



NATIONAL SECURITY AUTHORITY

Version 1.1

**SIM of the mobile device for electronic signing
through secure WEB/WAP or PKCS#11 interfaces**

10 April 2009

This English version of the Slovak document No. 584/2009/IBEP-007 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

Department of Information Security and Electronic Signature

Budatínska č. 30, P.O. BOX 16, 850 07 Bratislava 57

<http://www.nbusr.sk/>

E-mail: info@nbusr.sk

Content

1	Introduction.....	4
2	Scope.....	4
3	References.....	5
4	Abbreviations	5
5	Assumptions for performing the signing process.....	6
6	Components and their usage.....	6
6.1	Encrypting private key	6
6.2	Signing private key	6
6.3	Format of encrypted data being sent to SIM card.....	6
Table 1	A type of encrypted data in SMS TEDData in ASN.1	7
6.4	DigiID of the user	8
Table 2	CMS AdES signature in ASN.1 – DigiID	9
Table 3	Complete SignedData in ASN.1	9
7	Signing and signature verification.....	10
Annex A (Informative)	Examples of data for signing in SIM.....	11
A.1	Textual file of trustworthy certificates signed in DigiID.p7m file.....	11
A.2	DigiID.p7m file encoded in BASE64	11
A.3	DigiID.p7m file displayed in ASN.1 dump	12
Annex B (Informative)	Bibliography	21
Annex C	History.....	23

1 Introduction

Application communication with Secure-Signature-Creation-Devices (SSCD) in environment which is not secure requires the use of a secure channel whose creation needs an enormous data exchange for the initial key exchange and the device and signing application authentication. When using the mobile device whose SIM card contains private keys for electronic signature needs, the secure channel creation based on procedures from EN 14890 is practically not realizable due to a small capacity of data which can be transferred by means of SMS message and number of exchanged SMS messages for the secure channel creation through Secure Messaging.

2 Scope

The purpose of the present standard is to create the profile for PKCS#11 and WEB/WAP interfaces to define the minimal necessary requirements for exchanged data types between PKCS#11, WEB/WAP interfaces and SIM application which can be used in (qualified) electronic signature creation. Expected model consists of the standard user application or web signing application communicating through PKCS#11 interfaces and the mobile device with the SIM card which receives and sends data to communication interfaces interconnected with PKCS#11 especially by means of SMS. The document defines the protocol which supersedes Secure Messaging defined in EN 14890 in order to allow the use of the mobile device with the SIM card as the device providing the secure signing after the receipt of the request for signing with the authorization code and entering the authorization code and the signing PIN for creation of the signature itself.

The functionality of PKCS#11 interface could be included in servers accessible through secure WEB/WAP interfaces which provide trustworthy services e.g. for government or commercial organizations. Examples could be banks offering electronic payments for any goods and services or public portals for citizens.

This standard is issued pursuant to Article 10 part 1 (j) of Act No. 215/2002 Coll. on Electronic Signature.

3 References

References to documents defining used types and methods.

- [1] ETSI TS 101 733: "Electronic Signatures and Infrastructures (ESI); CMS Advanced Electronic Signatures (CAAdES)".
- [2] ETSI TR 102 272: "Electronic Signatures and Infrastructures (ESI); ASN.1 format for signature policies".
- [3] RFC 5280 (2008) "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".
- [4] RFC 3739 (2004): "Qualified Certificates Profile".
- [5] ETSI TS 101 862: "Qualified Certificate Profile".
- [6] RFC 3852 (2004): "Cryptographic Message Syntax (CMS)".
- [7] IETF RFC 3161 (2001): "Internet X.509 Public Key Infrastructure Time-Stamp Protocol (TSP)".
- [8] RFC 2560 (1999): "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP".
- [9] NSA Qualified Electronic Signature Formats
- [10] EN 14890-1:2008: "Application Interface for smart cards used as Secure Signature Creation Devices - Part 1: Basic services".
- [11] RFC 2044 (1996): UTF-8, a transformation format of Unicode and ISO 10646
- [12] ITU-T RECOMMENDATION X.509 (08/2005) | ISO/IEC 9594-8:2005 "Public-key and attribute certificate frameworks".

4 Abbreviations

ASCII	American Standard Code for Information Interchange
ASN.1	Abstract Syntax Notation 1
CA	Certification Authority
CAAdES	CMS Advanced Electronic Signature
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List
DER	Distinguished Encoding Rules (for ASN.1)
OID	Object Identifier
QC	Qualified Certificate
SHA-1	Secure Hash Algorithm 1
SMS	Short Message Service
SMS-C	Short Message Service Centre
SMS gateway	based on SS7 connectivity to route SMS (international termination model)
SS7	Signaling System #7
SSCD	Secure-Signature-Creation Device
URL	Uniform Resource Locator
QES	Qualified Electronic Signature

5 Assumptions for performing the signing process

The basic assumption to perform the signing by means of the SIM application is to issue two certificates for public keys whose private keys are generated and stored only on the SIM card wherefrom they cannot be exported. The first certificate is issued for a signing key (RSA/EC-(G)DSA) and the second one is issued for an encrypting key (RSA max 1024bit). The certificate for the encrypting key is used to encrypt the request for signing being sent by the signing application by means of SMS to the SIM card. The encrypted request contains a hash of data that should be signed and additional two pieces of data: a hash of a signer's certificate and the authorization code which is known only to a signer after being displayed by the signing application before the encryption of the SMS request for signature. The SIM application by decrypting the received SMS obtains the authorization code and subsequently displays the request for entering the authorization code to the signer. If the signer enters the incorrect authorization code, the signing is rejected what ensures the protection against the signature based on a fake request. The SIM application must be also able to display the hash value of data to be signed to distrustful users on request, so the signer can ensure that data to be signed were not forged. Entering the authorization code which was generated and displayed by the signing application for the signer is sufficient for common users.

6 Components and their usage

6.1 Encrypting private key

The encrypting private key is under control of the SIM application and the SIM application through this key decrypts the received SMS request containing the hash of data to be signed, the authorization code and the hash of the signer's certificate. The encrypting key is fully under control of the signing SIM application what means that it cannot be used by anybody else except by the signing SIM application and therefore the PIN entering is not required. The encrypting private key must not be exportable and is used only for decryption of requests for signature. The SIM application which uses the encrypting key must not allow the external processing of decrypted data of the authorization code beyond the SIM card in order to prevent the attack with faked data containing the same authorization code. The SIM application allows only the usage of decrypted data which have contained the correct authorization code, i.e. the authorization code which was identical with the authorization code entered by the signer in the mobile device.

6.2 Signing private key

The signing key can be used for signing the hash value of data to be signed only after entering the correct authorization code and the signing PIN for the key identified on the basis of identification through the hash of the signer's certificate. The signing key must be stored only on the SIM card and the card must not allow the key export in order to prevent its usage beyond the SIM card. The signing PIN must not be used for other purposes than the signing performed with the private signing key in order to prevent the misuse of the signing key by other operation which processes other data or keys on the SIM card. Due to the size of the answer it is recommended to use ECDSA (EC-GDSA) instead of RSA, so that the digital signature could be sent in one SMS.

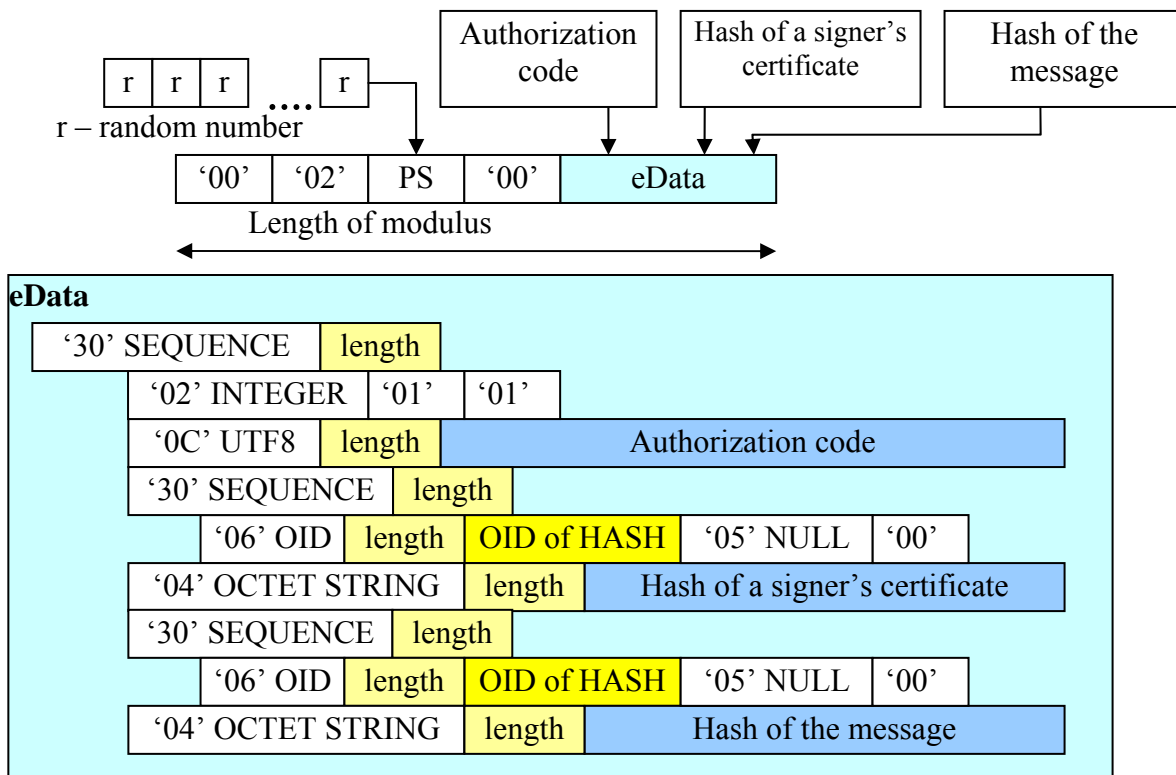
6.3 Format of encrypted data being sent to SIM card

SMS contains three items encrypted: the authorization code used to verify the authorization to data acquired from the decrypted SMS, the hash of the signer's certificate through which the SIM application finds the signing private key and the hash of data to be signed.

The SIM application after the data decryption from SMS asks the mobile device to enter the authorization code and continues in signing only in case the entered authorization code of the signer is identical with the authorization code from the decrypted SMS being generated/entered and displayed by the signing application to the signer before the request for signing through SMS was sent. In order to perform the signing process, the SIM application subsequently asks the mobile

device to display the information for the signer to enter the signing PIN and if the entered PIN is correct, the signing of the hash value of data to be signed, which were received in SMS, is performed and the signature in the new SMS is sent back to the device being connected with the signing application for example through PKCS#11 interfaces.

The format of data from the received SMS which are encrypted, for example by means of RSA with the key size of 1024, is described in the following picture. Due to the short time period while the encrypted data are used, RSA key with the size of 1024 bits is sufficient for SMS encryption.



Encrypted data eData of the type TEdData which are defined as a message M in formatting according to RFC 3447 PKCS#1, version 2.1, chapter 7.2.1 "EME-PKCS1-v1_5".

The formatting is according to RFC 3447 PKCS#1: '00 02' || PS || '00' || M; where PS is a sequence of octets consisting of pseudo randomly generated sequence of octets not containing zero. The sequence of octets must consist of n octets where n is the size of private key modulus intended for decryption. The assumption for the secure communication is based on the fact that the "Authorization code" and "Hash of the message" are known only to the signer before performing the signing and the attacker cannot forge both pieces of data.

Table 1 A type of encrypted data in SMS TEdData in ASN.1

	ASN.1	Info
1.	TEdData ::= SEQUENCE {	
2.	version INTEGER,	Current version 1 (1)
3.	authorizationCode UTF8String,	Authorization code displayed only to the signer.
4.	signingCertDigestAlgorithm AlgorithmIdentifier,	OID of hash algorithm for the hash of the signer's certificate
5.	signingCertDigest Digest	hash of the signer's certificate
6.	messageDigestAlgorithm AlgorithmIdentifier,	OID of hash algorithm for the hash of data being signed in SIM

	ASN.1	Info
7.	messageDigest Digest }	hash of data being signed in SIM
8.	utf8String UTF8String (SIZE (1..MAX))	
9.	AlgorithmIdentifier ::= SEQUENCE { algorithm OBJECT IDENTIFIER, parameters ANY DEFINED BY algorithm OPTIONAL }	For example OID for SHA-1 (1 3 14 3 2 26) and the parameter NULL
10.	Digest ::= OCTET STRING	

6.4 DigiID of the user

The user of the SIM application for signing obtains two certificates issued by the certification authority which falls under the trustworthy root certificate. It means that the user trusts the root certificate and obviously his two certificates which allow his digital identification. To prevent the user of the SIM application for signing from permanent setting up the certificates which he trusts and from configuration of signing and verifying applications, there are issued and stored certificates for him/her in such format and with such data which allow an automatic operation of applications with the minimal requirements for the user.

The encrypting certificate of the owner of the SIM application contains the telephone number for sending SMS to the SIM card in the field *subject* and the signing certificate may also contain an e-mail address to allow the electronic mail signing and verifying. The telephone number format is defined by ITU-T E.123, also mentioned in RFC 3966 and in the document “Qualified Certificate Formats”, Table 5, line 27, <http://www.nbusr.sk/en/electronic-signature/approved-formats/index.html>.

Before the registration process termination when both certificates are issued to the SIM card owner, the SIM card owner is asked to create the first electronic signature which allows to verify the successfulness of registration process and at the same time it creates the DigiID file of trust in certificates for the user so the user can try the functionality of his/her signing SIM application. The SIM card owner creates an integrity signature defined in the document “Qualified Electronic Signature Formats” in Annex D <http://www.nbusr.sk/en/electronic-signature/approved-formats/index.html> where the hash values of signing, encrypting and root certificates are recorded into a textual file. The textual file is signed by internal signature type and is stored as CMS AdES signature type in “DigiID.p7m” file which contains these 3 certificates, as a minimum, in the field *certificates* of the signature.

The data stored in the file “DigiID.p7m” (and signed by the user) are necessary for the whole functionality of PKCS#11 interface through which the signing application requires the electronic signature signing and verification and communicates with the SIM application.

The signing application through PKCS#11 interface which is configured by means of the “DigiID.p7m” file reads the following data:

- the signing certificate (The signing certificate is used to verify signatures and in some applications also to log in the user when data signed by the user himself with his signing certificate are the only one trusted, it means the list of certificates whose references in the form of hash values are in the signed file in “DigiID.p7m”.)
- the signing algorithm and the key size (The signing algorithm and the key size are used by the user for signing and these data are obtained from the certificate of the signature “DigiID.p7m”.)
- the hash algorithm and its parameters (The hash algorithm and its parameters are used by the user for signing and these data are obtained from the signature “DigiID.p7m”.)
- the list of trustworthy certificates (The list of trustworthy certificates are trusted by the signer and can be used automatically for signature verification without permanent trust confirmation by the signer.)
- the encrypting certificate (The encrypting certificate contains the telephone number to which SMS with the request for signing is sent. The request is sent directly by means of SMS through

PKCS#11 interfaces unless the other ways of communication with the SIM application are used (e.g. through USB/serial cable, infrared...)

Table 2 CMS AdES signature in ASN.1 – DigiID

ASN.1
ContentInfo ::= SEQUENCE {
contentType ContentType, -- id-signedData
content [0] EXPLICIT ANY DEFINED BY contentType }

Table 3 Complete SignedData in ASN.1

	ASN.1	Info	Must
1.	SignedData ::= SEQUENCE {		
2.	version CMSVersion,		
3.	digestAlgorithms DigestAlgorithmIdentifiers,		
4.	encapContentInfo SEQUENCE {		
5.	eContentType ContentType,	id-data RFC 3852	
6.	eContent [0] EXPLICIT OCTET STRING OPTIONAL },	Textual file in UTF-8 coding containing URL for file names and hash values of these files of trustworthy certificates.	X
7.	certificates [0] IMPLICIT CertificateSet OPTIONAL,	Signing, encrypting and root certificates.	X
8.	crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,		
9.	signerInfos SET OF		
10.	SEQUENCE { -- SignerInfo	Signature of the SIM card owner with his signing certificate.	X
11.	version CMSVersion,		
12.	sid SignerIdentifier,		
13.	digestAlgorithm DigestAlgorithmIdentifier,		
14.	signedAttrs [0] IMPLICIT SET SIZE(1..MAX) OF		
15.	SEQUENCE { -- Attribute	It MUST contain at least these attributes: content-type, message-digest, signingCertificate or signingCertificateV2.	X
16.	attrType OBJECT IDENTIFIER,		
17.	attrValues SET OF AttributeValue } OPTIONAL,		
18.	signatureAlgorithm SignatureAlgorithmIdentifier,		
19.	signature OCTET STRING, -- SignatureValue		
20.	unsignedAttrs[1]IMPLICIT SET SIZE(1..MAX) OF		
21.	SEQUENCE {		
22.	attrType OBJECT IDENTIFIER,		
23.	attrValues SET OF AttributeValue } OPTIONAL } }		

The user of the signing application which communicates through PKCS#11 interface must have the PKCS#11 library configured before the first use, for example through the graphic interface being accessible by using the menu over the icon located in sysTray where the user sets the “DigiID.p7m” file location or enters URL to obtain “DigiID.p7m”. If the signer uses WEB/WAP to fill out a form and to sign it, he enters URL for “DigiID.p7m” or he loads this file to WEB/WAP through WEB interface. To simplify the signing procedure the operator, who provides the SIM cards containing the signing and encrypting private keys with the application for signing, could offer a standard directory with the “DigiID.p7m” file for example in the following form <http://www.operator.com/sk/telephonenumber/DigiID.p7m>.

7 Signing and signature verification

Before the first signing or signature verification the first step is to setup the path to the “DigiID.p7m” file in WEB/WAP interface or PKCS#11 libraries. Then the interface can automatically read all the important information from the file “DigiID.p7m” (as described in clause 6.4) without asking the answers on questions from the user e.g. about the selection of the user’s signing certificate. Then the processes for the users are the same in the signing application as if a smart card or USB token were connected to a computer. Before the request for signing is submitted, there is generated and displayed the authorization code to the user in PKCS#11 interface; or it is entered by the user as an attribute CKA_LABEL and is inserted into PKCS#11 interface by the application before encryption and by sending the SMS request for signing. After receiving SMS on the mobile device of the signer, SMS is automatically decrypted and the signer is invited to enter the same authorization code as displayed in the signing application. The SIM application performs the signing process only in case the decrypted authorization code from SMS and the entered authorization code on the mobile device are the same. More advanced users can have the hash value of data being signed displayed. But for common users this functionality is accessible only on request by using the button in the dialogue box which requires entering the authorization code.

After that the signer enters the signing PIN and the mobile device signs and sends the signed hash in SMS back to PKCS#11 or WEB interfaces. WEB server applications or PKCS#11 libraries check the signed hash by using the key from signer’s certificate and provide it to the signing application which creates the final electronic signature where the signer's certificate and the signed document are inserted by the signing application.

While verifying signatures the application for signature verification reads the list of certificates stored in “DigiID.p7m” through PKCS#11 interfaces and requires the login confirmation by selection of the signing certificate which is used to verify “DigiID.p7m” signature. So, it may be expected that the signer is able to recognize his/her signing certificate and therefore he/she can confirm that the signing certificate being displayed in his/her application belongs to him/her. After the confirmation the application can trust everything being verified by this certificate. After the confirmation and signature verification of the “DigiID.p7m” file, the application reads the list of trustworthy certificates which are not necessary to be further verified by the user because the signer has confirmed his/her trust by his/her signature and so the user is prevented from difficult and heavily understandable procedure of setting up the trust in different certificates. The signer of the “DigiID.p7m” file and the user of the verifying application is one and the same person so it means that he/she trusts his/her own signature.

After the user has logged in the verifying application according to procedure described in the previous paragraph, verification processes of different signatures whose signing certificates were issued by certification authorities falling under the same trustworthy root certificates which are trusted by the logged-in signer can follow.

Annex A (Informative) Examples of data for signing in SIM

A.1 Textual file of trustworthy certificates signed in DigiID.p7m file

A message consists of a simple sequence of attributes FILE, HASH and NOTICE. FILE and HASH are mandatory attributes, NOTICE is an optional attribute. The sequence must always start by the FILE attribute. One line must contain only one attribute. All lines contain only ASCII characters. The lines are separated by CRLF. The message is stored in the *encapContentInfo* field within *SignedData*. Rules mentioned above define an integrity type of the signature.

The signed message for DigiID creation, signed with the integrity signature (*.p7m), includes information about files containing trusted certificates.

CRLF character (13) + character (10)

FILE = < file name >CRLF

HASH (< algorithm >: < OID of algorithm >) = < file hash – capital letters >CRLF

NOTICE= < note, e. g. a name or type of certificate... >CRLF

Example:

```
FILE=http://www.operator.com/sk/tel00421123123123/sign.cer
HASH(SHA1:1 3 14 3 2 26)=E0FC2B90315FC4C62E85A76B00B8F5FBAF6CC334
NOTICE=Signature
FILE=http://www.operator.com/sk/tel00421123123123/crypt.cer
HASH(SHA1:1 3 14 3 2 26)=541E1CF31EFCB2A650E7AED94A1E5C73DBB8D3B6
NOTICE=Encryption
FILE=http://www.operator.com/sk/tel00421123123123/mobileCA.cer
HASH(SHA1:1 3 14 3 2 26)=23DE8E7554544B8841914A9F7DE79FE59FA236A1
NOTICE=Root CA
```

A.2 DigiID.p7m file encoded in BASE64

```
MIME-Version: 1.0
Content-Type: application/octet-stream; name="DigiID.p7m"
Content-Transfer-Encoding: base64
Content-Disposition: attachment; filename="DigiID.p7m"
```

```
MIIOlgYJKoZIhvcNAQcCoIIOhzCCDoMCAQExCzAJBgUrDgMCGGUAMIIBTAYJKoZIhvcNAQcBoIIB
PQSCATLGSUx6XHNpZ24uY2VyDQpIQVNIKFNIQTE6MSAzIDE0IDMgMiAyNik9RTBGQzJCOTAz
MTVgQzRDNjJFODVBNzZCMDBCOEY1RkRBRjZDQzMzNA0KTK9USUNFPVNpZ25hdHVyZQ0KRklMRT1F
OlxjcnlwdC5jZXINckhBU0goU0hBMT0xIDMgMTQgMyAyIDI2KT0lNDFFMUNGmzFFRkNCmKE2NTBF
N0FFRdk0QTFNFUM3M0RCQjhEM0I2DQpOT1RjQ0U9RW5jcnlwdGlvbG0KRklMRT1F0lxtb2JpbGVD
QS5jZXINckhBU0goU0hBMT0xIDMgMTQgMyAyIDI2KT0yM0RFOEU3NTU0NTQ0Qjg4NDE5MTRBOUY3
REU3OUZFNtLlQ0TlZnKExDQpOT1RjQ0U9Um9vdCBDQ0K0IILMTCCA5kwggKB0AMCAQICCGxda5a2
EvhpmA0GCSqGSIb3DQEBBQUAMEwxCzAJBgNVBAYTAlNLMRMwEQYDVQHEwpcCmF0aXNsYXZzMRQw
...
ggHmAgEBMfgwTDELMAKGA1UEBhMCU0sxZARBgNVBAcTCKJyYXRpc2xhdmExFDASBgNVBAoTC0V4
YWlwGUGT3JnMRiEAYDVQQDEwlnb2JpbGUgQ0ECCGxda5a2EvhpmAMAKGBSs0AwIaBQCggekWGAYJ
KoZIhvcNAQkDMQsGCSqGSIb3DQEHATAcBgkqhkiG9w0BCQUxDxcNMDgWMDA5MTQ0MDAyWjAjbGkq
hkiG9w0BCQQxG9Uq779FuD6G6BxwTm8z0g5RB3R0aIwgYkGCyqGSIb3DQEJEAIMMxOweDB2MHQE
FDuTPBoAIYg9r740Mcb9ecuvQH4fMFwwUKROMEwxCzAJBgNVBAYTAlNLMRMwEQYDVQHEwpcCmF0
aXNsYXZzMRQwEgYDVQKEwtFeGFtcGxlIE9yZzESMBAGA1UEAxMjTW9iaWxliENBAGhsXWuWthL4
aTANBgkqhkiG9w0BAQEFAASBgHgCnJf36UcgejzW5ugSvXnRDVbJTlsmDr9OSlyy5gPqk5UK5MAH
cPlc/tjuNcIWsvtaZUq00MaJPCnWa0nq42dl8TAnnppqKB8m2eLr2Ye7zffJHaeEo11E3ILLNcxy1
mtUTEDbVgR/Y72ZTf8jDvpX1EpJ/BdZFu0CfSNMphjPj
```

A.3 DigiID.p7m file displayed in ASN.1 dump

```
SEQUENCE {
. OBJECT IDENTIFIER signedData (1 2 840 113549 1 7 2)
. [0] {
. . SEQUENCE {
. . . INTEGER 1
. . . SET {
. . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER sha1 (1 3 14 3 2 26)
. . . . . NULL
. . . . . }
. . . . . }
. . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER data (1 2 840 113549 1 7 1)
. . . . . [0] {
. . . . . . OCTET STRING
. . . . . . 'FILE=http://www.operator.com/sk/tel00421123123123/sign.cer'
. . . . . . 'HASH(SHA1:1 3 14 3 2 26)=E0FC2B90315FC4C62E85A76B00B8F5FBAF6CC334'
. . . . . . 'NOTICE=Signature'
. . . . . . 'FILE=http://www.operator.com/sk/tel00421123123123/crypt.cer'
. . . . . . 'HASH(SHA1:1 3 14 3 2 26)=541E1CF31EFCB2A650E7AED94A1E5C73DBB8D3B6'
. . . . . . 'NOTICE=Encryption'
. . . . . . 'FILE=http://www.operator.com/sk/tel00421123123123/mobileCA.cer'
. . . . . . 'HASH(SHA1:1 3 14 3 2 26)=23DE8E7554544B8841914A9F7DE79FE59FA236A1'
. . . . . . 'NOTICE=Root CA'
. . . . . . }
. . . . . . }
. . . . . [0] {
. . . . . . SEQUENCE {
. . . . . . . SEQUENCE {
. . . . . . . [0] {
. . . . . . . . INTEGER 2
. . . . . . . . }
. . . . . . . . INTEGER 6C 5D 6B 96 B6 12 F8 69
. . . . . . . . SEQUENCE {
. . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . sha1withRSAEncryption (1 2 840 113549 1 1 5)
. . . . . . . . . NULL
. . . . . . . . . }
. . . . . . . . SEQUENCE {
. . . . . . . . . SET {
. . . . . . . . . . SEQUENCE {
. . . . . . . . . . . OBJECT IDENTIFIER countryName (2 5 4 6)
. . . . . . . . . . . PrintableString 'SK'
. . . . . . . . . . . }
. . . . . . . . . . . }
. . . . . . . . . . . SET {
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . OBJECT IDENTIFIER localityName (2 5 4 7)
. . . . . . . . . . . . . PrintableString 'Bratislava'
. . . . . . . . . . . . . }
. . . . . . . . . . . . . }
. . . . . . . . . . . . SET {
. . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . OBJECT IDENTIFIER organizationName (2 5 4 10)
. . . . . . . . . . . . . . PrintableString 'Example Org'
. . . . . . . . . . . . . . }
. . . . . . . . . . . . . . }
. . . . . . . . . . . . . SET {
. . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . OBJECT IDENTIFIER commonName (2 5 4 3)
. . . . . . . . . . . . . . . PrintableString 'Mobile CA'
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . }
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . UTCTime 31/03/2008 13:08:58 GMT
. . . . . . . . . . . . . UTCTime 29/03/2018 13:08:58 GMT
. . . . . . . . . . . . . }
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . SET {
. . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . OBJECT IDENTIFIER countryName (2 5 4 6)
. . . . . . . . . . . . . . . PrintableString 'SK'
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . }
. . . . . . . . . . . . SET {
. . . . . . . . . . . . . . SEQUENCE {
```

```
. . . . . OBJECT IDENTIFIER commonName (2 5 4 3)
. . . . . PrintableString 'Peter Rybar'
. . . . . }
. . . . . }
. . . . . SET {
. . . . . SEQUENCE {
. . . . . . . . . . OBJECT IDENTIFIER telephoneNumber (2 5 4 20)
. . . . . . . . . . PrintableString '+421 123 123123'
. . . . . . . . . . }
. . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . . . . . . . rsaEncryption (1 2 840 113549 1 1 1)
. . . . . . . . . . . . . . . . NULL
. . . . . . . . . . . . . . . . }
. . . . . . . . . . BIT STRING, encapsulates {
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . INTEGER
. . . . . . . . . . . . . . . . . . . . . . 00 AF 46 BA A8 35 EF C5 F1 05 32 DA BF F6 16 AC
. . . . . . . . . . . . . . . . . . . . . . 85 A6 14 A1 A1 BB 57 5D B6 5E BF 4F 67 0E 07 88
. . . . . . . . . . . . . . . . . . . . . . 05 5A 09 CF 3A B3 2F C2 45 5A 5E 7E C5 DC A0 13
. . . . . . . . . . . . . . . . . . . . . . E6 31 33 7E 9E 0D BE BB 6B 03 C2 E6 36 7D BB 36
. . . . . . . . . . . . . . . . . . . . . . D4 47 48 9A 11 8C 3C 64 AC 86 AC 53 C3 51 DA 7F
. . . . . . . . . . . . . . . . . . . . . . 54 EA 2E E7 7E 36 32 6F 13 37 4A 96 A1 9B 13 08
. . . . . . . . . . . . . . . . . . . . . . 01 2C 77 B4 BB FD 0C 44 6C 7A EC 54 CB D8 56 E6
. . . . . . . . . . . . . . . . . . . . . . 73 C8 EE 00 78 54 0B 86 34 5B 8D FB 8E B4 7A 05
. . . . . . . . . . . . . . . . . . . . . . ED
. . . . . . . . . . . . . . . . . . . . . . INTEGER 65537
. . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . [3] {
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 92 C5 7A F9 F3 67 DD BF EE 00 98 8B 97 5C C6 DA
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 36 C8 55 56
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER keyUsage (2 5 29 15)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . BOOLEAN TRUE
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . BIT STRING 1 unused bit
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . '0000011'B
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . Error: Spurious zero bits in bitstring.
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER subjectAltName (2 5 29 17)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . [1] 'pr@mailbox.sk'
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER basicConstraints (2 5 29 19)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . BOOLEAN TRUE
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . SEQUENCE {}
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . authorityKeyIdentifier (2 5 29 35)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . [0]
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 01 35 AE 07 FF DE 68 E3 6C F5 8F CB 69 0B 61 61
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . D1 1C B2 70
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER
```

```

. . . . . cRLDistributionPoints (2 5 29 31)
. . . . . OCTET STRING, encapsulates {
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . [0] {
. . . . . [0] {
. . . . . [6] 'http://ca.example.sk/crls/ca20080330.crl'
. . . . . }
. . . . . }
. . . . . }
. . . . . }
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . authorityInfoAuthorization (1 3 6 1 5 5 7 1 1)
. . . . . OCTET STRING, encapsulates {
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . caIssuers (1 3 6 1 5 5 7 48 2)
. . . . . [6] 'http://ca.example.sk/crls/ca20080330.p7c'
. . . . . }
. . . . . }
. . . . . }
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
. . . . . OCTET STRING, encapsulates {
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER anyPolicy (2 5 29 32 0)
. . . . . }
. . . . . }
. . . . . }
. . . . . }
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . sha1withRSAEncryption (1 2 840 113549 1 1 5)
. . . . . NULL
. . . . . }
. . . . . BIT STRING
. . . . . AF 3E 93 E6 A1 83 79 46 3E DE 8D B3 E0 0E 44 37
. . . . . 45 AA 17 B3 6D 10 83 66 D7 F8 F3 56 C3 30 92 67
. . . . . BA 0C 50 4D 54 47 3C 9B D9 79 A0 E7 3F 93 89 71
. . . . . 37 77 07 ED 1D E1 E5 9B 58 BE FB E4 8A 03 5A 2B
. . . . . 0A F3 AB 92 B8 17 D6 74 10 FC C2 51 61 D5 EC 71
. . . . . ED 06 44 1D A0 92 87 16 76 B5 CE 2A 2D B0 A6 97
. . . . . A1 8B 55 1D B1 67 E7 5F CF 6D D0 04 4E 6A 8F 25
. . . . . 03 5B 13 FA 3F 29 79 E4 4B 46 A8 00 D6 99 80 E3
. . . . . AA 98 A9 3B 2F 1F 91 6B 1F 82 9C 74 D9 3D 31 9E
. . . . . C3 D4 7A E3 26 5B ED F9 F6 03 F1 3E 75 03 6E BB
. . . . . 90 F2 DC 0F 5C 51 18 90 88 5A AA 8B 17 7B 38 F7
. . . . . 05 57 29 1B 16 09 19 61 8A 52 1C F9 38 9A 1A 42
. . . . . 0C 84 B7 69 67 6F 89 0F BE 72 AE 81 DB 35 DF 46
. . . . . B8 42 3E E1 5B DD 47 D4 C6 44 83 F2 C9 1A E5 DE
. . . . . D2 E9 BF 86 BE CE 01 BA 65 B8 3D 9A 9A 03 23 46
. . . . . 90 B4 0B C6 A1 DC E0 7A D6 C1 6C 54 5F 2F 83 46
. . . . . }
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . [0] {
. . . . . INTEGER 2
. . . . . }
. . . . . INTEGER 51 AD 7C 91 9B 2F 94 78
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . sha1withRSAEncryption (1 2 840 113549 1 1 5)
. . . . . NULL
. . . . . }
. . . . . SEQUENCE {
. . . . . SET {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER countryName (2 5 4 6)
. . . . . PrintableString 'SK'
. . . . . }
. . . . . }
. . . . . }

```



```
. . . . . SET {
. . . . . SEQUENCE {
. . . . . . . . . . OBJECT IDENTIFIER localityName (2 5 4 7)
. . . . . . . . . . PrintableString 'Bratislava'
. . . . . . . . . . }
. . . . . . . . . . }
. . . . . SET {
. . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . OBJECT IDENTIFIER organizationName (2 5 4 10)
. . . . . . . . . . . . . . . . PrintableString 'Example Org'
. . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . }
. . . . . SET {
. . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . OBJECT IDENTIFIER commonName (2 5 4 3)
. . . . . . . . . . . . . . . . PrintableString 'Mobile CA'
. . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . . UTCTime 31/03/2008 13:08:58 GMT
. . . . . . . . . . UTCTime 29/03/2018 13:08:58 GMT
. . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . . SET {
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER countryName (2 5 4 6)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . PrintableString 'SK'
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . SET {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER commonName (2 5 4 3)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . PrintableString 'Peter Rybar'
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . SET {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER telephoneNumber (2 5 4 20)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . PrintableString '+421 123 123123'
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . rsaEncryption (1 2 840 113549 1 1 1)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . NULL
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . BIT STRING, encapsulates {
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . INTEGER
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 00 C8 0E BD 53 AD 87 84 B4 40 2F A6 19 F4 09 24
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 8C A0 FE 9E 9B 7A C1 C3 B8 CE 72 86 60 6D 74 5F
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 5D 5B C6 90 F5 C4 EA 77 10 B5 68 7B 9D 8B 93 53
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 86 74 CC 96 B1 1E 0E B8 DA 53 AF 32 09 74 7D FF
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . B7 41 41 E7 97 79 24 2C 95 7F 98 40 5D 63 2A 86
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . E6 B3 EC 76 57 8E D7 1C 51 E8 66 52 9C 0B D9 73
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 55 79 4F 96 C6 14 BC 5C 4E 7A A7 93 5B F4 6A 5A
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . A4 55 C1 DD FB 99 00 22 48 7B 3B 65 2E 21 E9 D5
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 1F 24 FD C7 92 D8 01 33 44 DA 52 49 8A D3 A8 F4
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . F5 6B 7F 90 51 9D 53 58 00 0A 47 1A 54 5B CD 2A
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . B0 73 9E 16 0C 0B E1 2F 8C EE 98 65 D2 3F 70 BF
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . B8 A8 B7 EB FB C7 1A 7B 0C 25 E0 13 8A D1 07 2C
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 9B A1 BC BF 80 EA 09 DE 8B BA BF 3D AD 2C A4 85
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 6E 4D 67 D6 3C 75 59 39 C0 2F 4E EC 80 A6 CA D2
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . BC CA F1 AF C7 A4 84 F9 D7 40 E8 EF 7E EB D0 76
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 49 30 B7 E2 3E F7 DD 13 17 5A 60 5C 54 32 AA 44
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . C3
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . INTEGER 65537
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . . . }
. . . . . [3] {
. . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OBJECT IDENTIFIER subjectKeyIdentifier (2 5 29 14)
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . OCTET STRING
. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . A9 5C AB 95 BB 82 AA C2 E6 37 59 0B 11 22 02 40
```

```
. . . . . 63 A3 6F 92
. . . . . }
. . . . . }
. . . . . SEQUENCE {
. . . . . . OBJECT IDENTIFIER keyUsage (2 5 29 15)
. . . . . . . BOOLEAN TRUE
. . . . . . . OCTET STRING, encapsulates {
. . . . . . . . BIT STRING 4 unused bits
. . . . . . . . '1100'B
. . . . . . . }
. . . . . . }
. . . . . SEQUENCE {
. . . . . . . OBJECT IDENTIFIER subjectAltName (2 5 29 17)
. . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . SEQUENCE {
. . . . . . . . . . [1] 'pr@mailbox.sk'
. . . . . . . . . . }
. . . . . . . . }
. . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . OBJECT IDENTIFIER basicConstraints (2 5 29 19)
. . . . . . . . . BOOLEAN TRUE
. . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . SEQUENCE {}
. . . . . . . . . . }
. . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . authorityKeyIdentifier (2 5 29 35)
. . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . [0]
. . . . . . . . . . . . . 01 35 AE 07 FF DE 68 E3 6C F5 8F CB 69 0B 61 61
. . . . . . . . . . . . . D1 1C B2 70
. . . . . . . . . . . . . }
. . . . . . . . . . . . }
. . . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . cRLDistributionPoints (2 5 29 31)
. . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . [0] {
. . . . . . . . . . . . . . [0] {
. . . . . . . . . . . . . . . [6] 'http://ca.example.sk/crls/ca20080330.crl'
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . . . }
. . . . . . . . . . . . }
. . . . . . . . . . . }
. . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . authorityInfoAuthorization (1 3 6 1 5 5 7 1 1)
. . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . OBJECT IDENTIFIER
. . . . . . . . . . . . . . . caIssuers (1 3 6 1 5 5 7 48 2)
. . . . . . . . . . . . . . . [6] 'http://ca.example.sk/crls/ca20080330.p7c'
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . }
. . . . . . . . . . . }
. . . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . . OBJECT IDENTIFIER certificatePolicies (2 5 29 32)
. . . . . . . . . . OCTET STRING, encapsulates {
. . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . SEQUENCE {
. . . . . . . . . . . . . . OBJECT IDENTIFIER anyPolicy (2 5 29 32 0)
. . . . . . . . . . . . . . . }
. . . . . . . . . . . . }
. . . . . . . . . . . }
. . . . . . . . . . }
. . . . . . . . . }
. . . . . SEQUENCE {
. . . . . . . . . OBJECT IDENTIFIER
```



```
. . . . . OBJECT IDENTIFIER
. . . . . signingCertificate (1 2 840 113549 1 9 16 2 12)
. . . . . SET {
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . OCTET STRING
. . . . . 3B 93 3D B3 80 21 88 3D AF BE 28 31 C6 FD 79 CB
. . . . . AF 40 7E 1F
. . . . . SEQUENCE {
. . . . . SEQUENCE {
. . . . . [4] {
. . . . . SEQUENCE {
. . . . . SET {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . countryName (2 5 4 6)
. . . . . PrintableString 'SK'
. . . . . }
. . . . . }
. . . . . SET {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . localityName (2 5 4 7)
. . . . . PrintableString 'Bratislava'
. . . . . }
. . . . . }
. . . . . SET {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . organizationName (2 5 4 10)
. . . . . PrintableString 'Example Org'
. . . . . }
. . . . . }
. . . . . SET {
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER
. . . . . commonName (2 5 4 3)
. . . . . PrintableString 'Mobile CA'
. . . . . }
. . . . . }
. . . . . }
. . . . . }
. . . . . INTEGER 6C 5D 6B 96 B6 12 F8 69
. . . . . }
. . . . . }
. . . . . }
. . . . . }
. . . . . SEQUENCE {
. . . . . OBJECT IDENTIFIER rsaEncryption (1 2 840 113549 1 1 1)
. . . . . NULL
. . . . . }
. . . . . OCTET STRING
. . . . . 78 02 9E 37 F7 E9 47 20 7A 3C D6 E6 E8 12 BD 79
. . . . . D1 0D 56 C9 4E 5B 26 0E BF 4E 4A 5C B2 E6 03 EA
. . . . . 93 95 0A E4 C0 07 70 F2 DC FE D8 EE 35 C2 16 B2
. . . . . FB 5A 65 4A B4 38 C6 89 3D C9 D6 6B 49 EA E3 67
. . . . . 65 F1 30 27 9E 9A 6A 28 1F 26 D9 E2 EB D9 87 BB
. . . . . CD F1 49 1D A7 84 A3 5D 44 DC 82 CD 73 1C B5 9A
. . . . . D5 13 10 36 D5 81 1F D8 EF 66 53 7F C8 C3 BE 95
. . . . . F5 12 92 7F 05 D6 45 BB 40 9F B0 D3 29 1E 3A 49
. . . . . }
. . . . . }
. . . . . }
. . . . . }
```

Annex B (Informative) Bibliography

Basic documents of the Slovak Republic legislation for electronic signature

<http://www.nbusr.sk/en/electronic-signature/legislation/index.html>

Qualified electronic signature formats

<http://www.nbusr.sk/en/electronic-signature/approved-formats/index.html>

Certification path creation and verification

<http://www.nbusr.sk/en/electronic-signature/verification/index.html>

- IETF RFC 4158 "Internet X.509 Public Key Infrastructure: Certification Path Building"

NOTE: Available at <http://www.rfc-archive.org/getrfc.php?rfc=4158>

- IETF RFC 5217 "Multi-Domain PKI Interoperability" July 2008

NOTE: Available at <http://www.rfc-archive.org/getrfc.php?rfc=5217>

- IETF RFC 4853 (2007): "Cryptographic Message Syntax (CMS) Multiple Signer Clarification"
- IETF RFC 3447 (2003): "Public-Key Cryptography Standards (PKCS) #1: RSA Cryptography Specifications Version 2.1"
- ISO/IEC 8825-1:1998, Information technology — ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER)
- ISO/IEC 19794-2:2005, Biometrics — Biometric Data Interchange Formats — Part 2: Finger Minutiae
- IETF RFC 3279 (2002): "Algorithms and Identifiers for the Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile"
- IETF RFC 4055 (2005): "Additional Algorithms and Identifiers for RSA Cryptography for use in the Internet X.509 Public Key Infrastructure - Certificate and Certificate Revocation List (CRL) Profile"
- IETF RFC 3281 (2002): "An Internet Attribute Certificate profile for Authorization"
- IETF RFC 3370 (2002): "Cryptographic Message Syntax (CMS) Algorithms"
- ETSI TR 102 038: "TC Security - Electronic Signatures and Infrastructures (ESI); XML format for signature policies"
- ETSI TS 101 861: "Time stamping profile"
- EN 14890-2:2008: "Application Interface for smart cards used as Secure Signature Creation Devices - Part 2: Additional Services"
- ETSI TS 101 456: "Electronic Signatures and Infrastructures (ESI); Policy requirements for certification authorities issuing qualified certificates"
- ETSI TS 102 042: "Electronic Signatures and Infrastructures (ESI); Policy requirements for certification authorities issuing public key certificates"
- CWA 14167-1: "Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures - Part 1: System Security Requirements"
- CWA 14167-2: "Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures - Part 2: Cryptographic module for CSP Signing Operations with Backup - Protection Profile"
- CWA 14167-3: "Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures - Part 3: Cryptographic module for CSP key generation services - Protection profile (CMCKG-PP)"

- CWA 14167-4: "Security Requirements for Trustworthy Systems Managing Certificates for Electronic Signatures - Part 4: Cryptographic module for CSP signing operations - Protection profile - CMCSO PP"
- W3C Recommendation (10 June 2008): "XML Signature Syntax and Processing (Second Edition)"

NOTE: Available at <http://www.w3.org/TR/xmlsig-core/>

- W3C Recommendation (10 December 2002): "XML Encryption Syntax and Processing"

NOTE: Available at <http://www.w3.org/TR/2002/REC-xmlenc-core-20021210/>

- CWA 14169: "Secure Signature-Creation Devices "EAL 4+""
- IETF RFC 4949 "Internet Security Glossary, Version 2" August 2007

NOTE: Available at <http://www.rfc-archive.org/getrfc.php?rfc=4949>

- NIST X.509 path validation test suite

NOTE: Available at <http://csrc.nist.gov/pki/testing/x509paths.html> <http://csrc.nist.gov/pki/testing/pathdiscovery.html>

- Object Identifier (OID) Repository: ITU-T X.660 & X.670 Recommendation series (or ISO/IEC 9834 series of International Standards)

NOTE: Available at <http://www.oid-info.com/>

- FESA – Forum of European Supervisory Authorities,

NOTE: Available at <http://www.fesa.rtr.at>

- OID tree structure,

NOTE: Available at <http://www.darmstadt.gmd.de/secude/Doc/htm/oidgraph.htm>

- Common ISIS-MTT Specification for interoperable PKI applications. Version 1.1. 16 March 2004
- Internet Draft "X.509 Public Key Infrastructure: Additional Algorithms and Identifiers for DSA and ECDSA"

NOTE: Available at <http://tools.ietf.org/html/draft-ietf-pkix-sha2-dsa-ecdsa-05>

- Internet Draft "X.509 Public Key Infrastructure Time-Stamp Protocol (TSP) "

NOTE: Available at <http://tools.ietf.org/html/draft-ietf-pkix-rfc3161bis-01>

- TeleTrusT Deutschland e. V., "OID-Liste",

NOTE: Available at <http://www.teletrust.de/index.php?id=171>

European Commission <http://ec.europa.eu/>

- Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures

NOTE: Available at

http://eur-lex.europa.eu/smartapi/cgi/sga_doc?smartapi!celexapi!prod!CELEXnumdoc&lg=EN&numdoc=31999L0093&model=guichett

- IDABC stands for Interoperable Delivery of European eGovernment Services to public Administrations, Businesses and Citizens. - eSignature Agenda & Presentations

NOTE: Available at <http://ec.europa.eu/idabc/en/document/7312>

- European Network and Information Security Agency (ENISA)

NOTE: Available at <http://www.enisa.europa.eu/>

- PKIX Status Pages <http://tools.ietf.org/wg/pkix/>

Annex C History

Version	Date of issuing	Note	Editor
Version 0.1.	16 December 2007	First edition (draft)	Peter Rybár, NSA
Version 1.0. No. 2739/2008/IBEP-003	31 March 2008	First edition	Peter Rybár, NSA
Version 1.1 No. 584/2009/IBEP-006	10 April 2009	Specification of authorization code usage	Peter Rybár, NSA