وزارة المواصلات والاتصالات
MINISTRY OF TRANSPORT AND COMMUNICATIONS

Electronic Signature
Overview & Specification

Version: 1.0

Author: Qatar Public Key Infrastructure Section

Document Classification: PUBLIC

Published Date: May 2018

Document Information

Date	Version	Reviewed By
01/05/2018	1.0	Qatar National PKI Team

Content

1. Legal Mandate(s)

Emiri decision No. (8) for the year 2016 sets the mandate for the Ministry of Transport and Communication (hereinafter referred to as "MOTC") provides that MOTC has the authority to supervise, regulate and develop the sectors of Information and Communications Technology (hereinafter "ICT") in the State of Qatar in a manner consistent with the requirements of national development goals, with the objectives to create an environment suitable for fair competition, support the development and stimulate investment in these sectors; to secure and raise efficiency of information and technological infrastructure; to implement and supervise e-government programs; and to promote community awareness of the importance of ICT to improve individual's life and community and build knowledge-based society and digital economy.

Article (22) of Emiri Decision No. 8 of 2016 stipulated the role of the Ministry in protecting the security of the National Critical Information Infrastructure by proposing and issuing policies and standards and ensuring compliance.

This guideline has been prepared taking into consideration current applicable laws of the State of Qatar. In the event that a conflict arises between this document and the laws of Qatar, the latter, shall take precedence. Any such term shall, to that extent be omitted from this Document, and the rest of the document shall stand without affecting the remaining provisions. Amendments in that case shall then be required to ensure compliance with the relevant applicable laws of the State of Qatar.

2. Overview

This document provides recommendations and requirements for successful implementation of electronic signature applications by introducing signature creation and verification processes, the main components of signature creation and verification applications. It also presents some key aspects related to the general context of the application (type of signature creation environment, lifetime of the electronic signature, different types of verification of an electronic signature).

3. Introduction

1. Objective of the document

The electronic signature is a process that ensures the authentication of the signer, the integrity of signed electronic documents and non-repudiation. This process makes it possible to reproduce on the computer network, the environment of trust of the physical exchanges (traditional mail), since it becomes impossible to repudiate an electronic document (a purchase order, a payment order...).

This document is a technical guide that specifies recommendations and requirements for the successful implementation of an electronic signature creation and verification application on electronic documents. This guide focuses on four main areas:

- Structure of the electronic signature.

- Electronic signature application environment.

- Electronic signature creation.

- Electronic signature verification.

2. Audience

This document is intended for developers wishing to implement electronic signature creation and verification applications, all organizations and users that intend to use Electronic Signatures. It can also serve anybody wanting to deepen on the aspects of electronic signature.

3. Legal framework of the electronic signature

This section introduces the electronic signature from its legal framework of the state Of Qatar. Law No. 16 of 2011 specifies that the electronic document is proof that it is preserved in its final form by a reliable method and is reinforced by an electronic signature. This law indicates in its article 28 that the electronic signature has the same legal value as the handwritten signature if it fulfills the following conditions:

- The signature creation information are identified with the signer and no other person.
- The signature creation information were, at the time of signing, under the control of the signer and of no other person.
- Any alteration to the electronic signature, made after the time of signing, is detectable.

- Where a purpose of the legal requirement for a signature is to provide assurance as to the integrity of the information to which it relates, any alteration made to that information after the time of signing is detectable.

4. Structure of the electronic signature

The electronic signature consists of the document to be signed, the signature certificate (s), the signed attributes, the value of the electronic signature and unsigned attributes. These components may be mandatory or optional depending on the format of the signature and / or the signature policy. Figure 1 provides a simplified overview of the electronic signature structure. The structure of the electronic signature is detailed in [Structure and electronic signature formats Standards].

1. Signed attributes

Signed (or signature) attributes are components that, when present, must be protected by an electronic signature. These attributes are signed together with the rest of the document content. Signed attributes can be:

- **Content Type**: This attribute ensures that the signed data can be presented to the verifier in the same way as the signer; the signature verification application must present the signer document in the correct format.
- **Message Digest**: Condensed document to sign
- **Signature Policy IDentifier**: Reference to the signature policy that was considered during signature generation
- **Commitment Type**: (In case, a signature policy is considered): it indicates the exact intent of the electronic signature (proof of origin, approval, creation, etc.). The commitment type can be specified in the signing policy.
- **Signing Time**: This is the date and time of signature declared by the signer. The timestamp provided by the time stamp authorities is a reliable source of time information.
- **Signer Location**: It allows the signer to indicate the place where the signer purportedly produced the signature.
- **Signing Certificate** (signer certificate or its digest): the signer certificate can be in the certificate field of the electronic signature structure. To avoid a certificate substitution attack, the identifier of the certificate is included in the signed attributes. This identifier can be either the signature certificate or its digest.
- **SignerRules**: this is the role declared by the signer.

Signed attributes may be required or optional depending on the electronic signature format and signature policy considered (optional).

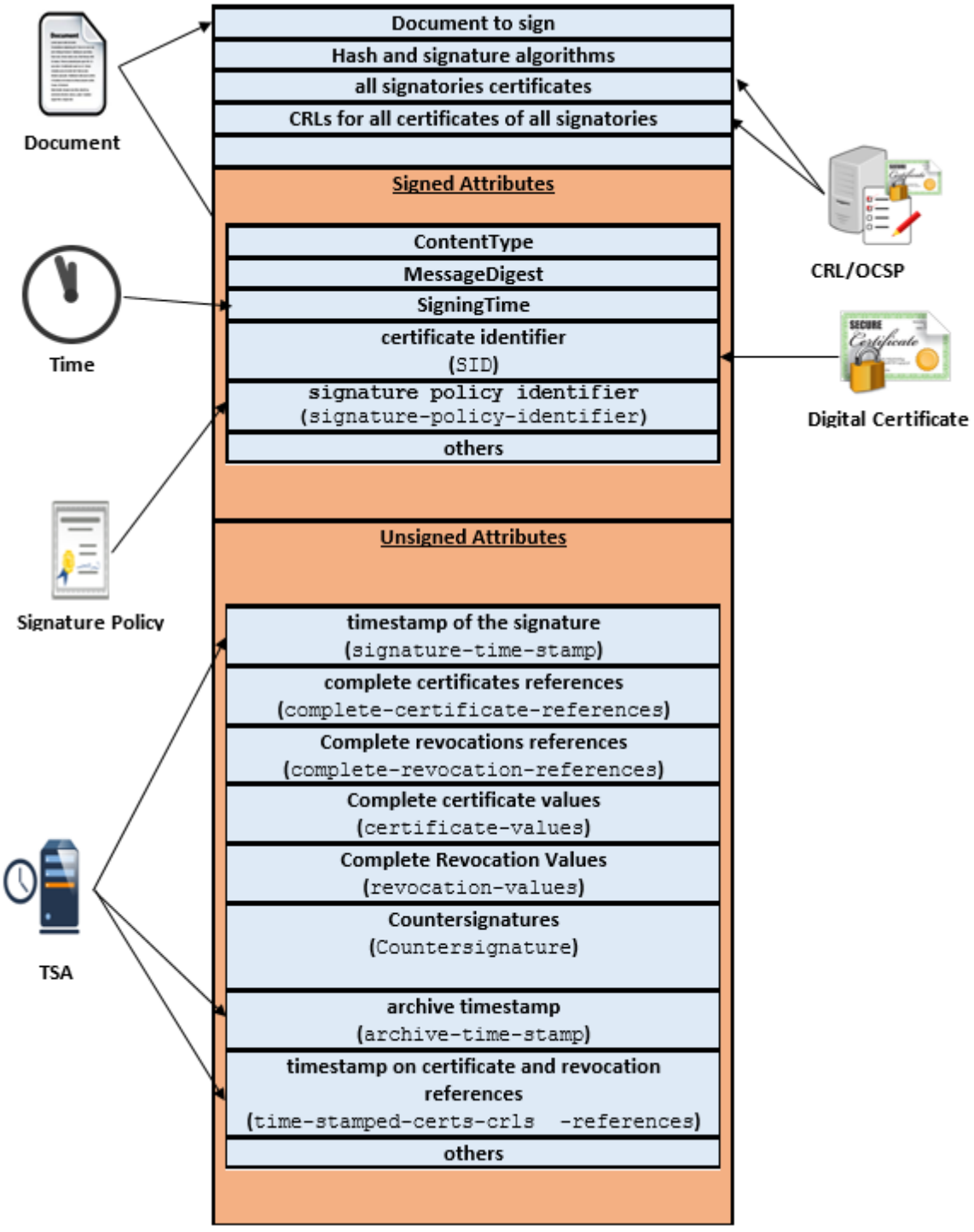

Figure 1: Electronic Signature Structure

2. Unsigned Attributes

Some attributes are not signed with the rest of the message content. These are:

- Timestamp of the signature.
- The complete certificate and revocation references.
- Countersignature that allows the signer to confirm the signed content as well as the previous signature.

3. Electronic Signature Formats

An electronic signature can exist in several formats [Structure and electronic signature formats Standards]:

- **CMS**: Is an IETF standard that defines a syntax for signing or encrypting any digital content. The CMS syntax is based on the ASN.1 [ASN.1] format, which is a formal notation for abstractly describing structured data.
- **CAdES**: Is an extension of the CMS standard. The extension concerns in particular the field of non-repudiation, by defining CMS formats for advanced electronic signatures likely to remain valid for long periods.
- **S / MIME**: Is an electronic signature format for mail encapsulated in MIME format.
- **XMLDSig**: Is defined by the W3C and the IETF. It uses XML as a medium to transport all the data forming the electronic signature. As a result, it benefits from all the flexibility of XML in terms of structuring data.
- **XAdES**: Defined by the W3C and the IETF, is an extension of the XMLDSig format. The extension concerns the field of non-repudiation, by defining XML formats for advanced electronic signatures that can remain valid for long periods.
- **PDF**: Is a format based on the CMS signature.
- **PAdES**: Is a format that is mainly based on the CAdES signature.

4. Multiple Signature

There are two types of multiple signatures that are [Structure and electronic signature formats Standards]:

i. Co-signature

A co-signature is used when all parties agree on the signed content. The validity of other signatures is not verified or confirmed. This type of signature consists of signing a file already signed by other parties with the same hierarchical level (example: a contract between two parties).

ii. Counter-signature

By using a counter-signature, the signer confirms the signed content as well as the previous signature. Indeed, this type of signature requires a hierarchical validation (example: Workflow approval. Employee → Manager → Service Provider).

5. Signature Creation Environment (SCE)

1. Definition

The overall objective of a Signature Creation Application is to generate an Advanced or, where applicable, a Qualified Electronic Signature that covers the signer document (SD), the signer's Certificate or, where applicable, Qualified Certificate (or a reference to it), and, conditionally, the Data Content Type of the SD.

A Signature Creation Environment (SCE) for the creation of Electronic Signatures includes a Signer interacting with a Signature Creation System (SCS) [CEN CWA 14170]. The Signature Creation System contains a Signature Creation Application (SCA), a Signature Creation Device (SCDev) (Or a Secure Signature Creation Device (SSCD) if a Qualified Electronic Signatures is to be created, with an associated Certificate or a Qualified Certificate if a Qualified Electronic Signatures is to be created). Figure 2 shows the overall schema of the signature creation environment.

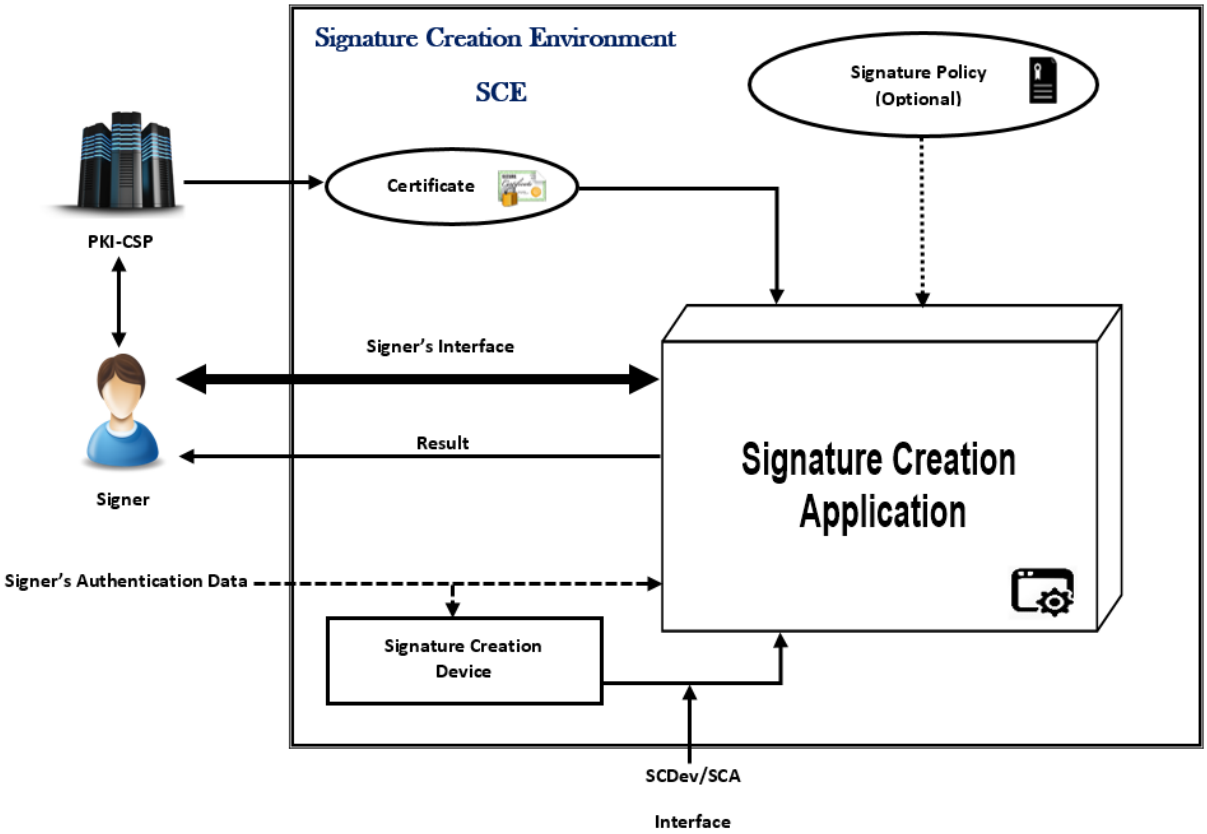

Figure 2: Signature Creation Environment (SCE)

2. Signature Creation Device (SCDev)

The SCDev, which is the signature creation device, may be a microcircuit card, a USB token, or a software component implanted in the host platform itself. Indeed, the electronic signature operation uses the private key associated with the digital certificate. The signature-creation device (SCDev) addressed by this document must be implemented in a separate piece of physical hardware, with its own processing capabilities for PIN code verification and for performing cryptographic functions.

In the case of the use of a microcircuit card with cryptographic capacity, the cryptographic calculations implementing the private signature key and making it possible to generate an electronic signature, are carried out in the SCDev and not in the SCA.

6. Signature Creation Application (SCA)

A signature creation application (SCA) enables the generation of an electronic signature from a document to be signed, the signer's certificate and a signature policy (optional).

1. Electronic Signature Creation Process

The process of creating an electronic signature, which is illustrated in Figure 3, is mainly formed by the following steps that must be performed in the following order:

- Establishment of trusted channels.
- Selection and displaying of the document to sign.
- Authentication of the signer.
- Selection and verification of the signer certificate.
- Displaying / selection of signed and unsigned attributes.
- Signature of the document (in accordance with a signature policy: optional).
- Sending or storing the signed document.
- Clear the memory cache and close established trust channels.

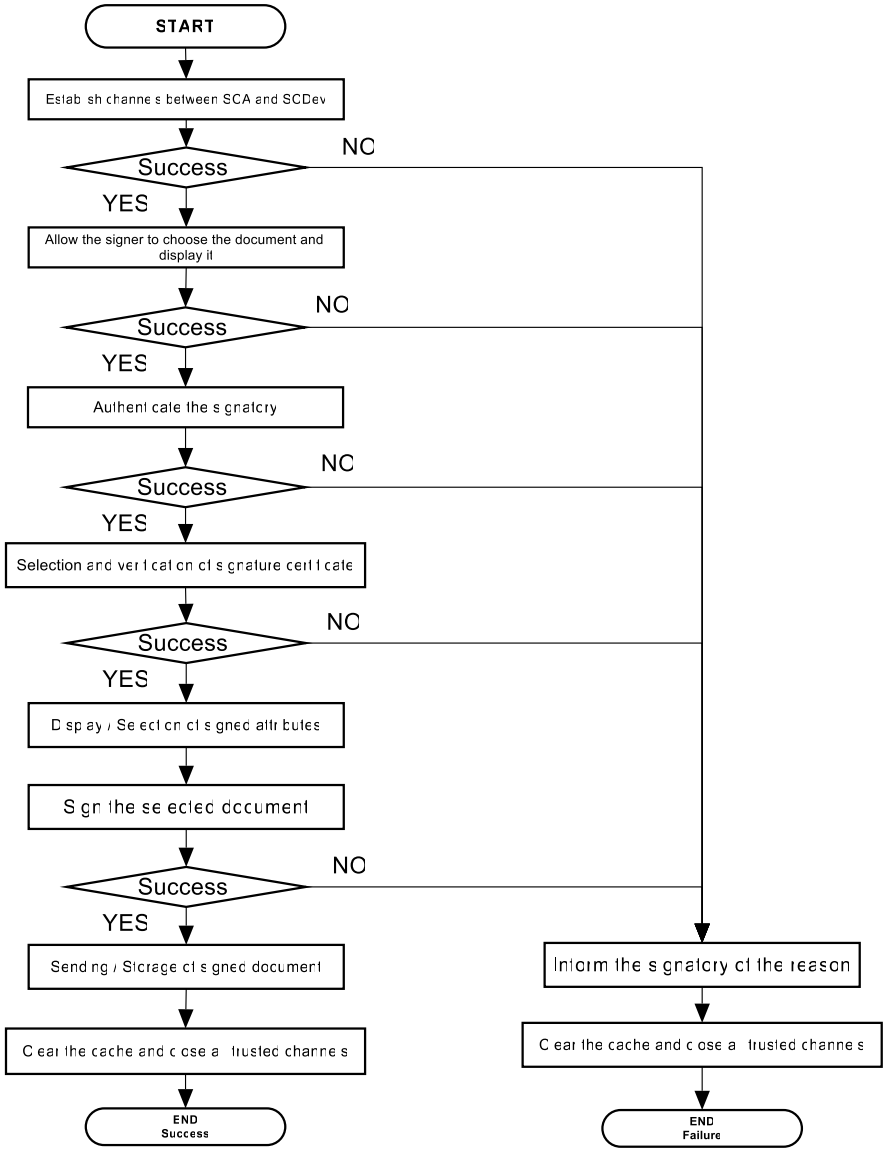

Figure 3: Electronic Signature Creation Process

i. Establishment of trusted channels

The main channels of trust that need to be established are:

- The first for the secure entry of the authentication data of the SCDev (in the case of a Token, the authentication data will be a PIN code).
- The second one for the secure communication of the digest of the document and the value of the signature between the SCA and the SCDev.

If the trust channels can not be established correctly, the signing application must inform the user. In this case, the signing process is interrupted, the signer informed of the reason, the cached information deleted.

The trust channel must:

- Guarantee the confidentiality and integrity of the signer's authentication data while they are in the SCA and when sending and receiving the SCDev.
- Guarantee the integrity of the components of the electronic signature, ie the data to be signed, while they are in the SCA and when sending and receiving the SCDev.

The set of components for establishing the first and second channels may include different software and/or hardware components installed on the operating system such as PKCS # 11 drivers or Cryptographic Service Providers (CSPs) defining a cryptographic interface that the SCA use to access the SCDev.

ii. Selection and displaying of the document to sign

- The SCA must allow the signer to choose the document he wishes to sign.
- The SCA must check if there is a signature attached to the document. If so, it must indicate to the signer what type they are: co-signature or counter-signature. Then, the SCA must ask the signer if he wishes to verify the signature (s) and which he wishes to check when there are several.
  - If the SCA will create a counter-signature and
    - that the signer does not want to verify the existing signature, the signature process must be stopped by itself, the cached information deleted and the trust channels closed.
    - If the signer wishes to verify the existing signature, the SCA must do and indicate the result of the verification. If the signature verification fails, the signing process must be halted by itself, the signer informed of the reason, the cached information deleted, and the trusted channels closed.
  - If the SCA will create a co-signature and
    - If the signer wishes to verify the existing signature, the SCA must do and indicate the result of the verification. If the signature check fails, the signer must decide whether to continue the signing process or abort it.
    - If the signer does not want to verify the existing signature, the application must continue the signing process.
- When the document contains hidden text (white on white, very small fonts, etc.) and / or active content (macros, external links, fields, etc.), the SCA must indicate this to the signer and ask him / her it must be removed from the document before signing.
  - If the signer's response is No, the signing process must be terminated by itself, the signer informed of the reason, the cached information removed, and the trusted channels closed.
  - If the signer's answer is Yes and
    - If no signature is attached to the document, the application must delete the hidden text and / or active content.

- If a signature is already attached to the document, then the signing process must be stopped by itself, the signer informed of the reason, the cached information deleted and the trusted channels closed.

- The document is presented to the signer.
- The signature application must ask the signer if he has read the contents of the document:
  - If the answer is yes, the signer confirms their intention to sign the posted document and proceeds to the next step.
  - Otherwise, the signer has the option to select another document or to abandon the signing process.

iii. Authentication of the signer

- The SCA must ask the signer if he really want to sign the document.
  - If the answer is yes, the signer confirms their intention to sign the posted document and proceeds to the next step.
  - Otherwise, the signing process is interrupted, the signer is informed of the reason, the cached information is deleted, and the trusted channels are closed.
- The signer must then enter his authentication data.
  - If signer authentication fails, the signing process is terminated by itself, the signer is informed of the reason, cached information is deleted, and closed trust channels are closed.
  - If the signer is authenticated, the signing process continues.

iv. Selection and verification of the signer certificate

- The SCA must allow the signer to choose the certificate that he wishes to use for the creation of the electronic signature. It must also display all the necessary information of the selected certificate such as the sender, the subject, the start and end date and the use of the key (Usage key). The SCA must also verify the validity of the certificate and indicate whether it is valid or not.
  - If the selected certificate is not valid, the SCA must allow the signer to choose another certificate, otherwise, the signature process must be halted by itself, the signer informed of the reason, the cached information removed and trusted channels closed.

The verification process of the signing certificate will be detailed in section [**VIII.4.iii**].

v. Displaying / selection of signed and unsigned attributes

- The SCA must allow the display of the signed attributes and may allow them to be selected and modified.
- The SCA can allow the display of unsigned attributes and can allow them to be selected and modified. The number and type of unsigned attributes depends on the format of the electronic signature.

vi. Electronic Signature Creation

The creation of the signature goes through the following steps:

- Creation of the signature structure: This step consists in choosing a signature format while respecting the fact that there are mandatory signed attributes and other optional ones in the case where these attributes will be used.
- Calculating the digest: It depends on the signed attributes. When they are absent, the digest is calculated by considering the document to be signed. In the opposite case, it is also necessary to calculate the digest of the signed attributes.
- Signing the digest: The resulting digest is signed by the signer's private key.
- Finalization of the signature structure: The structure of the electronic signature is created by including the calculated digest (Message Digest), the signed digest (value of the electronic signature) and, possibly, other necessary components such as timestamps, CRL, etc.

Once completed, the SCA must display a message for the signer (Example:"The document has been correctly signed").

vii. Sending or storing the signed document

The signed document is sent to a server or stored locally.

viii. Clearing memory cache and closing trusted channels

The memory cache is cleared and the established trust channels are closed.

2. SCA Components

The components of the SCA are as follows:

- **SDP** [**Signer Document Presentation Component**]: used for presenting the SD that the signer selects by the Signer Interaction Component.
- **SAV** [**Signature Attributes Viewer**]: used for viewing the Signature Attributes that the signer selects by the Signer Interaction Component and which will be signed together with the SD.
- **DTBSF** [**Data To Be Signed Formatter**]: which formats and sequences the SD or a hash of it together with the Signature Attributes and delivers the result to the Data Hashing Component.
- **SIC** [**Signer Interaction Component**]: through which the signer interacts with the SCA to control the signature creation process, and through which the SCA returns error and status messages to the signer. This interface is used for all interactions between the Signer and the SCA, including input/selection of the SD and Signature Attributes except the Signer's Authentication Data;
- **SAC** [**Signer's Authentication Component**] (e.g. a card terminal with PIN pad). This is used for presenting knowledge based Signer's Authentication Data and/or biometric features and preparation of the Signer's Authentication Data in such a way that they can be compared with Signer's Authentication Data held in the SCDev;
- **DHC** [**Data Hashing Component**]: for producing the DTBS Representation (which might be non-hashed, partially hashed or completely hashed as required by the SCDev). If the SCDev carries out all of the hash processing, then the task of this component is only to forward the DTBS Representation unchanged to the SCDev;
- **SSC** [**SCDev/SCA Communicator**] which manages the interaction between SCA and SCDev;
- **SSA** [**SCDev/SCA Authenticator**] which establishes a trusted path between SCDev and SCA. The presence of this component is conditional, i.e. it might only be present in SCAs that are

under the control of public service providers and where the trusted path cannot be established by organizational means.

Other components can be added, in the case of creating electronic signature by considering a signature policy:

- Signature policy Selection component;
- signature policy Administration component.

Figure 4 shows the different components of SCA and their interaction with the SCDev via the communication component with the SCDev that uses software and/or hardware components installed on the operating system (PKCS # 11 drivers, vendors cryptographic services (CSP), etc).

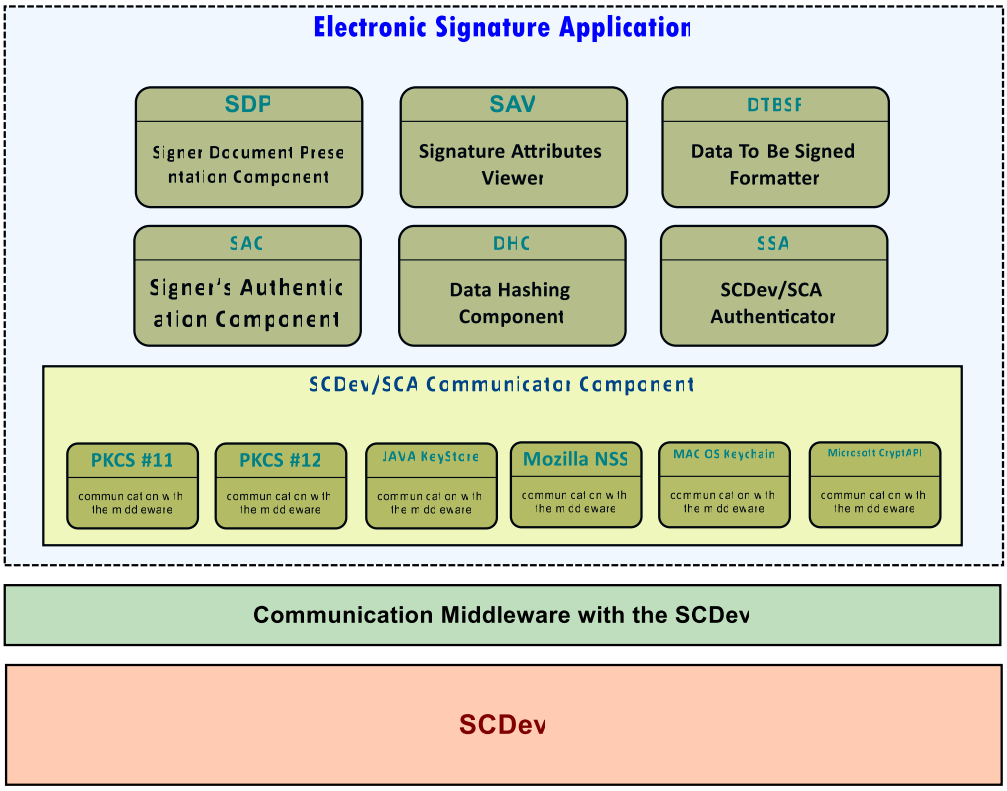

Figure 4: Signature creation application architecture (SCA)

The software components installed on the operating system to communicate with the SCDev for signature capabilities (eg PKCS # 11 drivers or Cryptographic Service Providers (CSPs) defining a cryptographic interface that the electronic signature application calls for access a module that actually generates the signature).

i. Signer Interaction Component

This interface (human-machine interface, API, etc.) allows the signer to:

- Select / de-select one or more documents to sign (including or not a signature);
- Select the attributes to be signed together with the document;
- Select the certificate (and therefore the private key) to use for signing;
- Enter its authentication data allowing the SCDev to activate the signing key;
- Select the signature policy to be applied (optional);
- Abort the signature creation process at any time, before sending the data to be signed to the SCDev;
- Express your consent to sign to ensure that the signer really wants to sign and that this is not the result of an involuntary or accidental action.

ii. Signer Document Presentation Component

The signer must be able to appreciate the content of the electronic document at the time of creation of the electronic signature. This component must allow, on request of the signer, the launch of a presentation application corresponding to the format of the document to be viewed. To do this, the Document Presentation Component manages the correspondence between the document formats that it accepts and the visualization applications.

This component must:

- Ensure that the document presented to the signer is the same as the one selected;
- Notify the signer if the specified content type is not supported by the component of viewing. In this case, it must allow the signer to abort the signing process;
- Notify the signer if all parts of the document can not be presented because of configuration, software, or hardware restrictions. In this case, it must allow the signer to abort the signing process;
- Inform the signer when the document already contains one or more signatures and allow him to see and verify the signature (s) as well as the information on the type of signature (co-signature / counter-signature).

iii. Signature Attributes Viewer

This component allows the signer to view the selected signed attributes before generating the electronic signature.

iv. Data To Be Signed Formatter & Data Hashing Components

These two components format the document to be signed (**DTBS**) as well as the attributes of the signature and hash it (**DHC**) to produce information called "Digest of formatted data to be signed" which will be sent to the SCDev.

These two components must:

- Produce the exact format of the electronic signature according to the signer's choice or signature policy, which contains all signed and unsigned attributes required to create the chosen electronic signature format;
- Ensure that PKI Authority Recommended Algorithms for electronic signature Standard recognizes the hashing and signature algorithms.

v. SCDev/SCA Communicator Component

In order to interact with the SCDev, the communicator component with the SCDev uses software and/or intermediate hardware components (middleware). This component provides the following functions:

- Obtain from the SCDev the references of the certificates usable by the signer;
- Indicate to the SCDev the signature key to activate;
- Transfer the digest data (Message Digest) associated with the signature to the SCDev;
- For each document to be signed, receive from the SCDev the electronic signature as well as the execution exit status relating to the failure or success of the signature creation process.

vi. Signature policy Selection/Administration Component

A signature policy is a set of rules for creating or verifying an electronic signature, according to which a signature can be valide or not. At the time of creation of the signature, a subset of the signature policy must be implemented. This subset defines the minimum requirements for the electronic signature to be accepted. These requirements include requirements on the signer's certificate such as:

- Information concerning the uses of the private key.
- Extensions required for the certificate.
- The types of commitment allowed by this policy.

The signature policy selection component allows, when creating the signature, to consider one or more signature policies stored in electronic form.

The signature policy administration component allows an authenticated administrator to manage all signature policies accepted by the SCA. This component can include the following functions:

- Add a signature policy.
- Removal a signature policy.

3. Signature Creation Environments

There are two types of signature creation environments (**SCE**). These two types of SCEs differ in the following two points:

- Control and possession of the signature creation system (**SCS**).
- Security level measures.

Figure 5 illustrates two distinct cases of control and possession of an SCS in different types of SCE that result in different security measures needed to fulfil the security requirements.

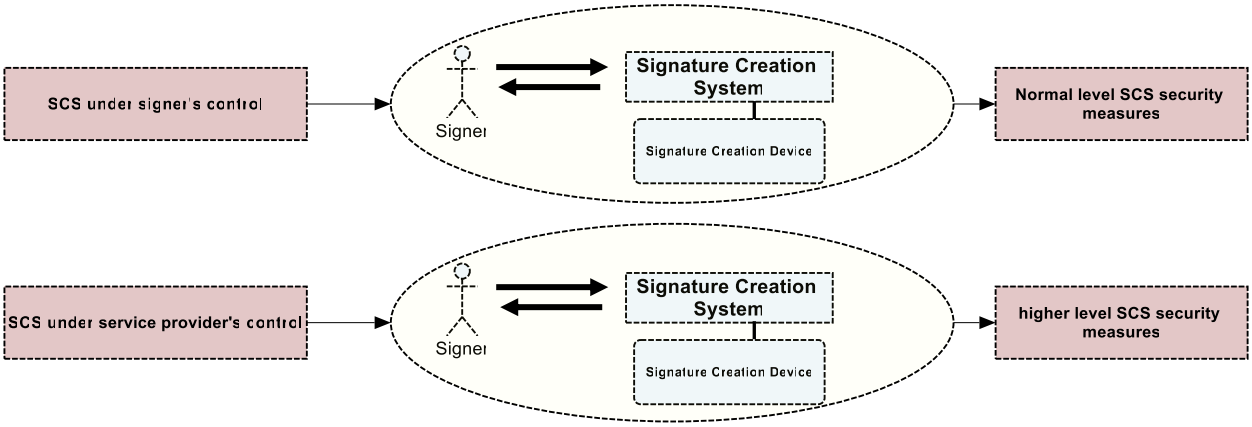

Figure 5: Signature Creation Environments

i. SCS under Signer's Control

A typical environment for the first case might be the home or the office, where the individual or the company has direct control of the SCS (e.g. an SCS implemented in a mobile phone). In this case, the security requirements may be met by organizational methods put in place or managed by the signer, and the technical means to ensure achievement of the security requirements may be more relaxed. For instance, in an extreme case, the Signer can use an isolated PC that is stored in a safe that can only be opened by the signer.

ii. SCS under a Service Provider's Control

A typical environment for the second case is where an SCS is located in a public place such as a railway station, bank or any other SCS that is operated by a service provider that is not necessarily related to or under the control of the signer. Without further technical security measures, this type of environment can suffer a number of other types of attack - e.g. replacement with a fake SCS. The technical requirements of SCSs operated in such public environments will necessarily be more stringent.

7. Signature verification Application (SVA)

1. SVA Environemet (SVE)

An SVA allows verification of an electronic signature generated by an SCA from the signed document and validation data. A signature verification application (SVA) validates an electronic signature against a set of validation constraints and outputs a validation report.

The signature verification environment consists of the following elements:

- SVA.
- The operating system of the physical machine running the SVA.
- Validation data providers.
- The verifier.

Figure 6 illustrates the overall schema of the signature verification environment.

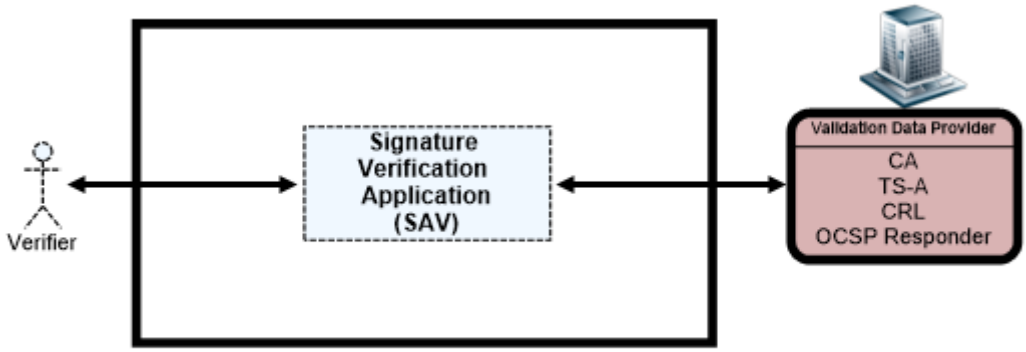

Figure 6: SVA Environment

2. Signature lifetime

The requirements during the verification process, and therefore the data that will be associated with the electronic signature and the type of verification, vary according to the expected life of the signature.

For this purpose, there are three different lifetimes for an electronic signature:

- **Ephemeral signature**: signatures that need not to be kept beyond the next meaningful revocation information issue time. I.e. responds mainly to the case where it must be controlled a few seconds, minutes or even hours after the moment of its creation. This case is found in particular during authenticated exchanges during computer transactions such as electronic payments.
- **Short-term signature**: signatures that are to be verified for a period of time that does not go beyond the signers' certificate expiration date. The electronic signature is systematically invalid beyond the validity period of the signer's certificate. This case can be found for example when signing a contract.
- **Long-term signature**: signatures that are expected to be verified beyond the signers' certificate expiration date and, possibly, even after the expiration date of the certificate of the signers' certificate-issuing CA.

3. Electronic Signature verfication types

There are two types of verification that can be considered on electronic signature:

i. The Initial verification

The initial verification corresponds to a first verification of the electronic signature performed within a time as short as possible after the reception of the electronic signature by the verifier. The target of evaluation is used in order to perform the control of the electronic signature according to a signature policy chosen by the verifier to ensure that the electronic signature is valid. During this operation, the validation data necessary to the verification of the signature are found in the electronic signature or collected by other means. These validation data include a time reference attesting the existence of the digital signature on a specified date. The validity of the other validation data is in particular controlled with respect to this reference date.

ii. The Subsequent verification

The subsequent verification corresponds to a verification of the electronic signature based on the validation data collected during the initial verification by applying a signature policy chosen by the verifier. This verification is performed whereas the time reference (e.g. a time-stamp token) positioning the digital signature in time is still valid. Note: a third type of verification should be considered if the time reference attached to the digital signature and/or the validation data during the initial verification are no more valid. This type of verification would imply a storing and/or a maintenance of the pieces of evidence. This third use case is not covered by this protection profile.

4. Electronic signature verification process

The electronic signature verification process, described in Figure 7, includes the following steps:

- Define if the document includes one or more signatures:
  - If the document contains more than one signature, the verifier must be able to select the signature he wishes to verify.
- Identify and present signature policy to be used during the remainder of the verification process (optional).
- Get the validation data:
  - If the verification is immediate, the SVA must obtain the necessary validation data,
  - If the verification is later, the validation data are in the structure of the electronic signature.
- Check the validity of the signer certificate.
- Check the signature.

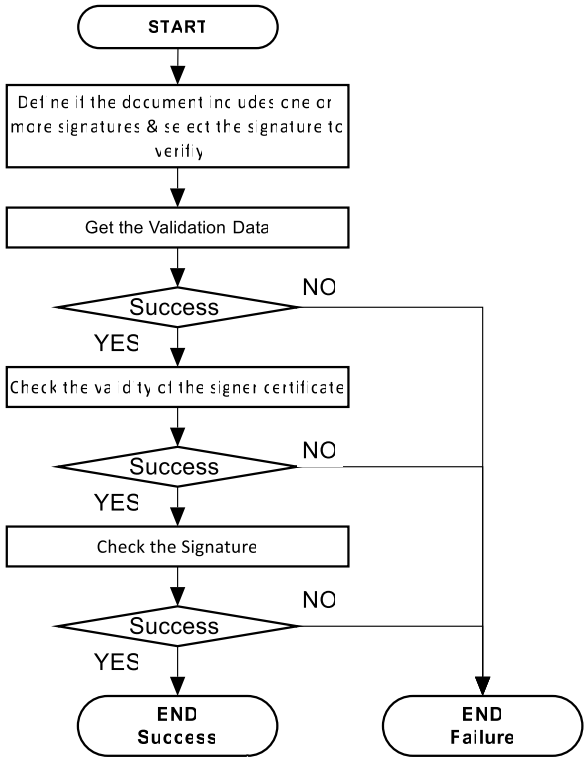

Figure 7: Electronic Signature verification process

i. Signature Policy Identification

The SVA must allow the control of the electronic signature according to a signature policy chosen by the verifier to ensure that the electronic signature is valid. A signature policy is a set of rules for the creation or the validation of electronic signatures, under which electronic signatures can be determined valid. It includes rules defining the signature attributes, which must be provided by the signer.

ii. Getting validation data

In order to validate an electronic signature, additional data must be obtained. They may be partially or fully provided by the SCA depending on the electronic signature format and signature policy. The verifier can also obtain it using the SVA.

The process of obtaining the necessary data includes the following steps:

- **Obtain signer's certificate**: This is an essential component in the signature verification process. It can be included in the structure of the electronic signature. If not, the verifier must be able to obtain it by other means. In order to identify it, the certificate identifier must always be part of the attributes contained in the structure of the electronic signature.
- **Obtain the answers of the CRL / OCSP**: these are essential components for the validation of the certificate of signature and certificates of the chain of certification. The verifier must be sure that the certificate used to create and verify the signature was valid at the time of signature creation or immediate signature verification.
- **Obtain other additional data** (time stamps, etc.) according to the requirements of the signature policy or according to the format of the signature chosen during the creation of the electronic signature.

The following table summarizes the main components of the validation data:

Component	Details
Revocation status information	This information proves that the certificate used by the signer was valid at the time of signing (or is valid at the time of verification). These informations can be obtained either from a CRL or from OCSP.
Certification chain	The certification chain is a chain of certificates that begins with the signer's certificate and ends at the top level with the root CA certificate.
Timestamp	The timestamp, which is obtained from a TSA, allows the dating of the electronic signature.

Table1: Validation data components

Validation data can be obtained in several ways:

- They can be obtained from a remote server (on a local or public network);
- They can be stored locally on the machine where the verification is carried out.
- They can be stored with the signature (depending on the format).

iii. Signer Certificate Validation

The validation process of the signing certificate goes through the following steps:

- Ensure the authenticity of the certificate to avoid a substitution attack of the certificate. This means that the certificate used for the verification should match the certificate identifier contained in the electronic signature.
- Define if the certificate is designed for signing (check the **KeyUsage** of the certificate).
- Determine if the time of verification is within the validity period of the certificate:
    - Secure proof is required at this stage of the verification process, i.e. a timestamp. Without secure proof, the signature, valid for the moment, can be declared invalid in the future due to a certificate expiry.
- Check the revocation status of the certificate.
- Build the certification chain.
- Check the certification chain: Verification of the certification chain can be influenced by the signature policy, which may, for example, require the use of specific root CAs.

iv. Signature Verification

The signature verification phase goes through the following steps:

- Produce a digest from the data related to the signature.
- Decrypt the signature value by the public key contained in the signer's certificate. The result of the decryption gives a digest.
- Compare the two obtained digests: if the two results are the same, the signature is valid.
- Present the signed content to the verifier.

The signing policy may impose additional requirements to the verification process that must be met.

5. SVA Components

The SVA comprises following functional components:

- The component managing the interaction with the verifier.
- The component of selection of the signature policy to be applied.
- The component controlling the invariance of the document's semantics.
- The component executing the viewer applications of documents.
- The component collecting and processing the validation data.
- The component of verification of digital signatures.
- The component of administration of the signature policies.

i. Component managing the interaction with the Verifier

This component will interface with the verifier and allow the following interactions:

- **Selection of the document to be verified by the verifier**: Enabling verifier to indicate which document and which electronic signatures he wishes to verify.
- **Selection of a signature policy to be applied** (Optional): Allowing verifier:
  - To explicitly select a signature policy to be applied (selected signature policy).
  - To use the signature policy referenced in the electronic signature.
- **Communication/presentation of the signature attributes to the verifier:** Allowing the verifier to consult the signature attributes present in the electronic signatures;
- **Communication of the execution status at the end of the verification process:** Enabling to communicate the execution status of the verification process to the verifier ;
- **Communication of the validation data to the verifier**: Enabling to export the validation data used during the verification of the electronic signatures to the verifier. This allows the verifier to backup these data for a later use.

ii. Component of selection of the signature policy to be applied (Optional)

The controls operated by the module of verification depend on a signature policy. The SVA determines the signature policy applied in the following way:

- If a signature policy has been explicitly selected by the verifier, then this selected signature policy will be applied, even if a signature policy is referenced in the electronic signature. If the policy referenced by the electronic signatures is different from the signature policy applied, the SVA informs the verifier.
- If the signature policy has not been defined by the SVA, then the applied signature policy will be the policy referenced in the electronic signature, if such a reference is present. The referenced policy will then be returned to the SVA in order to verify if necessary that the policy is appropriate for the context of operations.
- If no signature policy were preselected and if no signature policy is referenced in the electronic signatures.

iii. Component controlling the invariance of the document's semantics

A document to be signed can contain variable fields or active code which depends on external parameters and which thus can be different according to the context where the document is viewed. A signer could thus sign an electronic document whose contents may vary according to the context where it is viewed. This can mislead the verifier who receives the signature. He could view a document semantically different from the one displayed to the signer. Thus, the contents of the documents must be controlled to attest that its semantics does not depend on external parameters.

iv. Component collecting and processing the validation data

In compliance with the applied signature policy, this component provides the following functions:

- Verification of the compliance of the signed attributes.
- Positioning of the digital signature in time.
- Construction of a valid certification path.
- Verification of the validity of the certification path.

These functions are implemented in an iterative way as long as a valid certification path could not be constructed.

v. Component of verification of digital signatures

This component is a cryptographic component supporting the algorithms (hash and verification of signature) necessary to the verification of the digital signatures implied in the verification process. The digital signatures to be verified are, among others:

- The digital signature of the document.
- Digital signatures contained in the certificates constituting the certification path.
- The digital signature of the auto signed root certificate.
- Digital signatures associated with the collected validation data (CRL, OCSP responses).

vi. Component of administration of the signature policies

This component shall be able to include either none, or some, or all the following functions:

- The addition of a policy;
- The deletion of a policy.

References

- [**CEN CWA 14170**] CEN CWA 14170, Security requirements for signature creation applications, May 2004.
- [**CEN CWA 14171**] CEN CWA 14171, General guidelines for electronic signature verification, May 2004.
- [**DCSSI PP-ACSE**] DCSSI PP-ACSE, Protection profile– Electronic Signature Creation Application, May 2011.
- [**DCSSI PP-MVSE**] DCSSI PP-MVSE, Protection profile– Electronic Signature Verification Module, July 2008.
- [**RFC 2560**] IETF RFC 2560, X.509 Internet Public Key Infrastructure online certificate Status Protocol – OCSP, June 2013.
- [**RFC 5126**] IETF RFC 5126, CMS Advanced Electronic Signatures (CAdES), March 2008.
- [**RFC 3275**] IETF RFC 3275, (Extensible Markup Language) XML-Signature Syntax and Processing, March 2002.
- [**RFC 5652**] IETF RFC 5652, Cryptographic Message Syntax (CMS), September 2009.
- [**RFC 6818**] IETF RFC 8618, Internet X.509 Internet Public Key Infrastructure, Certificate and revocation list (CRL) profile, January 2013.
- [**RFC 5751**] IETF RFC 5751, Secure/Multipurpose Internet Mail Extensions(S/MIME), Version 3.2 Message Specification, January 2010.

- [**ETSI EN 319 132**] ETSI EN 319 132 (Part1, Part2), Electronic Signatures and Infrastructures (ESI), XAdES digital signatures, April 2016.

June 2009.

- [**ETSI EN 319 122**] ETSI EN 319 122 (Part1, Part2), Electronic Signatures and Infrastructures (ESI), CAdES digital signatures, April 2016.
- [**ETSI EN 319 142**] ETSI EN 319 142 (Part1, Part2), Electronic Signatures and Infrastructures (ESI), PAdES digital signatures, April 2016.
- [**ISO 32000-1**] ISO 32000-1:2008, Document management, Portable document format, part 1 : PDF 1.7.

APPENDIX 1

Definitions

Digital Certificate: A document issued by a certification service provider certifying the valid link between a signer identity and its signing key.

Certification Authority (CA): A trusted authority to create and assign electronic certificates.

Time Stamping Authority (TSA): A time stamping authority issues time countermarks on data presented to it. It thus guarantees the date that is affixed to an electronic document.

Electronic signature: Refers to data in electronic form, which is logically associated with other data in electronic form and which is used by the signer to sign (letters, numbers, symbols or others affixed to a data message, which uniquely identify the signer from others in order to indicate the signer's approval on the data message).

Signer: a person that has legal right to access signature creation information, and acts either on its own behalf or on behalf of the person it represents to use the signature creation information to create an electronic signature.

Electronic signature creation data: The signer-specific elements, such as private cryptographic keys, used by him to create an electronic signature.

Electronic signature verification data: Items, such as public cryptographic keys, used to verify the electronic signature.

Timestamp: Service provided by a time stamping authority, where a signed proof of the existence of specific data before a given time.

Revocation list (CRL): Signed list indicating the set of certificates that are no longer considered valid by the issuing CA.

Object Identifier (OID): A suite of electronic or alpha electronic characters, registered in accordance with ISO / IEC 9834, which uniquely identifies an object or class of objects in the envelope of an electronic signature.

Electronic Signature Policy: A signing policy is a set of rules for creating and validating an electronic signature. Indeed, its purpose is to ensure that the signer and the verifier obtain the same results. The signature policy can be explicitly identified in the structure of the electronic signature (using a specific OID) or implicitly defined using external documents or semantics of the signed data.

APPENDIX 2

Acronyms

API: Application Programming Interface
ASN.1: Abstract Syntax Notation One
CA: Certification Authority
CAdES: CMS Advanced Electronic Signature formats
CMS: Cryptographic Message Syntax
CRL: Certification Revocation List
CSP: Cryptographic Service Provider
CEN: European Committee for Standardization
CWA: CEN Workshop Agreements
OASIS DSS: OASIS Digital Signature Services
ETSI: European Telecommunications Standards Institute
IETF: The Internet Engineering Task Force
ISO: International Standards Organization
OCSP: Online Certificate Status Protocol
OID: Object IDentifier
PDF: Portable Document Format
PAdES: PDF Advanced Electronic Signatures
PKCS: Public-Key Cryptography Standards
PKCS#11: An API defining a generic interface for cryptographic device
RFC: Request For Comment
SCA: Signature Creation Application
SCDev: Signature Creation Device
SCE: Signature Creation Environment
SCS: Signature Creation System
S/MIME: Secure / Multipurpose Internet Mail Extensions
SVA: Signature Verification Application
TSA: Time Stamping Authority
XAdES: XML Advanced Electronic Signatures
XML: eXtensible Markup Language
XMLDSig: XML Digital Signature
W3C: World Wide Web Consortium.
WYSIWYS: What You See Is What You Sign

APPENDIX 3

Qualified Electronic Signature

1- Definition

An advanced electronic signature that is created by a qualified electronic signature creation device, and which is based on a qualified certificate for electronic signatures.

2- Qualified electronic signature Creation device

- Definition: An electronic signature creation device that meets the requirements laid down.
- Requirements:
  - **-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.

3- Qualified certificate for electronic signature

- Definition: means a certificate for electronic signatures, which is issued by a qualified trust service provider and meets the requirements laid down.
- Requirements: 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;
- **(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.