EUROPEAN COMMISSION

DIGIT
Connecting Europe Facility

# CEF eSignature DSS

Version 0.02

# Qualified electronic signature (QES) validation algorithm

Document Status:

| Status |
|---|
| [Draft/Approved/Final] |

Summary of Changes:

| Version | Date | Created by | Short Description of Changes |
|---|---|---|---|
| 0.01 | 14/05/2018 | Olivier Barette | Creation |
| 0.02 | 17/07/2018 | Olivier Barette | Update |
|  |  |  |  |

# Table of Contents

# 1. Introduction

The following algorithm has been implemented in the DSS open-source library in version 5.3, and represents the Connecting Europe Facility's (CEF) eSignature Building Block's interpretation of the eIDAS Regulation's requirements for the validation of qualified and advanced signatures and seals.

This algorithm has been designed following discussions and meetings with experts involved in the field, in the context of the CEF eSignature Building Block. This algorithm, however, should not be considered as a standard, but rather as guidelines for implementers, or parties interested in understanding how QES validation is implemented in DSS.

Note that an ETSI standard, TS 119 172-4, is currently being drafted with the aim of standardizing a "signature validation policy for European qualified electronic signatures/seals using trusted lists".

The algorithm below focuses on determining 3 sub-conclusions:

- Whether the certificate is qualified
- What is the type of this certificate
- Whether the corresponding private key is protected by a QSCD.

These sub-conclusions are important for handling the eIDAS Regulation's Article 32.1(a), (b) and (f) (and corresponding article 40 for e-seals) with the aim of determining whether an electronic signature or seal can be considered as QESig / QESeal / AdESig-QC / AdESeal-QC / AdESig / AdESeal / AdES(?). Note, however, that verifying compliance against Articles 26 and 36 (requirements for advanced electronic signatures and advanced electronic seals) are outside of the scope of the present document.

For the sake of simplicity, the algorithm focuses on the case where the time of signing is after 01/07/2016. Note that this time of signing is the "best possible time", for which a proof of existence is available. By default, it is the validation time (current time). If any proof of existence of signature is found, the lowest trusted time is used (signature-timestamp)

# 2. Definitions and abbreviations

These definitions and abbreviations are based on ETSI TR 119 001 and ETSI EN 319 102-1.

## 2.1. General definitions and abbreviations

sigCert: signing certificate. The certificate corresponding to the private key that was used to produce a digital signature.

QC: qualified certificate.

WSA: web site authentication (type of certificate).

Type of QC: type of a qualified certificate. Three types are currently defined by the eIDAS Regulation: for electronic signature, for electronic seal, for web site authentication.

TL: trusted list.

TSP: trust service provider.

LOTL: list of the trusted lists.

## 2.2. Definitions and abbreviations regarding signature levels

AdES: digital signature that is either a CAdES signature, or a PAdES signature or a XAdES signature. A digital signature is defined as data appended to, or a cryptographic transformation of a data unit that allows a recipient of the data unit to prove the source and integrity of the data unit and protect against forgery e.g. by the recipient.

> Note: Regulation (EU) No 910/2014 (the "eIDAS" Regulation) defines the terms electronic signature, advanced electronic signature, electronic seals and advanced electronic seal. These signatures and seals are usually created using digital signature technology. The present document aims at supporting the Regulation (EU) No 910/2014 [i.15] for creation and validation of advanced electronic signatures and seals when they are implemented as AdES digital signatures.

AdESig: AdES supported by a non-qualified certificate for electronic signatures.

AdESeal: AdES supported by a non-qualified certificate for electronic seals.

AdESig-QC: AdES supported by a qualified certificate for electronic signatures.

AdESeal-QC: AdES supported by a qualified certificate for electronic seals.

AdES(?): AdES supported by a non-qualified certificate for which the type could not be determined. That is, the algorithm could not determine whether it is a certificate for electronic signatures, for electronic seals, or even for web site authentication.

QES: AdES supported by a qualified certificate, with the corresponding private key protected by a QSCD.

QESig: QES where the certificate is for electronic signatures.

QESeal: QES where the certificate is for electronic seals.

## 2.3. Definitions and abbreviations regarding the content of a certificate

QcCompliance: individual QcStatement standardized by ETSI EN 319 412-5 that can be present in the qcStatements extension of a X.509 certificate. The presence of this QcStatement claims that the certificate is an EU qualified certificate that is issued according to Directive 1999/93/EC [i.3] or the Annex I, III or IV of the Regulation (EU) No 910/2014 whichever is in force at the time of issuance. Its formal syntax is id-etsi-qcs-QcCompliance.

QcType: individual QcStatement standardized by ETSI EN 319 412-5 that can be present in the qcStatements extension of a X.509 certificate. The presence of this QcStatement claims that an EU qualified certificate is issued as one specific types according to Annexes I, III or IV of the Regulation (EU) No 910/2014 when used in combination with the QcCompliance defined above. When used on its own it indicates that it is used for the purposes of electronic signatures, seals or web sites for "nonqualified

certificates" within the context of Regulation (EU) No 910/2014. Its formal syntax is id-etsi-qcs-QcType. For the moment, 3 values are defined:

- id-etsi-qct-esign (for the purpose of electronic signatures)
- id-etsi-qct-eseal (for the purpose of electronic seals)
- id-etsi-qct-web (for web site authentication)

  Note: This statement, without the one defined in clause 4.2.1 of ETSI EN 319 412-5, can be potentially used in other regulatory environments which use electronic signature, electronic seal or web site with the same meaning.

QcQSCD: individual QcStatement standardized by ETSI EN 319 412-5 that can be present in the qcStatements extension of a X.509 certificate. The presence of this Qcstatement claims that the private key related to the certified public key resides in a QSCD according to the Regulation (EU) No 910/2014 [i.8] or a secure signature creation device (SSCD) as defined in the Directive 1999/93/EC [i.3]. Its formal syntax is id-etsi-qcs-QcSSCD.

QCP and QCP+: certificate policies defined in ETSI TS 101 456.

- QCP (qcp-public): a certificate policy for qualified certificates issued to the public.
- QCP+ (qcp-public-with-sscd): a certificate policy for qualified certificates issued to the public, requiring use of secure signature-creation devices.

## 2.4. Definitions and abbreviations regarding the content of a trusted list

More information can be found in version 2.1.1 of ETSI TS 119 612.

CA/QC: Service type identifier http://uri.etsi.org/TrstSvc/Svctype/CA/QC: trust service issuing qualified certificates.

SDI: Service Digital Identity unambiguously identifying the trust service. The standard imposes it to contain at least one certificate (if the service uses PKI public-key technology). In the present document, the SDI is said to be "catching" a sigCert if a path can be found from the sigCert up to the SDI.

Sie:aSI:ForXX: Service information extension / additional service information that specifies for which type of certificates the service is provided. ForXX is an abbreviation for ForeSignatures / ForeSeals / ForWebSiteAuthentication.

Sie:Q:QcStatement : Service information extension / qualification that is composed of one or more criteria and a qualifier. The qualifier applies to the sigCert only if the sigCert meets the criteria. In the present document, "a Sie:Q:QcStatement is present" shall be understood as "a Sie:Q:QcStatement that applies to the sigCert is present".

Sie:Q:notQualified : Service information extension / qualification that is composed of a criteria and a qualifier. The qualifier applies to the sigCert only if the sigCert meets the criteria. In the present document, "a Sie:Q:notQualified is present" shall be understood as "a Sie:Q:notQualified that applies to the sigCert is present". The criteria are then said to "catch" this sigCert.

Sie:Q:QCForXX: Service information extension / qualification that is composed of criteria and a qualifier. The qualifier applies to the sigCert if and only if the sigCert is qualified and meets the criteria. In the present document, "a Sie:Q:QCForXX is present" shall be understood as "a Sie:Q:QCForXX that applies to the sigCert is present". The criteria are then said to "catch" this sigCert. QCForXX is an abbreviation for QCForESig / QCForESeal / QCForWSA.

Sie:Q:QCXXQSCD : Service information extension / qualification that is composed of criteria and a qualifier. The qualifier applies to the sigCert only if the sigCert is qualified and meets the criteria. In the present document, "a Sie:Q:QCXXQSCD is present" shall be understood as "a Sie:Q:QCXXQSCD that applies to the sigCert is present". The criteria are then said to "catch" this sigCert. QCXXQSCD is an abbreviation for QCWithQSCD / QCNoQSCD / QCQSCDStatusAsInCert / QCQSCDManagedOnBehalf.

## 2.5. Other definitions and abbreviations

QSCD : qualified signature creation device, as defined in the Regulation (EU) No 910/2014 (the "eIDAS" Regulation).

EUOJ: Official Journal of the European Union.

# 3. UNDERLYING PRINCIPLES

The algorithm presented in the next section is based on the following principles:

## 3.1. CA/QC trust service(s) matching the sigCert

A CA/QC entry in a TL will correspond to the sigCert if and only if:

- A path can be found from the sigCert up to the SDI of this entry.
- The type of the sigCert is in line with the type of qualified certificates this service is issuing, taking into account possible overruling of the TL (see below the section on overruling). Even if the SDI is matching, a CA/QC entry will not be related to a sigCert if the corresponding types (Sie:aSI:ForXX and QcType respectively) are not matching.
- The certificate is confirmed to be qualified, taking into account possible overrule in the TL entry (see below the section on overruling). Even if the SDI is matching, a CA/QC entry will not be related to a sigCert if the sigCert is not qualified.

For instance, when looking for the CA/QC entry that catches the sigCert:

- Several CA/QC entries with the same catching SDI but with different Sie:aSI:ForXX may exist. The entry actually catching the sigCert will be the one with the appropriate Sie:aSI:ForXX.
- One CA/QC entry may exist with a catching SDI. This entry will not be catching the sigCert because the sigCert is not qualified.
- One CA/QC entry may exist together with a CA/PKC, with the same catching SDI and the same Sie:aSI:ForXX. If the certificate is not qualified, the CA/QC entry will not be considered as catching it (and so will not be considered as applicable).

A further check should be performed to rule out unapplicable cross-certification or root signing (e.g. outside of EU): When present, the organizationIdentifier attribute of the issuer of the sigCert, or the issuerAltName field, should match the TSP name or the TSP trade name of the TSP service entry.

## 3.2. Overruling by the TL

- Sie:Q:QcStatement or Sie:Q:notQualified qualifier in the TL overrules the QcCompliance statement present in the sigCert, if any

  Note: As stated in its definition above, it is also subject to the condition that the corresponding criteria is catching this sigCert.

- Sie:Q:QCXXQSCD qualifier(s) in the TL overrule(s) the QSCD statement present in the sigCert, if any.

Note: As stated in its definition above, it is also subject to the conditions that the corresponding criteria is catching this sigCert, and that this sigCert is concluded to be qualified.

- Sie:Q:QCForXX qualifier in the TL overrules the QcType statement present in the sigCert, if any.

  Note: As stated in its definition above, it is also subject to the conditions that the corresponding criteria is catching this sigCert, and that this sigCert is concluded to be qualified.

## 3.3. Two moments in time to be considered under Article 32 and Article 40 of eIDAS

There are two moments in time to be considered when validating a signature under Article 32 (and a seal under Article 40):

- Time of issuance of the certificate, due to Article 32 1(b);
- Time of signing, due to Article 32 1(a) and Article 32 1(f) among others.

Note: the type of the certificate at the time of issuance and at the time of signing regarding shall be the same. The algorithm will raise an error if the type has changed in between.

## 3.4. Before eIDAS vs. under eIDAS

The fields to be checked, as well as the manner in which to check them, in both the sigCert and the TL differ depending on whether the moment in time under consideration is before the entry into force of eIDAS, or after.

Note that the entry into force of eIDAS is 1 July 2016 0:00:00 CET, due to the eIDAS Regulation being signed in Brussels. This translates to 30 June 2016 22:00:00 in UTC.

Fields to be checked in the sigCert and that differ before and under eIDAS are:

- QCP and QCP+ are alternatives to QcCompliance and/or QcQSCD, but only before eIDAS. They are ignored under eIDAS.
  - As for the algorithm, QCP is considered as an alternative to the presence of QcCompliance, and QCP+ is considered as an alternative to the presence of both QcCompliance and QcQSCD.
  - Any combinations of QCP/QCP+/QcCompliance/QcQSCD are accepted, so including QCP and QCP+ simultaneously.
- qcp-natural-qscd and qcp-legal-qscd policy OIDs are not considered as alternatives to QcCompliance and/or QcQSCD. They are ignored by the algorithm.

Fields to be checked in the TL and that differ before and under eIDAS are:

- QCForXX is ignored before eIDAS with a warning, as the only type existing before eIDAS is for electronic signature
- The qualified status of a TSP is designated:
    - Before eIDAS by "undersupervision", "supervisionincessation", or "accredited".
    - Under eIDAS by "granted".

## 3.5. Qualified status of a certificate is lost if the qualified status of the issuing trust service is withdrawn

The qualified status of a certificate is not immutable after its issuance, as this status is dependent on the qualified status of the issuing trust service. If the trust service loses its qualified status, the already-issued certificates will lose their qualified status starting from the same date and time.

## 3.6. Interpretation of QcType in the sigCert

QcType can be present without QcCompliance in the sigCert. Depending on the QcStatement in the TL, it will either determine the type of QES (QESig or QESeal) or the type of AdES (AdESig or AdESeal).

Absence of QcType is interpreted as presence of QcType id-etsi-qct-esign (i.e. certificate for electronic signatures).

> Note: A certificate for WSA is not accepted either as a certificate for e-signatures nor as a certificate for e-seals.

# 4. PRELIMINARY STEPS AND CHECKS

There can be problems:

- When accessing and validating the LOTL and the TLs.
- With incompatible statements in the TL.
- With incompatible statements between the content of the sigCert and the TL.

The subsections below describe consistency checks related to these potential problems. If such consistency check fails:

- In most of the cases, a warning is included in the validation report.
- In some critical cases, a full stop of the process, with the result being an error.

## 4.1. Access LOTL and all TLs

- General checks on LOTL
  - Availability: if not available immediately, ensure a certain freshness.
  - Not expired (listIssue date, nextUpdate).
  - Correctly signed (based on EUOJ and pivot LOTL(s)).
- General checks on TL
  - Availability: if not available immediately, ensure a certain freshness.
  - Not expired (listIssue date, nextUpdate date).
  - Correctly signed (based on information present in the LOTL).
- Failures of these checks are reported as warnings in the validation report.

## 4.2. Identify CA/QC service(s) in the TL as trust anchor(s), and detect inconsistencies

Based on all TLs, identify the CA/QC entry(ies) to which a path can be built from the sigCert. This part is further detailed in the next section.

The following coherence checks on this TL entry(ies) are performed:

- Sie:aSI:ForXX shall not be set for a service status entry before eIDAS. The algorithm will ignore it and raise a warning.
- Sie:Q:QcForXX shall not be set for a service status entry before eIDAS. The algorithm will ignore it and raise a warning.
- The following Sie:Q:* statements are mutually exclusive and will raise an error:
  - QcStatement and NotQualified for the same sigCert under consideration.
  - QcForeSig, QcForeSeal, QcForWSA for the same sigCert under consideration.
  - QcForLegalPerson, QcForeSig for the same sigCert under consideration.

- One shall not be able to conclude both QSCD and not QSCD. The following combinations are inconsistent:
    - QcNoQSCD together with any of the following statements: QcWithQSCD, QcQSCDManagedOnBehalf or QcQSCDStatusAsInCert,
    - QcWithQSCD and QcQSCDStatusAsInCert,
    - QcQSCDManagedOnBehalf and QcQSCDStatusAsInCert.
    - The same 3 combinations, with "QSCD" replaced by "SSCD" in all statements.
- Sie:aSI:ForXX and Sie:Q:QcForXX shall be consistent, i.e. if Sie:Q:QcForXX is forcing certificates to be for eSignature / eSeal / WSA, then a corresponding Sie:aSI:ForXX shall be declared for that trust service.
- The organizationIdentifier or the issuerAltName of the sigCert shall match the TSP Name or the TSP trade name. Note: It could be located at other places in the sigCert, but the algorithm only checks these 2.
-

# 5. Main algorithm

The algorithm is composed of 3 main parts, based on Articles 32 and 40 of the eIDAS Regulation:

- Determining if the sigCert was a QC for eSig / eSeal (and is valid) at the time of issuance.
- Determining if the sigCert was a QC for eSig / eSeal and is related to a QSCD (and is valid) at the time of signing.
- Determining if the signature is an AdES. This step in based on ETSI EN 319 102-1 and will not be developed further in the present document.

Provided that it is parametrized with a date, the first 2 parts can be factorized into:

- Determining the qualified status of the sigCert, and its type.
- At the time of signing: If the sigCert is qualified, determining if the corresponding private key is protected by a QSCD.

For executing these steps, one has to find which CA/QC entry(ies) is(are) corresponding to the sigCert. Because of the overruling of the trusted list on the sigCert content, several CA/QC entries may catch the sigCert at first, and so the algorithm considers all these CA/QC entry(ies) catching the sigCert, irrespective of the service status, and irrespective of the Sie:aSI:ForXX type. For instance:

- A CA/QC Sie:aSI:ForeSignatures might catch a qualified sigCert with QcType eSeal if there is an overruling Sie:Q:QcForeSig catching this sigCert and forcing it to be for electronic signatures.
- A CA/QC might catch a non-qualified sigCert if there is an overruling Sie:Q:QcStatement catching this sigCert and forcing it to be qualified.

For each applicable entry, conclusion based on the 3 subsections below can be: QESig / QESeal / AdESig-QC / AdESeal-QC / AdESig / AdESeal / AdES(?).

However, after considering all these CA/QC entries corresponding to the sigCert, applying for each the rules presented in the following subsections, there should remain only one applicable CA/QC entry and one conclusion:

- If there is no applicable CA/QC entry left, the sigCert is considered as not qualified, and its type is QcType.
- If there is more than one applicable CA/QC entry left, and their conclusions are different, it raises an error and the algorithm stops.
- If there is more than one applicable CA/QC entry left, and their conclusions are identical, it raises a warning and the algorithm conclusion is this one.

## 5.1. Qualified status of the sigCert

- If the CA/QC entry under consideration is not in granted status, the sigCert is not qualified.

- If the CA/QC entry under consideration is overruling the qualified status of the sigCert (Sie:Q:QcStatement or Sie:Q:NotQualified), then the conclusion on the sigCert qualification is based on the CA/QC entry, whatever is present in the sigCert.
- If the CA/QC entry under consideration is not overruling the qualified status of the sigCert, then the conclusion on the sigCert qualification is based on the sigCert content:
  - Before eIDAS, the algorithm considers both the QcCompliance and the QCP / QCP+ OIDs
  - Under eIDAS, the algorithm considers the QcCompliance only.

## 5.2. Type of the sigCert

- If the last subsection concluded on the sigCert being not qualified, then the type of the sigCert is as declared by the QcType.
- If the last subsection concluded on the sigCert being qualified, this conclusion is only applicable provided that the type of certificate does match. The type declared in the certificate and the type declared in the CA/QC entry under consideration do match if:
  - There is an overruling Sie:Q:QcForXX catching the sigCert. The conclusion on the type is then XX, whatever is present in the sigCert.
  - There is no overruling Sie:Q:QcForXX catching the sigCert, and the QcType of the sigCert belongs to the list of Sie:aSI:ForXX present in the CA/QC entry. The conclusion is on the type is then QcType. In case of absence of QcType, the QcType is considered to be eSig.

  If these two types do not match, then the CA/QC entry is not applicable, and the algorithm removes it from the list of potential CA/QC entries.

There cannot be 2 or more QcType present in a QC under eIDAS. If this happens, the sigCert is rejected and an error is raised.

## 5.3. QSCD status

If sigCert is not qualified, no conclusion is drawn on QSCD. The algorithm proceeds to this step only if in the last subsection the sigCert was concluded to be qualified. Then:

- If there is an overruling Sie:Q:QCXXQSCD catching the sigCert, conclusion on QSCD is drawn from Sie:Q:QCXXQSCD, whatever is present in the sigCert.
- If no overruling Sie:Q:QCXXQSCD is catching the sigCert , conclusion on QSCD is drawn based on the sigCert content:
  - Before eIDAS, the algorithm considers both the QcQSCD and the QCP+ OID.

- Under eIDAS, the algorithm considers the QcQSCD only.

# 6. CONTACT INFORMATION

CEF eSignature Support Team

Support for DSS is provided through JIRA. Questions, issues or feature requests can be posted on:

**https://ec.europa.eu/cefdigital/tracker/projects/DSS/summary**

For more information on the provided support, refer to:

**https://ec.europa.eu/cefdigital/wiki/display/CEFDIGITAL/eSignature+Service+desk**

For general question on the CEF eSignature building block:

**CEF-ESIGNATURE-SUPPORT@ec.europa.eu**