person.

Residue Class
The residue class of a number a modulo of a number m, designated as a module, containing the set of all numbers that, in the case of division by m, leave the same residue as a. All of the numbers in the residue class are thus differentiated from each other by an integer multiple of the module. Thus, one writes $a + m\mathbb{Z}$ for the residue class a modulo m.

Re-Signing
The security of all popular signature systems is based on the unproven assumption that the solution of the problem at the basis, such as the factorisation problem or the discrete logarithm problem is ”difficult” (see one-way function). The used algorithms and parameters can also loose their suitability as security measures on account of growing computing power or improved algorithms so that their power as evidence in conjunction with a digital signature would be reduced over time. Thus, § 15 [VDG17] stipulates that in case of qualified electronic signatures or qualified electronic seals or qualified electronic time stamps a signature or seal or time stamp renewal has to be done if the signed or sealed or time stamped data are needed in signed or sealed or time stamped form for a period of time that lasts longer than the period during which the algorithms and belonging parameters used for its creation and verification can be considered suitable. In this case the signature or seal or time stamp renewal shall be finished with a new qualified electronic signature or qualified electronic seal or qualified electronic time stamp prior to the time at which the suitability of the algorithms and related parameters will end. This must occur with suitable new algorithms or associated parameters, include older signatures, seals or time stamps and bear for example a new qualified electronic time stamp. More details are to be found in [ETSI-119511(v1.1.1)] and [ETSI-119512].

Revocation List
A revocation list is created by a trust service provider pursuant to eIDAS-Regulation [(EU)910/2014] and published in a directory service. It contains information about which certificates were revoked by the certificate owner or other authorized authorities. A widely accepted format for revocation lists was specified in X.509 and profiled in more detail in [RFC5280].

RIPEMD-160
RIPEMD-160 [DoBP96, ISO10118-3] is a hash algorithm developed in the scope of the RIPE project supported by the EU (RACE Integrity Primitive Evaluation 1988-1992) by Hans Dobbertin, Antoon Bosselaers and Bart Preneel.

Root Certificate
A root certificate is a self signed certificate generated by a certificate authority.

Role
A role contains at least one required permission.

RSA
The RSA algorithm [RSA78], which is named after its inventors (Rivest, Shamir, and Adleman) is an asymmetric cryptographic algorithm that can be used for encryption and the creation of digital signatures. The security of this procedure is based on the cryptographic assumption that the factoring problem cannot be solved efficiently for large composite numbers.

SAML-Assertion
A SAML-assertion is a digital identity, specified in [SAML(v1.0), SAML(v1.1), SAML(v2.0)], by which a successful authentication, the belonging of an attribute to a special entity or a required permission of an entity is confirmed by a SAML-authority.

Security Assertion Markup Language (SAML)
SAML is a XML-based Assertion Markup Language for security relevant claims, standardised by OASIS. The different specifications in the context of SAMLS are to be found under `http://www.oasis-open.org/committees/security/`.

SAML-Authority

A SAML-authority is an Identity Provider, which creates SAML-assertions (see [SAML-Glos(v2.0)]).

Secret Key

Secret keys are used together with symmetrical cryptographic algorithms. Unlike private keys used in the case of asymmetric cryptographic algorithms, the entire secret key material is known to all communication partners. Because the signer cannot keep the key material under his sole control (see *Article 26(c)* in the eIDAS-Regulation [(EU)910/2014]) , no advanced electronic signature can be realised with symmetrical cryptographic algorithms - the authenticity and integrity can be ensured, but not the non-repudiation.

Secure Electronic Signature Creation Device

Pursuant to *Article 3(23)* and *Article 3(32)* in the eIDAS-Regulation [(EU)910/2014], a ”secure electronic signature creation device”, also called qualified electronic signature creation device or Qualified Electronic Seal Creation Device,”means an electronic signature creation device, which meets the requirements laid down in *Annex II*” in the eIDAS-Regulation [(EU)910/2014].

Secure Socket Layer (SSL)

SSL is a protocol developed originally by Netscape for the secure transmission of data that is used especially for the secure transmission of websites between the web server and the browser.

Security Token

The term ”security token” has multiple meanings. In the framework of Web Service Security (see [WS-Security(v1.1)]) it is a sequence of claims. On the other side, a ”token” may also be a hardware-based PSE, as for example a smart card.

Shell Model

The shell model is a validity model for digital signatures in which a signature is only considered valid if at the time of the *verification* of the signature all of the certificates in the certificate path were valid. The shell model is particularly well suited to verify digital signatures for the purpose of authentication. Also see the hybrid model and the chain model.

Service-Oriented Architecture

A service-oriented architecture is a software infrastructure, in which program moduls are implemented as loosely coupled services, which communicate with each other via webservice-interfaces and enable to build distributed applications incorporating complex systems, running on different operating systems and written in different languages, over a network.

Service Provider

A service provider offers special services in a service-orientierted architecture, see also [eIDAS-TL].

Service Provisioning Markup Language (SPML)

The service provisioning markup language (SPML) [SPML(v1.0), SPML(v2)] is a framework specified from OASIS for exchanges of attributes between with cooperating organizations for user, resources and services .

SHA-2

The Secure Hash Algorithm SHA-2 embodies a collection of four cryptographic hash functions:

- SHA-224: 28 bytes (224 Bit) long hash value,
- SHA-256: 32 bytes (256 Bit) long hash value,
- SHA-384: 48 bytes (384 Bit) long hash value,
- SHA-512: 64 bytes (512 Bit) long hash value.

This standard was standardised by the US- National Institute for Standards and Technology (NIST).

SHA-3

In 2015, the cryptographic Hash Algorithm SHA-3 was standardised by the US- National Institute for Standards and Technology (NIST) as an alternative to SHA-2. This standard provides the following SHA3-versions:

- SHA3-224: 28 bytes (224 Bit) long hash value,
- SHA3-256: 32 bytes (256 Bit) long hash value,
- SHA3-384: 48 bytes (384 Bit) long hash value,
- SHA3-512: 64 bytes (512 Bit) long hash value,
- SHAKE128: variable hash length,
- SHAKE256: variable hash length.

Signatory

Pursuant to *Article 3 (9)* in the eIDAS-Regulation [(EU)910/2014], a ”signatory means a natural person who creates an electronic signature”.

Signature Algorithm

A signature algorithm is an asymmetric cryptographic algorithm that is used to create digital signatures. The most popular signature algorithms include RSA, DSA and ECDSA.

Signature Creation Device

A signature creation device is hardware or software in which private keys required for the creation of signatures can be stored as in a PSE and also applied. Smart Cards, HSMs or standard computer systems can be used as signature creation devices; in doing so the private key is saved, for example, as a file encrypted with a password in PKCS #12 format. Secure signature creation devices are needed for the creation of qualified electronic signatures.

Signature Format

In order that the recipient of a signed message may verify a digital signature, the sender of the signature must create the signature using a standardised signature format. In this case, one differentiates between raw low-level signature formats and extended high-level signature formats. The basic difference is that high-level signature formats can contain additional information as well as the raw signature data: e.g. the time of the signature creation and the certificate needed for the validation of the signature or a reference to it.

The Commission Implementing Decision (EU) [(EU)2015/1506] of 8 September 2015 lays ”down specifications relating to formats of advanced electronic signatures and advanced electronic seals to be recognised by public sector bodies pursuant to Articles 27(5) and 37(5) of Regulation (EU) No 910/2014 ..” (see [(EU)910/2014, ETSI-319122-1(v1.2.1), ETSI-319142-1(v1.1.1)] and [(EU)2015/1506]). Pursuant to this Commission Implementing Decision [(EU)2016/650], four binding high-level-signature formats are defined: CAdES (see [ETSI-103173(v2.2.1)]), XAdES (see [ETSI-103171(v2.1.1)]), PAdES (see [ETSI-103172(v2.2.2)]) and ASiC (see [ETSI-103174(v2.2.1)]).

Signature Validation
The signature verification includes two different verification steps. In the first step, the mathematical validity of the low-level signature is verified to prove the integrity. In the second step, the entire signature is verified with regard to the validity model at its basis to prove the authenticity. This procedure, which can be complex under certain circumstances, includes the examination of whether the certificate used for creation of the signature is valid at the reference point in time, i.e. the time of signature creation for a qualified electronic signature, and the current point in time for a signature used for authentication. During the verification of the validity of a certificate it is asked whether the low-level signature created by the issuing authority is mathematically valid, whether the certificate extensions defined in [RFC5280] were done correctly, whether a certificate path to a trustworthy root certificate or trust anchor can be built, and whether the certificate has been revoked.

Simple Electronic Signature
A ”simple electronic signature” is understood to be an electronic signature that does not fulfil all of the requirements of an advanced electronic signature.

Smart Card
A chip card with an integrated processor is also called a smart card. It can serve as a PSE in order to store trustworthy certificates and private keys in a secure manner and also act as a (secure signature creation device).

S/MIME
Secure Multipurpose Internet Mail Extensions (S/MIME) is a format for encrypting and signing emails and email attachments in MIME format [RFC2045] originally developed by the laboratories of RSA Security Inc., now standardised in IETF in [RFC8550, RFC8551] based on PKCS #7/CMS.

Software Publishing Certificate (SPC)
An SPC is a data structure used in the scope of Microsoft Authenticode Technology that is used during the signing of program code. It consists of a PKCS #7 structure in which one or several X.509 certificates are located.

Subexponential Run Time
An Algorithm with an asymptotic run time shorter than exponential has a subexponential run time. One often uses the $L_n[u, v]$-function for the description of the run time of this algorithm.

Subgroup
A subgroup U of a group G is a subset of G which, in turn, builds a group with the group operation of G. The order of such a subgroup is always a divisor of the group order (see [Buch10, Theorem 2.10.2]).

Supervisory Body
Pursuant to Article 17 of the eIDAS-Regulation ([(EU)910/2014]) ”Member States shall designate a supervisory body established in their territory or, upon mutual agreement with another Member State, a supervisory body established in that other Member State. That body shall be responsible for supervisory tasks in the designating Member State.”

Tested and Certified Products
The Commission Implementing Decision [(EU)2016/650] of 25 April 2016 defines standards for the security assessment of qualified secure signature creation device or Electronic Seal Creation Device pursuant to *Article 30(3) and 39(2)* of the eIDAS Regulation [(EU)910/2014], which are deployed in the environment of qualified electronic signatures or qualified electronic seals (see [EN419211-1, EN419211-2, EN419211-3, EN419211-4, EN419211-5, EN419211-6]). These requirements shall be strictly fulfilled by the trust service providers.

Text Form
For the text form defined in § 126b [BGB] ”the declaration must be delivered in a document or another manner in letters suitable for long-term rendition that names the declaring person and makes the conclusion of the declaration recognisable by means of reproduction of the signatures or in another way.” For example, text form can also be achieved with an unsigned electronic document, e.g. one in PDF-format.

Time Assertions
Time-stamp token or Evidence Record

Time Stamp
See electronic time stamp.

Time Stamp Protocol (TSP)
TSP is a client-server protocol standardised in [RFC3161] (updated by [RFC5816]) by IETF for the issuance of time stamps.

Time Stamp Authority (TSA)
A time stamp service issues time stamps. Often, the time stamp protocol specified within IETF is used in doing so.

Time Stamp Token
data object defined in [RFC3161], representing a time stamp

Transmission Control Protocol (TCP)
The TCP specified in [RFC793] is a reliable, connection-oriented transport protocol protocol in computer networks that is also used in the Internet.

Transport Layer Security
The transport layer security protocol [RFC8446] was developed by IETF the successor to the SSL-protocol.

Trustworthiness

”The attribute of a person or enterprise that provides confidence to others of the qualifications, capabilities, and reliability of that entity to perform specific tasks and fulfill assigned responsibilities” [CNSSI-4009].

Trust Anchor

Pursuant to [ETSI-119001(v1.2.1)], a ”trusted anchor” is an ”entity that is trusted by a relying party and used for validating certificates in certification paths”.

Trusted List

Pursuant to [ETSI-119001(v1.2.1)], a ”trusted list”[4] ”provides information about the status and the status history of the Trust Services from Trust Service Providers regarding compliance with the applicable requirements and the relevant provisions of the applicable legislation”.

Trust Service

”Trust service” ”means an electronic service normally provided for remuneration which consists of:

(a) the creation, verification, and validation of electronic signatures, electronic seals or electronic time stamps, electronic registered delivery services and certificates related to those services, or

(b) the creation, verification and validation of certificates for website authentication; or

(c) the preservation of electronic signatures, seals or certificates related to those services”

(see *Article 3 (16)* in the eIDAS-Regulation [(EU)910/2014]).

Trust Service Practice Statement

Pursuant to [ETSI-119001(v1.2.1)], a trust service practice statement is a ”statement of the practices that a trust service provider employs in providing a trust service.”

Trust Service Policy

Pursuant to [ETSI-119001(v1.2.1)], a trust service policy is a ”set of rules that indicates the applicability of a trust service to a particular community and/or class of application with common security requirements.”

Trust Service Provider

”Trust service provider’ means a natural or a legal person who provides one or more trust services either as a qualified or as a non-qualified trust service provider” (see *Article 3 (19)* in the eIDAS-Regulation [(EU)910/2014]). See also Qualified Trust Service Provider.

Uniform Resource Identifier (URI)

An uniform resource identifier (URI) is a string, which fullfils the syntax, defined in [RFC3986], and which has the purpose to identify an abstract or physical resource.

[4] European Commission page on EU Member States Trusted Lists: https://ec.europa.eu/digital-agenda/en/eu-trusted-lists-certification-service-providers

United Nations Centre for Trade Facilitation and Electronic Business (UN/CEFACT)
CEFACT `http://www.unece.org/cefact/`, sponsored by the United Nations, has the goal of promoting, simplifying, and harmonizing international trade. It is responsible, among other things, for the international data standard EDIFACT and is one of initiators of ebXML.

Unlinkability
Unlinkability means the incapability of stating the relation between two partial identities of an entity (see [PfHa07]).

Validation Data
Pursuant to *Article 3(40)* in the eIDAS-Regulation [(EU)910/2014], a ”validation data” means electronic seal validation data or electronic signature validation data.

Validity Model
The validity model determines under which circumstances a digital signature is considered valid. The most popular validity models include the shell model, the hybrid model and the chain model.

Web Service
A web service is a software component, which can uniquely be identified by an uniform resource identifier (URI) and whose inferfaces are defined and described by the web services description language.

Web Services Description Language (WSDL)
The web services description language (WSDL) [WSDL(v2.0)] defines a platform-neutral XML-specification for the description of web services, independent from programming languages and communication protocols.

World Wide Web Consortium (W3C)
The World Wide Web Consortium (W3C) (`http://www.w3.org`) develops specifications, guidelines, software and tools that are conducive for exploiting the potential of the Web.

Written Form
The written form is standardised in civil law in § 126 [BGB]. It stipulates that written documents are signed personally by the issuer. The written form can be replaced regularly by the so-called electronic form pursuant to § 126 clause 3 [BGB] that is achieved according to § 126a [BGB] by means of qualified electronic signatures. Thus, the written form can be replaced by using qualified electronic signatures, and one can do without costly paper processes.

X.500
[X.500] is a recommendation developed by ITU for a (global) directory service in which the entries are organised in a hierarchical directory tree, the so-called ”directory information tree (DIT)”, and are addressed by their distinguished names. The ”directory access protocol (DAP)”, specified in [X.519], is intended for access to the entries in this directory. Because this protocol is comparatively complex, today one usually uses the simpler lightweight directory access protocol (LDAP) to access the entries in the directory services.

X.509
X.509 is a recommendation developed by the ITU [X.509] for a framework for authentication with the use of asymmetric cryptographic algorithms. In this standard, very common, widely used formats for certificates and revocation lists are specified in particular.

XAdES (digital) signature
Pursuant to [ETSI-119001(v1.2.1)], an ”XAdES (digital) signature” means a ”digital signature that satisfies the requirements specified within [ETSI-319132-1(v1.2.1)] or [ETSI-319132-2(v1.1.1)]”.

XML Digital Signature (XML-DSig)
For the digital signature of data in XML-format, a specific signature format was developed by a work group of W3C, [XML-DSig, RFC3275]. Compared to the generic signature format PKCS #7, with which data in any format can be signed, XML-DSig offers a higher level of flexibility that is needed in order to completely exploit the potential of XML in the scope of digital signatures.

Bibliography

[1999/93/EC] *Directive 1999/93/EC of the European Parliament and of the Council on a Community framework for electronic signatures).* http://data.europa.eu/eli/dir/1999/93/oj, 1999.

[2012/0146/COD] European Commission. *Procedure 2012/0146/COD.* http://eur-lex.europa.eu/procedure/EN/2012_146, June 2012.

[9492/21] European Commission. *Report from the Commission to the European Parliament and the Council.* https://www.parlament.gv.at/dokument/XXVII/EU/63534/imfname_11071235.pdf, 2021.

[95/46/EC] *Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data.* http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=CELEX:31995L0046:en:HTML, 1995.

[AK] Dr. U.; Dr. D Hühnlein Ahmad, J.; Korte. *Vertrauenswürdige E-Akte auf Basis von TR-RESISCAN & TR-ESOR.* https://www.ecsec.de/fileadmin/Ecsec-files/pub//DACH2018-E-Akte.pdf, 2018.

[AKS04] Manindra Agrawal, Neeraj Kayal, and Nitin Saxena. *PRIMES is in P. Annals of Mathematics*, volume 160(2):781–793. https://www.cse.iitk.ac.in/users/manindra/algebra/primality_v6.pdf, 2004.

[AMV90] G. Agnew, R.Mullin, and S.Vanstone. *Improved digital signature scheme based on discrete exponentiation. Electronic Letters*, volume 26:1024–1025. https://digital-library.theiet.org/content/journals/10.1049/el_19900663, 1990.

[ANSI-X9.31] American National Standards Institute (ANSI). *Digital Signatures Using Reversible Public Key Cryptography for the Financial Services Industry (rDSA).* Public Key Cryptography for the Financial Services Industry – X9.31, September 1998.

[ANSI-X9.62] American National Standards Institute (ANSI). *Public Key Cryptography for the Financial Services Industry: the Elliptic Curve Digital Signature Algorithm (ECDSA).* Public Key Cryptography for the Financial Services Industry – X9.62, 2005.

[BaSh96] Eric Bach and Jeffrey Shallit. *Algorithmic Number Theory*, volume 1 (MIT Press, 1996). https://mitpress.mit.edu/books/algorithmic-number-theory-volume-1.

[BCCN01] E. Brier, C. Clavier, J.S. Coron, and D. Naccache. *Cryptanalysis of RSA signatures with fixed pattern padding.* In Joe Kilian (editor), *Advances in Cryptology – CRYPTO 2001*, volume 2139 of *Lecture Notes in Computer Science*, pages 276–292 (Springer-Verlag, 2001).

[Bern09] Daniel J. Bernstein et al. *Post-Quantum Cryptography*. In *Post-Quantum Cryptography* (Springer, 2009). https://link.springer.com/book/10.1007/978-3-540-88702-7.

[BeRo93] Mihir Bellare and Paul Rogaway. *Random Oracles are Practical: a Paradigm for Designing Efficient Protocols*. In *1st ACM Conference on Computer and Communications Security*, pages 62–73 (1993).

[BeRo96] Mihir Bellare and Paul Rogaway. *The Exact Security of Digital Signatures - How to Sign with RSA and Rabin*. In *Advances in Cryptology – EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 399–416 (Springer-Verlag, 1996).

[BeRo98] Mihir Bellare and Paul Rogaway. *PSS: Provably Secure Encoding Method for Digital Signatures*. Einreichung zur IEEE P1363 Arbeitsgruppe. https://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.79.3835&rep=rep1&type=pdf, 1998.

[Bert01] Andreas Bertsch. *Digitale Signaturen*. Xpert.press (Springer Verlag, 2001). ISBN 3-540-42351-6, https://www.springer.com/de/book/9783540423515.

[BGB] *Bürgerliches Gesetzbuch*. RGBl 1896, 195, Neugefasst durch Bek. v. 2. 1.2002 I 42, 2909; 2003, 738; zuletzt geändert durch Art. 1 G v. 21. 4.2005 I 1073. http://bundesrecht.juris.de/bundesrecht/bgb/, 1896.

[Blake99] Ian Blake et al. *Elliptic Curves in Cryyptology*. In *Elliptic Curves in Cryyptology*, London Mathematical Society Lecture Note Series 265 (Cambridge University Press, 2013). https://www.cambridge.org/core/books/elliptic-curves-in-cryptography/16A2B60636EFA7EBCC3D5A5D01F28546.

[Blei96] Daniel Bleichenbacher. *Generating EIGamal Signatures Without Knowing the Secret Key*. In *Advances in Cryptology – EUROCRYPT '96*, volume 1070 of *Lecture Notes in Computer Science*, pages 10–18 (Springer-Verlag, 1996).

[BLP93] Joe Buhler, Hendrik Lenstra, and Carl Pomerance. *Factoring integers with the number fields sieve*. In A.K. Lenstra and H.W. Lenstra (editors), *The Developement of the Number Field Sieve*, volume 1554 of *Lecture Notes in Mathematics*, pages 50–94 (Springer-Verlag, 1993). https://www.researchgate.net/publication/225539588_Factoring_integers_with_the_number_field_sieve.

[BoDu99] Dan Boneh and Glen Durfee. *Cryptanalysis of RSA with private key d less than $N^{0.292}$*. In *Advances in Cryptology – EUROCRYPT '99*, volume 1592 of *Lecture Notes in Computer Science*, pages 213–229 (Springer-Verlag, 1999).

[Born02] Folkmar Bornemann. *Ein Durchbruch für "Jedermann"*. DMV-Mitteilungen. https://www-m3.ma.tum.de/foswiki/pub/M3/Allgemeines/FolkmarBornemannPublications/aks.pdf, 2002.

[BOU-2022] Fabrice Boudot, Pierrick Gaudry, Aurore Guillevic, Nadia Heninger, Emmanuel Thomé, and Paul Zimmermann. *The state of the art in integer factoring and breaking public key cryptography*. https://hal.science/hal-03691141/file/cryptography.pdf, 2022.

[BSI-319401-AssP1] Federal Office for Information Security. *Criteria for Assessing Trust Service Providers against ETSI Policy Requirements, Part 1: Assessment Criteria for all TSP - ETSI EN 319 401*. https://www.bundesnetzagentur.de/EVD/SharedDocuments/Downloads/QES/Assessment-Handbuch_ETSI_319_401.pdf?__blob=publicationFile&v=3.

[BSI-319511-AssP2] Federal Office for Information Security. *Criteria for Assessing Trust Service Providers against ETSI Policy Requirements, Part 2: Assessment Criteria providing long-term preservation of digital signatures or general data using digital signature techniques - ETSI TS 119 511*. https://www.bundesnetzagentur.de/EVD/SharedDocuments/Downloads/QES/Assessment-Handbuch_ETSI_119_511.pdf;jsessionid=E73A98F6D5613FA18BBD2ADF5625C554?__blob=publicationFile&v=3.

[BSI-eIDAS-nPA] Federal Office for Information Security. *eIDAS Notification of the German eID*. https://www.bsi.bund.de/EN/Topics/ElectrIDDocuments/German-eID/eIDAS-notification/eIDAS_notification_node.html, August 2017.

[BSI-Q20] Federal Office for Information Security (BSI). *Status of Quantum Computer Development*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/Studien/Quantencomputer/P283_QC_Studie-V_1_2.html, June 2020.

[BSI-Sig-Leit] Bundesamt für Sicherheit in der Informationstechnik. *Leitlinie für digitale Signatur-/Siegel-, Zeitstempelformate sowie technische Beweisdaten (Evidence Record)*. Version 1.0. https://www.bundesnetzagentur.de/EVD/SharedDocuments/Downloads/QES/BSI_TR_03125.pdf?__blob=publicationFile&v=1, 2020.

[BSI-TR-03125] Federal Office for Information Security. *TR-ESOR Preservation of Evidence of Cryptographically Signed Document*. Technical Guideline (main document), Version 1.2.2 and higher. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-C1] Federal Office for Information Security. *BSI Technical Guideline (BSI-TR-03125) Preservation of Evidence of Cryptographically Signed Documents Annex TR-ESOR-C.1: Conformity Test Specification (Level 1 – Functional Conformity)*. Version 1.2.2 and higher. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-C2] Federal Office for Information Security. *BSI Technical Guideline BSI-TR-03125 (TR-ESOR) C.2: Conformity Test Specification (Level 2 Technical Conformity*. Technical guideline (BSI-TR-03125), Version 1.2.2 and higher. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-E] BSI TR 03125 E. *BSI Technical Guideline (BSI-TR-03125) Annex TR-ESOR-E: Concretisation of the Interfaces on the Basis of the eCard-API-Framework Version 1.2.2 und higher*. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-ERS] Federal Office for Information Security. *BSI Technical Guideline (BSI-TR-03125) Preservation of Evidence of Cryptographically Signed Documents - EvidenceRecord Profiling pursuant to RFC4998 and RFC6283*. Version 1.2.2 and higher. http://www.bsi.bund.de/EN/tr-esor, 2022.

[BSI-TR-03125-F] BSI TR 03125 F. *BSI Technical Guideline BSI-TR-03125 Annex TR-ESOR-F: Formats, v1.2.2 and higher*. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-lt] Federal Office for Information Security. *Guideline for Data Preservation pursuant to BSI TR-03125 TR-ESOR – Guidance for public authorities and businesses*. Eine Handlungshilfe für Behörden und Unternehmen (BSI-TR-ESOR-LEIT), Version 1.2.1 und 1.2.2. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-M3] BSI TR 03125 M.3. *BSI Technical Guideline BSI-TR-03125 Annex TR-ESOR-M.3: ArchiSig-Module, v1.2.2 and higher*. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-PEPT] BSI TR 03125 PEPT. *BSI Technical Guideline 03125 Preservation of Evidence of Cryptographically Signed Documents Annex TR-ESOR-PEPT: Preservation Evidence Policy Template for TR-ESOR (PEPT), v1.2.1 and higher*. https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Publikationen/TechnischeRichtlinien/TR03125/BSI_TR_03125_Anlage_PEPT_V1_2_1_higher.pdf?__blob=publicationFile&v=2.

[BSI-TR-03125-TRANS] BSI TR 03125 TRANS. *Federal Office for Information Security. BSI Technical Guideline BSI-TR-03125, Preservation of Evidence of Cryptographically Signed Documents, Annex TR-ESOR-TRANS*. http://www.bsi.bund.de/EN/tr-esor.

[BSI-TR-03125-VR] BSI TR 03125 VR. *BSI Technical Guideline BSI-TR-03125, Preservation of Evidence of Cryptographically Signed Document, Annex TR-ESOR-VR, v1.2.1 and higher*. http://www.bsi.bund.de/EN/tr-esor.

[BSL04] D. Boneh, H. Shacham, and B. Lynn. *Short signatures from the Weil pairing*. *Journal of Cryptology*, volume 17(4):297–319. http://crypto.stanford.edu/~dabo/abstracts/weilsigs.html, 2004.

[Buch10] Johannes Buchmann. *Einführung in die Kryptographie* (Springer–Verlag, 2010). https://link.springer.com/book/10.1007/978-3-642-11186-0.

[BuWi88] Johannes Buchmann and Hugh C. Williams. *A key-exchange system based on imaginary quadratic fields*. *Journal of Cryptology*, volume 1(3):107–118. https://www.researchgate.net/publication/220478946_A_Key-Exchange_System_Based_on_Imaginary_Quadratic_Fields, 1988.

[CC] CCMB. *Common Criteria for Information Technology Security Evaluation*. Version 3.1, part 1-3. http://www.commoncriteriaportal.org/cc/, 2017.

[CEG87] David Chaum, Jan-Hendrik Evertse, and Jeroen van de Graaf. *An Improved Protocol for Demonstrating Possession of Discrete Logarithms and some Generalizations*. In *Advances in Cryptology – EUROCRYPT '87*, volume 304 of *Lecture Notes in Computer Science*, pages 127–141 (Springer-Verlag, 1988). https://www.researchgate.net/publication/221348326_An_Improved_Protocol_for_Demonstrating_Possession_of_Discrete_Logarithms_and_Some_Generalizations.

[CHJ99] D. Coppersmith, S. Halevi, and C. Jutla. *ISO 9796-1 and the new forgery strategy*. Research contribution to IEEE P1363. http://mpqs.free.fr/attack9796.pdf, 1999.

[CNS99] J.S. Coron, D. Naccache, and J.P. Stern. *On the security of RSA Padding*. In Michael Wiener (editor), *Advances in Cryptology – CRYPTO '99*, volume 1666 of *Lecture Notes in Computer Science*, pages 1–18 (Springer-Verlag, 1999). http://www.crypto-uni.lu/jscoron/publications/padding.pdf.

[CNSSI-4009] Committee on National Security Systems (CNSS). *Glossary*. https://rmf.org/wp-content/uploads/2017/10/CNSSI-4009.pdf, april 2015.

[COM(2012)238] European Commission. *Proposal for a regulation of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market*. http://eurlex.europa.eu/LexUriServ/LexUriServ.do?uri=COM:2012:0238:FIN:EN:PDF, June 2012.

[COMMON-PKI-BNetzA] T7 e.V. and TeleTrusT e.V. *COMMON PKI SPECIFICATIONS FOR INTEROPERABLE APPLICATIONS - Common PKI V2.0*. https://www.bundesnetzagentur.de/EVD/SharedDocuments/Downloads/QES/Common_PKI_v2.0_02.pdf, Januar 2009.

[CP-Col17] Gaetan Leurent et al. *From Collisions to Chosen-Prefix Collisions Application to Full SHA-1*. IACR ePrint Archive Report. https://eprint.iacr.org/2019/459.pdf.

[DaLu05] Magnus Daum and Stefan Lucks. *The Story of Alice and her Boss: Hash Functions and the Blind Passenger Attack*. Beitrag zur Rump-Session der Eurocrypt 2005. http://www.cits.rub.de/imperia/md/content/magnus/rump_ec05.pdf, 2005.

[Damg89] Ivan Damgård. *A Design Principle for Hash Functions*. In *Advances in Cryptology – CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 416–427 (Springer-Verlag, 1990). https://link.springer.com/book/10.1007/0-387-34805-0.

[DeOd85] Yvo Desmedt and Andrew Odlyzko. *A chosen text attack on the RSA cryptosystem and some discrete logarithm schemes*. In *Advances in Cryptology – CRYPTO '85*, volume 218 of *Lecture Notes in Computer Science*, pages 516–522 (Springer-Verlag, 1986). https://link.springer.com/chapter/10.1007/3-540-39799-X_40.

[DiHe76] Whitfield Diffie and Martin E. Hellman. *New Directions in Cryptography*. *IEEE Transactions on Information Theory*, volume 22(6):644–654, 1976.

[Dili18] Leo Ducas et al. *CRYSTALS-Dilithium:A Lattice-Based Digital Signature Scheme*. IACR Transactions on Cryptographic Hardware and Embedded Systems. https://tches.iacr.org/index.php/TCHES/article/view/839/791.

[DIN16560-15] Deutsches Institut für Normung (DIN). *EDIFACT - Anwendungsregeln - Teil 15: Anwendung des Service-Nachrichtentyps AUTACK zur Übermittlung von Integritäts- und Authentizitätsinformationen über versendete Nutzdaten*. DIN 16560-15, Beuth Verlag, Juli 2003.

[DoBP96] H. Dobbertin, A. Bosselaers, and B. Preneel. *RIPEMD-160, a strengthened version of RIPEMD*. In Dieter Gollmann (editor), *Fast Software Encryption: Third International Workshop Cambridge, UK, February 21–23 1996 Proceedings*, volume 1039 of *LNCS*, pages 71–82 (Springer, 1996). https://link.springer.com/content/pdf/10.1007%2F3-540-60865-6_44.pdf.

[Dubu00] Olivier Dubuisson. *ASN.1 – Communication between Heterogeneous Systems* (OSS Nokalva, 2000). ISBN:0-12-6333361-0.

[(EC)765/2008] *Regulation (EC) No 765/2008 of the European Parliament and of the Council of 9 July 2008 setting out the requirements for accreditation and market surveillance relating to the marketing of products and repealing Regulation (EEC) No 339/93 (Text with EEA relevance)*. http://data.europa.eu/eli/reg/2008/765/oj, 2008.

[EHS09] Jan Eichholz, Detlef Hühnlein, and Jörg Schwenk. *SAMLizing the European Citizen Card*. In *Proceedings of* BIOSIG 2009: Biometrics and Electronic Signatures, volume 155 of *Lecture Notes in Informatics (LNI)*, pages 105–117 (GI-Edition, 2009). http://www.ecsec.de/pub/SAMLizing-ECC.pdf.

[eIDAS-CN] CEF Digital. *eIDAS Cooperation Network space*. https://ec.europa.eu/cefdigital/wiki/display/EIDCOMMUNITY/Cooperation+Network+Resources, 2018.

[eIDAS-CNO-16-01] eIDAS Cooperation Network. *Opinion No. 1/2016 of the Cooperation Network on version 1.0 of the eIDAS Technical specifications*. https://ec.europa.eu/cefdigital/wiki/pages/viewpage.action?pageId=29655984, January 2016.

[eIDAS-CNO-16-02] eIDAS Cooperation Network. *Opinion No. 2/2016 of the Cooperation Network on version 1.1 of the eIDAS Technical specifications*. https://ec.europa.eu/cefdigital/wiki/pages/viewpage.action?pageId=37750723, December 2016.

[eIDAS-CR] eIDAS Technical Subgroup. *eIDAS - Cryptographic requirements for the Interoperability Framework - TLS and SAML*. Version 1.0. https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eidas_-_crypto_requirements_for_the_eidas_interoperability_framework_v1.0.pdf, November 2015.

[eIDAS-IA] eIDAS Technical Subgroup. *eIDAS Interoperability Architecture*. Version 1.2. https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20Interoperability%20Architecture%20v.1.2%20Final.pdf, September 2019.

[eIDAS-SAML-AP] eIDAS Technical Subgroup. *eIDAS SAML Attribute Profile*. Version 1.2. https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20SAML%20Attribute%20Profile%20v1.2%20Final.pdf, September 2019.

[eIDAS-SAML-MF] eIDAS Technical Subgroup. *eIDAS SAML Message Format*. Version 1.2. https://ec.europa.eu/cefdigital/wiki/download/attachments/82773108/eIDAS%20SAML%20Message%20Format%20v.1.2%20Final.

pdf?version=3&modificationDate=1571068651727&api=v2, August 2019.

[eIDAS-TL] *eIDAS Trusted List Browser*. https://esignature.ec.europa.eu/efda/tl-browser/#/screen/home.

[ElGa85] Taher ElGamal. *A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms*. *IEEE Transactions on Information Theory*, volume 31(4):469–472. https://link.springer.com/chapter/10.1007/3-540-39568-7_2, 1985.

[EN419211-1] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 1: Overview*. EN 419 211 (Part 1). https://www.en-standard.eu/csn-en-419211-1-protection-profiles-for-secure-signature-creation-device-part-1-overview/, 2014.

[EN419211-2] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 2: Device with key generation*. EN 419 211 (Part 2). https://www.en-standard.eu/csn-en-419211-2-protection-profiles-for-secure-signature-creation-device-part-2-device-with-key-generation/, 2013.

[EN419211-3] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 3: Device with key import*. EN 419 211 (Part 3). https://www.en-standard.eu/csn-en-419211-3-protection-profiles-for-secure-signature-creation-device-part-3-device-with-key-import/, 2013.

[EN419211-4] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 4: Extension for device with key generation and trusted channel to certificate generation application*. EN 419 211 (Part 4). https://www.en-standard.eu/csn-en-419211-4-protection-profiles-for-secure-signature-creation-device-part-4-extension-for-device-with-key-generation-and-trusted-channel-to-certificate-generation-application/, 2013.

[EN419211-5] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 5: Extension for device with key generation and trusted channel to signature creation application*. EN 419 211 (Part 5). https://www.en-standard.eu/csn-en-419211-5-protection-profiles-for-secure-signature-creation-device-part-5-extension-for-device-with-key-generation-and-trusted-channel-to-signature-creation-application/, 2013.

[EN419211-6] European Comittee for Standardisation (CEN). *Protection profiles for secure signature creation device – Part 6: Extension for device with key import and trusted channel to signature creation application*. EN 419 211 (Part 6). https://www.en-standard.eu/csn-en-419211-6-protection-profiles-for-secure-signature-creation-device-part-6-extension-for-device-with-key-import-and-trusted-channel-to-signature-creation-application/, 2014.

[ENISA-IR] European Union Agency for Network and Information Security (ENISA). *Proposal for Article 19 Incident reporting.* https://www.enisa.europa.eu/publications/technical-guideline-for-incident-reporting, December 2015.

[ENISA-SL] European Union Agency for Network and Information Security (ENISA). *Security framework for TSPs - Guidelines on maintaining appropriate security level.* https://www.enisa.europa.eu/topics/trust-services/guidelines/appropriate_security_level, November 2016.

[ENISA-STS] European Union Agency for Network and Information Security (ENISA). *Guidelines on supervision of qualified trust service providers.* Version 0.4. https://www.enisa.europa.eu/topics/trust-services/guidelines/supervision_tsps, September 2016.

[ENSIA] Sławomir Górniak – ENISA. *Standardisation in the field of Electronic Identities and Trust Service Providers.* https://www.enisa.europa.eu/publications/standards-eidas, 2015.

[ETSI-019510(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Scoping study and framework for standardization of long-term data preservation services, including preservation of/with digital signatures.* ETSI SR 019 510, Version 1.1.1. https://www.etsi.org/deliver/etsi_sr/019500_019599/019510/01.01.01_60/sr_019510v010101p.pdf, May 2017.

[ETSI-101733(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES).* ETSI TS 101 733, Version 2.2.1. https://www.etsi.org/deliver/etsi_ts/101700_101799/101733/02.02.01_60/ts_101733v020201p.pdf, April 2013.

[ETSI-101903(v1.4.2)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); XML Advanced Electronic Signatures (XAdES).* ETSI TS 101 903, Version 1.4.2. https://www.etsi.org/deliver/etsi_ts/101900_101999/101903/01.04.02_60/ts_101903v010402p.pdf, December 2010.

[ETSI-102778-2(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 2: PAdES Basic - Profile Based on ISO 32000-1.* ETSI TS 102 778-2, Version 1.2.1. https://www.etsi.org/deliver/etsi_ts/102700_102799/10277802/01.02.01_60/ts_10277802v010201p.pdf, July 2009.

[ETSI-102778-3(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 3: PAdES Enhanced - PAdES-BES and PAdES-EPES Profiles.* ETSI TS 102 778-3, Version 1.2.1. https://www.etsi.org/deliver/etsi_ts/102700_102799/10277803/01.02.01_60/ts_10277803v010201p.pdf, July 2010.

[ETSI-102778-4(v1.1.2)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature*

Profiles; Part 4: PAdES Long Term - PAdES LTV Profile. ETSI TS 102 778-4, Version 1.1.2. https://www.etsi.org/deliver/etsi_ts/102700_102799/10277804/01.01.02_60/ts_10277804v010102p.pdf, December 2009.

[ETSI-102778-6(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PDF Advanced Electronic Signature Profiles; Part 6: Visual Representations of Electronic Signatures*. ETSI TS 102 778-6, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/102700_102799/10277806/01.01.01_60/ts_10277806v010101p.pdf, July 2010.

[ETSI-103171(v2.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); XAdES Baseline Profile*. ETSI TS 103 171, Version 2.1.1. https://www.etsi.org/deliver/etsi_ts/103100_103199/103171/02.01.01_60/ts_103171v020101p.pdf, March 2012.

[ETSI-103172(v2.2.2)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PAdES Baseline Profile*. ETSI TS 103 172, Version 2.2.2, update in progress. https://www.etsi.org/deliver/etsi_ts/103100_103199/103172/02.02.02_60/ts_103172v020202p.pdf, April 2013.

[ETSI-103173(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); CAdES Baseline Profile*. ETSI TS 103 173, Version 2.2.1, (update in preparation). https://www.etsi.org/deliver/etsi_ts/103100_103199/103173/02.02.01_60/ts_103173v020201p.pdf, April 2013.

[ETSI-103174(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); ASiC Baseline Profile*. ETSI TS 103 174, Version 2.2.1. https://www.etsi.org/deliver/etsi_ts/103100_103199/103174/02.02.01_60/ts_103174v020201p.pdf, June 2013.

[ETSI-119001(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); The Framework for Standardization of Signatures; Definitions and Abbreviations*. ETSI TR 119 001, Version 1.2.1, update in preparation. https://www.etsi.org/deliver/etsi_tr/119000_119099/119001/01.02.01_60/tr_119001v010201p.pdf, March 2016.

[ETSI-119102-1(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Procedures for Creation and, Validation of AdES Digital Signatures; Part 1: Creation and Validation*. ETSI TS 119 102-1, Version 1.2.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11910201/01.02.01_60/ts_11910201v010201p.pdf, August 2018.

[ETSI-119102-2(v1.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI);Procedures for Creation and Validation of AdES Digital Signatures; Part 2: Signature Validation Report*. ETSI TS 119 102-2, Version 1.3.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11910202/01.03.01_60/ts_11910202v010301p.pdf, September 2021.

[ETSI-119122-3(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); CAdES Digital Signatures; Part 3: Incorporation of Evidence Record Syntax (ERS) mechanisms in CAdES*. ETSI TS 119 122-3, Version 1.1.1. http://www.etsi.org/deliver/etsi_ts/119100_119199/11912203/01.01.01_60/ts_11912203v010101p.pdf, January 2017.

[ETSI-119132-3(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); XAdES digital signatures; Part 3: Incorporation of Evidence Record Syntax (ERS) mechanisms in XAdES*. ETSI TS 119 132-3, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11913203/01.01.01_60/ts_11913203v010101p.pdf, January 2021.

[ETSI-119172-1(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Signature Policies; Part 1: Building Blocks and Table of Contents for Human Readable Signature Policy Documents*. ETSI TS 119 172-1, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11917201/01.01.01_60/ts_11917201v010101p.pdf, July 2015.

[ETSI-119172-2(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Signature Policies; Part 2: XML format for signature policies*. ETSI TS 119 172-2, Version 1.1.1. http://www.etsi.org/deliver/etsi_ts/119100_119199/11917202/01.01.01_60/ts_11917202v010101p.pdf, December 2019.

[ETSI-119172-3(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Signature Policies; Part 3: ASN.1 format for signature policies*. ETSI TS 119 172-3, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11917203/01.01.01_60/ts_11917203v010101p.pdf, December 2019.

[ETSI-119172-4(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Signature Policies; Part 4: Signature applicability rules (validation policy) for European qualified electronic signatures/seals using trusted lists*. ETSI TS 119 172-4, Version 1.1.1, May 2021.

[ETSI-119182-1(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); JAdES digital signatures built on JSON Web Signatures; Part 1: Building blocks and JAdES baseline signatures*. ETSI TS 119 182-1, March 2021.

[ETSI-119312(v1.4.2)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Cryptographic Suites*. ETSI TS 119 312, Version 1.4.2. https://www.etsi.org/deliver/etsi_ts/119300_119399/119312/01.04.02_60/ts_119312v010402p.pdf, February 2022.

[ETSI-119441(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy requirements for TSP providing signature validation services*. ETSI TS 119 441, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/119400_119499/119441/01.01.01_60/ts_119441v010101p.pdf, August 2018.

[ETSI-119511(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy and security requirements for trust service providers providing long-term preservation of digital signatures or general data using digital signature techniques.* ETSI TS 119 511, Version 1.1.1. https://www.etsi.org/deliver/etsi_ts/119500_119599/119511/01.01.01_60/ts_119511v010101p.pdf, June 2019.

[ETSI-119512] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Protocols for trust service providers providing long-term data preservation services.* ETSI TS 119 512, Version 1.1.2. https://www.etsi.org/deliver/etsi_ts/119500_119599/119512/01.01.02_60/ts_119512v010102p.pdf, October 2020.

[ETSI-119612(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Trusted Lists.* ETSI TS 119 612, Version 2.2.1. http://www.etsi.org/deliver/etsi_ts/119600_119699/119612/02.02.01_60/ts_119612v020201p.pdf, April 2016.

[ETSI-319102-1(v1.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 1: Creation and Validation.* ETSI EN 319 102-1, Version 1.3.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31910201/01.03.01_60/en_31910201v010301p.pdf, November 2021.

[ETSI-319102-2(v1.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Procedures for Creation and Validation of AdES Digital Signatures; Part 2: Signature Validation Report.* ETSI EN 319 102-2, Version 1.3.1. https://www.etsi.org/deliver/etsi_ts/119100_119199/11910202/01.03.01_60/ts_11910202v010301p.pdf, September 2021.

[ETSI-319122-1(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); CAdES Digital Signatures; Part 1: Building Blocks and CAdES Baseline Signatures.* ETSI EN 319 122-1, Version 1.2.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31912201/01.02.01_60/en_31912201v010201p.pdf, October 2021.

[ETSI-319122-2(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); CAdES Digital Signatures; Part 2: Extended CAdES Signatures.* ETSI EN 319 122-2, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31912202/01.01.01_60/en_31912202v010101p.pdf, April 2016.

[ETSI-319132-1(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); XAdES Digital Signatures; Part 1: Building Blocks and XAdES Baseline Signatures.* ETSI EN 319 132-1, Version 1.2.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31913201/01.02.01_60/en_31913201v010201p.pdf, February 2022.

[ETSI-319132-2(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); XAdES Digital Signatures; Part 2: Extended XAdES Signatures.* ETSI EN 319 132-2, Version

1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31913202/01.01.01_60/en_31913202v010101p.pdf, April 2016.

[ETSI-319142-1(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PAdES Digital Signatures; Part 1: Building Blocks and PAdES Baseline Signatures.* ETSI EN 319 142-1, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31914201/01.01.01_60/en_31914201v010101p.pdf, April 2016.

[ETSI-319142-2(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); PAdES Digital Signatures; Part 2: Additional PAdES Signatures Profiles.* ETSI EN 319 142-2, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31914202/01.01.01_60/en_31914202v010101p.pdf, April 2016.

[ETSI-319162-1(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 1: Building Blocks and ASiC Baseline Containers.* ETSI EN 319 162-1, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31916201/01.01.01_60/en_31916201v010101p.pdf, April 2016.

[ETSI-319162-2(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Associated Signature Containers (ASiC); Part 2: Additional ASiC Containers.* ETSI EN 319 162-2, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319100_319199/31916202/01.01.01_60/en_31916202v010101p.pdf, April 2016.

[ETSI-319401(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); General Policy Requirements for Trust Service Providers.* ETSI EN 319 401, Version 2.2.1. https://www.etsi.org/deliver/etsi_en/319400_319499/319401/02.01.01_60/en_319401v020101p.pdf, April 2018.

[ETSI-319401(v2.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); General Policy Requirements for Trust Service Providers.* ETSI EN 319 401, Version 2.3.1. https://www.etsi.org/deliver/etsi_en/319400_319499/319401/02.03.01_60/en_319401v020301p.pdf, Mai 2021.

[ETSI-319403(v2.2.2)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Trust Service Provider Conformity Assessment - Requirements for Conformity Assessment Bodies Assessing Trust Service Providers.* ETSI EN 319 403, Version 2.2.2. https://www.etsi.org/deliver/etsi_en/319400_319499/319403/02.02.02_60/en_319403v020202p.pdf, August 2015.

[ETSI-319411-1(v1.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy and Security Requirements for Trust Service Providers Issuing Certificates; Part 1: General Requirements.* ETSI EN 319 411-1, Version 1.3.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941101/01.03.01_60/en_31941101v010301p.pdf, May 2021.

[ETSI-319411-2(v2.4.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Trust Service Providers issuing certificates; Part 2: Requirements for trust service providers issuing EU qualified certificates.* ETSI EN 319 411-2, Version 2.4.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941102/02.04.01_60/en_31941102v020401p.pdf, November 2021.

[ETSI-319412-1(v1.4.4)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 1: Overview and Common Data Structures.* ETSI EN 319 412-1, Version 1.4.4. https://www.etsi.org/deliver/etsi_en/319400_319499/31941201/01.04.04_60/en_31941201v010404p.pdf, May 2021.

[ETSI-319412-2(v2.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 2: Certificate Profile for Certificates Issued to Natural Persons.* ETSI EN 319 412-2, Version 2.2.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941202/02.02.01_60/en_31941202v020201p.pdf, July 2020.

[ETSI-319412-3(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 3: Certificate Profile for Certificates Issued to Legal Persons.* ETSI EN 319 412-3, Version 1.2.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941203/01.02.01_60/en_31941203v010201p.pdf, July 2020.

[ETSI-319412-4(v1.2.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 4: Certificate Profile for Web Site Certificates.* ETSI EN 319 412-4, Version 1.2.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941204/01.02.01_60/en_31941204v010201p.pdf, September 2021.

[ETSI-319412-5(v2.3.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Certificate Profiles; Part 5: QCStatements.* ETSI EN 319 412-5, Version 2.3.1. https://www.etsi.org/deliver/etsi_en/319400_319499/31941205/02.03.01_60/en_31941205v020301p.pdf, April 2020.

[ETSI-319421(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy and Security Requirements for Trust Service Providers Issuing Time-Stamps.* ETSI EN 319 421, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319400_319499/319421/01.01.01_60/en_319421v010101p.pdf, March 2016.

[ETSI-319521(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Policy and security requirements for Electronic Registered Delivery Service Providers.* ETSI EN 319 521, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319500_319599/319521/01.01.01_60/en_319521v010101p.pdf, February 2019.

[ETSI-319422(v1.1.1)] European Telecommunications Standards Institute (ETSI). *Electronic Signatures and Infrastructures (ESI); Time-Stamping Protocol and Time-Stamp Token Profiles.* ETSI EN 319 422, Version 1.1.1. https://www.etsi.org/deliver/etsi_en/319400_319499/319422/01.01.01_60/en_319422v010101p.pdf, March 2016.

[EU-QSCD-2021] European Commission. *Compilation of Member States notification on SSCDs and QSCDs.* https://esignature.ec.europa.eu/efda/notification-tool/#/screen/browse/list/QSCD_SSCD, 2021.

[(EU)182/2011] *Regulation (EC) No 182/2011 of the European Parliament and of the Council of of 16 February 2011 laying down the rules and general principles concerning mechanisms for control by Member States of the Commission's exercise of implementing powers).* http://data.europa.eu/eli/reg/2011/182/oj, 2011.

[(EU)2015/1501] *Commission Implementing Regulation (EU) 2015/1501 of 8 September 2015 on the interoperability framework pursuant to Article 12(8) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance)).* http://data.europa.eu/eli/reg_impl/2015/1501/oj, 2015.

[(EU)2015/1502] *Commission Implementing Regulation (EU) 2015/1502 of 8 September 2015 on setting out minimum technical specifications and procedures for assurance levels for electronic identification means pursuant to Article 8(3) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance).* http://data.europa.eu/eli/reg_impl/2015/1502/oj, 2015.

[(EU)2015/1505] *Commission Implementing Decision (EU) 2015/1505 of 8 September 2015 laying down technical specifications and formats relating to trusted lists pursuant to Article 22(5) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance).* http://data.europa.eu/eli/dec_impl/2015/1505/oj, 2015.

[(EU)2015/1506] *Commission Implementing Decision (EU) 2015/1506 of 8 September 2015 laying down specifications relating to formats of advanced electronic signatures and advanced seals to be recognised by public sector bodies pursuant to Articles 27(5) and 37(5) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance).* http://data.europa.eu/eli/dec_impl/2015/1506/oj, 2015.

[(EU)2015/1984] *Commission Implementing Decision (EU) 2015/1984 of 3 November 2015 defining the circumstances, formats and procedures of notification pursuant to Article 9(5) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (notified under document C(2015) 7369) (Text with EEA relevance).* http://data.europa.eu/eli/dec_impl/2015/1984/oj, 2015.

[(EU)2015/296] *Commission Implementing Decision (EU) 2015/296 of 24 February 2015 establishing procedural arrangements for cooperation between Member States on electronic identification pursuant to Article 12(7) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the in-*

ternal market Text with EEA relevance. http://data.europa.eu/eli/dec_impl/2015/296/oj, 2015.

[(EU)2015/806] *Commission Implementing Regulation (EU) 2015/806 of 22 May 2015 laying down specifications relating to the form of the EU trust mark for qualified trust services (Text with EEA relevance).* http://data.europa.eu/eli/reg_impl/2015/806/oj, 2015.

[(EU)2016/650] *Commission Implementing Decision (EU) 2016/650 of 25 April 2016 laying down standards for the security assessment of qualified signature and seal creation devices pursuant to Articles 30(3) and 39(2) of Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market (Text with EEA relevance).* http://data.europa.eu/eli/dec_impl/2016/650/oj, 2016.

[(EU)2016/679] *Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance).* http://data.europa.eu/eli/reg/2016/679/oj, 2016.

[(EU)2021/1153] *Regulation (EU) 2021/1153 of the European Parliament and of the Council of 7 July 2021 establishing the Connecting Europe Facility and repealing Regulations (EU) No 1316/2013 and (EU) No 283/2014 (Text with EEA relevance).* http://data.europa.eu/eli/reg/2021/1153/oj, 2021.

[(EU)910/2014] *Regulation (EU) No 910/2014 of the European Parliament and of the council on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC).* http://data.europa.eu/eli/reg/2014/910/oj, 2014.

[F-Col17] M. Stevens, E. Bursztein, P. Karpman, A. Albertini, and Y Markov. *The First Collision for Full SHA-1.* In *In Advances in Cryptology–CRYPTO 2017: 37th Annual International Cryptology Conference*, volume Part I 37 of *Lecture Notes in Computer Science*, pages 570–596 (Springer, 2017).

[Falcon20] Pierre-Alain Fouque et al. *FALCON: Fast-Fourier Lattice-based Compact Signatures over NTRU.* https://falcon-sign.info/falcon.pdf.

[FIPS180-4] United States of America National Institute for Standards and Technology (NIST). *Secure Hash Standard (SHS).* Federal Information Processing Standard (FIPS) Publication 180-4, 01.08.2015. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.180-4.pdf, August 2015.

[FIPS186-4] United States of America National Institute for Standards and Technology (NIST). *Digital Signature Standard (DSS).* Federal Information Processing Standard (FIPS) Publication 186-4. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.186-4.pdf, July 2013.

[FIPS202] United States of America National Institute for Standards and Technology (NIST). *SHA-3 Standard: Permutation-Based Hash and Extendable-Output Functions.* Federal Information Processing Standard (FIPS) Publication 180-1. https://nvlpubs.nist.gov/nistpubs/FIPS/NIST.FIPS.202.pdf, August 2015.

[FiSh86] Amos Fiat and Adi Shamir. *How To Prove Yourself: Practical Solutions to Identification and Signature Problems.* In *Advances in Cryptology – CRYPTO '86*, volume 263 of *Lecture Notes in Computer Science*, pages 186–194 (Springer-Verlag, 1987). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.8796.

[FMH99] G. Frey, M. Müller, and H.G. Rück. *The Tate Pairing and the Discrete Logarithm Applied to Elliptic Curve Cryptosystems. IEEE Transactions on Information Theory*, volume 45(5):1717–1719. https://ieeexplore.ieee.org/document/771254, 1999.

[GaPe20] G. Leurent and T. Peyrin. *SHA-1 is a shambles: First chosen-prefix collision on SHA-1 and application to the PGP web of trust.* In Proceedings of the 29th USENIX Conference on Security Symposium. https://eprint.iacr.org/2020/014.pdf, 2020.

[Gaus01] Carl Friedrich Gauß. *Disquisitiones Arithmeticae* (Springer–Verlag, 1801). Nachdruck, 1986, ISBN 0-387-96254-9, https://www.math.uni-bielefeld.de/~sieben/Disquisitiones.ocr.pdf.

[GIS05] Max Gebhardt, Georg Illies, and Werner Schindler. *A Note on the Practical Value of Single Hash Collisions for Special File Formats.* Beitrag zum "Chryptographic Hash Workshop" des NIST, 31. Oktober - 1. November 2005, Gaithersburg, Maryland. https://dl.gi.de/bitstream/handle/20.500.12116/24792/GI-Proceedings-77-41.pdf, 2005.

[Gold01] Oded Goldreich. *Foundations of Cryptography – Volume 1* (Cambridge University Press, 2001). ISBN 0-521-79172-3.

[Gord93] D. Gordon. *Discrete logarithms in $\mathbb{F}_p$ using the number field sieve. SIAM J. Discrete Math*, volume 6:124–138, 1993.

[Gord98] Daniel M. Gordon. *A Survey of Fast Exponentiation Methods. J. Algorithms*, volume 27(1):129–146. http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.24.8878, 1998.

[HAK18] D. Hühnlein; J. Ahmad; U. Korte. *Vertrauenswürdige Digitalisierung auf Basis von TR-RESISCAN und TR-ESOR.* DACH-Security, Gelsenkirchen, 2018.

[HAK19] D. Hühnlein; J. Ahmad; U. Korte. *Sichere Digitalisierung mit TR-RESISCAN und TR-ESOR - Auf dem Weg zur digitalen Bundesverwaltung.* DUD. https://link.springer.com/article/10.1007/s11623-019-1093-7, april 2019.

[HHS19] Hühnlein Detlef ; Hühnlein Tina ; Schuberth Sebastian ; Wich Tobias ; Lottes René ; Otto Florian ; Crossley Neil. *How to harmonise local and remote signing.* Open Identity Summit. Gesellschaft für Informatik , page 25-36. https://dl.gi.de/handle/20.500.12116/20991, 2019.

[HMP94] Patrick Horster, Markus Michels, and Holger Petersen. *Meta-ElGamal signature schemes.* In *2nd ACM Conference on Computers and Communications Security*, pages 96–107 (ACM Press, 1994). http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.31.9387.

[HMP95] Patrick Horster, Markus Michels, and Holger Petersen. *Das Meta-ElGamal Signaturverfahren und seine Anwendungen*. In *Verläßliche Informationssysteme, VIS '95*, pages 207–228 (Vieweg Verlag, 1995). https://link.springer.com/chapter/10.1007/978-3-322-91094-3_14.

[Hueh01] Detlef Hühnlein. *Faster Generation of NICE-Schnorr-type Signatures*. In *The Cryptographers' Track at RSA Conference 2001*, volume 2020 of *Lecture Notes in Computer Science*, pages 1–12 (Springer-Verlag, 2001).

[Hueh04a] Detlef Hühnlein. *How to Qualify Electronic Signatures and Time Stamps*. In Sokratis K. Katsikas, Stefanos Gritzalis, and Javier Lopez (editors), *Public Key Infrastructure, First European PKI Workshop: Research and Applications, EuroPKI 2004, Samos Island, Greece, June 25-26, 2004, Proceedings*, volume 3093 of *Lecture Notes in Computer Science*, pages 314–321 (Springer Verlag, 2004). http://www.ecsec.de/pub/2004_PKI.pdf.

[Hueh04b] Detlef Hühnlein. *Kryptosysteme auf Basis imaginärquadratischer Nicht-Maximalordnungen*. Dissertation. http://elib.tu-darmstadt.de/diss/000521/, 2004.

[HuMe00] Detlef Hühnlein and Johannes Merkle. *An efficient NICE-Schnorr-type cryptosystem*. In *Practice and Theory in Public Key Cryptography, PKC 2000*, volume 1751 of *Lecture Notes in Computer Science*, pages 14–27 (Springer-Verlag, 2000). https://link.springer.com/chapter/10.1007/978-3-540-46588-1_2.

[IEEE-P1363] *IEEE-P1363: (inactive-received) Standard Specifications for Public Key Cryptography*. https://standards.ieee.org/ieee/1363/2049/, August 2000.

[ISO10118-3] ISO/IEC. *ISO/IEC 10118-3: Information Technology – Security Techniques – Hash functions – Part 3: Dedicated hash functions*. International Standard, 2018.

[ISO10646-1] ISO/IEC. *ISO/IEC 10646-1: Information technology – Universal Multiple-Octet Coded Character Set (UCS) – Part 1: Architecture and Basic Multilingual Plane*. International Standard, 2000.

[ISO14533-1] *Processes, data elements and documents in commerce, industry and administration — Long term signature profiles — Part 1: Long term signature profiles for CMS Advanced Electronic Signatures (CAdES)*. International Standard, 2014.

[ISO14533-2] *Processes, data elements and documents in commerce, industry and administration — Long term signature — Part 2: Profiles for XML Advanced Electronic Signatures (XAdES)*. International Standard, 2021.

[ISO14533-3] *Information technology – Long term signature profiles for EDI Data and Electronic Documents – Part 2: Long term signature profiles for XML Advanced Electronic Signatures (XAdES)*. ISO/TC 154 Standard, 2017.

[ISO14533-4] *Information technology – Long term signature profiles for EDI Data and Electronic Documents – Part 2: Long term signature profiles for XML Advanced Electronic Signatures (XAdES)*. ISO/TC 154 Standard, 2019.

[ISO14888-3] *Information technology - Security techniques - Digital signatures with appendix - Part 3: Discrete logarithm based mechanisms.* ISO/IEC JTC 1/SC 27, 2018.

[ISO15408-1] *Information technology — Security techniques — Evaluation criteria for IT security — Part 1: Introduction and general model.* ISO/IEC JTC 1/SC 27, 2009.

[ISO15408-2] *Information security, cybersecurity and privacy protection — Evaluation criteria for IT security — Part 2: Security functional components.* ISO/IEC JTC 1/SC 27, 2008.

[ISO15408-3] *Information technology — Security techniques — Evaluation criteria for IT security — Part 3: Security assurance components.* ISO/IEC JTC 1/SC 27, 2008.

[ISO18014-1] *ISO/IEC 18014-1: Information technology – Security techniques – Time-stamping services – Part 1: Framework.* International Standard, 2008.

[ISO18045] *ISO/IEC 18045: Information technology – Security techniques – Methodology for IT security evaluation.* International Standard, 2008.

[ISO24727-3] ISO/IEC. *Identification cards – Integrated circuit cards programming interfaces – Part 3: Application programming interface, ISO/IEC 24727-3.* International Standard, 2008.

[ISO32000-1] *ISO 32000: Document management – Portable document format – Part 1: PDF 1.7.* International Standard. http://www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/PDF32000_2008.pdf, 2008.

[ISO32000-2:2020] *Document management — Portable document format — Part 2: PDF 2.0.* International Standard, 2020.

[ISO9735-1] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 1: Syntax rules common to all parts.* ISO 9735-1 (Second edition 2002-07-01), july 2002.

[ISO9735-10] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) – Part 10: Syntax service directories*, july 2014.

[ISO9735-2] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 2: Syntax rules specific to batch EDI.* ISO 9735-2 (Second edition 2002-07-01), july 2002.

[ISO9735-3] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 3: Syntax rules specific to interactive EDI.* ISO 9735-3 (Second edition 2002-07-01), july 2002.

[ISO9735-4] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 4: Syntax and service report message for batch EDI (message type – CONTRL).* ISO 9735-4 (Second edition 2002-07-01), last confirmed 2019, july 2002.

[ISO9735-5] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 5: Security rules for batch EDI (authenticity, integrity and non-repudiation of origin).* ISO 9735-5 (Second edition 2002-07-01), july 2002.

[ISO9735-6] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 6: Secure authentication and acknowledgement message (message type - AUTACK)*, july 2002.

[ISO9735-7] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 7: Security rules for batch EDI (confidentiality)*, july 2002.

[ISO9735-8] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 8: Associated data in EDI*, july 2002.

[ISO9735-9] Joint ISO/TC 154 UN/CEFACT Syntax Working Group (JSWG). *Electronic data interchange for administration, commerce and transport (EDIFACT) – Application level syntax rules (Syntax version number: 4, Syntax release number: 1) –Part 9: Security key and certificate management message (message type - KEYMAN)*, july 2002.

[ISO9796-2] *ISO-IEC 9796-2: Information Technology - Security Techniques -Digital Signature Schemes Giving Message Recovery – Part 2: Integer Factorization Based Mechanisms.* International Standard, december 2010.

[BSI-ITSEC] *Kriterien für die Bewertung der Sicherheit von Systemen der Informationstechnik (updated by [CC]).* https://www.bsi.bund.de/SharedDocs/Downloads/DE/BSI/Zertifizierung/ITSicherheitskriterien/itsec-en_pdf.pdf?__blob=publicationFile&v=1, August 1991.

[JoLe03] Antoine Joux and Reynald Lercier. *Improvements to the general number field sieve for discrete logarithms in prime fields. A comparison with the gaussian integer method. Mathematics of Computation*, volume 72:953–967, 2003.

[JoMe99] D. Johnson and A. Menezes. *The Elliptic Curve Digital Signature Algorithm (ECDSA).* SV. https://link.springer.com/article/10.1007/s102070100002, August 2001.

[Joux13] Antoine Joux. *A new index calculus algorithm mit complexity L(1/4+o(1)) in very scall charactristic*. http://eprint.iacr.org/2010/006.pdf, 2013.

[KAF10] Thorsten Kleinjung, Kazumaro Aoki, Jens Franke, Arjen K. Lenstra, Emmanuel Thome, Joppe W. Bos, Pierrick Gaudry, Alexander Kruppa, Peter L. Montgomery, Dag Arne Osvik, Herman te Riele, Andrey Timofeev, and Paul Zimmermann. *Factorization of a 768-bit RSA modulus*. Version of February 18, 2010. http://eprint.iacr.org/2010/006.pdf, 2010.

[KCDSA99] KCDSA Task Force Team. *The Korean Certificate-based Digital Signature Algorithm*. https://www.sciencedirect.com/science/article/abs/pii/S0045790699000117, August 1999.

[KHS13] U. Korte; D. Hühnlein; S. Schwalm. *ertrauenswürdige und beweiswerterhaltende elektronische Langzeitspeicherung auf Basis von DIN 31647 und BSI-TR-03125*. Informatik, GI-LNI, P220,ISBN 978-3-88579-614-5, S. 550-566. https://dl.gi.de/handle/20.500.12116/20778, 2014.

[KHS14a] U. Korte; D. Hühnlein; S. Schwalm. *Standards for the preservation of evidence and trust*. Proceedings Archiving, Springfield 2014 S. 9-14. https://library.imaging.org/admin/apis/public/api/ist/website/downloadArticle/archiving/11/1/art00003.

[KHS14b] U. Korte; D. Hühnlein; S. Schwalm. *Standards und Lösungen zur langfristigen Beweiswerterhaltung*. Proceedings DACH-Security S. 46-58. https://www.syssec.at/de/veranstaltungen/archiv/dachsecurity2014/papers/DACH_Security_2014_Paper_12A2, 2014.

[KHS15] U. Korte; D. Hühnlein; S. Schwalm. *Ersetzendes Scannen und Beweiswerterhaltung mitSAP*. Proceedings DACH-Security, S. 72-85. https://www.syssec.at/de/veranstaltungen/archiv/dachsecurity2015/papers/DACH_Security_2015_Paper_13A1.pdf, Frechen 2015.

[KHS16] U. Korte; D. Hühnlein; S. Schwalm; T. Kusber. *Beweiswerterhaltung im Kontext eIDAS - eine Case Study*. DACH-Security, S. 379-392. https://www.researchgate.net/publication/313116574_Beweiswerterhaltung_im_Kontext_eIDAS_-_eine_Case_Study, Frechen 2015.

[KHSKPW17] U. Korte; D. Hühnlein; S. Schwalm; T. Kusber; M. Prechtl; B. Wild. *Datenpakete zur Informationsund Beweiswerterhaltung. Ein Vergleich*. DACH-Security S. 291-303. https://www.syssec.at/en/veranstaltungen/dachsecurity2017/papers/DACH_Security_2017_Paper_22A2.pdf, Frechen 2017.

[KKS18] U. Korte; T. Kusber; S. Schwalm. *Vertrauenswürdiges E-Government – Anforderungen und Lösungen zur beweiswerterhaltenden Langzeitspeicherung*. 23. Archivwissenschaftliches Kolloqium. https://www.researchgate.net/publication/330856704_Vertrauenswurdiges_E-Government-Anforderungen_und_Losungen_zur_beweiswerterhaltenden_Langzeitspeicherung, Marburg 2018.

[KKS21] U. Korte; T. Kusber; S. Schwalm. *Anforderungen und Lösungen zur beweiswerterhaltenden Langzeitspeicherung*. 23. Archivwissenschaftliches Kolloquium der Archivschule Marburg, 2021.

[Klim05] Vlastimil Klima. *Finding MD5 Collisions - a Toy For a Notebook*. Cryptology ePrint Archive: Report 2005/075. http://eprint.iacr.org/2005/075, März 2005.

[Kobl87] Neal Koblitz. *Elliptic Curve Cryptosystems*. *Mathematics of Computation*, volume 48(177):203–209, 1987.

[Kobl89] Neal Koblitz. *Hyperelliptic cryptosystems*. *Journal of Cryptology*, volume 1(3):139–150. https://link.springer.com/article/10.1007/BF02252872, 1989.

[Kobl94] Neil Koblitz. *A Course in Number Theory and Cryptography*, volume 114 of *Graduate Texts in Mathematics* (Springer–Verlag, 1994), 2 edition.

[KS16] S. Schwalm T. Kusber. *Elektronische Langzeitspeicherung als SOA-Dienst – Kernelement eines vertrauenswürdigen Informationsmanagements*. INFORMATIK S. 869-882. https://dl.gi.de/handle/20.500.12116/1195, 2016.

[KSKE22] Tomasz Kusber; Steffen Schwalm; Ulrike Korte; Mario Engel. *Langfristige Beweissicherheit und Vertrauenswürdigkeit digitaler Unterlagen (qualifizierte) Bewahrungsdienste nach eIDAS, ETSI und TR-ESOR*. Datenschutz und Datensicherheit, 1/2022.

[KSKK20] Kalinda Shamburger Tomasz Kusber, Steffen Schwalm and Ulrike Korte. *Criteria for trustworthy digital transactions – Blockchain/ DLT between eIDAS, GDPR, Data and Evidence Preservation*. Open Identity Summit, Lecture Notes in Informatics (LNI), Gesellschaft für Informatik, Bonn 2020.

[KSKS21] Tomasz Kusber; Steffen Schwalm; Dr. Ulrike Korte; Kalinda Shamburger. *Records Management and Long-Term Preservation of Evidence in DLT*. Lecture Notes in Informatics (LNI), GI, Open Identity Summit 2021.

[kyber] *Cryptographic Suite for Algebraic Lattices*. https://pq-crystals.org/kyber/.

[Lens87] Hendrik W. Lenstra. *Factoring integers with elliptic curves*. *Annals of Mathematics*, volume 126:649–673, 1987.

[LeWe05] Arjen K. Lenstra and Benne de Weger. *On the possibility of constructing meaningful hash collisions for public keys*. In Colin Boyd and Juan Manuel González Nieto (editors), *Information Security and Privacy, 10th Australasian Conference, ACISP 2005, Brisbane, Australia, July 4-6, 2005, Proceedings*, volume 3574 of *Lecture Notes in Computer Science*, pages 267–279 (Springer, 2005).

[LiNi86] Rudolf Lidl and Harald Niederreiter. *Introduction to finite fields and their applications* (Cambridge University Press, 1986).

[LiLe98] Chae Hoon Lim and Pil Joong Lee. *A Study on the Proposed Korean Digital Signature Algorithm*. In Kazuo Ohta and Pei Dingyi (editors), *Advances in Cryptology – ASIACRYPT '98*, volume 1514 of *Lecture Notes in Computer Science*, pages 175–186 (Springer-Verlag, 1998).

[LLMP93] Arjen K. Lenstra, Hendrik W. Lenstra, Mark S. Manasse, and John M. Pollard. *The number field sieve*. In A.K. Lenstra and H.W. Lenstra (editors), *The Developement of the Number Field Sieve*, volume 1554 of *Lecture Notes in Mathematics*, pages 11–42 (Springer-Verlag, 1993).

[M460] European Commission. *Standardisation Mandate to the European Standardisation Organisations CEN, CENELEC and ETSI in the Field of Information and Communication Technologies applied to Electronic Signatures*. https://www.etsi.org/images/files/ecmandates/m460.pdf, December 2009.

[May05] Alexander May. *Computing the RSA Secret Key Is Deterministic Polynomial Time Equivalent to Factoring*. In Matthew K. Franklin (editor), *Advances in Cryptology – CRYPTO 2004*, volume 3152 of *Lecture Notes in Computer Science*, pages 213–219 (Springer-Verlag, 2004). https://link.springer.com/chapter/10.1007/978-3-540-28628-8_13.

[Mene93] Alfred J. Menezes. *Elliptic Curve Public Key Cryptosystems* (Kluwer Academic Publishers, 1993).

[Merk79] Ralph Charles Merkle. *Secrecy, Authentication, and Public Key Systems*. Stanford University Information Systems Laboratory Technical Report 1979-1. Https://www.merkle.com/papers/Thesis1979.pdf, June 1979.

[Merk80] Ralph C. Merkle. *Protocols for public key cryptosystems*. In *Symposium on Security and Privacy, Oakland, CA, USA*, pages 122–134 (1980). https://www.researchgate.net/publication/220713913_Protocols_for_Public_Key_Cryptosystems.

[Merk89] Ralph C. Merkle. *One way hash functions and DES*. In *Advances in Cryptology – CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 428–446 (Springer-Verlag, 1990). https://link.springer.com/chapter/10.1007/0-387-34805-0_40.

[Mill85] Victor S. Miller. *Use of Elliptic Curves in Cryptography*. In *Advances in Cryptology – CRYPTO '85*, volume 218 of *Lecture Notes in Computer Science*, pages 417–426 (Springer-Verlag, 1986).

[Misa98] J. Misarsky. *How (Not) to Design RSA Signature Schemes*. In *Practice and Theory in Public Key Cryptography, PKC '98*, volume 1431 of *Lecture Notes in Computer Science*, pages 14–28 (Springer-Verlag, 1998).

[MMM+19] Vladislav Mladenov, Christian Mainka, Karsten Meyer zu Selhausen, Martin Grothe, and Jorg Schwenk. *Attacks bypassing the signature validation in PDF*. Vulnerability Report, Chair for Network and Data Security, 08.11.2018. https://www.nds.ruhr-uni-bochum.de/media/ei/veroeffentlichungen/2019/02/12/report.pdf, 2018.

[MNP96] Markus Michels, David Naccache, and Holger Petersen. *GOST 34.10 – A Brief Overview of Russiaś DSA*. *Computers & Security*, volume 15(8):725–732. https://www.sciencedirect.com/science/article/pii/S0167404896000168, 1996.

[ModTerm] Modinis IDM Study Team. *Common Terminological Framework for Interoperable Electronic Identity Management*. Modinis Study on

Identity Management in eGovernment – Consultation Paper, Version 2.01. http://ec.europa.eu/information_society/activities/ict_psp/documents/eid_terminology_paper.pdf, 2005.

[MOV91] Alfred Menezes, Tatsuaki Okamoto, and Scott Vanstone. *Reducing elliptic curve logarithms to logarithms in a finite field*. In *Proceedings of the twenty-third annual ACM symposium on Theory of computing*, pages 80–89 (ACM Press, 1991). ISBN 0-89791-397-3.

[MOV97] Alfred J. Menezes, Paul C. van Oorschot, and Scott A. Vanstone. *Handbook of Applied Cryptography*. http://www.cacr.math.uwaterloo.ca/hac/, 1996.

[MTW04] Alfred Menezes, Edlyn Teske, and Annegret Weng. *Weak Fields for ECC*. In Tatsuaki Okamoto (editor), *Topics in Cryptology - CT-RSA 2004, The Cryptographers Track at the RSA Conference 2004*, volume 2964 of *Lecture Notes in Computer Science*, pages 366–386 (Springer Verlag, 2004). ISBN 3-540-20996-4.

[Nebe00] G. Nebe. *Faktorisieren großer Zahlen*. In *Jahresbericht der Deutschen Mathematiker-Vereinigung*, volume 102, pages 1–14 (B.G. Teubner, Stuttgart, Leipzig, 2000).

[NgSh02] P. Q. Nguyen and I. E. Shparlinski. *The insecurity of the Digital Signature Algorithm with partially known nonces*. *Journal of Cryptology*, volume 15(3):151–156. https://link.springer.com/content/pdf/10.1007/s00145-002-0021-3.pdf, 2002.

[NgSh03] P. Q. Nguyen and I. E. Shparlinski. *The Insecurity of the Elliptic Curve Digital Signature Algorithm with Partially Known Nonces*. *Designs, Codes and Cryptography*, volume 30(2):201–217, 2003.

[Nguy04] Phong Q. Nguyen. *Can We Trust Cryptographic Software? Cryptographic Flaws in GNU Privacy Guard v1.2.3*. In *Advances in Cryptology – EUROCRYPT 2000*, volume 3027 of *Lecture Notes in Computer Science*, pages 555–570 (Springer, 2004).

[NIST-8413] David Cooper et. al. Gorjan Alagic, Daniel Apon. *Status Report on the Third Round of the NIST Post-Quantum Cryptography Standardization Process*. https://doi.org/10.6028/NIST.IR.8413-upd1, 2022.

[NyRu94] Kaisa Nyberg and Rainer A. Rueppel. *Message recovery for signature schemes based on the discrete logarithm problem*. In *Advances in Cryptology – EUROCRYPT '94*, volume 950 of *Lecture Notes in Computer Science*, pages 182–193 (Springer-Verlag, 1995).

[OASIS-VR] D. Hühnlein. *OASIS DSS v1.0 Profile for Comprehensive Multi-Signature Verification Reports Version 1.0, Committee Specification*. http://docs.oasis-open.org/dss-x/profiles/verificationreport/oasis-dssx-1.0-profiles-vr-cs01.pdf, 2010.

[OkUc98] Tatsuaki Okamoto and Shigenori Uchiyama. *A New Public-Key Cryptosystem as secure as Factoring*. In *Advances in Cryptology – EUROCRYPT '98*, volume 1403 of *Lecture Notes in Computer Science*, pages 308–318 (Springer-Verlag, 1998). https://link.springer.com/content/pdf/10.1007/BFb0054135.pdf.

[Pail99] Pascal Paillier. *A Trapdoor Permutation Equivalent to Factoring*. In Hideki Imai and Yuliang Zheng (editors), *Public Key Cryptography, Second International Workshop on Practice and Theory in Public Key Cryptography, PKC '99, Kamakura, Japan, March 1-3, 1999, Proceedings*, volume 1560 of *Lecture Notes in Computer Science*, pages 219–222 (Springer-Verlag, 1999).

[Pala04] Otto Palandt. *Bürgerliches Gesetzbuch – Mit Einführungsgesetz (Auszug), Produkthaftungsgesetz, ErbbaurechtsVO, Wohnungseigentumsgesetz, HausratsVO*, volume 63. Auflage (Verlag C.H. Beck, 2004). ISBN 3-406-51035-3.

[PAuswVwV] *Allgemeine Verwaltungsvorschrift zur Durchführung des Personalausweisgesetzes und der Personalausweisverordnung (Personalausweisverwaltungsvorschrift*. http://www.verwaltungsvorschriften-im-internet.de/bsvwvbund_16122019_DGI220105131.htm, December 2019.

[PDF(v1.3)] Adobe Systems Incorporated. *PDF Reference – Second Edition – Adobe Portable Document Format Version 1.3*. Addison Wesley, ISBN 0-201-61588-6, Juli 2000.

[PDF(v1.4)] Adobe Systems Incorporated. *PDF Reference – Third Edition – Adobe Portable Document Format Version 1.4*. Addison-Wesley, ISBN 0-201-75839-3. https://www.adobe.com/content/dam/acom/en/devnet/pdf/pdfs/pdf_reference_archives/PDFReference.pdf, November 2001.

[PDF(v1.5)] Adobe Systems Incorporated. *PDF Reference – Fourth Edition – Adobe Portable Document Format Version 1.5*, August 2003.

[PDF(v1.6)] Adobe Systems Incorporated. *PDF Reference – Fifth Edition – Adobe Portable Document Format Version 1.6*, November 2004.

[PfHa07] Andreas Pfitzmann and Marit Hansen. *Anonymity, Unlinkability, Undetectability, Unobservability, Pseudonymity, and Identity Management – A Consolidated Proposal for Terminology*. Version v0.29. http://dud.inf.tu-dresden.de/literatur/Anon_Terminology_v0.29.pdf, 2007.

[PKCS12] RSA Laboratories. *PKCS #12: Personal Information Exchange Syntax Standard - Version 1.1 - (RFC 7292)*. Public Key Cryptography Standards – PKCS #12. Last update 2020-01-21, https://datatracker.ietf.org/doc/html/rfc7292, Juni 2014.

[PKCS1(v2.2)] RSA Laboratories. *PKCS #1: RSA Encryption Standard - Version 2.2*. Public Key Cryptography Standards – PKCS #1 v2.2. Last updated 2020-01-21, https://datatracker.ietf.org/doc/html/rfc8017, Juni 2016.

[PKCS7(v1.5)] RSA Laboratories. *PKCS #7: Cryptographic Message Syntax Standard - Version 1.5*. Public Key Cryptography Standards – PKCS #7. ftp://ftp.rsasecurity.com/pub/pkcs/doc/pkcs-7.doc, November 1993.

[PKCS11] OASIS Standard. *PKCS #11 Cryptographic Token Interface Base Specification Version 3.0*. Public Key Cryptography Standards – PKCS #11. https://docs.oasis-open.org/pkcs11/pkcs11-base/v3.0/os/pkcs11-base-v3.0-os.html, june 2020.

[Poll74] John M. Pollard. *Theorems on Factorization and Primality Testing.* In *Proceedings of Cambridge Philosophy Society*, volume 76 (Cambridge University Press, 1974).

[Poll75] John M. Pollard. *A Monte Carlo method for factorization.* *BIT*, volume 15:331–334, 1975.

[Poll78] John M. Pollard. *Monte Carlo methods for index computation (mod p).* *Mathematics of Computation*, volume 32(143):918–924, 1978.

[Pome85] Carl Pomerance. *The Quadratic Sieve Factoring Algorithm.* In Thomas Beth, Norbert Cot, and Ingemar Ingemarsson (editors), *Advances in Cryptology*, volume 209 of *Lecture Notes in Computer Science*, pages 169–182 (Springer-Verlag, 1985).

[RaEf02] Wolfgang Rankl and Wolfgang Effing. *Handbuch der Chipkarten* (Carl Hanser Verlag, München, Wien, 2002). 4. Auflage, ISBN 3-446-22036-4.

[RFC793] University of Southern California Information Sciences Institute. *Transmission Control Protocol.* Request For Comments – RFC 793. https://datatracker.ietf.org/doc/html/rfc793, September 1981.

[RFC1321] Ron Rivest. *The MD5 Message-Digest Algorithm.* Request For Comments – RFC 1321. http://www.ietf.org/rfc/rfc1321.txt, April 1992.

[RFC1951] P. Deutsch. *DEFLATE Compressed Data Format Specification version 1.3.* Request For Comments – RFC 1951. https://datatracker.ietf.org/doc/html/rfc1951, Mai 1996.

[RFC2045] N. Freed and N. Borenstein. *Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies.* Request For Comments – RFC 2045. https://datatracker.ietf.org/doc/html/rfc2045, November 1996.

[RFC2315] B. Kaliski. *PKCS #7: Cryptographic Message Syntax - Version 1.5.* Request For Comments – RFC 2315. http://www.ietf.org/rfc/rfc2315.txt, 1998.

[RFC2634] P. Hoffman. *Enhanced Security Services for S/MIME.* Request For Comments – RFC 2630. Upated by [RFC5035], https://datatracker.ietf.org/doc/html/rfc2634, Juni 1999.

[RFC3125] J. Ross, D. Pinkas, and N. Pope. *Electronic Signature Policies.* Request For Comments – RFC 3125. http://www.ietf.org/rfc/rfc3125.txt, September 2001.

[RFC3161] C. Adams, P. Cain, D. Pinkas, and R. Zuccherato. *Internet X.509 Public Key Infrastructure – Time-Stamp Protocol (TSP).* Request For Comments – RFC 3161. http://www.ietf.org/rfc/rfc3161.txt, August 2001.

[RFC3275] D. Eastlake, J. Reagle, and D. Solo. *(Extensible Markup Language) XML-Signature Syntax and Processing.* Request For Comments – RFC 3275. http://www.ietf.org/rfc/rfc3275.txt, March 2002.

[RFC3647] S. Chokhani, W. Ford, R. Sabett, C. Merrill, and S. Wu. *Internet X.509 Public Key Infrastructure – Certificate Policy and Certification Practices Framework*. Request For Comments – RFC 3647. http://www.ietf.org/rfc/rfc3647.txt, November 2003.

[RFC3986] T. Berners-Lee, R. Fielding, and L. Masinter. *Uniform Resource Identifier (URI): Generic Syntax*. Request For Comments – RFC 3986. https://datatracker.ietf.org/doc/html/rfc3986, Januar 2005.

[RFC4510] K. Zeilenga (Ed.). *Lightweight Directory Access Protocol (LDAP): Technical Specification Road Map*. Request For Comments – RFC 4510. https://datatracker.ietf.org/doc/html/rfc4510, June 2006.

[RFC4880] J. Callas, L. Donnerhacke, H. Finney, D. Shaw, and R. Thayer. *OpenPGP Message Format*. Request For Comments – RFC 4880. https://datatracker.ietf.org/doc/html/rfc4880, November 2007.

[RFC4998] T. Gondrom, R. Brandner, and U. Pordesch. *Evidence Record Syntax (ERS)*. Request For Comments – RFC 4998. https://datatracker.ietf.org/doc/html/rfc4998, August 2007.

[RFC5035] J. Schaad. *Enhanced Security Services (ESS) Update: Adding CertID Algorithm Agility*. Request For Comments – RFC 5035. https://datatracker.ietf.org/doc/html/rfc5035, August 2007.

[RFC5280] D. Cooper, S. Santesson, S. Farrell, S. Boeyen, R. Housley, and W. Polk. *Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile*. Request For Comments – RFC 5280. https://datatracker.ietf.org/doc/html/rfc5280, Mai 2008.

[RFC5272] M. Myers J. Schaad. *Certificate Management over CMS (CMC)*. Request For Comments – RFC 5272. https://tools.ietf.org/html/rfc5272, Juni 2008.

[RFC5652] R. Housley. *Cryptographic Message Syntax (CMS)*. Request For Comments – RFC 5652. https://datatracker.ietf.org/doc/html/rfc5652, September 2009.

[RFC5698] T. Kunz, S. Okunick, and U. Pordesch. *Data Structure for the Security Suitability of Cryptographic Algorithms (DSSC)*. Request For Comments – RFC 5698. http://www.ietf.org/rfc/rfc5698.txt, November 2009.

[RFC5755] S. Turner S. Farrell, R. Housley. *An Internet Attribute Certificate Profile for Authorization*. Request For Comments – RFC 5755. http://www.ietf.org/rfc/rfc5755.txt, January 2010.

[RFC5816] S. Santesson and N. Pope. *ESSCertIDv2 Update for RFC 3161*. Request For Comments – RFC 5816. https://datatracker.ietf.org/doc/html/rfc5816, August 2010.

[RFC6283] A. Jerman Blazic, S. Saljic, and T. Gondrom. *Extensible Markup Language Evidence Record Syntax (XMLERS)*. Request For Comments – RFC 6283. http://www.ietf.org/rfc/rfc6283.txt, July 2011.

[RFC6960] S. Santesson, M. Myers, R. Ankney, A. Malpani, S. Galperin, and C. Adams. *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSPl*. Request For Comments – RFC 6960. https://datatracker.ietf.org/doc/html/rfc6960, June 2013.

[RFC7159] T. Bray. *The JavaScript Object Notation (JSON) Data Interchange Format*. Request For Comments – RFC 7159. https://datatracker.ietf.org/doc/html/rfc7159, March 2014.

[RFC7230] Roy T. Fielding and Julian Reschke. *Hypertext Transfer Protocol (HTTP/1.1): Message Syntax and Routing*. RFC 7230, june 2014.

[RFC7515] M. Jones, J. Bradley, and N. Sakimura. *JSON Web Signature (JWS)*. Request For Comments – RFC 7515. https://www.ietf.org/rfc/rfc7515.txt, May 2015.

[RFC8017] Kathleen Moriarty, Burt Kaliski, Jakob Jonsson, and Andreas Rusch. *PKCS #1: RSA Cryptography Specifications Version 2.2*. RFC 8017, november 2016.

[RFC8259] T. Bray. *The JavaScript Object Notation (JSON) Data Interchange Format*. Request For Comments – RFC 8259. https://www.ietf.org/rfc/rfc8259.txt, December 2017.

[RFC8391] Andreas Hülsing et al. *XMSS: Extended hash-based signatures*. Internet Research Task Force (IRTF). Https://tools.ietf.org/html/rfc8391https://tools.ietf.org/html/rfc8391, May 2018.

[RFC8446] Eric Rescorla. *The Transport Layer Security (TLS) Protocol Version 1.3*. RFC 8446, august 2018.

[RFC8550] Jim Schaad, Blake C. Ramsdell, and Sean Turner. *Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Certificate Handling*. RFC 8550, april 2019.

[RFC8551] Jim Schaad, Blake C. Ramsdell, and Sean Turner. *Secure/Multipurpose Internet Mail Extensions (S/MIME) Version 4.0 Message Specification*. RFC 8551, april 2019.

[RFC8554] D. McGrew et al. *Leighton-Micali Hash-Based Signatures*. Internet Research Task Force (IRTF). Https://tools.ietf.org/html/rfc8554, April 2019.

[Ries94] Hans Riesel. *Prime Numbers and Computer Methods for Factorization*, volume 126 of *Progress in Mathematics* (Birkhäuser, 1994), 2 edition.

[RO07] A. Rossnagel; RAin Dr. Stefanie Fischer-Dieskau; Silke Jandt; Michael Knopp. *Langfristige Aufbewahrung elektronischer Dokumente, Anforderungen und Trends*. Baden-Baden. https://dl.gi.de/handle/20.500.12116/1195, 2007.

[Romp90] J. Rompel. *One-way functions are necessary and sufficient for secure signatures*. In *Proceedings of the twenty-second annual ACM symposium on Theory of computing*, pages 387 – 394 (Association for Computing Machinery (ACM), 1990). ISBN:0-89791-361-2.

[RoSc05] Alexander Rossnagel and Paul Schmücker (editors). *Beweiskräftige und sichere Langzeitarchivierung elektronisch signierter Dokumente – Ergebnisse des Forschungsvorhabens ArchiSig* (Verlagsgruppe Hüthig, Jehle, Rehm, 2005).

[RSA200] Jens Franke, Friedrich Bahr, M. Böhm, Thorsten Kleinjung, Peter L. Montgomery, and Herman te Riele. *Announcement: Factorization of RSA200*. http://www.loria.fr/~zimmerma/records/rsa200, 2005.

[RSA640] Heise. *RSA-640 geknackt*. Meldung vom 09.11.2005. https://www.heise.de/newsticker/meldung/RSA-640-geknackt-146417.html, 2005.

[RSA78] Ronald L. Rivest, Adi Shamir, and Leonard Adleman. *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*. *Communications of the ACM*, volume 21(2):120–126, 1978.

[SaAr98] T. Satoh and K. Araki. *Fermat quotients and the polynomialtime discrete log algorithm for anomalous elliptic curves*. *Comm. Math. Univ. Sancti. Pauli*, volume 47:81–92, 1998.

[SAML-Glos(v2.0)] Jeff Hodges, Rob Philpott, and Eve Maler. *Glossary for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS Standard, 15.03.2005. http://docs.oasis-open.org/security/saml/v2.0/saml-glossary-2.0-os.pdf, 2005.

[SAML(v1.0)] Phillip Hallam-Baker and Eve Maler. *Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML)*. OASIS Standard, 05.11.2002. http://www.oasis-open.org/committees/download.php/2290/oasis-sstc-saml-1.0.zip, 2002.

[SAML(v1.1)] Eve Maler, Prateek Mishra, and Rob Philpott. *Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V1.1*. OASIS Standard, 02.09.2003. http://www.oasis-open.org/committees/download.php/3406/oasis-sstc-saml-core-1.1.pdf, 2003.

[SAML(v2.0)] Scott Cantor, John Kemp, Rob Philpott, and Eve Maler. *Assertions and Protocol for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS Standard, 15.03.2005. http://docs.oasis-open.org/security/saml/v2.0/saml-core-2.0-os.pdf, 2005.

[SBK+17] Marc Stevens, Elie Bursztein, Pierre Karpman, Ange Albertini, and Yarik Markov. *The first collision for full SHA-1*. In *Advances in Cryptology – CRYPTO 2017: 37th Annual International Cryptology Conference*, pages 570–596 (Springer, 2017). https://shattered.io/static/shattered.pdf.

[Sch17] S. Schwalm. *sustainable electronic business*. Open Identity Summit, Lecture Notes in Informatics (LNI), Proceedings S. 131-144. https://dl.gi.de/handle/20.500.12116/3571, 2017.

[SchHue19] Claudia Göbel, Dr. Siegfried Kaiser, Steffen Schwalm, Dr. Detlef Hühnlein, Enrico Entschew, Jürgen Prummer, Markus Schuster, Nils Britze, Rebekka Weiß, Tatami M. Michalek, and Thorsten Brand. *eIDAS und der*

ECM-Markt. Bitkom - Bundesverband Informationswirtschaft, Telekommunikation und neue Medien e.V. https://www.bitkom.org/sites/default/files/2019-06/190618_lf_ecm_eidas_web.pdf, 2019.

[Schn15] Bruce Schneier. *NSA Plans for a Post-Quantum World.* Schneier on Security. Https://www.schneier.com, August 2015.

[Schn89] Claus P. Schnorr. *Efficient identification and signatures for smart cards*. In *Advances in Cryptology – CRYPTO '89*, volume 435 of *Lecture Notes in Computer Science*, pages 239–252 (Springer-Verlag, 1990).

[Schn91] Claus P. Schnorr. *Efficient Signature Generation by Smart Cards*. *Journal of Cryptology*, volume 4(3):161–174. http://publikationen.ub.uni-frankfurt.de/volltexte/2005/1203/pdf/schnorr.pdf, 1991.

[ScLe84] Claus Peter Schnorr and Hendrik W. Lenstra, Jr. *A Monte Carlo Factoring Algorithm With Linear Storage*. *Mathematics of Computation*, volume 43(167):289–311. https://www.ams.org/journals/mcom/1984-43-167/S0025-5718-1984-0744939-5/, 1984.

[Shan72] Daniel Shanks. *The infrastructure of a real quadratic field and its applications*. In *Proceedings of Number Theory Conference, Boulder 1972*, pages 217–224 (1972).

[Shor97] Peter W. Shor. *Polynomial-Time Algorithms for Prime Factorization and Discrete Logarithms on a Quantum Computer*. *SIAM J. Sci. Statist. Comput.*, volume 26:1484–1509. https://doi.org/10.1137/S0036144598347011, 1999.

[SigG] *Gesetz über Rahmenbedingungen für elektronische Signaturen und zur Änderung weiterer Vorschriften, vom 16.05.2001*. BGBl. 2001 Teil I Nr. 22, S. 876 ff, Geändert durch Art. 1 G v. 4. 1.2005 I 2. http://bundesrecht.juris.de/bundesrecht/sigg_2001/, 2001.

[Silv87] Robert D. Silverman. *The multiple polynomial quadratic sieve*. *Mathematics of Computation*, volume 48:329–339, 1987.

[SiSu98] Joseph H. Silverman and Joe Suzuki. *Elliptic Curve Discrete Logarithms and the Index Calculus*. In *Advances in Cryptology – Proceedings of Asiacrypt'98*, volume 1514 of *Lecture Notes in Computer Science*, pages 110–125 (Springer-Verlag, 1998).

[SKH14] Detlef Hühnlein Steffen Schwalm, Ulrike Korte. *Standards for the Preservation of Evidence and Trust for Electronic Records*. Archiving 2014. ISBN: 978-0-89208-309-1, 2014.

[Smar99] Nigel P. Smart. *The Discrete Logarithm Problem on Elliptic Curves of Trace One*. *Journal of Cryptology*, volume 12(3):193–196. https://link.springer.com/content/pdf/10.1007/s001459900052.pdf, 1999.

[SOGISV1.2] Senior Officials Group Information Systems Security SOGIS Crypto Working Group. *SOGIS Agreed Cryptographic Mechanisms Certificate Management over CMS (CMC) Updates, V1.2*. Publication. https://www.sogis.eu/documents/cc/crypto/SOGIS-Agreed-Cryptographic-Mechanisms-1.2.pdf, January 2020.

[SPHINCS18] Andreas Hülsing et al. *SPHINCS+*. https://sphincs.org.

[SPML(v1.0)] D. Rolls. *OASIS Service Provisioning Markup Language (SPML) Version 1.0*. OASIS Standard. http://www.oasis-open.org/committees/download.php/4137/os-pstc-spml-core-1.0.pdf, Oktober 2003.

[SPML(v2)] G. Cole. *OASIS Service Provisioning Markup Language (SPML) Version 2*. OASIS Standard. http://www.oasis-open.org/committees/download.php/17708/pstc-spml-2.0-os.zip, April 2006.

[StGB] *Strafgesetzbuch*. RGBl 1871, 127, neugefasst durch Bek. v. 13.11.1998 I 3322, zuletzt geändert Artikel 2 des Gesetzes vom 22. November 2021. http://bundesrecht.juris.de/bundesrecht/stgb/, 1871.

[Taka98] Tsuyoshi Takagi. *Fast RSA-Type Cryptosystem Modulo $p^k q$*. In Hugo Krawczyk (editor), *Advances in Cryptology – CRYPTO '98*, volume 1462 of *Lecture Notes in Computer Science*, pages 318–326 (Springer-Verlag, 1998). https://link.springer.com/chapter/10.1007/BFb0055738.

[Tesk98] Edlyn Teske. *New Algorithms for Finite Abelian Groups*. ISBN 3-8265-4045-X, 1998.

[TFEU] *Consolidated versions of the Treaty on European Union and the Treaty on the Functioning of the European Union - Consolidated version of the Treaty on the Functioning of the European Union - Protocols - Annexes - Declarations annexed to the Final Act of the Intergovernmental Conference which adopted the Treaty of Lisbon, signed on 13 December 2007 - Tables of equivalences*. http://eur-lex.europa.eu/legal-content/EN/TXT/HTML/?uri=CELEX:12012E/TXT, October 2012.

[TUDarmstadt] TU Darmstadt. *OIDs des Fachgebietes CDC der Informatik*. https://www.hrz.tu-darmstadt.de/services/it_services/oids/oids_der_informatik_cdc/index.en.jsp.

[Vaud96] Serge Vaudenay. *Hidden Collisions on DSS*. In *Advances in Cryptology – CRYPTO '96*, volume 1109 of *Lecture Notes in Computer Science*, pages 83–88 (Springer-Verlag, 1996). https://link.springer.com/chapter/10.1007/3-540-68697-5_7.

[VDG17] *Vertrauensdienstegesetz VDG, Artikel 1 des Gesetzes zur Durchfuehrung der Verordnung (EU) Nr. 910/2014 des Europaeischen Parlaments und des Rates vom 23. Juli 2014 ueber elektronische Identifizierung und Vertrauensdienste fuer elektronische Transaktionen im Binnenmarkt und zur Aufhebung der Richtlinie 1999/93/EG (eIDAS-Durchfuehrungsgesetz), Bundesgesetzblatt Jahrgang 2017 Teil I Nr. 52, published in Bonn*. https://www.gesetze-im-internet.de/vdg/BJNR274510017.html, July 2017.

[Wagn94] Klaus Wagner. *Einführung in die Theoretische Informatik* (Springer–Verlag, 1994). ISBN 3-540-58139-1.

[WaYu05] Xiaoyun Wang and Hongbo Yu. *How to Break MD5 and Other Hash Functions*. In *Advances in Cryptology – EUROCRYPT 2005*, volume 3494 of *Lecture Notes in Computer Science*, pages 19–35 (Springer, 2005). https://link.springer.com/chapter/10.1007/11426639_2.

[Webe96] Damian Weber. *Computing discrete logarithms with the number field sieve*. In Henri Cohen (editor), *Algorithmic Number Theory, ANTS-II*, volume 1122 of *Lecture Notes in Computer Science* (Springer-Verlag, 1996). https://link.springer.com/chapter/10.1007/3-540-61581-4_70.

[WeKa14] Friedl Weiss and Clemens Kaupa. *European Union Internal Market Law* (Cambridge University Press, 2014).

[Will82] Hugh C. Williams. *A $p + 1$ Method of Factoring*. *Mathematics of Computation*, volume 39:225–234. https://www.ams.org/journals/mcom/1982-39-159/S0025-5718-1982-0658227-7/, 1982.

[WS-Security(v1.1)] Anthony Nadalin, Chris Kaler, Ronald Monzillo, and Phillip Hallam-Baker. *Web Services Security: SOAP Message Security 1.1*. OASIS Standard, 01.02.2006. http://www.oasis-open.org/committees/download.php/16790/wss-v1.1-spec-os-SOAPMessageSecurity.pdf, 2006.

[WS-Trust(v1.3)] Anthony Nadalin, Marc Goodner, Martin Gudgin, Abbie Barbir, and Hans Granqvist. *WS-Trust 1.3*. OASIS Standard, 19.03.2007. http://docs.oasis-open.org/ws-sx/ws-trust/200512/ws-trust-1.3-os.pdf, 2007.

[WSDL(v2.0)] R. Chinnici, J.-J. Moreau, A. Ryman, and S. Weerawarana. *Web Services Description Language (WSDL) – Version 2.0 Part 1: Core Language*. W3C Recommendation. http://www.w3.org/TR/wsdl20/, June 2007.

[WVKS18] M. Weber; T Vogt; W. Krogel; S. Schwalm. *DIN NID 15 WG 2: Records Management nach ISO 15489. Einführung und Anleitung*. https://www.beuth.de/de/publikation/records-management-nach-iso-15489/270032872, Berlin 2018.

[WYY05a] Xiaoyun Wang, Yiqun Lisa Yin, and Hongbo Yu. *Finding Collisions in the Full SHA-1*. In *Advances in Cryptology – CRYPTO 2005*, Lecture Notes in Computer Science (Springer, 2005). https://link.springer.com/chapter/10.1007/11535218_2.

[X.408] ITU-T. *ITU-T Recommendation X.408*. Message Handling Systems: Encoded Information Type Conversion Rules. https://www.itu.int/rec/T-REC-X.408-198811-I/en, 1988.

[X.500] ITU-T. *ITU-T Recommendation X.500 (2019) - ISO-IEC 9594-1:2019*. Information technology - Open Systems Interconnection - The Directory: Overview of Concepts, Models, and Services. https://www.itu.int/itu-t/recommendations/rec.aspx?rec=X.500, 2019.

[X.501] ITU-T. *ITU-T Recommendation X.501 (2008) - ISO-IEC 9594-1:2008*. Information technology - Open Systems Interconnection - The Directory: Models. http://www.itu.int/rec/T-REC-X.501-200811-I, 2008.

[X.509] ITU-T. *ITU-T Recommendation X.509*. Information Technology - Open systems Interconnection - The Directory - Public-key and attribute certificate frameworks : Corrigendum 1. https://www.itu.int/rec/T-REC-X.509-202110-I!Cor1, October 2021.

[X.519] ITU-T. *ITU-T Recommendation X.519 (2019)*. Information technology - Open Systems Interconnection - The Directory: Protocol Specifications. https://www.itu.int/rec/T-REC-X.519-201910-I, October 2019.

[X.660] ITU-T. *ITU-T Recommendation X.660*. Information technology - Procedures for the operation of object identifier registration authorities: General procedures and top arcs of the international object identifier tree. https://www.itu.int/rec/T-REC-X.660-201107-I, July 2011.

[X.680] ITU-T. *ITU-T Recommendation X.680 (2021) | ISO/IEC 8824-1:2021*. Information Technology - Abstract Syntax Notation One (ASN.1): Specification of Basic Notation. https://www.itu.int/rec/T-REC-X.680-202102-I, 2021.

[X.690] ITU-T. *ITU-T Recommendation X.690 (2021) | ISO/IEC 8824-1:2021*. Information Technology - ASN.1 Encoding Rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER). https://www.itu.int/rec/T-REC-X.690-202102-I, 2021.

[X.691] ITU-T. *ITU-T Recommendation X.691 (2021) | ISO/IEC 8824-1:2021*. Information technology - ASN.1 encoding rules: Specification of Packed Encoding Rules (PER). https://www.itu.int/rec/T-REC-X.691-202102-I, 2021.

[X.693] ITU-T. *ITU-T Recommendation X.693 (2021) | ISO/IEC 8824-1:2021*. Information Technology - ASN.1 Encoding Rules: XML Encoding Rules (XER). https://www.itu.int/rec/T-REC-X.693-202102-I, 2021.

[XAdES] J. C. Cruellas, G. Karlinger, D. Pinkas, and J. Ross. *XML Advanced Electronic Signatures (XAdES)*. W3C Recommendation. http://www.w3.org/TR/XAdES/, Februar 2003.

[XML-DSig] D. Eastlake, J. Reagle, D. Solo, F. Hirsch, M. Nyström, T. Roessler, and K. Yiu. *XML-Signature Syntax and Processing Version 1.1*. W3C Recommendation. http://www.w3.org/TR/xmldsig-core/, April 2013.

[XML(v1.0)] François Yergeau, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, and Eve Maler. *Extensible Markup Language (XML) 1.0*. W3C Recommendation, Fifth Edition. http://www.w3.org/TR/xml/, November 2008.

[XML(v1.1)] François Yergeau, John Cowan, Tim Bray, Jean Paoli, C. M. Sperberg-McQueen, and Eve Maler. *Extensible Markup Language (XML) 1.1*. W3C Recommendation, Second Edition. http://www.w3.org/TR/xml11/, August 2006.

[XPath] Jonathan Robie, Michael Dyck, and Josh Spiegel. *XML Path Language (XPath) Version 3.1*. W3C Recommendation. http://www.w3.org/TR/xpath, March 2017.

[XSL] S. Adler, A. Bergl, J. Caruso, S. Deach, P. Grosso, E. Gutentag, A. Milowski, S. Parnell, J. Richman, and S. Zilles. *Extensible Stylesheet Language (XSL)*. W3C Proposed Recommendation. http://www.w3.org/TR/2001/PR-xsl-20010828/, August 2001.

[XSLT] J. Clark. *XSL Transforms (XSLT) Version 2.0*. W3C Recommendation. http://www.w3.org/TR/xslt, March 2021.

[ZPO] *Zivilprozessordnung in der Fassung der Bekanntmachung vom 5. Dezember 2005 (BGBl. I S. 3202; 2006 I S. 431; 2007 I S. 1781), die zuletzt durch Artikel 11 Absatz 15 des Gesetzes vom 18. Juli 2017 (BGB. I S. 2745) geändert worden ist.*