NATIONAL SECURITY AUTHORITY

Version 4.0

Formats of certificates and qualified certificates

10 July 2014

This English version of the Slovak document No. 3109/2014/IBEP/OA-001 is for reference purposes only. In case of conflict between the English translation and the original Slovak version, the Slovak version shall prevail and supersedes the English translation as the original version. Therefore, only the NSA Deliverables published by NSA in their original language shall be used for evaluation of products and technical judgement.

NATIONAL SECURITY AUTHORITY

Information Security and Electronic Signature Department

Budatínska 30, 850 07 Bratislava 57

http://www.nbusr.sk/

e-mail: podatelna@nbusr.sk

Contents

1 Introduction

In the Qualified Electronic Seal and Signature verification [1, 2, 6, 9, 10, 16, 18, and 19] (hereinafter referred to as QES) the basic assumption is to verify qualified certificate validity correctly [3, 4, 5, 11, 12, 13, 17, 22 and 23]. In order to verify the qualified certificate unambiguously it is necessary to define unambiguous rules for contents, identification and the qualified certificate use.

2 Scope

The standard "Formats of certificates and qualified certificates" is issued in accordance with Article 3 (1) of the NSA Decree No. 131/2009 Coll. on the format, content and administration of certificates and qualified certificates and the format, periodicity and method of issuing a list of revoked qualified certificates (on certificates and qualified certificates) as amended (hereinafter referred to as the NSA Decree No. 131/2009 Coll.).

The purpose of the present document is to determine technical requirements for individual types of qualified certificates and maintenance certificates to ensure the compatibility and unified environment of the electronic signature in the Slovak Republic with respect to the electronic signature environment particularly in the countries of the European Union.

The document defines the unambiguous format and content of the qualified certificate, the qualified certificate for seal and the maintenance certificate, further it defines the method of the qualified certificate identification pursuant to Article 7 (3) of the Act No. 215/2002 Coll. on Electronic Signature and on the amendment and supplementing of certain acts as amended (hereinafter referred to as the Act).

3 References

References to documents defining used types and procedures.

[1] ETSI TS 101 733 Electronic Signature Formats (CAdES)

[2] ETSI TR 102 272 ASN.1 format for signature policies

[3] RFC 5280 X.509 PKI Certificate and Certificate Revocation List 5-2008

[4] RFC 3739 Qualified Certificates Profile 3-2004

[5] ETSI TS 101 862 Qualified Certificate Profile

[6] RFC 5652 Cryptographic Message Syntax 9-2009

[7] RFC 3161 Time-Stamp Protocol (TSP) 8-2001

[8] RFC 6960 X.509 PKI Online Certificate Status Protocol 6-2013

[9] NSA Qualified Electronic Seal and Signature Formats

[10] ITU-T RECOMMENDATION X.509 | ISO/IEC 9594-8

[11] ETSI TS 119 412-2 Profiles for Trust Service Providers issuing certificates; Part 2: Certificate Profile for certificates issued to natural persons

[12] ETSI TR 102 437 Guidance on TS 101 456

[13] ETSI EN 319 411-2 Policy and security requirements for TSP issuing certificates; Part 2: Policy requirements for CA issuing qualified certificates

[14] ETSI EN 319 411-3 Policy and security requirements for TSP issuing certificates; Part 3: Policy requirements for CA issuing public key certificates

[15] ETSI TS 119 612 Trusted Lists

[16] ETSI TS 101 903 XML Advanced Electronic Signatures (XAdES)

[17] RFC 2560 X.509 PKI Online Certificate Status Protocol 6-1999

[18] CD 2014/148/EU COMMISSION IMPLEMENTING DECISION of 17 March 2014 amending Decision 2011/130/EU establishing minimum requirements for the cross-border processing of documents signed electronically by competent authorities under Directive 2006/123/EC of the European Parliament and of the Council on services in the internal market

[19] ETSI TS 102 778-3 Part 3: PAdES Enhanced – PadES-BES and PAdES-EPES Profiles

[20] ISO/IEC 3166 Codes for the representation of countries

[21] RFC 3647 Internet X.509 PKI Certificate Policy and Certification Practices Framework

[22] CD 2013/662/EU COMMISSION IMPLEMENTING DECISION of 14 October 2013 amending Decision 2009/767/EC as regards the establishment, maintenance and publication of trusted lists of certification service providers supervised/accredited by Member States

[23] Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC

4 Abbreviations

ACA	Accredited Certification Authority
ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation 1
CA	Certification Authority
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List
DER	Distinguished Encoding Rules (for ASN.1)
eIDAS	Regulation (EU) No 910/2014 of the European Parliament and of the Council on electronic identification and trust services for electronic transactions in the internal market
ESS	Enhanced Security Services (enhances CMS)
GMT	Greenwich Mean Time
HTTP	Hyper Text Transfer Protocol
ISO	International Organization for Standardization
LDAP	Lightweight Directory Access Protocol
MIME	Multipurpose Internet Mail Extensions
NSA	National Security Authority
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PAdES	PDF Advanced Electronic Signature
PKCS	Public Key Cryptographic Standards, Standards published by RSA, Labs.
PKIX	Internet X.509 Public Key Infrastructure
QC	Qualified Certificate
QCP SK	Qualified Certificate Policy of Slovakia
SSCD	Secure-Signature-Creation Device
TSA	Time-Stamping Authorities
TSP	Time Stamp Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
XML	eXtensible Markup Language
QES	Qualified Electronic Signature or Qualified Electronic Seal

5 A type of a certificate and identity of a person in a certificate

Pursuant to Article 3 (8) of the NSA Decree No.131/2009 Coll. qualified certificates and maintenance certificates must contain an identifier of the certificate policy of accredited certification services, i.e. OID with the value QCP SK '1 3 158 36061701 0 0 0 1 2 2'. This identifier must be indicated in the certificate policy extension of the certificate. This identifier is also the identifier of the certificate policy which may be indicated only in qualified certificates and maintenance certificates being issued by CAs accredited by the NSA. Certificates of an end user in the extension of the certificate policy whose OID is '1 3 158 36061701 0 0 0 1 2 2' should contain in the item *userNotice-explicitText* (*utf8String* format) a text with the information on the certificate type according to the Slovak legislation written in English as well as Slovak languages with the introductory text "EN:" and "SK:" containing mainly: "A certificate is issued as a **qualified** certificate / a qualified certificate of **mandate** / a qualified certificate **for seal** / a **maintenance** certificate pursuant to the Act No. 215/2002 Coll. and the NSA Decree No. 131/2009 Coll." The content of the certificate policy is defined in the Annex 1 of the NSA Decree No.133/2009 Coll. and the recommendation for the certificate policy content is indicated in RFC 3647.
The qualified certificate of the person associates the identity of the private key owner with the public key used to verify its seals and signatures while **all data contained in the qualified certificate have been verified as valid in the time of the certificate issuance by the certification service provider**. These data are stored mainly in items of the subject name *Certificate-tbsCertificate-subject-RelativeDistinguishedName*.

The content of the item *commonName* is informative and provides short information on the name and possibly the certificate type for a verifier and a signatory. The item *commonName* must be included and must occur only once.

5.1 Types of qualified certificates

A qualified certificate of a physical person pursuant to Article 3 (4) of the NSA Decree No. 131/2009 Coll. must contain as a minimum:

- a given name in *givenName*, and a surname in *surname* or a pseudonym in *pseudonym*, and
- an additional identifier in *serialNumber* in the form of "PNO" ensuring unambiguity of identity data of a subject to whom a qualified certificate was issued; it may also contain "IDC" and "PAS".

A certificate of mandate pursuant to Article 3 (5) of the NSA Decree No. 131/2009 Coll. in addition to the above defined items which are identifying a mandatary must also contain:

- Identity data of a person on whose behalf a mandatary acts (hereinafter referred to as the mandator) where a name **"MANDANT"** written in capital letters must be placed in front of mandator's data in each item of a mandator (following the initial characters of the type, e. g. "PASSK- MANDANT"):
  - identity data of a public body: a name in *organizationName* and one kind of identity information of "VAT", "NTR" or "SZ:" type at least in one *serialNumber* or
  - identity data of a person indicated in items *givenName*, *surname* and at least in one item *serialNumber* (one kind of information in the form of "PNO"); it may also contain "IDC" and "PAS".

- Identity data of a public body or a person where a mandatary performs his/her activities according to a special regulation or performs his/her function according to a special regulation are defined in the item *organizationName* and at least in one item *serialNumber* containing one kind of identity information of "VAT", "NTR" or "SZ:" type (pursuant to Article 7 (3), letter c) of the Act).

- Indication of authorization pursuant to Article 10a (2), letter a) in connection with Article 7 (4) of the Act is provided as the last number of OID (1.3.158.36061701.1.1.xyz) of the certificate policy in the certificate extension. The optional item *userNotice-explicitText* contains the *utf8String* with 200 characters as a maximum of the authorization being published on the NSA website in the list of authorizations. It is recommended to provide the number and the authorization text in both languages: in English with the introductory text "EN:" and in Slovak with the introductory text "SK:" in the item *userNotice-explicitText*. Management of the content of the list of authorizations is not a subject of the present document and therefore the details can be found on the NSA website http://www.nbusr.sk/sk/elektronicky-podpis/zoznam-opravneni.1.html. In the item *cPSuri* there is a reference to "Certification Practice Statement" (CPS) which is containing information on methods used for verification of the authorization being identified by OID according to a list of documents used in proof of this authorization and is included in the list of authorizations on the NSA website. Accredited Certification Authority can have more documents of CPS which indicate the steps of how ACA proceeds when issuing the certificate for authorization being identified by OID, whereas it refers to authorizations and requirements published in the NSA list.

A qualified certificate for seal pursuant to Article 3 (6) of the NSA Decree No. 131/2009 Coll. must contain a name, as a minimum, in the item *organizationName* and one identity kind of information of "VAT", "NTR" or "SZ:" type at least in one item *serialNumber*. The certificate for seal must not contain *givenName*, *surname* or *pseudonym* (what definitely differentiates it from qualified certificates and certificates of mandate issued to a physical person).

The reference to the person's identity in the item *serialNumber* must be indicated in the format consisting of **two parts** which are separated by one blank space (ASCII character 0x20) or by a dash "-" (ASCII character 0x2D). The dash "-" instead of the character "blank space" must be present in the item *serialNumber* in certificates which will be issued since 1 September 2014, whilst only one reference to the person's identity must be indicated in one item *serialNumber* and the qualified certificate must contain, as a minimum, one item *serialNumber* with the reference to the person's identity.

The first part of the item *serialNumber* consists of 3 initial characters specifying the type of the reference to the identity.
Three initial characters identify the type of the reference to the identity:
1. "PAS" for identification based on passport number,
2. "IDC" for identification based on identity card number,
3. "PNO" for identification based on personal number; Slovak citizens and foreigners have the personal number assigned pursuant to Act No.301/1995 Coll. on personal number,
4. "SZ:" for identification based on a set of characters assigned pursuant to Article 27 (4) of the Act No. 540/2001 Coll. on state statistics as amended by Act No. 55/2010 Coll.,
5. "VAT" for identification based on tax identification number,
6. "NTR" for identification based on company registration number.

The following two characters containing the country code according to ISO 3166 (for Slovakia "SK") identify the country which issued the data used in the second part. These data are provided on the basis of legislative requirements defined in the respective country.

The second part of the item *serialNumber* consists of data whose type is determined by 3 initial characters.

Examples of the content of *serialNumber*: "PASSK P3000180", "IDCSK-SP989783", "SZ:SK-MANDANT 123123" or "SZ:SK-123123".

In case of "PNO" the personal number which is composed of numbers from a decimal system without the slash between the first and second part of the personal number is used according to Article 5 (2) of the Act. The personal number shall have 10 or 9 places (referring to personal numbers assigned to by 31 December 1953) pursuant to Act No.301/1995 Coll. on personal number (examples: *serialNumber*: "PNOSK 9959199999", "PNOSK 535919999", "PNOSK-MANDANT 535919999").

5.2 Recommendations

Accredited certification service providers are recommended to use, as a minimum, one of the above mentioned references to person's identity when issuing the qualified certificate in compliance with the legislation of the Slovak Republic not excluding the case when the certificate will not be used in communication with public bodies.

Each qualified certificate of a physical person should also contain an **identity card number** or a **passport number** used in registration process when issuing the certificate (recommendation due to preparation of a secondary legislation to a Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC).

An example of the identity card number:
IDCSK-P6553199999 - serialNumber(2.5.4.5)

Data in the certificate should contain only necessary data, e.g. a permanent address should be optional.

If SSCD or an application for signature/seal creation contains or will contain more certificates of the same type or other certificate types, the user may decide which one he/she will use based on the text in the item *commonName*. The item *commonName* is informative and should begin with the following name tag:
The name tag in the form of ZZZX, where ZZZ are 3 capital letters and X is a distinguishing number, if there are more same certificates on the device, for example a successive certificate.

ZZZ acquires, as a minimum, the following values:

- for qualified "QES" (Qualified Electronic Signature, Qualified Electronic Seal),
- for encrypted "ENC" (encryption) and
- for authenticated "AUT" (authentication).

Examples:
QES J. C. Cruellas - *commonName*(2.5.4.3)
QES2 J. C. Cruellas - *commonName*(2.5.4.3)
ENC J. C. Cruellas - *commonName*(2.5.4.3)
AUT J. C. Cruellas - *commonName*(2.5.4.3)

The certificate of mandate may contain a name tag "MANDATÁR" (mandatary) followed by an abridged name of the mandatary and a name tag "OPRÁVNENIE" (authorization) followed by an authorization number and possibly by an abridged text of the authorization from the list of authorizations published on the NSA website.
An example:
MANDATÁR J. C. Cruellas OPRÁVNENIE 346 Legal Representative -commonName(2.5.4.3)

6 Formats of qualified certificates and maintenance certificates

Table 1. Basic format of X.509 qualified certificate

	Record in ASN.1	Short description
1.	`Certificate ::= SEQUENCE {`	
2.	`tbsCertificate TBSCertificate,`	Certificate data signed by CA.
3.	`signatureAlgorithm AlgorithmIdentifier,`	Signature algorithm identifier and algorithm parameters if required by the algorithm. Certification Authority uses the algorithm for signing *tbsCertificate*.
4.	`signature BIT STRING }`	Certificate signature.

Table 2. TBSCertificate

	Record in ASN.1	Short description
1.	`TBSCertificate ::= SEQUENCE {`	
2.	`version [0] EXPLICIT Version DEFAULT v1,`	Certificate version must be in v3 (value 2).
3.	`serialNumber CertificateSerialNumber,`	It is a positive serial number of the certificate with a maximum size of 20 bytes. $1 \leq \text{serialNumber} < 2^{159}$
4.	`signature AlgorithmIdentifier,`	Exactly the same content as in Table 1, line 3.
5.	`issuer Name,`	It is a name of the certificate issuer (CA). *Issuer* together with *serialNumber* (line 3) must identify the issued certificate **unambiguously.** The issuer name must always contain *countryName* where CA has its residence and the name *organizationName*. Certificates issued for a new key pair of the issuer shall contain one kind of information of the "VAT", "NTR", "SZ:", "PAS", "PNO" or "IDC" type in one item *serialNumber*, as a minimum. From 1 September 2014 it will be required for each certificate issued for a new key pair. The name must consist of attributes, in particular *countryName*, *organizationName*, *organizationalUnitName*, *distinguishedNameQualifier*, *stateOrProvinceName*, *commonName*, *serialNumber* and *domainComponent*. It can also consist of other attributes such as *localityName, title, surname, givenName, initials, pseudonym* and *generationQualifier*. The text is non-empty in items *DirectoryString* and *UTF8String* coding must be used.
6.	`validity Validity,`	It is a period of the certificate validity (from, to). The format is initiated in *UTCTime* and it must be encoded in *GeneralizedTime* (YYYYMMDDhhmmssZ) from 2050. The format must contain seconds and is expressed in Zulu time zone (Universal Coordinated Time).
7.	`subject Name,`	The name of a subject to whom a certificate was

		issued (DN to whom the certificate is issued) must be unambiguous in CA during the whole CA existence. The same name can be also used in other certificates issued for the same person but it must not be used in certificates issued for another person. The reference to person's identity of a private key owner, to whom the certificate was issued, is placed in the item *seriaNumber* and the format is defined in chapter 5. *Name* must contain *countryName*, *serialNumber*, *commonName*, for a physical person *(surname* and *givenName)* or *pseudonym;* if it is not a physical person then *organizationName. Name* contains items in particular: *countryName, commonName*, *surname*, *givenName*, *pseudonym, serialNumber, organizationName, organizational-UnitName, stateOrProvincename, localityName* and *title.* If *pseudonym* is initiated in *Name,* then *surname* and *givenName* must not be initiated. If the pseudonym is initiated in the item *commonName*, then the word PSEUDONYM must be appended after or before the given pseudonym. *Rfc822Name* should not be used in *Name* but it is necessary to use an attribute from a certificate extension *subjectAltNames*. The text is non-empty in items *DirectoryString* and *UTF8String* coding must be used.
8.	`subjectPublicKeyInfo SubjectPublicKeyInfo,`	It is the public key of a subject to whom a certificate was issued being in DER coding and the algorithm identifier for which this key is intended.
9.	`issuerUniqueID [1] IMPLICIT UniqueIdentifier OPTIONAL,`	Not used.
10.	`subjectUniqueID [2] IMPLICIT UniqueIdentifier OPTIONAL,`	Not used.
11.	`extensions [3] EXPLICIT Extensions OPTIONAL }`	Certificate extensions.

Table 3. Name

	Record in ASN.1	**Short description**
1.	`Name ::= CHOICE { RDNSequence }`	Components of *Name*.
2.	`RDNSequence ::= SEQUENCE OF RelativeDistinguishedName`	
3.	`RelativeDistinguishedName ::= SET OF AttributeTypeAndValue`	
4.	`AttributeTypeAndValue ::= SEQUENCE { type AttributeType, value AttributeValue }`	
5.	`AttributeType ::= OBJECT IDENTIFIER`	
6.	`AttributeValue ::= ANY DEFINED BY AttributeType`	

Table 4. DirectoryString for text storing

	Record in ASN.1	Short description
1.	`DirectoryString ::= CHOICE {`	*UTF8String* must be used.
2.	`teletexString TeletexString (SIZE (1..MAX)),`	
3.	`printableString PrintableString (SIZE (1..MAX)),`	It can be used in case of backward compatibility need and the chain can be presented through *PrintableString* coding in a character set without the loss of information (without diacritics), otherwise *UTF8String* text coding must be used.
4.	`universalString UniversalString (SIZE (1..MAX)),`	
5.	`utf8String UTF8String (SIZE (1..MAX)),`	It is a mandatory format apart from the exception given in line 3.
6.	`bmpString BMPString (SIZE (1..MAX)) }`	

Table 5. X.501 attributes used e.g. in Name

	Name	OID	ASN.1 type	Max size
1.	commonName	{id-at 3}	DirectoryString	64
2.	surName	{id-at 4}	DirectoryString	64
3.	givenName	{id-at 42}	DirectoryString	64
4.	serialNumber	{id-at 5}	PrintableString	64
5.	title	{id-at 12}	DirectoryString	64
6.	organizationName	{id-at 10}	DirectoryString	64
7.	organizationalUnitName	{id-at 11}	DirectoryString	64
8.	businessCategory	{id-at 15}	DirectoryString	128
9.	streetAddress	{id-at 9}	DirectoryString	128
10.	postalCode	{id-at 17}	DirectoryString	40
11.	localityName	{id-at 7}	DirectoryString	128
12.	stateOrProvinceName	{id-at 8}	DirectoryString	128
13.	countryName	{id-at 6}	PrintableString (SIZE(2))	2 ISO 3166 code
14.	distinguishedNameQualifier	{id-at 46}	PrintableString	64
15.	initials	{id-at 43}	DirectoryString	64
16.	generationQualifier	{id-at 44}	DirectoryString	64
17.	emailAddress	{pkcs-9 1}	IA5String	128
18.	domainComponent	{0 9 2342 19200300100 1 25}	IA5String	Description in [RFC2247]
19.	postalAddress	{id-at 16}	SEQUENCE SIZE(1..6) OF DirectoryString	6 x 30 in [RFC3739] 1. Street Number 2. Zip Code Locality 3. State
20.	pseudonym	{id-at 65}	DirectoryString	64
21.	dateOfBirth	{id-pda 1}	GeneralizedTime	YYYYMMDD000000Z
22.	placeOfBirth	{id-pda 2}	DirectoryString	128
23.	gender	{id-pda 3}	PrintableString (SIZE(1))	"M" "F"
24.	countryOfCitizenship	{id-pda 4}	PrintableString (SIZE(2))	2 ISO 3166 code
25.	countryOfResidence	{id-pda 5}	PrintableString (SIZE(2))	2 ISO 3166 code
26.	nameAtBirth	{id-isismtt-at 14}	DirectoryString	64
27.	telephoneNumber	{id-at 20}	PrintableString (SIZE(32))	32 ITU-T Rec. E.123 "+44 582 10101"

Table 6. GeneralName

	Record in ASN.1	Short description
1.	`GeneralName ::= CHOICE {`	
2.	`  otherName  [0] IMPLICIT OtherName,`	It is used for data identification of other type than types used below.
3.	`  rfc822Name [1] IMPLICIT IA5String,`	It is the e-mail address according to the format "addr-spec" [RFC2822] without “< >” in the form "local-part@domain"
4.	`  dNSName     [2] IMPLICIT IA5String,`	It is the Internet domain name specified in [RFC1034]
5.	`  x400Address       [3] IMPLICIT ORAddress,`	(it is not used) X400 address specified in ITU-T X.411
6.	`  directoryName     [4] EXPLICIT Name,`	X500 name
7.	`  ediPartyName      [5] IMPLICIT EDIPartyName,`	(it is not used) The name from Electronic Data Exchange System.
8.	`  uniformResourceIdentifier[6] IMPLICIT IA5String,`	URI specified in [RFC1630] contains the Uniform Resource Names (URNs) and also URLs. (http:// ...) Permitted URL formats are specified in [RFC1738] and [RFC2255].
9.	`  iPAddress  [7] IMPLICIT OCTET STRING,`	It is the IP address in IPv4 [RFC791] or in IPv6 [RFC2460] format
10.	`  registeredID     [8] IMPLICIT OBJECT IDENTIFIER }`	(it is not used) A registered OBJECT IDENTIFIER (identifying e.g. the organization)

11.	`OtherName ::= SEQUENCE {`
	`  type-id    OBJECT IDENTIFIER,`
	`  value [0] EXPLICIT ANY DEFINED BY type-id`
	`}`
		Defined above.

12.	`EDIPartyName ::= SEQUENCE {`
	`  nameAssigner     [0] EXPLICIT DirectoryString OPTIONAL,`
	`  partyName  [1] EXPLICIT DirectoryString`
	`}`
		Defined above.

Table 7. Extension - certificate extensions

	Record in ASN.1	Short description
1.	`Extensions  ::=  SEQUENCE SIZE (1..MAX) OF Extension`	It is a non-empty list of certificate extensions.
2.	`Extension  ::=  SEQUENCE  {`	
3.	`  extnID      OBJECT IDENTIFIER,`	It is OID specifying the extendable information type.
4.	`  critical    BOOLEAN DEFAULT FALSE,`	If there is TRUE, then the extension is critical and the application must know how to process the information, otherwise the certificate is refused.
5.	`  extnValue   OCTET STRING` `}`	It is the DER coded value of extendable information.

Table 8. Certificate extensions

	X.509 basic extensions	OID	Short description	Critical
1.	AuthorityKeyIdentifier	{2 5 29 35}	It is the public key identifier of CA which issued the certificate.	MUST NOT
2.	SubjectKeyIdentifier	{2 5 29 14}	It is the public key identifier of a subject to whom the certificate was issued.	MUST NOT
3.	KeyUsage	{2 5 29 15}	It defines the purpose of a private key whose public key is a part of the certificate.	MUST
4.	PrivateKeyUsagePeriod	{2 5 29 16}	It enables to determine another private key usage period than is the usage period of the certificate.	MUST NOT
5.	CertificatePolicies	{2 5 29 32}	It identifies certificate policies under which the certificate was issued. Qualified certificates and maintenance certificates issued by CA being accredited by the NSA must also contain OID *QCP SK* (1 3 158 36061701 0 0 0 1 2 2).	MAY
6.	PolicyMappings	{2 5 29 33}	It confirms that issuing CA considers its policy to be equivalent to CA policy for which the certificate is issued.	MUST NOT
7.	SubjectAltNames	{2 5 29 17}	It is the alternative (technical) name of a subject to whom the certificate was issued, e. g. OtherName, e-mail, DNS name, IP address, URI, etc.	SHOULD NOT
8.	IssuerAltNames	{2 5 29 18}	It is the alternative (technical) name of the certificate issuer, e.g. OtherName, e-mail, DNS name, IP address, URI, etc.	SOULD NOT
9.	SubjectDirectoryAttributes	{2 5 29 9}	The extension contains more detailed X.500 attributes of a subject to whom the certificate was issued.	MUST NOT
10.	BasicConstraints	{2 5 29 19}	It identifies CA certificate. The certification path can be limited in the CA certificate; e.g. **zero** means that CA issues only user certificates.	MUST in CA certificate.
11.	NameConstraints	{2 5 29 30}	CA determines the scope of changes in certificate names that must be verified in all	MUST

			subject names (or alternative names) in the succeeding certificates in the certification path.	
12.	PolicyConstraints	{2 5 29 36}	It can be used in CA certificates for limiting of certification path validation.	MUST
13.	ExtendedKeyUsage	{2 5 29 37}	It determines more accurately the purpose of the public key usage in the certificate. It is verified independently of *KeyUsage* certificate extension.	MAY But it MUST be in TSP certificate.
14.	CRLDistributionPoints	{2 5 29 31}	It determines the way and where from it is possible to obtain CRL.	SHOULD NOT
15.	AuthorityInfoAccess	{1 3 6 1 5 5 7 1 1}	It determines (http:// … p7c, cer or also ldap://...) the address for obtaining certificates issued for the certificate issuer and the OCSP address.	MUST NOT
16.	SubjectInfoAccess	{1 3 6 1 5 5 7 1 11}	Services which are provided by subject. It determines (http:// … p7c, cer or also ldap://...) the address for obtaining CA certificates issued by the certificate subject and the address for obtaining a time stamp.	MUST NOT
	RFC3739 QC extensions			
17.	BiometricInfo	{1 3 6 1 5 5 7 1 2}	It includes references to biometric information with the purpose of confirming the validity, e.g. the address for the picture of a subject to whom the certificate was issued and hash (digital fingerprint) from the picture.	MUST NOT
18.	QCStatements	{1 3 6 1 5 5 7 1 3}	A certificate statement to confirm that the certificate is a qualified certificate pursuant to a particular technical standard and contains limitations on qualified certificate usage.	SHOULD NOT

Table 9. End User certificate extensions

	Extension	Presence	The scope of values
1.	AuthorityKeyIdentifier	MUST	It must be the same as initiated in *subjectKeyIdentifier* of the certificate issuer. *AuthorityKeyIdentifier.keyIdentifier = SubjectKeyIdentifier* and if the certification path is created ambiguously, then it must also contain authorityCertIssuer and authorityCertSerialNumber.
2.	*SubjectKeyIdentifier*	MUST	20 byte hash SHA1 from *BIT STRING subjectPublicKeyInfo* (without *tag, length* and unused bits) is recommended (it is calculated from generated public key and is used as a supporting index (ambiguous – possible SHA-1 collision should not cause errors in the applications).
3.	*KeyUsage*	MUST	a) End entity: Only *nonRepudiation* bit must be in qualified certificates. If the certificate is determined for other purposes (e. g. for integrity verification or for maintenance), then *digitalSignature* bit can be also set. For maintenance: b) CRL: If the certificate is issued for signing of *indirect* CRL, then it only contains *crlSign* bit. c) OCSP: If the certificate is issued for signing of OCSP, then it only contains *nonRepudiation* bit and in the *ExtendedKeyUsage* extension it must only contain *id-kp-OCSPSigning.* d) TSP: If the certificate is issued for signing of TSP, then it only contains *nonRepudiation* bit and in the *ExtendedKeyUsage* extension it must only contain *id-kp-timeStamping.* In these cases the BIT STRING DER coding is presented as one individual octet (because *decipherOnly* is not set) due to unambiguity.
4.	*PrivateKeyUsagePeriod*	MAY	In accordance with RFC 5280.
5.	*CertificatePolicies*	MUST	For the end user of the qualified certificate it also determines the purpose for which the certificate is issued. If it contains *UserNotice,* then the application must be able to display it during the certificate usage. Maintenance certificates and qualified certificates issued by CA being accredited by the NSA must contain the certificate policy *QCP SK* (1 3 158 36061701 0 0 0 1 2 2) as a minimum. If the documents CPS and CP are in PDF, they should be secured according to ETSI TS 102 778-3 Part 3 [19].
6.	*PolicyMappings*	MUST NOT	In accordance with RFC 5280.
7.	*SubjectAltNames*	MAY	In accordance with RFC 5280.
8.	*IssuerAltNames*	MAY	In accordance with RFC 5280.
9.	*SubjectDirectory-Attributes*	MAY	In this extension qualified certificates CAN contain identification data issued by the state office (verified for example according to ID card or passport, etc.) and also data which are not found in the item *Subject*, e.g. *dateOfBirth, placeOfBirth,gender, countryOfCitizenship, countryOfResidence* defined in RFC 3739 and *nameAtBirth*.
10.	*BasicConstraints*	MAY	In accordance with RFC 5280.
11.	*NameConstraints*	MUST	In accordance with RFC 5280.

		NOT	
12.	*PolicyConstraints*	MUST NOT	In accordance with RFC 5280.
13.	*ExtendedKeyUsage*	MAY	In accordance with RFC 5280. In TSP certificate it must only contain *id-kp-timeStamping*. In OCSP certificate it must only contain *id-kp-OCSPSigning.*
14.	*CRLDistributionPoints*	MUST	It must contain HTTP address and can contain LDAP address but in that case it must also contain a "host" address of LDAP server at least in one LDAP address. How to fill out items for *direct* CRL and *indirect* CRL is defined in the NSA document "Formats of certificate revocation list and confirming the status and validity of certificates". In certificates in the NSA root subtree the *direct* CRL for the certificate issuer must precede the *indirect CRL* for the NSA certificate that issues CRL for certificates issued by accredited CA.
15.	*AuthorityInfoAccess*	MUST (The field DOES NOT HAVE TO be present in *self signed root* certificates)	If the signature does not contain a complete certification path while verifying the signature, then CA certificates for certificate signature verification, stored in HTTP address in ".p7c" or ".cer" file, are necessary for certification path creation. The file ".p7c"is the empty CMS signature containing a list of certificates that can be used for the certificate verification. The address for CA certificates must contain HTTP address and can contain LDAP address but in that case it must also contain the internet address of LDAP server at least in one LDAP address. It can also contain HTTP address for OCSP (Online Certificate Status Protocol).
16.	*subjectInfoAccess*	MAY	Information about the subject. So far it is used only in CA certificates.
17.	*BiometricInfo*	MAY	In accordance with RFC 3739.
18.	*QCStatements*	MUST	It is the unambiguous identification of the qualified certificate type of the end user for QES (Qualified Electronic Signatures) creation by means of OID in *QCStatements* in accordance with RFC 3739 and especially with ETSI TS 101 862. The qualified certificate **must** contain OID identifier *id-etsi-qcs-QcCompliance* and **if** the qualified certificate **contains** the certificate policy OID *QCP SK* (1 3 158 36061701 0 0 0 1 2 2), it should also contain *id-etsi-qcs-QcSSCD*. OID *id-etsi-qcs-QcSSCD* has been obligatorily required since 1 July 2010 until a new OID according to eIDAS regulation, which shall repeal the Directive on electronic signature, replaces it. The certificate for seal may contain OID *id-etsi-qcs-QcSSCD*. The qualified certificate may contain the limits on transaction values *etsi-qcs-QcLimitValue*, for which the certificate can be used. Significance of OID identifiers: • *esi4-qcStatement-1 (id-etsi-qcs-QcCompliance)* a

			certificate is a qualified certificate. • *esi4-qcStatement-4 (id-etsi-qcs-QcSSCD)* a qualified certificate contains a public key corresponding to a private key which is stored in a secure device SSCD while the private key cannot be exported from the secure device and is under a sole control of a person to whom the qualified certificate was issued. • *esi4-qcStatement-2 (etsi-qcs-QcLimitValue)* a limit on transaction value An example: SEQUENCE { PrintableString 'EUR' - - 978, ISO 4217 Currency Code INTEGER 1 - - amount INTEGER 2 - - exponent } - - value = amount * 10^exponent

Annex A (informative) Examples of qualified certificates

A.1 Example of the end user's qualified certificate

```
   0 1596: SEQUENCE {
   4 1316:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2    -- version
         :       }
  13    2:     INTEGER 3088   -- serialNumber
  17   13:     SEQUENCE {     -- signature
  19    9:       OBJECT IDENTIFIER
         :         sha256WithRSAEncryption (1 2 840 113549 1 1 11)
  30    0:       NULL
         :       }
  32   83:     SEQUENCE {     -- issuer
  34   11:       SET {
  36    9:         SEQUENCE {
  38    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
  43    2:           PrintableString 'EU'
         :           }
         :         }
  47   22:       SET {
  49   20:         SEQUENCE {
  51    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
  56   13:           UTF8String 'Test locality'
         :           }
         :         }
  71   26:       SET {
  73   24:         SEQUENCE {
  75    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
  80   25:           UTF8String 'Narodny bezpecnostny urad'
         :           }
         :         }
  99   16:       SET {
 101   14:         SEQUENCE {
 103    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 108    7:           UTF8String 'Test CA'
         :           }
         :         }
 258   21:       SET {
 260   19:         SEQUENCE {
 262    3:           OBJECT IDENTIFIER serialNumber (2 5 4 5)
                                    -- organization unique identification data
 267   14:           PrintableString 'NTRSK-36061701'
         :           }
         :         }
         :       }
 117   30:     SEQUENCE {
 119   13:       UTCTime 21/11/2006 15:21:16 GMT -- notBefore
 134   13:       UTCTime 21/11/2007 15:21:16 GMT -- notAfter
         :       }
 149  129:     SEQUENCE {     -- subject
 152   11:       SET {
 154    9:         SEQUENCE {
 156    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 161    2:           PrintableString 'EU'
         :           }
         :         }
 165   22:       SET {
 167   20:         SEQUENCE {
 169    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
 174   13:           UTF8String 'Test locality'
```

```
         :           }
         :         }
 189   20:       SET {
 191   18:         SEQUENCE {
 193    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 198   15:           UTF8String 'QES Peter Rybar'
         :           }
         :         }
 211   14:       SET {
 213   12:         SEQUENCE {
 215    3:           OBJECT IDENTIFIER surname (2 5 4 4)
 220    5:           UTF8String 'Rybar'
         :           }
         :         }
 227   14:       SET {
 229   12:         SEQUENCE {
 231    3:           OBJECT IDENTIFIER givenName (2 5 4 42)
 236    5:           UTF8String 'Peter'
         :           }
         :         }
 243   13:       SET {
 245   11:         SEQUENCE {
 247    3:           OBJECT IDENTIFIER title (2 5 4 12)
 252    4:           UTF8String 'Ing.'
         :           }
         :         }
 258   21:       SET {
 260   19:         SEQUENCE {
 262    3:           OBJECT IDENTIFIER serialNumber (2 5 4 5)
                             -- Personal unique identification data
 267   16:           PrintableString 'PNOSK 1234567889'
         :           }
         :         }
         :       }
 281  290:     SEQUENCE {     -- subjectPublicKeyInfo
 285   13:       SEQUENCE {
 287    9:         OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
 298    0:         NULL
         :         }
 300  271:       BIT STRING, encapsulates {
 305  266:         SEQUENCE {
 309  257:           INTEGER
         :             00 8A 32 3D 82 25 5D 31 DB CD 59 8B A2 8C 82 56
         :             0D F5 DE 0F 5A 3F 83 4F 88 41 62 7E 27 56 A6 88
         :             D8 8F CB D8 07 65 EE 32 3C C3 E8 46 15 3C F6 00
         :             C8 A8 67 43 1E CD 1D BF 32 DB 2B EC 33 BC 63 2D
         :             50 4E 1C 76 66 E7 88 C5 68 58 87 ED E1 DF 46 26
         :             BC 21 76 BF 91 33 54 0F BE 45 82 92 8D 41 31 1D
         :             A8 83 8E F1 EB 57 2A 5F 53 DA F9 FE 3A 28 CD FE
         :             25 CE E3 FA BD 0D 0E 9E DA 06 C6 93 CC 0D CF FF
         :                     [ Another 129 bytes skipped ]
 570    3:           INTEGER 65537
         :           }
         :         }
         :       }
 575  745:     [3] {          -- extensions
 579  741:       SEQUENCE {
 583   31:         SEQUENCE {
 585    3:           OBJECT IDENTIFIER authorityKeyIdentifier (2 5 29 35)
 590   24:           OCTET STRING, encapsulates {
 592   22:             SEQUENCE {
 594   20:               [0]
         :                 38 B3 D9 00 A5 F6 81 B9 4B 0D 9B 2A DB 4D 65 67
         :                 B1 A5 1B CC
```

```
        :                 }
        :               }
        :             }
616   29:           SEQUENCE {
618    3:             OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
623   22:             OCTET STRING, encapsulates {
625   20:               OCTET STRING
        :                 38 94 23 FE 2B 7E 2C C8 5B 1E E3 D4 19 87 C6 54
        :                 6A 93 BC B0
        :               }
        :             }
647   14:           SEQUENCE {
649    3:             OBJECT IDENTIFIER keyUsage (2 5 29 15)
654    1:             BOOLEAN TRUE   -- critical
657    4:             OCTET STRING, encapsulates {
659    2:               BIT STRING 6 unused bits
        :                 '10'B (bit 1)    -- nonRepudiation
        :               }
        :             }
663   36:           SEQUENCE {
665    3:             OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
670   29:             OCTET STRING, encapsulates {
672   27:               SEQUENCE {
674    8:                 SEQUENCE {       -- QCP public + SSCD
676    6:                   OBJECT IDENTIFIER '0 4 0 1456 1 1'
        :                   }
684   15:                 SEQUENCE {       -- QCP SK
686   13:                   OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
        :                   }
        :                 }
        :               }
        :             }
701    9:           SEQUENCE {
703    3:             OBJECT IDENTIFIER basicConstraints (2 5 29 19)
708    2:             OCTET STRING, encapsulates {
710    0:               SEQUENCE {}  -- end entity certificate
        :               }
        :             }
712  265:           SEQUENCE {
716    3:             OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
721  256:             OCTET STRING, encapsulates {
725  253:               SEQUENCE {
728   40:                 SEQUENCE {
730   38:                   [0] {
732   36:                     [0] {
734   34:                       [6] 'http://qes.test.eu/test/testca.crl'
        :                       }
        :                     }
        :                   }
770  109:                 SEQUENCE {
772  107:                   [0] {
774  105:                     [0] {
776  103:                       [6]
        :                       'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
        :                       'n,l=Test locality,c=EU?certificateRevocationList'
        :                       ';binary'
        :                       }
        :                     }
        :                   }
881   98:                 SEQUENCE {
883   96:                   [0] {
885   94:                     [0] {
887   92:                       [6]
        :                       'ldap:///cn=Test CA,o=Test organization,l=Test lo'
```

```
         :                    'cality,c=EU?certificateRevocationList;binary'
         :                      }
         :                    }
         :                  }
         :                }
         :              }
         :            }
 981  303:          SEQUENCE {
 985    8:            OBJECT IDENTIFIER authorityInfoAccess (1 3 6 1 5 5 7 1 1)
 995  289:            OCTET STRING, encapsulates {
 999  285:              SEQUENCE {
1003   36:                SEQUENCE {
1005    8:                  OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
1015   24:                  [6] 'http://ocsp.test.eu/ocsp'
         :                  }
                           -- certificate and cross-certificates
1041   46:                SEQUENCE {
1043    8:                  OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1053   34:                  [6] 'http://qes.test.eu/test/testca.p7c'
         :                  }
1089  103:                SEQUENCE {
1091    8:                  OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1101   91:                  [6]
         :                    'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
         :                    'n,l=Test locality,c=EU?caCertificate;binary'
         :                  }
1194   92:                SEQUENCE {
1196    8:                  OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1206   80:                  [6]
         :                    'ldap:///cn=Test CA,o=Test organization,l=Test lo'
         :                    'cality,c=EU?caCertificate;binary'
         :                  }
         :                }
         :              }
         :            }
1288   34:          SEQUENCE {
1290    8:            OBJECT IDENTIFIER qcStatements (1 3 6 1 5 5 7 1 3)
1300   22:            OCTET STRING, encapsulates {
1302   20:              SEQUENCE {
1304    8:                SEQUENCE {      -- etsi-qcs-QcCompliance
1306    6:                  OBJECT IDENTIFIER '0 4 0 1862 1 1'
         :                  }
1314    8:                SEQUENCE {      -- etsi-qcs-QcSSCD
1316    6:                  OBJECT IDENTIFIER '0 4 0 1862 1 4'
         :                  }
         :                }
         :              }
         :            }
         :          }
         :        }
         :      }
1324   13:    SEQUENCE {      -- signatureAlgorithm
1326    9:      OBJECT IDENTIFIER
         :        sha256WithRSAEncryption (1 2 840 113549 1 1 11)
1337    0:      NULL
         :      }
1339  257:    BIT STRING      -- signatureValue
         :      A3 64 00 CC E5 ED 4A 20 5E D4 AD D9 E3 49 76 53
         :      3E 22 8D EB 5E B2 D3 53 B6 FB F2 58 21 4A 66 8C
         :      41 BC 6A D2 58 AC 1D 6A 09 F2 0C 1A 52 8E 67 15
         :      6B F0 4F AB 98 A9 C8 85 A4 19 1F 17 06 2D F1 45
         :      93 FC 7A 97 D3 1D 75 F3 3E E0 DA 5C 3D 50 02 4C
         :      68 7B AD 3B DB 92 6B 62 DC 65 64 50 98 F8 6B 55
         :      25 4B EC D6 6C 49 8E 7A 0A B8 C2 8E 2E 43 1D 52
```

```
        :     3F 37 A8 CC C6 FE D0 03 FC 55 87 A4 65 82 68 8E
        :             [ Another 128 bytes skipped ]
        :   }
```

A.2 Example of the qualified certificate of mandate

```
SEQUENCE {
  SEQUENCE {
    [0] {
      INTEGER 2  -- version
      }
    INTEGER 30882      -- serialNumber
    SEQUENCE {   -- signature
      OBJECT IDENTIFIER
        sha256WithRSAEncryption (1 2 840 113549 1 1 11)
      NULL
      }
    SEQUENCE {   -- issuer
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER countryName (2 5 4 6)
          PrintableString 'EU'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER localityName (2 5 4 7)
          UTF8String 'Test locality'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER organizationName (2 5 4 10)
          UTF8String 'Narodny bezpecnostny urad'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER commonName (2 5 4 3)
          UTF8String 'Test CA'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER serialNumber (2 5 4 5)
                    -- organization unique identification data
          PrintableString 'NTRSK-36061701'
          }
        }
      }
    SEQUENCE {
      UTCTime 21/11/2006 15:21:16 GMT -- notBefore
      UTCTime 21/11/2007 15:21:16 GMT -- notAfter
      }
    SEQUENCE {   -- subject
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER countryName (2 5 4 6)
          PrintableString 'EU'
          }
        }
      SET {
        SEQUENCE {
```

```
          OBJECT IDENTIFIER localityName (2 5 4 7)
          UTF8String 'Test locality'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER commonName (2 5 4 3)
          UTF8String 'MANDATÁR Peter Rybar OPRÁVNENIE 346 Zákonný zástupca'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER surname (2 5 4 4)
          UTF8String 'MANDANT Pekar'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER givenName (2 5 4 42)
          UTF8String 'MANDANT Jan'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER title (2 5 4 12)
          UTF8String 'MANDANT Ing.'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER surname (2 5 4 4)
          UTF8String 'Rybar'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER givenName (2 5 4 42)
          UTF8String 'Peter'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER title (2 5 4 12)
          UTF8String 'Ing.'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER serialNumber (2 5 4 5)
                    -- personal unique identification data
          PrintableString 'PNOSK-1234567889'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER serialNumber (2 5 4 5)
                    -- personal unique identification data
          PrintableString "PNOSK-MANDANT 535919999"'
          }
        }
      }
    SEQUENCE {   -- subjectPublicKeyInfo
      SEQUENCE {
        OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
```

```
        NULL
        }
      BIT STRING, encapsulates {
        SEQUENCE {
          INTEGER
            00 8A 32 3D 82 25 5D 31 DB CD 59 8B A2 8C 82 56
            0D F5 DE 0F 5A 3F 83 4F 88 41 62 7E 27 56 A6 88
            D8 8F CB D8 07 65 EE 32 3C C3 E8 46 15 3C F6 00
            C8 A8 67 43 1E CD 1D BF 32 DB 2B EC 33 BC 63 2D
            50 4E 1C 76 66 E7 88 C5 68 58 87 ED E1 DF 46 26
            BC 21 76 BF 91 33 54 0F BE 45 82 92 8D 41 31 1D
            A8 83 8E F1 EB 57 2A 5F 53 DA F9 FE 3A 28 CD FE
            25 CE E3 FA BD 0D 0E 9E DA 06 C6 93 CC 0D CF FF
                    [ Another 129 bytes skipped ]
          INTEGER 65537
          }
        }
      }
    [3] {       -- extensions
      SEQUENCE {
        SEQUENCE {
          OBJECT IDENTIFIER authorityKeyIdentifier (2 5 29 35)
          OCTET STRING, encapsulates {
            SEQUENCE {
              [0]
                38 B3 D9 00 A5 F6 81 B9 4B 0D 9B 2A DB 4D 65 67
                B1 A5 1B CC
              }
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
          OCTET STRING, encapsulates {
            OCTET STRING
              38 94 23 FE 2B 7E 2C C8 5B 1E E3 D4 19 87 C6 54
              6A 93 BC B0
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER keyUsage (2 5 29 15)
          BOOLEAN TRUE      -- critical
          OCTET STRING, encapsulates {
            BIT STRING 6 unused bits
              '10'B (bit 1)  -- nonRepudiation
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
          OCTET STRING, encapsulates {
            SEQUENCE {
              SEQUENCE {          -- QCP public + SSCD
                OBJECT IDENTIFIER '0 4 0 1456 1 1'
                }
              SEQUENCE {          -- QCP SK
                OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
                  SEQUENCE {
                    SEQUENCE {
                      OBJECT IDENTIFIER unotice (1 3 6 1 5 5 7 2 2)
                      SEQUENCE {
                        UTF8String 'EN: Qualified certificate of mandate
pursuant to Act No. 215/2002 Coll. and Decree No. 131/2009 Coll. SK:
Kvalifikovaný mandátny certifikát podľa zákona č. 215/2002 Z. z. a vyhlášky č.
131/2009 Z. z.'
                        }
```

```
                        }
                      }
                    }
                  SEQUENCE {           -- Authorisation 346 Legal Representative
                    OBJECT IDENTIFIER '1 3 158 36061701 1 1 346'
                      SEQUENCE {
                        SEQUENCE {
                          OBJECT IDENTIFIER cps (1 3 6 1 5 5 7 2 1)
                          IA5String http://qes.test.eu/resources/cps1_346
                          }
                        SEQUENCE {
                          OBJECT IDENTIFIER unotice (1 3 6 1 5 5 7 2 2)
                          SEQUENCE {
                            UTF8String 'EN: Authorization 346, Legal
Representative, pursuant to Article 10a of the Act No. 215/2002 Coll. SK:
Oprávnenie 346, Zákonný zástupca, podľa § 10a zákona č. 215/2002 Z. z.'
                            }
                          }
                        }
                    }
                  }
                }
              }
            SEQUENCE {
              OBJECT IDENTIFIER basicConstraints (2 5 29 19)
              OCTET STRING, encapsulates {
                SEQUENCE {}      -- end entity certificate
                }
              }
            SEQUENCE {
              OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
              OCTET STRING, encapsulates {
                SEQUENCE {
                  SEQUENCE {
                    [0] {
                      [0] {
                        [6] 'http://qes.test.eu/test/testca.crl'
                        }
                      }
                    }
                  SEQUENCE {
                    [0] {
                      [0] {
                        [6]
                      'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
                      'n,l=Test locality,c=EU?certificateRevocationList'
                      ';binary'
                        }
                      }
                    }
                  SEQUENCE {
                    [0] {
                      [0] {
                        [6]
                      'ldap:///cn=Test CA,o=Test organization,l=Test lo'
                      'cality,c=EU?certificateRevocationList;binary'
                        }
                      }
                    }
                  }
                }
              }
            SEQUENCE {
              OBJECT IDENTIFIER authorityInfoAccess (1 3 6 1 5 5 7 1 1)
```

```
          OCTET STRING, encapsulates {
            SEQUENCE {
              SEQUENCE {
                OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
                [6] 'http://ocsp.test.eu/ocsp'
                }
           -- certificate and cross-certificates
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6] 'http://qes.test.eu/test/testca.p7c'
                }
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6]
                  'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
                  'n,l=Test locality,c=EU?caCertificate;binary'
                }
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6]
                  'ldap:///cn=Test CA,o=Test organization,l=Test lo'
                  'cality,c=EU?caCertificate;binary'
                }
              }
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER qcStatements (1 3 6 1 5 5 7 1 3)
          OCTET STRING, encapsulates {
            SEQUENCE {
              SEQUENCE {          -- etsi-qcs-QcCompliance
                OBJECT IDENTIFIER '0 4 0 1862 1 1'
                }
              SEQUENCE {          -- etsi-qcs-QcSSCD
                OBJECT IDENTIFIER '0 4 0 1862 1 4'
                }
              }
            }
          }
        }
      }
    }
  SEQUENCE {          -- signatureAlgorithm
    OBJECT IDENTIFIER
      sha256WithRSAEncryption (1 2 840 113549 1 1 11)
    NULL
    }
  BIT STRING          -- signatureValue
    A3 64 00 CC E5 ED 4A 20 5E D4 AD D9 E3 49 76 53
    3E 22 8D EB 5E B2 D3 53 B6 FB F2 58 21 4A 66 8C
    41 BC 6A D2 58 AC 1D 6A 09 F2 0C 1A 52 8E 67 15
    6B F0 4F AB 98 A9 C8 85 A4 19 1F 17 06 2D F1 45
    93 FC 7A 97 D3 1D 75 F3 3E E0 DA 5C 3D 50 02 4C
    68 7B AD 3B DB 92 6B 62 DC 65 64 50 98 F8 6B 55
    25 4B EC D6 6C 49 8E 7A 0A B8 C2 8E 2E 43 1D 52
    3F 37 A8 CC C6 FE D0 03 FC 55 87 A4 65 82 68 8E
            [ Another 128 bytes skipped ]
  }
```

A.3 Example of the qualified certificate for seal

```
 SEQUENCE {
   SEQUENCE {
```

```
[0] {
  INTEGER 2  -- version
  }
INTEGER 30883      -- serialNumber
SEQUENCE {   -- signature
  OBJECT IDENTIFIER
    sha256WithRSAEncryption (1 2 840 113549 1 1 11)
  NULL
  }
SEQUENCE {   -- issuer
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER countryName (2 5 4 6)
      PrintableString 'EU'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER localityName (2 5 4 7)
      UTF8String 'Test locality'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER organizationName (2 5 4 10)
      UTF8String 'Narodny bezpecnostny urad'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER commonName (2 5 4 3)
      UTF8String 'Test CA'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER serialNumber (2 5 4 5)
                -- organisation unique identification data
      PrintableString 'NTRSK-36061701'
      }
    }
  }
SEQUENCE {
  UTCTime 21/11/2006 15:21:16 GMT -- notBefore
  UTCTime 21/11/2007 15:21:16 GMT -- notAfter
  }
SEQUENCE {   -- subject
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER countryName (2 5 4 6)
      PrintableString 'EU'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER localityName (2 5 4 7)
      UTF8String 'Test locality'
      }
    }
  SET {
    SEQUENCE {
      OBJECT IDENTIFIER commonName (2 5 4 3)
      UTF8String 'Podatelna'
      }
```

```
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER organizationName (2 5 4 10)
          UTF8String 'Narodny bezpecnostny urad'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER organizationalUnitName (2 5 4 11)
          UTF8String 'SIBEP'
          }
        }
      SET {
        SEQUENCE {
          OBJECT IDENTIFIER serialNumber (2 5 4 5)
                    -- unique identification data
          PrintableString 'NTRSK-36061701'
          }
        }
      }
    SEQUENCE {   -- subjectPublicKeyInfo
      SEQUENCE {
        OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
        NULL
        }
      BIT STRING, encapsulates {
        SEQUENCE {
          INTEGER
            00 8A 32 3D 82 25 5D 31 DB CD 59 8B A2 8C 82 56
            0D F5 DE 0F 5A 3F 83 4F 88 41 62 7E 27 56 A6 88
            D8 8F CB D8 07 65 EE 32 3C C3 E8 46 15 3C F6 00
            C8 A8 67 43 1E CD 1D BF 32 DB 2B EC 33 BC 63 2D
            50 4E 1C 76 66 E7 88 C5 68 58 87 ED E1 DF 46 26
            BC 21 76 BF 91 33 54 0F BE 45 82 92 8D 41 31 1D
            A8 83 8E F1 EB 57 2A 5F 53 DA F9 FE 3A 28 CD FE
            25 CE E3 FA BD 0D 0E 9E DA 06 C6 93 CC 0D CF FF
                    [ Another 129 bytes skipped ]
          INTEGER 65537
          }
        }
      }
    [3] {        -- extensions
      SEQUENCE {
        SEQUENCE {
          OBJECT IDENTIFIER authorityKeyIdentifier (2 5 29 35)
          OCTET STRING, encapsulates {
            SEQUENCE {
              [0]
                38 B3 D9 00 A5 F6 81 B9 4B 0D 9B 2A DB 4D 65 67
                B1 A5 1B CC
              }
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
          OCTET STRING, encapsulates {
            OCTET STRING
              38 94 23 FE 2B 7E 2C C8 5B 1E E3 D4 19 87 C6 54
              6A 93 BC B0
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER keyUsage (2 5 29 15)
```

```
            BOOLEAN TRUE      -- critical
            OCTET STRING, encapsulates {
              BIT STRING 6 unused bits
                '10'B (bit 1)  -- nonRepudiation
              }
            }
          SEQUENCE {
            OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
            OCTET STRING, encapsulates {
              SEQUENCE {
                SEQUENCE {          -- QCP public + SSCD
                  OBJECT IDENTIFIER '0 4 0 1456 1 1'
                  }
                SEQUENCE {          -- QCP SK
                  OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
                    SEQUENCE {
                      SEQUENCE {
                        OBJECT IDENTIFIER unotice (1 3 6 1 5 5 7 2 2)
                        SEQUENCE {
                          UTF8String 'EN: A certificate is issued as a qualified
certificate for seal pursuant to the Act No. 215/2002 Coll. and the NSA Decree
No. 131/2009 Coll. SK: Kvalifikovaný systémový certifikát podľa zákona č.
215/2002 Z. z. a vyhlášky č. 131/2009 Z. z.'
                          }
                        }
                      }
                  }
                }
              }
            }
          SEQUENCE {
            OBJECT IDENTIFIER basicConstraints (2 5 29 19)
            OCTET STRING, encapsulates {
              SEQUENCE {}      -- end entity certificate
              }
            }
          SEQUENCE {
            OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
            OCTET STRING, encapsulates {
              SEQUENCE {
                SEQUENCE {
                  [0] {
                    [0] {
                      [6] 'http://qes.test.eu/test/testca.crl'
                      }
                    }
                  }
                SEQUENCE {
                  [0] {
                    [0] {
                      [6]
                    'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
                    'n,l=Test locality,c=EU?certificateRevocationList'
                    ';binary'
                      }
                    }
                  }
                SEQUENCE {
                  [0] {
                    [0] {
                      [6]
                    'ldap:///cn=Test CA,o=Test organization,l=Test lo'
                    'cality,c=EU?certificateRevocationList;binary'
                      }
```

```
                }
              }
            }
          }
        }
        SEQUENCE {
          OBJECT IDENTIFIER authorityInfoAccess (1 3 6 1 5 5 7 1 1)
          OCTET STRING, encapsulates {
            SEQUENCE {
              SEQUENCE {
                OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
                [6] 'http://ocsp.test.eu/ocsp'
                }
             -- certificate and cross-certificates
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6] 'http://qes.test.eu/test/testca.p7c'
                }
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6]
                  'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
                  'n,l=Test locality,c=EU?caCertificate;binary'
                }
              SEQUENCE {
                OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
                [6]
                  'ldap:///cn=Test CA,o=Test organization,l=Test lo'
                  'cality,c=EU?caCertificate;binary'
                }
              }
            }
          }
        SEQUENCE {
          OBJECT IDENTIFIER qcStatements (1 3 6 1 5 5 7 1 3)
          OCTET STRING, encapsulates {
            SEQUENCE {
              SEQUENCE {          -- etsi-qcs-QcCompliance
                OBJECT IDENTIFIER '0 4 0 1862 1 1'
                }
              SEQUENCE {          -- etsi-qcs-QcSSCD
                OBJECT IDENTIFIER '0 4 0 1862 1 4'
                }
              }
            }
          }
        }
      }
    }
  SEQUENCE {          -- signatureAlgorithm
    OBJECT IDENTIFIER
      sha256WithRSAEncryption (1 2 840 113549 1 1 11)
    NULL
    }
  BIT STRING          -- signatureValue
    A3 64 00 CC E5 ED 4A 20 5E D4 AD D9 E3 49 76 53
    3E 22 8D EB 5E B2 D3 53 B6 FB F2 58 21 4A 66 8C
    41 BC 6A D2 58 AC 1D 6A 09 F2 0C 1A 52 8E 67 15
    6B F0 4F AB 98 A9 C8 85 A4 19 1F 17 06 2D F1 45
    93 FC 7A 97 D3 1D 75 F3 3E E0 DA 5C 3D 50 02 4C
    68 7B AD 3B DB 92 6B 62 DC 65 64 50 98 F8 6B 55
    25 4B EC D6 6C 49 8E 7A 0A B8 C2 8E 2E 43 1D 52
    3F 37 A8 CC C6 FE D0 03 FC 55 87 A4 65 82 68 8E
           [ Another 128 bytes skipped ]
```

```
   }
```

A.4 Example of the time stamp certificate for Qualified Electronic Signature

```
   0 1541: SEQUENCE {
   4 1261:   SEQUENCE {
   8    3:     [0] {
  10    1:       INTEGER 2    -- version
         :       }
  13    1:     INTEGER 33     -- serialNumber
  16   13:     SEQUENCE {     -- signature
  18    9:       OBJECT IDENTIFIER
         :         sha256WithRSAEncryption (1 2 840 113549 1 1 11)
  29    0:       NULL
         :       }
  31   83:     SEQUENCE {     -- issuer
  33   11:       SET {
  35    9:         SEQUENCE {
  37    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
  42    2:           PrintableString 'EU'
         :           }
         :         }
  46   22:       SET {
  48   20:         SEQUENCE {
  50    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
  55   13:           UTF8String 'Test locality'
         :           }
         :         }
  70   26:       SET {
  72   24:         SEQUENCE {
  74    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
  79   25:           UTF8String 'Narodny bezpecnostny urad'
         :           }
         :         }
  98   16:       SET {
 100   14:         SEQUENCE {
 102    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
 107    7:           UTF8String 'Test CA'
         :           }
         :         }
 258   21:       SET {
 260   19:         SEQUENCE {
 262    3:           OBJECT IDENTIFIER serialNumber (2 5 4 5)
                                 -- organization unique identification data
 267   14:           PrintableString 'NTRSK-36061701'
         :           }
         :         }
         :       }
 116   30:     SEQUENCE {
 118   13:       UTCTime 21/07/2006 08:48:17 GMT -- notBefore
 133   13:       UTCTime 20/07/2011 08:48:17 GMT -- notAfter
         :       }
 148   91:     SEQUENCE {     -- subject
 150   11:       SET {
 152    9:         SEQUENCE {
 154    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 159    2:           PrintableString 'EU'
         :           }
         :         }
 163   22:       SET {
 165   20:         SEQUENCE {
 167    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
 172   13:           UTF8String 'Test locality'
         :           }
```

```
        :          }
187   30:        SET {
189   28:          SEQUENCE {
191    3:            OBJECT IDENTIFIER organizationName (2 5 4 10)
196   21:            UTF8String 'Test TSA organization'
        :            }
        :          }
219   20:        SET {
221   18:          SEQUENCE {
223    3:            OBJECT IDENTIFIER commonName (2 5 4 3)
228   11:            UTF8String 'TSA for QES'
        :            }
        :          }
        :        }
241  290:      SEQUENCE {     -- subjectPublicKeyInfo
245   13:        SEQUENCE {
247    9:          OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
258    0:          NULL
        :          }
260  271:        BIT STRING, encapsulates {
265  266:          SEQUENCE {
269  257:            INTEGER
        :              00 D5 53 25 C5 4B 2B 83 9B FB 6A CE 4E 17 ED FA
        :              F5 32 B2 F1 85 C3 72 97 E4 F2 76 58 D9 19 1F 39
        :              B9 40 6C 87 1B 24 7C E4 9B E8 91 9D 71 4A C2 CF
        :              00 4F 7B 46 8E E1 C1 65 31 2C C6 DC B5 40 F7 3B
        :              77 0D EB B1 F1 A2 48 10 E5 3F 5B B7 12 AA 52 B4
        :              1E CB 13 C6 4C 14 E1 3E FC 1F 89 BA 3F A1 0C 9C
        :              32 0E BC 61 89 88 17 FE 75 4C F5 FF 0C BC 4E 9A
        :              52 B5 BE D0 F3 DD E0 83 C4 EB 94 7E 72 C1 33 CD
        :                      [ Another 129 bytes skipped ]
530    3:            INTEGER 65537
        :            }
        :          }
        :        }
535  730:      [3] {          -- extensions
539  726:        SEQUENCE {
543   31:          SEQUENCE {
545    3:            OBJECT IDENTIFIER authorityKeyIdentifier (2 5 29 35)
550   24:            OCTET STRING, encapsulates {
552   22:              SEQUENCE {
554   20:                [0]
        :                  38 B3 D9 00 A5 F6 81 B9 4B 0D 9B 2A DB 4D 65 67
        :                  B1 A5 1B CC
        :                }
        :              }
        :            }
576   29:          SEQUENCE {
578    3:            OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
583   22:            OCTET STRING, encapsulates {
585   20:              OCTET STRING
        :                35 78 32 78 AF D2 98 62 FF 05 9E B9 3D 9C 2A 0C
        :                95 8E 18 52
        :              }
        :            }
607   11:          SEQUENCE {
609    3:            OBJECT IDENTIFIER keyUsage (2 5 29 15)
614    4:            OCTET STRING, encapsulates {
616    2:              BIT STRING 6 unused bits
        :                '10'B (bit 1)    -- nonRepudiation
        :              }
        :            }
620   36:          SEQUENCE {
622    3:            OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
```

```
627   29:            OCTET STRING, encapsulates {
629   27:              SEQUENCE {
631   15:                SEQUENCE {      -- QCP SK
633   13:                  OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
        :                  }
648    8:                SEQUENCE {      -- NCP+: Normalized Certificate Policy
                                         -- requiring a secure user device
650    6:                  OBJECT IDENTIFIER '0 4 0 2042 1 2'
        :                  }
        :                }
        :              }
        :            }
658    9:          SEQUENCE {
660    3:            OBJECT IDENTIFIER basicConstraints (2 5 29 19)
665    2:            OCTET STRING, encapsulates {
667    0:              SEQUENCE {} -- end entity certificate
        :              }
        :            }
669   22:          SEQUENCE {
671    3:            OBJECT IDENTIFIER extKeyUsage (2 5 29 37)
676    1:            BOOLEAN TRUE   -- critical
679   12:            OCTET STRING, encapsulates {
681   10:              SEQUENCE {
683    8:                OBJECT IDENTIFIER timeStamping (1 3 6 1 5 5 7 3 8)
        :                }
        :              }
        :            }
693  265:          SEQUENCE {
697    3:            OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
702  256:            OCTET STRING, encapsulates {
706  253:              SEQUENCE {
709   40:                SEQUENCE {
711   38:                  [0] {
713   36:                    [0] {
715   34:                      [6] 'http://qes.test.eu/test/testca.crl'
        :                      }
        :                    }
        :                  }
751  109:                SEQUENCE {
753  107:                  [0] {
755  105:                    [0] {
757  103:                      [6]
        :                      'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
        :                      'n,l=Test locality,c=EU?certificateRevocationList'
        :                      ';binary'
        :                      }
        :                    }
        :                  }
862   98:                SEQUENCE {
864   96:                  [0] {
866   94:                    [0] {
868   92:                      [6]
        :                      'ldap:///cn=Test CA,o=Test organization,l=Test lo'
        :                      'cality,c=EU?certificateRevocationList;binary'
        :                      }
        :                    }
        :                  }
        :                }
        :              }
        :            }
962  303:          SEQUENCE {
966    8:            OBJECT IDENTIFIER authorityInfoAccess (1 3 6 1 5 5 7 1 1)
976  289:            OCTET STRING, encapsulates {
980  285:              SEQUENCE {
```

```
 984   36:               SEQUENCE {
 986    8:                 OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
 996   24:                 [6] 'http://ocsp.test.eu/ocsp'
         :                 }
                         -- certificate and cross-certificates
1022   46:               SEQUENCE {
1024    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1034   34:                 [6] 'http://qes.test.eu/test/testca.p7c'
         :                 }
1070  103:               SEQUENCE {
1072    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1082   91:                 [6]
         :                   'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
         :                   'n,l=Test locality,c=EU?caCertificate;binary'
         :                 }
1175   92:               SEQUENCE {
1177    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1187   80:                 [6]
         :                   'ldap:///cn=Test CA,o=Test organization,l=Test lo'
         :                   'cality,c=EU?caCertificate;binary'
         :                 }
         :               }
         :             }
         :           }
         :         }
         :       }
         :     }
1269   13:   SEQUENCE {
1271    9:     OBJECT IDENTIFIER
         :       sha256WithRSAEncryption (1 2 840 113549 1 1 11)
1282    0:     NULL
         :     }
1284  257:   BIT STRING
         :     27 2F 6F FA CC 27 7F 8A E1 6C 0B 10 A5 0F 31 E7
         :     02 5F 15 5C C6 B6 31 87 12 F6 D7 6B 37 E8 62 B2
         :     4E FF B7 92 5A F4 EF 31 90 4C 67 B9 46 DF CC 05
         :     CC BF F6 63 A7 45 D7 5F 92 0D 8E BB B7 4A 3C A3
         :     BD 20 4E 93 F0 35 28 79 FC 79 79 77 EF 4A E0 83
         :     E3 3D 4B 70 CE 25 BE 80 8D E5 E8 E9 2A 67 86 CF
         :     4D 79 EB 5F 9D 54 6B 9D AD AA DC 81 37 63 7D 39
         :     56 26 E3 9B 2C E1 A6 14 8C E9 BD 20 35 0F A8 7A
         :             [ Another 128 bytes skipped ]
         :   }
```

A.5 Example of CA certificate

```
  0 1926: SEQUENCE {
  4 1646:   SEQUENCE {
  8    3:     [0] {
 10    1:       INTEGER 2    -- version
         :       }
 13    2:     INTEGER 8558   -- serialNumber
 17   13:     SEQUENCE {     -- signature
 19    9:       OBJECT IDENTIFIER
         :         sha256WithRSAEncryption (1 2 840 113549 1 1 11)
 30    0:       NULL
         :       }
 32   85:     SEQUENCE {     -- issuer
 34   11:       SET {
 36    9:         SEQUENCE {
 38    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 43    2:           PrintableString 'EU'
         :           }
```

```
        :         }
 47   22:       SET {
 49   20:         SEQUENCE {
 51    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
 56   13:           UTF8String 'Test locality'
        :           }
        :         }
 71   26:       SET {
 73   24:         SEQUENCE {
 75    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
 80   17:           UTF8String 'Test organization'
        :           }
        :         }
 99   18:       SET {
101   16:         SEQUENCE {
103    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
108    9:           UTF8String 'Test Root'
        :           }
        :         }
        :       }
119   30:     SEQUENCE {
121   13:       UTCTime 24/08/2006 11:18:13 GMT -- notBefore
136   13:       UTCTime 24/08/2011 11:17:27 GMT -- notAfter
        :       }
151   83:     SEQUENCE {     -- subject
153   11:       SET {
155    9:         SEQUENCE {
157    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
162    2:           PrintableString 'EU'
        :           }
        :         }
166   22:       SET {
168   20:         SEQUENCE {
170    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
175   13:           UTF8String 'Test locality'
        :           }
        :         }
190   26:       SET {
192   24:         SEQUENCE {
194    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
199   25:           UTF8String 'Narodny bezpecnostny urad'
        :           }
        :         }
218   16:       SET {
220   14:         SEQUENCE {
222    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
227    7:           UTF8String 'Test CA'
        :           }
        :         }
        :       }
258   21:       SET {
260   19:         SEQUENCE {
262    3:           OBJECT IDENTIFIER serialNumber (2 5 4 5)
                              -- organization unique identification data
267   14:           PrintableString 'NTRSK-36061701'
        :           }
        :         }
        :       }
236  290:     SEQUENCE {     -- subjectPublicKeyInfo
240   13:       SEQUENCE {
242    9:         OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
253    0:         NULL
        :         }
255  271:       BIT STRING, encapsulates {
```

```
 260  266:          SEQUENCE {
 264  257:            INTEGER
       :              00 B2 D4 6C 5A D4 86 6B E5 BA 28 4F 87 4D F7 E2
       :              20 78 5F 02 E8 0F FA E3 EF EF 2A 09 CC 4A 74 0A
       :              36 4E B5 95 C1 FC 59 23 36 85 E6 6B 5F 6F 29 84
       :              9A 3F D1 08 ED 30 2E 17 3B 96 23 E8 67 68 C4 68
       :              78 84 D4 D2 AA 56 37 47 0F AD 1C E1 88 B5 39 5D
       :              B6 C1 22 30 08 BA DB 0D BB BF 53 2A 8E AD 48 E6
       :              D0 12 DB 02 A9 F9 DF AA E4 E9 01 A3 DD 39 E2 0C
       :              C5 C0 A0 EC 36 5C 93 99 AC 31 4B 09 18 71 31 FF
       :                      [ Another 129 bytes skipped ]
 525    3:            INTEGER 65537
       :            }
       :          }
       :        }
 530 1120:      [3] {          -- extensions
 534 1116:        SEQUENCE {
 538   31:          SEQUENCE {
 540    3:            OBJECT IDENTIFIER authorityKeyIdentifier (2 5 29 35)
 545   24:            OCTET STRING, encapsulates {
 547   22:              SEQUENCE {
 549   20:                [0]
       :                  06 DA 89 E7 D3 8E 53 3A 79 77 E9 EB F9 A6 B6 32
       :                  65 3F 46 24
       :                }
       :              }
       :            }
 571   29:          SEQUENCE {
 573    3:            OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
 578   22:            OCTET STRING, encapsulates {
 580   20:              OCTET STRING
       :                38 B3 D9 00 A5 F6 81 B9 4B 0D 9B 2A DB 4D 65 67
       :                B1 A5 1B CC
       :              }
       :            }
 602   14:          SEQUENCE {
 604    3:            OBJECT IDENTIFIER keyUsage (2 5 29 15)
 609    1:            BOOLEAN TRUE   -- critical
 612    4:            OCTET STRING, encapsulates {
 614    2:              BIT STRING 1 unused bit
       :                '1100000'B -- keyCertSign, cRLSign
       :              }
       :            }
 618   29:          SEQUENCE {
 620    3:            OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
 625    1:            BOOLEAN TRUE   -- critical
 628   19:            OCTET STRING, encapsulates {
 630   17:              SEQUENCE {
 632   15:                SEQUENCE { -- QCP SK
 634   13:                  OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
       :                  }
       :                }
       :              }
       :            }
 649   69:          SEQUENCE {
 651    3:            OBJECT IDENTIFIER policyMappings (2 5 29 33)
 656    1:            BOOLEAN TRUE   -- is optional
 659   59:            OCTET STRING, encapsulates {
 661   57:              SEQUENCE {
 663   23:                SEQUENCE {
                            -- issuerDomainPolicy  - QCP SK
 665   13:                  OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
                            -- subjectDomainPolicy - QCP public + SSCD
 680    6:                  OBJECT IDENTIFIER '0 4 0 1456 1 1'
```

```
        :                 }
 688   30:               SEQUENCE {
                           -- issuerDomainPolicy  - QCP SK
 690   13:                 OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
                           -- subjectDomainPolicy - QCP SK
 705   13:                 OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
        :                  }
        :                }
        :              }
        :            }
 720   18:          SEQUENCE {
 722    3:            OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 727    1:            BOOLEAN TRUE   -- critical
 730    8:            OCTET STRING, encapsulates {
 732    6:              SEQUENCE {
 734    1:                BOOLEAN TRUE     -- CA certificate
 737    1:                INTEGER 1        -- pathLenConstraint
        :                }
        :              }
        :            }
 740   15:          SEQUENCE {
 742    3:            OBJECT IDENTIFIER policyConstraints (2 5 29 36)
 747    1:            BOOLEAN TRUE   -- critical
 750    5:            OCTET STRING, encapsulates {
 752    3:              SEQUENCE {
 754    1:                [0] 00           -- requireExplicitPolicy SkipCerts
        :                }
        :              }
        :            }
 757  272:          SEQUENCE {
 761    3:            OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
 766  263:            OCTET STRING, encapsulates {
 770  259:              SEQUENCE {
 774   42:                SEQUENCE {
 776   40:                  [0] {
 778   38:                    [0] {
 780   36:                      [6] 'http://qes.test.eu/test/testroot.crl'
        :                      }
        :                    }
        :                  }
 818  111:                SEQUENCE {
 820  109:                  [0] {
 822  107:                    [0] {
 824  105:                      [6]
        :                      'ldap://qes.test.eu/cn=Test Root,o=Test organizat'
        :                      'ion,l=Test locality,c=EU?certificateRevocationLi'
        :                      'st;binary'
        :                      }
        :                    }
        :                  }
 931  100:                SEQUENCE {
 933   98:                  [0] {
 935   96:                    [0] {
 937   94:                      [6]
        :                      'ldap:///cn=Test Root,o=Test organization,l=Test '
        :                      'locality,c=EU?certificateRevocationList;binary'
        :                      }
        :                    }
        :                  }
        :                }
        :              }
        :            }
1033  310:          SEQUENCE {
1037    8:            OBJECT IDENTIFIER authorityInfoAccess (1 3 6 1 5 5 7 1 1)
```

```
1047  296:           OCTET STRING, encapsulates {
1051  292:             SEQUENCE {
1055   37:               SEQUENCE {
1057    8:                 OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
1067   25:                 [6] 'http://ocsp.test.eu/ocspr'
         :                 }
                         -- certificate and cross-certificates
1094   48:               SEQUENCE {
1096    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1106   36:                 [6] 'http://qes.test.eu/test/testroot.p7c'
         :                 }
1144  105:               SEQUENCE {
1146    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1156   93:                 [6]
         :                   'ldap://qes.test.eu/cn=Test Root,o=Test organizat'
         :                   'ion,l=Test locality,c=EU?caCertificate;binary'
         :                 }
1251   94:               SEQUENCE {
1253    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1263   82:                 [6]
         :                   'ldap:///cn=Test Root,o=Test organization,l=Test '
         :                   'locality,c=EU?caCertificate;binary'
         :                 }
         :               }
         :             }
         :           }
1347  303:         SEQUENCE {
1351    8:           OBJECT IDENTIFIER subjectInfoAccess (1 3 6 1 5 5 7 1 11)
1361  289:           OCTET STRING, encapsulates {
1365  285:             SEQUENCE {
1369   36:               SEQUENCE {
1371    8:                 OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
1381   24:                 [6] 'http://ocsp.test.eu/ocsp'
         :                 }
                         -- certificate and cross-certificates
1407   46:               SEQUENCE {
1409    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1419   34:                 [6] 'http://qes.test.eu/test/testca.p7c'
         :                 }
1455  103:               SEQUENCE {
1457    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1467   91:                 [6]
         :                   'ldap://qes.test.eu/cn=Test CA,o=Test organizatio'
         :                   'n,l=Test locality,c=EU?caCertificate;binary'
         :                 }
1560   92:               SEQUENCE {
1562    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1572   80:                 [6]
         :                   'ldap:///cn=Test CA,o=Test organization,l=Test lo'
         :                   'cality,c=EU?caCertificate;binary'
         :                 }
         :               }
         :             }
         :           }
         :         }
         :       }
         :     }
1654   13:   SEQUENCE {       -- signatureAlgorithm
1656    9:     OBJECT IDENTIFIER
         :       sha256WithRSAEncryption (1 2 840 113549 1 1 11)
1667    0:     NULL
         :     }
1669  257:   BIT STRING        -- signatureValue
         :     C0 D9 11 F1 AE 65 1E C3 76 B3 7C 6B A3 6C 06 E1
```

```
         :     52 2D 70 05 49 12 B4 5B 6E ED 94 B7 2A 64 3F 62
         :     41 16 C6 5D 3F 0D 4B CF 3E C4 4B C2 51 78 10 C6
         :     DF 8A F5 B9 6D 51 D5 5E 42 19 4B F7 86 B3 25 7A
         :     9B 1C F0 95 44 6E 81 1E 03 E0 58 11 A6 2B F5 02
         :     BC 97 4A 46 35 F2 7A 29 E7 95 EF 0B 7C A4 B1 A3
         :     8B DE 76 FE 4C A8 70 A0 5B D4 5A F9 B4 B1 0E A3
         :     F4 E1 C2 6A D3 6F CF 9A 84 F7 A3 00 28 8A 99 1B
         :             [ Another 128 bytes skipped ]
         :   }
```

A.6 Example of the root certificate

```
  0 1493: SEQUENCE {
  4 1213:   SEQUENCE {
  8    3:     [0] {
 10    1:       INTEGER 2    -- version
         :       }
 13    1:     INTEGER 1      -- serialNumber
 16   13:     SEQUENCE {     -- signature
 18    9:       OBJECT IDENTIFIER
         :         sha256WithRSAEncryption (1 2 840 113549 1 1 11)
 29    0:       NULL
         :       }
 31   85:     SEQUENCE {     -- issuer
 33   11:       SET {
 35    9:         SEQUENCE {
 37    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
 42    2:           PrintableString 'EU'
         :           }
         :         }
 46   22:       SET {
 48   20:         SEQUENCE {
 50    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
 55   13:           UTF8String 'Test locality'
         :           }
         :         }
 70   26:       SET {
 72   24:         SEQUENCE {
 74    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
 79   17:           UTF8String 'Test organization'
         :           }
         :         }
 98   18:       SET {
100   16:         SEQUENCE {
102    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
107    9:           UTF8String 'Test Root'
         :           }
         :         }
         :       }
118   30:     SEQUENCE {
120   13:       UTCTime 22/02/2005 16:13:37 GMT -- notBefore
135   13:       UTCTime 22/02/2015 15:43:57 GMT -- notAfter
         :       }
150   85:     SEQUENCE {     -- subject
152   11:       SET {
154    9:         SEQUENCE {
156    3:           OBJECT IDENTIFIER countryName (2 5 4 6)
161    2:           PrintableString 'EU'
         :           }
         :         }
165   22:       SET {
167   20:         SEQUENCE {
169    3:           OBJECT IDENTIFIER localityName (2 5 4 7)
```

```
174   13:           UTF8String 'Test locality'
        :           }
        :         }
189   26:       SET {
191   24:         SEQUENCE {
193    3:           OBJECT IDENTIFIER organizationName (2 5 4 10)
198   17:           UTF8String 'Test organization'
        :           }
        :         }
217   18:       SET {
219   16:         SEQUENCE {
221    3:           OBJECT IDENTIFIER commonName (2 5 4 3)
226    9:           UTF8String 'Test Root'
        :           }
        :         }
        :       }
237  290:     SEQUENCE {     -- subjectPublicKeyInfo
241   13:       SEQUENCE {
243    9:         OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
254    0:         NULL
        :         }
256  271:       BIT STRING, encapsulates {
261  266:         SEQUENCE {
265  257:           INTEGER
        :             00 F2 6F 8E C9 BD 3F 65 65 41 BE 5F DC 51 AB 4D
        :             C5 A4 8D E2 0C 4B 7C 52 75 9A 80 23 36 FB B4 53
        :             77 1D 8F D1 D7 BD DA 14 79 8E DB 13 51 66 C7 4A
        :             33 AD 0F 95 4F E8 83 BA 03 42 70 2E BE 9C F1 74
        :             6F 83 84 6C 5D F6 32 63 9E 6E DE 63 C0 DF 6B 31
        :             70 81 D6 21 BA D7 3A 81 F7 F1 95 7B C1 AA 36 39
        :             74 0B 2F F2 9B 6D 08 AA 05 A7 6C DA 2E 5B FD B5
        :             0D B8 FD 8B 75 53 9D A5 01 9E 1E E3 98 9B D3 29
        :                       [ Another 129 bytes skipped ]
526    3:           INTEGER 65537
        :           }
        :         }
        :       }
531  686:     [3] {          -- extensions
535  682:       SEQUENCE {
539   29:         SEQUENCE {
541    3:           OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
546   22:           OCTET STRING, encapsulates {
548   20:             OCTET STRING
        :               06 DA 89 E7 D3 8E 53 3A 79 77 E9 EB F9 A6 B6 32
        :               65 3F 46 24
        :             }
        :           }
570   14:         SEQUENCE {
572    3:           OBJECT IDENTIFIER keyUsage (2 5 29 15)
577    1:           BOOLEAN TRUE   -- critical
580    4:           OCTET STRING, encapsulates {
582    2:             BIT STRING 1 unused bit
        :               '1100000'B -- keyCertSign, cRLSign
        :             }
        :           }
586   26:         SEQUENCE {
588    3:           OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
593   19:           OCTET STRING, encapsulates {
595   17:             SEQUENCE {
597   15:               SEQUENCE { -- QCP SK
599   13:                 OBJECT IDENTIFIER '1 3 158 36061701 0 0 0 1 2 2'
        :                 }
        :               }
        :             }
```

```
         :           }
 614   15:         SEQUENCE {
 616    3:           OBJECT IDENTIFIER basicConstraints (2 5 29 19)
 621    1:           BOOLEAN TRUE   -- critical
 624    5:           OCTET STRING, encapsulates {
 626    3:             SEQUENCE {
 628    1:               BOOLEAN TRUE     -- CA certificate
         :               }
         :             }
         :           }
 631  272:         SEQUENCE {
 635    3:           OBJECT IDENTIFIER cRLDistributionPoints (2 5 29 31)
 640  263:           OCTET STRING, encapsulates {
 644  259:             SEQUENCE {
 648   42:               SEQUENCE {
 650   40:                 [0] {
 652   38:                   [0] {
 654   36:                     [6] 'http://qes.test.eu/test/testroot.crl'
         :                     }
         :                   }
         :                 }
 692  111:               SEQUENCE {
 694  109:                 [0] {
 696  107:                   [0] {
 698  105:                     [6]
         :                     'ldap://qes.test.eu/cn=Test Root,o=Test organizat'
         :                     'ion,l=Test locality,c=EU?certificateRevocationLi'
         :                     'st;binary'
         :                     }
         :                   }
         :                 }
 805  100:               SEQUENCE {
 807   98:                 [0] {
 809   96:                   [0] {
 811   94:                     [6]
         :                     'ldap:///cn=Test Root,o=Test organization,l=Test '
         :                     'locality,c=EU?certificateRevocationList;binary'
         :                     }
         :                   }
         :                 }
         :               }
         :             }
 907  310:         SEQUENCE {
 911    8:           OBJECT IDENTIFIER subjectInfoAccess (1 3 6 1 5 5 7 1 11)
 921  296:           OCTET STRING, encapsulates {
 925  292:             SEQUENCE {
 929   37:               SEQUENCE {
 931    8:                 OBJECT IDENTIFIER ocsp (1 3 6 1 5 5 7 48 1)
 941   25:                 [6] 'http://ocsp.test.eu/ocspr'
         :                 }
                         -- certificate and cross-certificates
 968   48:               SEQUENCE {
 970    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
 980   36:                 [6] 'http://qes.test.eu/test/testroot.p7c'
         :                 }
1018  105:               SEQUENCE {
1020    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
1030   93:                 [6]
         :                   'ldap://qes.test.eu/cn=Test Root,o=Test organizat'
         :                   'ion,l=Test locality,c=EU?caCertificate;binary'
         :                 }
1125   94:               SEQUENCE {
1127    8:                 OBJECT IDENTIFIER caIssuers (1 3 6 1 5 5 7 48 2)
```

```
1137   82:                 [6]
         :                   'ldap:///cn=Test Root,o=Test organization,l=Test '
         :                   'locality,c=EU?caCertificate;binary'
         :                 }
         :               }
         :             }
         :           }
         :         }
         :       }
         :     }
1221   13:   SEQUENCE {       -- signatureAlgorithm
1223    9:     OBJECT IDENTIFIER
         :       sha256WithRSAEncryption (1 2 840 113549 1 1 11)
1234    0:     NULL
         :     }
1236  257:   BIT STRING       -- signatureValue
         :     F2 3B 29 D1 58 61 09 BF 48 18 10 57 4B BA AF 87
         :     78 0B 29 F9 BA AE 41 DD F1 6C 7E 1B C9 29 3E F6
         :     4A E8 40 9C 6A DF 6B 70 E9 27 F8 A0 27 1B 90 3F
         :     7C 18 A1 76 48 1D 17 7C 6E 8F C2 6E EB D3 F4 A5
         :     1D A6 2F 37 DB D2 29 EA 11 5F 51 55 BF D4 52 FB
         :     85 71 8F 9A 58 D8 8F 4C 44 E3 51 CD 30 4F BE A1
         :     D9 BD 99 BD C8 CC 71 5C B5 D8 C4 95 B9 A1 8A 3A
         :     48 35 64 68 0B 0D A7 24 F0 D3 D4 EF 96 6F 96 72
         :             [ Another 128 bytes skipped ]
         :   }
```

Annex B (informative) Revisions made since previous version

B.1 Additional requirements

The following items have been added which significantly affect the requirements:

In chapter 5 there are defined requirements for identification of qualified certificates and certificates for mandate.

B.2 Updated requirements

The following items have been updated to extend choices or otherwise modify requirements:

The issuer name must contain an unambiguous identification for new certificates issued for a new key pair of the issuer in the item *serialNumber*.
Documents in PDF format containing for example CP or CPS should be protected as a minimum pursuant to requirements of ETSI TS 102 778-3 Part 3 [19].

B.3 Clarifications

The following items have been updated to clarify existing requirements:
None.

B.4 Editorial

A number of other editorial changes were made which do not affect the technical content of the present document:

New subchapters 5.1 and 5.2 have been added.

Annex C (informative) Bibliography

Basic documents of the Slovak legislation on electronic signature
http://www.nbusr.sk/sk/pravne-predpisy/elektronicky-podpis.1.html

NSA Standards
http://www.nbusr.sk/sk/elektronicky-podpis/standardy-nbu.1.html

Certification path building and validity verification of certificates
http://www.nbusr.sk/sk/elektronicky-podpis/overovanie.1.html

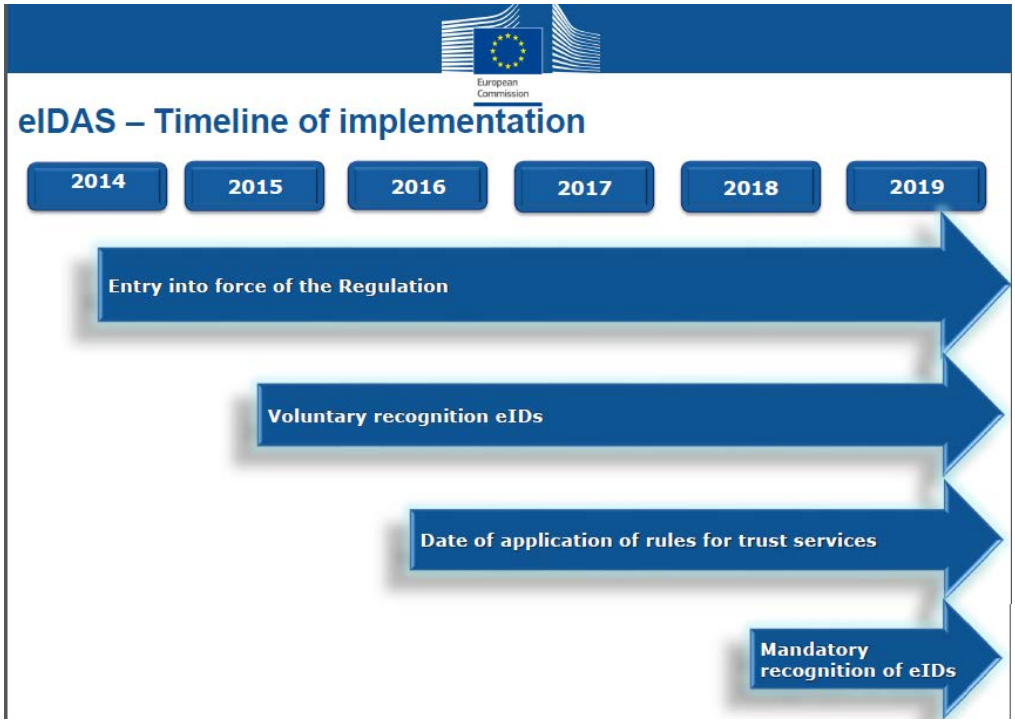
Commission Decisions on Electronic Signature:

COMMISSION IMPLEMENTING DECISION 2013/662/EU of 14 October 2013 amending Decision 2009/767/EC as regards the establishment, maintenance and publication of trusted lists of certification service providers supervised/accredited by Member States
http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2013:306:0021:0039:EN:PDF

COMMISSION IMPLEMENTING DECISION 2014/148/EU of 17 March 2014 amending Decision 2011/130/EU establishing minimum requirements for the cross-border processing of documents signed electronically by competent authorities under Directive 2006/123/EC of the European Parliament and of the Council on services in the internal market
http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2014:080:0007:0009:EN:PDF

EU legislation which repeals the Directive on electronic signature and requirements of national legislations adopted on the basis of this directive: http://ec.europa.eu/digital-agenda/en/trust-services-and-eid

Regulation (EU) No 910/2014 of the European Parliament and of the Council of 23 July 2014 on electronic identification and trust services for electronic transactions in the internal market and repealing Directive 1999/93/EC
http://eur-lex.europa.eu/legal-content/EN/ALL/?uri=uriserv:OJ.L_.2014.257.01.0073.01.ENG



Annex D History

Version	Date of issue	Note	Editor
Version 1.0.	30 September 2004	First edition	Peter Rybár
Version 1.1	14 August 2005	Second edition, revokes the edition 1.0	Peter Rybár, NSA
Version 1.2.	6 November 2005	Change of critical extensions	Peter Rybár, NSA **Translation** Vladimíra Iglarčíková
Version 2.0 No. 3198/2007/IBEP-001	8 April 2007	Adding of Reference to physical person's identity to X.509 qualified certificate	Peter Rybár, NSA
Version 2.1 No. 3198/2007/IBEP-017	18 September 2007	Exception for indication of the qualified certificate pursuant to Slovak legislation	Peter Rybár, NSA
Version 2.2 No. 2739/2008/IBEP-004	6 June 2008	Change of person's identification pursuant to amendment act on ES in 2008.	Peter Rybár, NSA
Version 3.0 No. 584/2009/IBEP-008	30 June 2009	Standards of qualified certificate formats prior to Version 3.0 are revoked. The change made pursuant to amendment decree in 2009.	Peter Rybár, NSA
Version 4.0 No. 3109/2014/IBEP/OA-001	10 July 2014	Change made pursuant to amendment of the legislation.	Peter Rybár, NSA