Resolution of comments on Drafts ETSI *EN 319 122-1 and* ETSI *EN 319 122-2 v0.0.3 – 31 May 2014*

CAdES

Foreword: Please note that the following disposition of comments is provided to the light of the current context of the m460 mandate, in particular with regards to Directive 1999/93/EC. It should be noted that such disposition should be reviewed to the light of the eIDAS Regulation.

Table of content

Comments on Draft ETSI *EN 319 122-1* V 0.0.3

Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES); Part 1: Core Specification

Comment number	**Clause/ Subclause**	**Paragraph Figure/ Table**	**Type of comment** (General/ Technical/Editorial)	**COMMENTS**	**Proposed change**	**OBSERVATIONS** on each comment submitted
Comment 1	4.1.3.1 (EN 319 122-1)		General	FEEDBACK TO EDITOR NOTE It is considered useful to include a referencing mechanism to the time-mark to allow automatic processing		**Finally, rejected:** The feedback was very useful. However, the STF finally decided to reject the possibility to include a reference to a time-mark into the signature, due to several reasons: 1) The reference can only be an unsigned reference, thus as long as it is not covered by another time-stamp, it has no sure information. 2) There is no specific format for time-marks. It could have multiple forms, like a within a Trusted Service Provider, a signed document, etc. As long as there is no information on in which form the time-stamp is stored, it is not feasible to provide a link to the time-mark that can be automatically processed.
Comment 2	4.1.3.2 (EN 319 122-1)		General	FEEDBACK TO EDITOR NOTE In our view, CAdES-A should be allowed even without including a timestamp / time-mark (CAdES-T) before the certificate expires or it is revoked		**Finally, rejected:** The feedback was very useful. However, the STF finally decided to reject the possibility to allow CAdES-A directly on CAdES-BES/EPES. The problem is that when using CAdES-A without CAdES-T, we don't know which date shall be used for the verification of the signature. Assuming we build CAdES-A directly on CAdES-BES/EPES. We would now have a time at which we know the signature was created, but all the added validation data will be created before that date. If is much clearer to first create CAdES-T and then extend it to CAdES-A.
Comment 3	**General**			XXX highly appreciates the activities at ETSI M/460 phase 2, which particularly address long term aspects of electronic signatures.	Therefore, it is proposed to enlarge the scope of the Draft ETSI <EN> <319 122-1 > V<V0.0.3 (2013-11) to cover as well the alternative approaches, which are based on the Evidence Record Syntax normalized in RFC 4998 and RFC 6283 and may be integrated with	**Declined** General disposition to all the comments derived from the general request of including ERS support in

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				However, in the current version of the proposed • "Draft EN 319 122-1 V0.0.3 (2013-11) Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES); Part 1: Core Specification" the **long-term-validation attribute including an Evidence Record type** is declared to be deprecated (p. 47ff), which is not optimal with respect to scalability because without the usage of Evidence Records each archived document requires independent archive time-stamps. Furthermore, this approach is not integrated with the international archival architectures standardized in • ISO 14721 "Space data and information transfer systems - Open archival information system - Reference model" and • ISO "14533-1:2012 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) (2012) • OASIS DSS v1.0 Profile for Comprehensive Multi-Signature Verification Reports Version 1.0 Committee Specification 01 (2010) and the German DIN-Standard and Technical Guideline • DIN 31647, Information and	archival systems based on ISO 14721 and ISO 14533 {C,X}AdES, OASIS DSS v1.0 Profile for Comprehensive Multi-Signature Verification Reports, DIN 31647 and TR 03125. **Proposed solution for CAdES:** The Evidence Record attribute shall be integrated into CAdES (as well as in XAdES and PAdES) as an ordinary attribute and not only as a deprecated attribute as it is actually done in chapter A.2.3 in the Draft ETSI <EN> <319 122-1 > V<V0.0.3 (2013-11 at the moment.	CAdES specification follows below. Nevertheless, this does not mean that some specific reaction or consideration is done to specific comments also derived from this general request, whenever is considered worth. The STF 458 made the resolutions copied below in its meeting held in 24/2/2014: 1. The STF 458 Area 1 Task 2 team proposes not to incorporate ERS management within all the AdES formats at this point in time. 2. The STF 458 Area 1 Task 2 team proposes to incorporate ERS management within ASiC packages so that signatures (CAdES, XAdES, PAdES?) that have been archived and preserved using ERS mechanisms, may be extracted from the archive, be packaged with the signed data objects, partial hash tree, and archive time-stamps, and be securely transferred to a different destination, where a relying party may still successfully validate the signatures. The new text will also provide guidance on the data objects that should also be securely archived within the ERS archive, for ensuring that the signature and all the required validation material is correctly preserved, and that once the signature and all the required validation material are extracted and incorporated to the ASiC package, the signature may be successfully validated. 3. The STF 458 Area 1 Task 2 team does not close the door to a potential incorporation of ERS within the different AdES formats, once analyzed the requirements for such an incorporation (which could also include an analysis of alternative archival systems), as all the different AdES formats include at this point in time extension mechanisms that would easily allow the definition of a potential new attribute (CAdES), property (XAdES), or dictionary (PAdES).. Note: The usage of long-term-validation attribute was deprecated since it does not handle the case where new unsigned attributes are added, e.g. a counter signature, after the addition of the LTV attribute.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				Documentation - Preservation of evidence of cryptographically signed electronic records (Beweiswerterhaltung kryptographisch signierter Dokumente), DIN draft standard. (2013) • Federal Office for Information Security (BSI): Technical Guideline 03125 Version 1.1: Preservation of Evidence of Cryptographically Signed Documents (TR-ESOR), available from from https://www.bsi.bund.de/EN/Publications/TechnicalGuidelines/TR03125/BSITR03125.html . (2011).		
Comment 4	**Motivation**		**T**	**Advantages of the Evidence Record syntax concept according RFC 4998 and RFC 6283:** • **Better Cost effectiveness and Performance:** ◦ Whereas XAdES-A requires one time-stamp per signature for a re-signing document the Evidence Record syntax standardised by IETF in RFC 4998 and RFC 6283 uses Merkle Hash Trees so that **only one time-stamp** is required for a complete re-signing **cycle of different documents.** • **Data Economy**		See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				• For any particular data object, the **hash tree can be reduced** to a few sets of hash values **(reduced hash trees**), which are sufficient to prove the existence of a single data object or data group**.** • **Data Protection** • Aspects with regard to **data protection technology** are also taken into account because with the ERS standard it is also **possible to delete parts of the document database** without compromising the conclusiveness of the remaining parts. • **Similar Processes independent of data formats** • The Evidence Record Syntax (ERS) specifies **similar processes** concerning generation, verification, time-stamp-renewal and hashtree-renewal of Evidence Records independent from the used data formats (e.g. CMS- or XML-based data formats) whereas the actual proposals for CAdES-A (e.g.		

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted

				archive-time-stamp-v3, ats-hash-index attribute) and XAdES-A (e.g. xadesv141:ArchiveTime-stamp element, xadesenv111:RenewedDigests and xadesv141: TimestampValidationData) look quite different.
				
				• **Combination of existing *AdES-A attributes with ERS is possible**
				  • E.g. ats-hash-index attribute could be a data object, which is part of the hashed data object group.
				  • E.g. the time-stamp of the root hash value of the ERS could be a Time-Stamping Authority (TSA) according to [RFC3161] or other data structures and protocols e.g. an xadesv141:ArchiveTimeStamp element or e.g. an archive-time-stamp-v3 attribute.
				• **Ordered list of POEs according to a clear life cycle concept and functional model**
				  • In the Evidence Record Syntax (RFC 4998 and 6283) there is a clear life cycle model and functional		

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				model. • Therefore, the ERS consists of a timely ordered and nested sequence of chains of Archive Timestamps (POEs) which facilitates the validation process. • In *AdES-A without Evidence Records and no timely ordered and nested POEs the validation process depends on low level data attributes and is more complicated (more test-cases in 8 steps, different status values, etc.) .		
Comment 5	**Use Cases**			**Use Cases:** - Preservation of the integrity and authenticity of digital records 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, especially for **Administration, Business and Science** in connection with - Secure electronic communication - Replacement through scanning - Documentation and		See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				analysis of processes - Electronic record and document management - Electronic filing and archiving - Proper administration - Electronic publication and promulgation of official leaves - **…** - **Exemplary Fields of Application** - E-Government - Pharmaceutical Industry - Electronic payment - Car - and Aircraft Industry - Health care - …		
Comment 6	**Conclusion**			**In most use cases it is a great advantage to have only one time-stamp for a complete re-signing cycle of many different documents and to have similar processes independent of the used data formats and data elements .**		See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 7	3.1	p. 13	E/T	**Current text:**	**Proposal: new definition:**	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
	Definitions			The definition of "Evidence Record" is missing.	**Evidence Record:** **Collection of evidence compiled for one or more given archived data objects over time. An Evidence Record includes all Archive Time-stamps (within structures of Archive Time-stamp Chains and Archive Time-stamp Sequences) and additional verification data, like certificates, revocation information, trust anchors, policy details, role information, etc." (see [4], p. 6)**	specifications.
Comment 8	Title of Chapter 4.1.3.2	p. 21	T	**Current text:** "CAdES-A with archive-time-stamp (ATSv3) attribute"	**Proposal:** "CAdES-A with archive-time-stamp (ATSv3 **or EvidenceRecord**) attribute"	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 9	Chapter 4.1.3.2	p. 21	T	**Current text:** "A CAdES Archival Electronic Signature (CAdES-A) with archive-time-stamp-v3 attribute, in accordance with the present document may be build on CAdES-BES, CAdES-EPES or any format above, by adding one or more archive-time-stamp-v3 attributes . .. The archive time-stamp gives an assurance when the signature existed already … The structure of the CAdES-A with ATSv3 form build on CAdES-T is shown in figure 4.	**Proposal:** "A CAdES Archival Electronic Signature (CAdES-A) with archive-time-stamp-v3 attribute **or EvidecneRecord attribute**, in accordance with the present document may be build on CAdES-BES, CAdES-EPES or any format above, by adding one or more archive-time-stamp-v3 attributes **or Evidence Record attribute**. … The archive time-stamp **or Evidence Record** gives an assurance when the signature existed already…. The structure of the CAdES-A with ATSv3 form **or Evidence Record** build on CAdES-T is shown in figure 4."	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 10	Figure 4	Illustration of CadES-A	T	**Current text in the right box of figure 4:** "Complete certificate and revocation data + Archive-time-stamp version 3 (ATSv3)"	**Proposal for the text in the right box of figure 4:** "Complete certificate and revocation data + rchive-time-stamp version 3 (ATSv3) **or** **Evidence Record**"	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 11	Chapter	1. Paragraph	T	**Current text:**	**Proposal:**	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
	6.5.1			"The ats-hash-index shall be carried as an unsigned attribute of the signature of the archive-time-stamp-v3 attribute (see clause 6.5.2)."	"The ats-hash-index shall be carried as an unsigned attribute of the signature of the archive-time-stamp-v3 attribute (see clause 6.5.2) **or the Evidence Record (see clause 6.5.3)**."	specifications.
Comment 12	Chapter 6.5	p. 34	T	**Current text:** 6.5 Archive validation data Where an electronic signature is required to last for a very long time, and the time-stamp token on an electronic signature (signature time-stamp or previous archival time-stamps) is in danger of being invalidated due to algorithm weakness or limits in the validity period of the TSA certificate, it may be required to time-stamp the electronic signature several times. When this is required, an archive time-stamp attribute may be required for the archive form of the electronic signature (CAdES-A). This archive time-stamp may be repeatedly applied over a period of time.	**Proposal:** 6.5 Archive validation data Where an electronic signature is required to last for a very long time, and the time-stamp token on an electronic signature (signature time-stamp or previous archival time-stamps) is in danger of being invalidated due to algorithm weakness or limits in the validity period of the TSA certificate, it may be required to time-stamp the electronic signature **or a data object group** with electronic signatures several times. When this is required, an archive time-stamp attribute **or an Evidence Record attribute** may be required for the archive form of the electronic signature (CAdES-A). This archive time-stamp **or Evidence Record attribute** may be repeatedly applied over a period of time.	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 13	**Chapter 6.5**	**New chapter 6.5.3** **Evidence Record**	**T**	**Current text:** A description of the Evidence Record attribute is missing.	**Proposal:** Please create a new chapter 6.5.3 The Evidence Record according to this document, **chapter 6.5.3 "The Evidence Record"**	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 14	**Chapter 7.4**	**p. 39**	**T**	**Current text:** "In addition it shall contain: • one or more archive-time-stamp-v3 attributes (see clause 6.5.2) to protect the complete certificate and revocation data, or/and • one or more archive time-stamp or long-term validation time-stamps defined in previous version of TS 101 733 [i.1], see	**Proposal:** "In addition it shall contain: • one or more archive-time-stamp-v3 attributes (see clause 6.5.2) **or EvidenceRecords (see clause 6.5.3)** to protect the complete certificate and revocation data, or/and • one or more archive time-stamp or long-term validation time-stamps defined in previous version of TS 101 733 [i.1], see also clauses A.2.2 and A.2.3."	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications. .

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				also clauses A.2.2 and A.2.3. … When a long-term-validation attribute is not present, applications claiming conformance to the present document shall generate an archive-time-stamp-v3 attributes (see clause 6.5.2) whenever a new archive time-stamp is required within the CAdES signature."	… When a long-term-validation attribute is not present, applications claiming conformance to the present document shall generate an archive-time-stamp-v3 attributes (see clause 6.5.2) **or an Evidence Record (see clause 6.5.3)** whenever a new archive time-stamp or **Evidence Record** is required within the CAdES signature."	
Comment 15	Chapter A.2.3	Last Paragraph	T	**Current text in the last paragraph of this chapter:** " If the PoE is an Evidence Record, then the process described above shall be done for every algorithm h_i in Evidence Record.digestAlgorithms , which will lead to a final digest computed by h_i . These final values for the different algorithms will be used as leaves in the hash tree building process of the Archive-Time-stamp of the Evidence Record as described in RFC 4998 [4]. If Evidence Record.digestAlgorithms contains only a single algorithm, the hash tree in the Archive-Time-stamp will contain only a single leave. All the hash functions contained in Evidence Record.digestAlgorithms shall also be included in the digestAlgorithms field of the SignedData ."	**The following sentence seems to be a misunderstanding:** "If Evidence Record.digestAlgorithms contains only a single algorithm, the hash tree in the Archive-Time-stamp will contain only a single leave." Proposal: Whereas CAdES-A requires one time-stamp per signature for re-signing one document the Evidence Record syntax standardised by IETF in RFC 4998 and RFC 6283 uses Merkle Hash Trees so that only one time-stamp is required for a complete re-signing cycle of different documents. In this case the leaves of the hash tree are hash values of one data object or the data objects in a group. Therefore even when the Evidence Record contains only a singe algorithm, there may be multiple leaves in the Merkle Hash Tree used by the Evidence Record.	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 16	E.2 Signature Format Definitions Using X.680 ASN.1 Syntax			**Current text:** -- Evidence Record Syntax RFC 4998 EvidenceRecord FROM ERS {iso(1) identified-organization(3) dod(6)	**Proposal:** Please insert also: According ASN.1-Module with **1997** Syntax: EvidenceRecord	**Declined** The evidence record is only used within a deprecated attribute. Adding a new ERS syntax would force people to include a new implementation for a deprecated attribute.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				internet(1) security(5) mechanisms(5) ltans(11) id-mod(0) id-mod-ers88(2) id-mod-ers88-v1(1)}" according to ASN.1-Module with 1988 Syntax	from ERS {iso(1) identified-organization(3) dod(6) internet(1) security(5) mechanisms(5) ltans(11) id-mod(0) id-mod-ers(1) id-mod-ers-v1(1) }	
Comment 17	**new** **Chapter 6.5.3**	**6.5.3 The Evidence Record** The following description is based on [4]. "The Evidence Record Syntax enables processing of several archive objects within a single processing pass using a hash tree technique and acquiring only one Time-Stamp to protect all archive objects. The leaves of the hash tree are hash values of the data objects in a group. A Time-Stamp is requested only for the root hash of the hash tree. The deletion of a data object in the tree does not influence the provability of others. For any particular data object, the hash tree can be reduced to a few sets of hash values, which are sufficient to prove the existence of a single data object. Similarly, the hash tree can be reduced to prove existence of a data group, provided all members of the data group have the same parent node in the hash tree." The Evidence Record Syntax (ERS) specifies processes for the generation and verification of Evidence Records. The standard defines in detail how re-signing and re-hashing, even **for large amounts of documents**, can be carried out automatically. Furthermore, the standard defines the data formats in which the Evidence Records are provided for an **unlimited period of time** and can be exchanged. Aspects with regard to **data protection technology** are also taken into account because with the ERS standard it is also possible to delete parts of the document database without compromising the conclusiveness of the remaining parts. Whereas CAdES-A requires one time-stamp per signature for a re-signing document the Evidence Record syntax standardised by IETF in RFC 4998 and RFC 6283 uses Merkle Hash Trees so that only one time-stamp is required for a complete re-signing cycle of a large amount of documents..		See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 18	**New Chapter 6.5.3.1**	6.5.3.1 Data Structures The Evidence Record attribute is an optional unsigned attribute. Several instances of this attribute may occur within the list of unsigned attributes. The Evidence Record attribute is a proof of existence (PoE) at a certain past date, computed over many signed archived data objects or archived data object groups of signed documents together with their signatures, including signed attributes and all other essential components of the signature. The Evidence Record contains an Archive Time-stamps Sequence, generated during a long archival period, and possibly useful data for validation. An **Archive Timestamp Sequence** is a part of the Evidence Record, which " is a sequence of Archive Timestamp Chains, where each				See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
		Archive Timestamp Chain preserves non-repudiation of the previous Archive Timestamp Chains, even after the hash algorithm used within the previous Archive Timestamp's hash tree became weak. Non-repudiation is preserved until the last Archive Timestamp of the last chain becomes invalid. The process of generating such an Archive Timestamp Sequence is called Hash-Tree Renewal. ([4], p.5)" An **Archive Timestamp Chain** is part of an Archive Timestamp Sequence, which "is a time-ordered sequence of Archive Timestamps, where each Archive Timestamp preserves non-repudiation of the previous Archive Timestamp, even after the previous Archive Timestamp becomes invalid. Overall non-repudiation is maintained until the new Archive Timestamp itself becomes invalid. The process of generating such an Archive Timestamp Chain is called Timestamp Renewal. ([4], p. 5)" An **Archive Timestamp** is "a timestamp and a list of hash values, which allow the verification of the existence of several data objects at a certain time.([4], p.5) …. The lists of hash values are generated by reduction of an ordered Merkle hash tree [MER1980]. 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 ([4], p. 11). A **Reduced Hashtree** contains lists of hash values, organized in PartialHashtrees for easier understanding. They can be derived by reducing a hash tree to the nodes necessary to verify a single data object. Hash values are represented as octet strings. If the optional attribute "reducedHashtree" is not present, the Archive-Timestamp simply contains an ordinary time-stamp.		
Comment 19	**New chapter 6.5.3.2**	6.5.3.2 Processes 6.5.3.2.1 Initial Archive-time-stamp in General According to ([4], p. 12), "the lists of hash values of an Archive Timestamp can be generated by building and reducing a Merkle hash tree [MER1980]. Such a hash tree can be built as follows: 1. Collect data objects to be timestamped. (Note1: If an ats-hash-index attribute is used according to chapter 6.5.1, the data object contains the concatenation of the SignedData.encapContentInfo.eContentType, the octets representing the hash of the signed data, the SignedData.signerInfos's item corresponding to the signature being archive-timestamped and a single instance of ATSHashIndex type in their binary encoded form without any modification and including the tag, length and value octets. The hashing process of such a data object including an ats-hash-index attribute is done according to Figure 5 of this document (p. 37). Note2: Instead of using an ats-hash-index attribute to secure the validationData, the validationDate of the signature of a data object could be stored in a data object which becomes part of a data object group together with the original data object , which has to be hashed according No. 3.) 2. Choose a secure hash algorithm H and generate hash values for the data objects. These values will be the leaves of the hash tree. 3. For each data group containing more than one document, its respective document hashes are binary sorted in ascending order, concatenated, and hashed. The hash values are the complete output from the hash algorithm, i.e., leading zeros are not removed, with the			See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications. The use of the ats-hash-index would allow to handle the case there new unsigned attributes are added. However, this would change the original hash tree. In the case where the ERS covers several documents, this has an influence of all copies of the ERS.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				most significant bit first. 4. If there is more than one hash value, place them in groups and sort each group in binary ascending order. Concatenate these values and generate new hash values, which are inner nodes of this tree. (If additional hash values are needed, e.g., so that all nodes have the same number of children, any data may be hashed using H and used.) Repeat this step until there is only one hash value, which is the **root node of the hash tree**. 5. Obtain **a time-stamp for this root hash value**. The hash algorithm in the time-stamp request MUST be the same as the hash algorithm of the hash tree, or the digestAlgorithm field of the Archive TimeStamp MUST be present and specify the hash algorithm of the hash tree". Note 1: A Time-stamp is a cryptographically secure confirmation generated by a Time-Stamping Authority (TSA), e.g., [RFC3161] , which specifies a structure for Time-Stamps and a protocol for communicating with a Time-Stamp Authority. Besides this, other data structures and protocols may also be appropriate, e.g. an archive-time-stamp-v3 attribute. Instead of using an archive-time-stamp-v3 attribute to secure the validationData, the validationData of the Archive Timestamp Sequence (e.g. certificates, revocation information, etc.) could be stored in the "cryptoInfo"-attribute of the Evidence Record ((see [4}, p. 10). 6.5.3.2.2 Validation of the Evidence Record According to ([4], p. 15ff), "an Archive Timestamp shall prove that a data object existed at a certain time, given by time-stamp. 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 time-stamp. …. 5. The verification of Archive Timestamp Chains and Archive Timestamp Sequences is done according to [4], p. 16ff. If a proof is necessary for more than one data object, steps 1 and 2 have to be done for all data objects to be proved. If an additional proof is necessary that the Archive Timestamp relates to a data object group (e.g., a document and all its signatures), it can be verified additionally, that only the hash values of the given data objects are in the first hash-value list." Note 1: When validating an Evidence Record using a time-stamp according the archive-time-stamp-v3 attribute (chapter 6.4.3), first the contained ats-hash-index of each such data object group shall be validated according to chapter 6.5.2.1.		

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
Comment 20	**New chapter 6.5.3.3**	6.5.3.3 Time-stamp Renewal According to ([4], p. 17), " The initial Archive Timestamp relates to a data object or a data object group. Before cryptographic algorithms that are used within the most recent Archive Timestamp (which is, at the beginning, the initial one) become weak or their time-stamp certificates become invalid, Archive Timestamps have to be renewed by generating a new Archive Timestamp. In the case of Time-stamp Renewal, the content of the time-stamp field of the old Archive Timestamp has to be hashed and timestamped by a new Archive Timestamp. The new Archive Timestamp MAY not contain a reducedHashtree field, if the time-stamp only simply covers the previous time-stamp. However, generally one can collect a number of old Archive Timestamps and build the new hash tree with the hash values of the content of their time-stamp fields. The new Archive Timestamp MUST be added to the Archive Timestamp Chain. This hash tree of the new Archive Timestamp MUST use the same hash algorithm as the old one, which is specified in the digestAlgorithm field of the Archive Timestamp or, if this value is not set (as it is optional), within the time-stamp itself." Note 1: Before incorporating a new time-stamp according to the archive-time-stamp-v3 Attribute , the SignedData of this time-stamp shall be extended to include any validation data, not already present, which is required for validating the signature being archive time-stamped. Validation data may include certificates, CRLs, OCSP responses, as required to validate any signed object within the signature including the existing signature, time-stamps, OCSP response and certificates and shall be included within the root SignedData.certificates , or SignedData.crls.			See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 21	**New chapter 6.5.3.4**	6.5.3.4 HashTree Renewal Before the hash algorithm used to build the hash trees in the Archive Timestamp loses its security properties, the Hash-Tree Renewal is required. In the case of a Hash-Tree Renewal, the Archive Timestamp and the archived data objects covered by the Archive Timestamp must be hashed and timestamped again, according to ([4], p. 18).				See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 22	**introduction**		**ed**	**It duplicates text from the scope.**	**Remove any text linked to the scope**	**Accepted**, Remove duplications, after resolution of ETSI 2
Comment 23	**scope**		**ed**	**Too long.**	**Review scope**	**Accepted**,

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				ETSI drafting rules: the scope defines without ambiguity the subject of the ETSI deliverable and the aspect(s) covered, thereby indicating the limits of applicability of the ETSI deliverable or particular parts of it... The "Scope" shall be succinct so that it can be used as a summary for bibliographic purposes	**Delete description of clauses. Reduce the first 7 paragraphs to describe concisely the subject of the standard and the limits of applicability.**	Remove clause description and reduce first 7 paragraphs.
Comment 24	**scope**		**tech**	**An ETSI standard cannot say it comply with legal provisions** The present document describes formats for advanced electronic signatures using ASN.1 (Abstract Syntax Notation 1) that remain valid over long periods, are compliant with the European Directive		**Accept removal** of "that remain valid over long periods, are compliant with the European Directive"
Comment 25	**4**		**ed**	**Clause 4 only contains 1 sub-clause.**	**Reduce the heading hierarchy to:** 4 Electronic Signature formats 4.1 Overview 4.2 CAdES Basic Electronic Signature (CAdES-BES) 4.3 CAdES Explicit Policy-based Electronic Signatures (CAdES-EPES) 4.4 Electronic Signature formats with validation data 4.4.1 Electronic Signature with time (CAdES-T) 4.4.2 CAdES-A with archive-time-stamp (ATSv3) attribute	**Accepted**

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
Comment 26	**4**		**ed**	**Clause 4 contains information which is redundant with other clauses. The standard needs to be as concise and precise as possible. Avoiding duplication of text is important**	**Merge and reduce what was in 4 and 4.1 to** **4.1 Overview :** The present document defines a number of Electronic Signature (ES) formats that shall build on CMS (RFC 5652 [**Erreur ! Source du renvoi introuvable.**]) by adding signed and unsigned attributes. These attributes are described in the present document and are either defined in CMS (RFC 5652 [**Erreur ! Source du renvoi introuvable.**]), ESS (RFC 2634 [**Erreur ! Source du renvoi introuvable.**] and RFC 5035 [**Erreur ! Source du renvoi introuvable.**]) or the present document. The attributes can be combined to generate different electronic signature forms. The following clauses definesfour forms of CMS-based advanced electronic signatures (CAdES), namely, the **Basic Electronic Signature** (CAdES-BES), **Explicit Policy-based Electronic Signature** (CAdES-EPES), the **Electronic Signature with Time** (CAdES-T) and the **Archival Electronic Signature** (CAdES-A). These forms are further profiled in CAdES Baseline Profile [**Erreur ! Source du renvoi introuvable.**]. The normative annex **Erreur ! Source du renvoi introuvable.** defines forms of CAdES signatures using attributes that contain references to validation data and attributes that encapsulate time-stamp tokens on the aforementioned references.	**Accepted**
Comment 27	**4 and 7**		**Ed/tech**	**information in 4 and 7 are redundant.**	**Keep only clause 4 and modify it to clearly specify mandatory and optional features**	**Accepted,**

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
					Proposal in attached revised draft	Do the same for annexes B and C
Comment 28	**4 and 5**		**tech**	**It is not clear which elements from RFC 5652 are mandatory.** **Shall CAdES formats comply with RFC 5652?** **The terminology "** A CAdES **Basic Electronic Signature** (CAdES-BES) shall consist of the following components: • The general CMS syntax and content type, as defined in RFC 5652 [**Erreur ! Source du renvoi introuvable.**] (see clauses **Erreur ! Source du renvoi introuvable.** and **Erreur ! Source du renvoi introuvable.**);" **And then** **Clause 5:** CAdES signatures build on Cryptographic Message Syntax (CMS), as defined in RFC 5652 **are not precise enough**	**Clearly specify what from RFC 5652 shall be supported.**	**Accepted,** State precisely what is taken from RFC 5652
Comment 29	**4 and 5**		**tech**	**Do not duplicate requirements.** **clause 4 says "**CMS `SignedData`, as defined in RFC 5652 [**Erreur ! Source du renvoi introuvable.**]," then in clause 5 we have a specific clause on SignedData which specifies what it shall be	**In clause 4: Remove any reference to RFC 5652 and only refer to the clause where the element is defined.** **Then in clause refer to RFC 5652 or the appropriate standard**	**Accepted**

Comment number	**Clause/ Subclause**	**Paragraph Figure/ Table**	**Type of comment** (General/ Technical/Editorial)	**COMMENTS**	**Proposed change**	**OBSERVATIONS** on each comment submitted
Comment 30	**5**		**tech**	**Is the text before 5.1 only informative?**		**Accepted,** Include normative text for basic CMS structure that can be references in CAdES-BES
Comment 31	**4.1.2**	**CAdES-EPES**	**tech**	**Is the following sentence a normative requirement?** **Or is this step imposed by another standard?** A counter-signature on a CAdES-EPES will be created after the signature and added encapsulated in the unsigned `countersignature attribute. The counter-signature covers the original CAdES-EPES signature.`	**Rephrase to express requirement or refer to standard imposing this**	This is not a normative text, but more an explanation how to place a counter signature in the figure. **Proposed solution:** Make a note out of it and add a reference to clause dealing with counter signatures
Comment 32	**4.1.3**		**ed**	**Introduce sub-header to cover the hanging paragraph at the beginning**	**Introduce sub header (using new numbering as suggested above)** **4.4.1 introduction**	**Accepted**
Comment 33	**4.1.3**		**tech**	**The purpose of the text before 4.1.3.1 is unclear.** **Is it the scope of the scope to specify what the validation data is (**validation data shall meet the requirements of the signature policy**?** **Is it in scope to define requirements on the signer and verifier (**The **validation data** may be collected by the signer and/or the verifier) otherwise a time-mark shall be available in an audit log: this is cover by 4.1.3.1 -> delete such	**Review the text before 4.1.3.1**	Was thought of as definition on what we mean by this. But it is probably better placed in EN 319 102. Proposed resolution: Shorten the text of this paragraphs and make reference to the definition in EN 319 102. **New text**: Validation of an electronic signature requires additional data to validate the electronic signature. This additional data is called **validation data**, as defined in EN 319 102 [xx], and includes: • Public Key Certificates (PKCs);

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				requirement from here		• revocation status information for each PKC; • trusted time-stamps or time-marks applied to the digital signature; and • when appropriate, the details of a signature policy to be used to verify the electronic signature. The present document defines unsigned attributes able to contain validation data that can be added to CAdES-BES and CAdES-EPES, leading to electronic signature formats that include validation data. Clauses 4.4.1 and 4.4.2 describe the basic formats including validation data. NOTE: When the `signature-policy-identifier` signed attribute is present, the validation data should meet the requirements of the signature policy.
Comment 34	**4.1.3.1**	**Figure 3**		**The figure is only an illustration. It cannot contain requirements (which are already part of the text)**	**Change right box to:** **Signature-time-stamp attribute or time mark managed and provisioned by the TSP** **Remove note 1.**	**Accepted**
Comment 35	**5 and all sub-clauses**		**tech**	**Appropriate verbal forms are not used to express normative requirements, recommendations, permissible action**	**Use appropriate verbal forms to express provisions (shall, should, may, can...)** **Replace "is as defined in" with "shall be as defined in"**	**Accepted**, revise section
Comment 36	**5**	**Note**	**ed**	**This is specified in RFC 5652. Delete.** **Does clause 2 of RFC 5652 apply, in particular " An implementation that conforms to this specification MUST**	**Delete the note** **If the clauses fully apply, suggest one clause specifying something like**	**Accepted**, delete note and give precise references to compliance with RFC 5652.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				implement the protection content, ContentInfo, and MUST implement the data, signed-data, and enveloped-data content types. The other content types MAY be implemented.?"	**CAdES shall comply with clauses 2, 3 and 4 of RFC 5652**	
Comment 37	**5.4**	**2nd paragraph**	**ed**	**Need to enhancing writing of provisions**	For the purpose of long-term validation, either the `eContent` should be present, or the data that is signed should be archived in such as way as to preserve any data encoding. NOTE 1: It is important that the OCTET STRING used to generate the signature remains the same every time either the verifier or an arbitrator validates the signature. NOTE 2: The `eContent` is optional in CMS: ▪ When it is present, this allows the signed data to be encapsulated in the `SignedData structure` which then contains both the signed data and the signature. However, the signed data can only be accessed by a verifier able to decode the ASN.1 encoded `SignedData` structure.	**Accepted**

Organization name	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
Comment 38	A.1.2.1		technical	At the Plugtests many participants discussed the right number of CrlOcspRef fields that shall be included in CompleteRevocationRefs. CAdES specification ETSI 101 733 V2.2.1 states: The complete-certificate-references attribute is an unsigned attribute. It references the full set of CA certificates that have been used to validate an ES with Complete validation data up to (but not including) the signer's certificate. Only a single instance of this attribute shall occur with an electronic signature. CompleteRevocationRefs attribute shall contain one CrlOcspRef field for the signing-certificate, followed by one for each OtherCertID in the CompleteCertificateRefs attribute. The second and subsequent CrlOcspRef fields shall be in the same order as the OtherCertID to which they relate. At least one of CRLListID or OcspListID or OtherRevRefs should be present for all but the "trusted" CA of the certificate path. The question could be raised about the CrlOcspRef for the root CA if the root CA reference is included in the CompleteCertificateRefs attribute. Some Plugtests participants included an empty reference in the CrlOcspRef for the root CA while some other participants didn't include any reference in the CrlOcspRef for the root CA. The clause "should be present for all but the trusted CA", means that it is required to put into revocation reference CRL/OCSP list for all certificates of the certificate path, but without revocation reference for trusted CA on the other hand the clause " followed by one for each OtherCertID in the ompleteCertificateRefs		**Proposed solution:** `CompleteRevocationRefs` shall contain one `CrlOcspRef` for the `signing-certificate`, followed by one for each `OtherCertID` in the `CompleteCertificateRefs` attribute, **except for the "trusted" CA**.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				attribute" means that that it is required to put into revocation reference CRL/OCSP list for all certificates whose reference is included in the complete-certificatereferences attribute.		
Comment 39				Some debate was devoted to which format of the OCSP response is preferable to be stored in the SignedData.OtherRevocationInfoFormat. 1. Identified by id-pkix-ocsp-basic of BasicOCSPResponse defined in clause 4.2.1of RFC 6960 2. Identified by id-ri-ocsp-response of OCSPResponse defined in clause 3 of RFC 5940 and in 4.2.1 of RFC 6960 When id-ri-ocsp-response is used then also some parts of OCSP protocol are included. Status from OCSP protocol is not protected and also not interesting in validation application. Only basic response as a signed object is important. For that reason the first option id-pkix-ocsp-basic seems to be preferable. A consequent/similar problem is that ETSI TS 101 733 does not specify what is the input for hash computation for the ocspRepHash field (complete-revocation-references attribute). Some participants used the entire OCSPResponse while other ones used only the BasicOCSPResponse. The same considerations reported above are still valid.		XAdES includes OCSP responses in fomr of `OCSPResponse`. RFC 5940 (from 2010) states that `OCSPResponse` shall be used. **Proposed solution**: To align with XAdES and RFC 5940, state that for when including an OCSP into the signature, `OCSPResonse` shall be used: • `SignedData.crls`: include the OCSPResponse within `other`, using id-ri-ocsp-response (1.3.6.1.5.5.7.16.2) • `revocation-values`: include the OCSPResponce within `otherRevVals` using id-ri-ocsp-response (1.3.6.1.5.5.7.16.2) • `complete-revocation-references:` include the reference on the digest computed on `OCSPResponse`. • Validation: Include a comment that in earlier versions of the document, the OCSPResponse could be also included as BasicOCSPRespons. Validation program should be able to handle this for backward compatibility.
Comment 40				Some participants asked about the correctness of including OCSP responses for TSA in RevocationValues. The problem relates to the correctness of including in RevocationValues any revocation material when the certificate to which it refers is not		**Proposed solution:** In the case of the ATSv3, this was clarified within a Note (see Comment 46).

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				referenced in CompleteCertificateRefs. A main conclusion of this topic is that RevocationValues holds the values of CRLs and OCSP referenced in the complete-revocation-references attribute and so revocation material for any certificate not referenced in CompleteCertificateRefs should not be included in RevocationValues (i.e. revocation material for TSA should be included in timestamptoken itself).		In the case of the RevocationValues, it is marked already in the latest version of the document: "This attribute may include the values of revocation data including CRLs and OCSPs for any TSUs that have provided the time-stamp tokens, if these certificates are not already included in the TSTs as part of the TSUs signatures. In this case, the unsigned attribute shall be added to the signedData of the relevant time-stamp token."
Comment 41				Some debate was devoted to which encoding shall be used for ATSHashIndex attribute. It seems that there is no clear requirement about ATSHashIndex BER or DER encoding in CAdES Mother Specification.		**Propose solution:** Add in clause 6.5.1: The ats-hash-index shall be DER encoded.

Comment 42				Many participants discussed about the right ASN.1 format for Signature Policies. The main debate concerned from which data the hash value of the signature policy should be calculated. There were two main positions.
				
				SignaturePolicy ::= SEQUENCE {
				
				signPolicyHashAlg AlgorithmIdentifier,
				
				signPolicyInfo SignPolicyInfo,
				
				signPolicyHash SignPolicyHash OPTIONAL
				
				}
				
				The hash is calculated over the DER value of the SignaturePolicy field without the outer type and length fields, and without the optional signPolicyHash field.
				
				1. It means the hash is calculated from the fields signPolicyHashAlg and signPolicyInfo and the hash value is
						The new CAdES document states that the hash shall be as defined in the document defining the signature policy.
						
						Proposed solution:
						
						This problem shall be corrected in the ASN.1 part of EN 319 172. Include this point into the issues list of that document.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted

				included in field signPolicyHash. Hash is calculated without taking into account the outer type and length fields of SignaturePolicy ::= SEQUENCE {
				
				2. It means the hash is calculated from the field signPolicyInfo and the hash value is included in field signPolicyHash. Hash is calculated without taking into account the outer type and length fields of
				
				SignPolicyInfo ::= SEQUENCE {
				
				Related to this topic there was a general agreement in remarking that the ETSI specification on signature policies must be re-worked and updated.		

Comment 43				There was a wide debate regarding signature validation when the fields' order of the DN of the issuer in the signing certificate is different from the one declared in ESSSigningCertificateV2.IssuerAndSerialNumber.Name.A Name is structured in hierarchical levels, each level (RelativeDistinguishedName) represented by a set, not ordered, of attributes and their values. A Name identifies a node of the tree of the directory (defined in X.501, which is interfaced using the LDAP protocol), each RelativeDistinguishedName is the name of a node; the position of the leaf node (Issuer / Subject) is the concatenation of the names of the nodes in the path from the root of directory to the leaf. Two Name are equal if they identify the same node, so if the sequence of intermediate nodes is the same. To decide whether two intermediate nodes are equal one must compare the attributes that make their RelativeDistinguishedName: the attributes must be equal in number, type and corresponding value, no matter in what		This is a general validation problem. **Suggested solution:** The problem shall be handled in EN 319 102.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				order they appear.		
Comment 44				There were some discussions between participants regarding signature validation results if signed attributes (that shall be DER encoded) are not ordered (ascending lexicographic order of BER encoding). Shall the validation of the signature fail even if the signature is correct from the point of view of cryptographic calculation in such case? Some participants relaxed their code to accept these signatures while other didn't consider them valid.		This is a CAdES specific validation problem. For a correct CMS implementation the ordering must be done correctly. **Suggested solution:** Add text to general part, that the order of signed attributes is important, with reference to CMS.

Comment 45				Some participants noticed that the syntax definition of ATSHashIndex seems to violate the ITU-T X.680 requirements for SEQUENCE type components.		The ASN.1 ATSHashIndex definition was wrong. Due to an untagged default entry, it is not possible to know which entry should be decoded. Also a default value for the hash algorithm is not optimal since the algorithm might get deprecated at some time.
						
						Proposed solution:
						
						Define an ats-hash-index-v2 attribute using the following syntax:
						
						ATSHashIndex ::= SEQUENCE {
						hashIndAlgorithm AlgorithmIdentifier,
						certificatesHashIndex SEQUENCE OF OCTET STRING,
						crlsHashIndex SEQUENCE OF OCTET STRING,
						unsignedAttrsHashIndex SEQUENCE OF OCTET STRING
						}
						
						Deprecate ats-hash-index.
						
						Add a reference to TS 119 312 for the selection of the hash algorithm and that it is recommended to use SHA-256.

Comment 46				One of the proposed test cases previewed that revocation material related to an ATSv3 should have been included within		We do not want to forbid to add the revocation data into the time-stamp token, since it might already be included by the TSA.

Comment number	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				the token itself (when generated). It was suggested during the Plugtest that ATSv3 never contains revocation material. Revocation material for ATSv3 must always be included in the root SignedData only.		**Proposed solution:** NOTE 5: The validation data of the ATSv3 can be added either into the time-stamp token or into the root SignedData. It is recommended to put the validation data into the root SignedData, if this is possible and if the validation data is not already included in the TSTs as part of the TSUs signatures.

Comments on Draft ETSI *EN 319 122-2* V 0.0.3

Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES); Part 2: Baseline Profile

Organization name	**Clause/ Subclause**	**Paragraph Figure/ Table**	**Type of comment** (General/ Technical/Editorial)	**COMMENTS**	**Proposed change**	**OBSERVATIONS** on each comment submitted
Comment 47	2.2 [i.6] (EN 319 122-2)		General	Decision 2009/767/EC has been amended	Include CD 2013/662/UE, amending the Commission Decision 2009/767/EC	**Accepted**
Comment 48	**General**		**T**	XXX highly appreciates the activities at ETSI M/460 phase 2, which address particularly long term aspects of electronic signatures. However in the current version of the proposed • "Draft EN 319 122-2 V0.0.3 (2013-11) Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAdES); Part 2: Baseline Profile" only covers approaches without an optional usability of Evidence Records according RFC 4998 and RFC 6283, which are not optimal with respect to scalability because without the usage of Evidence Records each archived document requires independent archive time stamps.		See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.
Comment 49				**Furthermore this approach is not** integrated with the international archival architectures standardized in • ISO 14721 "Space data and information transfer systems - Open archival information system - Reference model" and • ISO "14533-1:2012 Processes, data elements and documents in commerce, industry and administration -- Long term signature	Therefore it is proposed to enlarge the scope of the Draft ETSI EN 319 122 V0.0.3 (2013-11) to cover alternative approaches as well, which are based on the Evidence Record Syntax normalized in RFC 4998 and RFC 6283 and may be integrated with archival systems based on ISO 14721, ISO 14533 {C,X}AdES, OASIS DSS v1.0 Profile for Comprehensive Multi-Signature Verification Reports, DIN 31647 and BSI-TR03125 . **Proposed solution for CAdES:** The Evidence Record attribute shall be integrated in CAdES (as well as in in XAdES and PAdES) as an ordinary	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.

Organization name	Clause/ Subclause	Paragraph Figure/ Table	Type of comment (General/ Technical/Editorial)	COMMENTS	Proposed change	OBSERVATIONS on each comment submitted
				profiles -- Part 1: Long term signature profiles for CMS Advanced Electronic Signatures (CAdES) (2012) and • ISO 14533-2:2012 Processes, data elements and documents in commerce, industry and administration -- Long term signature profiles -- Part 2: Long term signature profiles for XML Advanced Electronic Signatures (XAdES) (2012) • OASIS DSS v1.0 Profile for Comprehensive Multi-Signature Verification Reports Version 1.0 Committee Specification 01 (2010) and the German DIN-Standard and Technical Guideline • DIN 31647, Information and Documentation - Preservation of evidence of cryptographically signed electronic records (Beweiswerterhaltung kryptographisch signierter Dokumente), DIN draft standard. (2013) • Federal Office for Information Security (BSI): Technical Guideline 03125 Version 1.1: Preservation of Evidence of Cryptographically Signed Documents (TR-ESOR), available from from https://www.bsi.bund.de/EN/Publications/TechnicalGuidelines/TR03125/BSITR03125.html . (2011).	attribute, not only as a deprecated attribute.	
Comment 50	Chapter 2.1	Normative references	E		Add: [7] IETF RFC 4998 (2007): "Evidence Record Syntax (ERS)"	See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.

<table>
<tr><th>Organization name</th><th>Clause/ Subclause</th><th>Paragraph Figure/ Table</th><th>Type of comment (General/ Technical/Editorial)</th><th>COMMENTS</th><th>Proposed change</th><th>OBSERVATIONS on each comment submitted</th></tr>
<tr><td>Comment 51</td><td>Chapter 4</td><td>Conformance Levels</td><td>T</td><td>Current Text:

LTA-Level profiles the incorporation of time-stamp tokens that allow validation of the signature long time after its generation. This level is understood to tackle the long term availability and integrity of the validation material.</td><td>Proposal:

LTA-Level profiles the incorporation of time-stamp tokens or Evidence Records that allow validation of the signature long time after its generation. This level is understood to tackle the long term availability and integrity of the validation material.</td><td>See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.</td></tr>
<tr><td>Comment 52</td><td>Chapter 9</td><td>Requirements for LTA-Level Conformance</td><td>T</td><td>Current Text:

When a long-term-validation attribute is not present, CAdES signatures conformant to LTA-Level shall be a signature conformant to the LT-Level to which one or more archive-time-stamp-v3 attributes (each one including one ats-hash-index unsigned attribute as specified in [1]) have been incorporated.</td><td>Proposal:

When a long-term-validation attribute is not present, CAdES signatures conformant to LTA-Level shall be a signature conformant to the LT-Level to which one or more archive-time-stamp-v3 attributes (each one including one ats-hash-index unsigned attribute as specified in [1]) or Evidence Records according [7] have been incorporated.</td><td>See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.</td></tr>
<tr><td>Comment 53</td><td></td><td></td><td></td><td>Current Text:

Evidence Record is missing in Table 13</td><td>Proposal:

Please add Evidence Record in Table 13</td><td>See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.</td></tr>
<tr><td>Comment 54</td><td></td><td></td><td></td><td></td><td>
<table>
<tr><td>Service / Protocol element</td><td>CAdES [1] Reference</td><td>Generator Requirements</td></tr>
<tr><td>Service: add archive time-stamp</td><td>Clause 6.5</td><td>M</td></tr>
<tr><td>archive-time-stamp-v3</td><td>Clause 5.2</td><td>O</td></tr>
<tr><td>Evidence Record</td><td>Clause 6.5.3</td><td>O</td></tr>
</table>
</td><td>See resolutions 1, 2 and 3 in Comment 3 on the incorporation of ERS within (C/P/X)AdES specifications.</td></tr>
</table>