NÁRODNÝ BEZPEČNOSTNÝ ÚRAD

Verzia 2.0

Formáty podpisových politík

19.12.2016

NÁRODNÝ BEZPEČNOSTNÝ ÚRAD

Sekcia informačnej bezpečnosti a elektronického podpisu

Budatínska č. 30, 851 06 Bratislava

http://www.nbu.gov.sk/

e-mail: podatelna@nbu.gov.sk

Obsah

1 Úvod

Členské štáty v súlade s podmienkami podľa vnútroštátneho práva na základe článku 17 ods. 5 nariadenia (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu a o zrušení smernice 1999/93/ES (ďalej len „nariadenie (EÚ) č. 910/2014“ alebo „nariadenie eIDAS“), vydajú pre dôveryhodnú infraštruktúru implementačné štandardy, ktoré mapujú požiadavky vnútroštátneho práva a EÚ legislatívy do technických postupov pre jej vykonateľnosť. Splnenie tejto požiadavky je realizované na viacerých úrovniach:

- Na základe požiadaviek podľa kapitoly II prílohy I vykonávacieho rozhodnutia Komisie (EÚ) 2015/1505 z 8. septembra 2015, ktorým sa ustanovujú technické špecifikácie a formáty týkajúce sa dôveryhodných zoznamov podľa čl. 22 ods. 5 nariadenia Európskeho parlamentu a Rady (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu a vnútroštátneho práva, je orgánom dohľadu (NBÚ) vydaná schéma dohľadu.
- Na základe § 11 ods. 1 písm. k) zákona č. 272/2016 Z. z. o dôveryhodných službách pre elektronické transakcie na vnútornom trhu a o zmene a doplnení niektorých zákonov (zákon o dôveryhodných službách) (ďalej len „zákon č. 272/2016 Z. z.“) vydáva NBÚ štandardy pre oblasť dôveryhodných služieb.

2 Predmet dokumentu

Tento implementačný štandard je vydaný na základe § 11 ods. 1 písm. m) zákona č. 272/2016 Z. z. podľa ktorého NBÚ vydáva, spravuje a zverejňuje podpisové politiky, ktoré obsahujú najmä zoznam algoritmov a ich minimálne parametre pre elektronický podpis a pre národné rozšírenia pre dôveryhodnú infraštruktúru a dôveryhodné služby, ktoré sa od ich zverejnenia NBÚ musia dodržiavať minimálne v styku so subjektom verejného sektora definovaného v článku 3 ods. 7 nariadenia (EÚ) č. 910/2014.

Dokument popisuje rámec používania podpisových politík a základnú množinu podpisových politík, ktorá sa v prípade potreby môže rozšíriť o ďalšie podpisové politiky pre QES a pre národné rozšírenia pre dôveryhodnú infraštruktúru a dôveryhodné služby.

Správne overovanie QES (vrátane elektronických časových pečiatok, certifikačnej cesty a hash hodnôt v objektoch zabezpečujúcich integritu) na základe podpisovej politiky, je kľúčovým predpokladom pre zabezpečenie kompatibility a jednotného prostredia pre QES a pre národné rozšírenia pre dôveryhodnú infraštruktúru a dôveryhodné služby v SR a v krajinách EÚ. Cieľom tohto dokumentu nebolo vytvorenie iba samostatného štandardu pre uvedenú oblasť, ale vytvorenie jednoznačného, minimálneho a záväzného profilu podpisových politík pre poskytovateľov dôveryhodných služieb, tvorcov aplikácií a samozrejme používateľov podľa požiadaviek nariadenia (EÚ) č. 910/2014.

Podpisové politiky sú dostupné na webovom sídle NBÚ:
http://www.nbu.gov.sk/doveryhodne-sluzby/doveryhodna-infrastruktura/podpisove-politiky/index.html

Podpisové politiky sú dôveryhodne zverejňované prostredníctvom zapečateného textového dokumentu, ktorý obsahuje sekvenciu http odkazu a hash hodnoty z odkazovaného dokumentu podpisovej politiky alebo dôveryhodného certifikátu.

Zapečatený zoznam odkazov pre automatické spracovanie je zverejnený na adrese webového sídla NBÚ:
http://ep.nbusr.sk/trusted_data/TrustedList.p7m

3 Odkazy

Odkazy na dokumenty, ktoré definujú použité typy a postupy.

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

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

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

[4] Schéma dohľadu - orgánu dohľadu NBÚ (pozri http://ep.nbusr.sk/kca/tsl/SchemaDohladu.pdf)

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

[6] ETSI TS 119 612 Trusted Lists

[7] NBU Dokumentácia TL X.509 XML schémy pre dôveryhodný zoznam (Pozri http://ep.nbusr.sk/kca/tsl/tlX509XMLSchemaDocumentation.pdf)

[8] ETSI EN 319 411-(1, 2, 3) Policy and security requirements for TSP issuing certificates

[9] RFC 5652 Cryptographic Message Syntax 9-2009

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

[11] Vykonávacie rozhodnutie Komisie (EÚ) 2015/1506 z 8. septembra 2015, ktorým sa ustanovujú špecifikácie týkajúce sa formátov zdokonalených elektronických podpisov a zdokonalených elektronických pečatí, ktoré môžu subjekty verejného sektora uznávať, podľa článkov 27 ods. 5 a 37 ods. 5 nariadenia Európskeho parlamentu a Rady (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu.

[12] Vykonávacie rozhodnutie Komisie (EÚ) 2015/1505 z 8. septembra 2015, ktorým sa ustanovujú technické špecifikácie a formáty týkajúce sa dôveryhodných zoznamov podľa článku 22 ods. 5 nariadenia Európskeho parlamentu a Rady (EÚ) č. 910/2014 o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu.

4 Skratky

ASN.1	Abstract Syntax Notation 1
CA	Certifikačná autorita (Certification Authority) – poskytovateľ kvalifikovanej dôveryhodnej služby vyhotovovania a overovania kvalifikovaných certifikátov pre elektronický podpis, elektronickú pečať a pre autentifikáciu webových sídiel
CAdES	CMS Advanced Electronic Signature
CMS	Cryptographic Message Syntax
CRL	Certificate Revocation List
DER	Distinguished Encoding Rules (for ASN.1)
eIDAS	Nariadenie Európskeho parlamentu a Rady o elektronickej identifikácii a dôveryhodných službách pre elektronické transakcie na vnútornom trhu
ESS	Enhanced Security Services (enhances CMS)
GMT	Greenwich Mean Time
HTTP	Hyper Text Transfer Protocol
ISO	International Organization for Standardization
MIME	Multipurpose Internet Mail Extensions
NBÚ	Národný bezpečnostný úrad
OCSP	Online Certificate Status Protocol
OID	Object Identifier
PAdES	PDF Advanced Electronic Signature
PKIX	Internet X.509 Public Key Infrastructure
QC	Qualified Certificate
QCP SK	Qualified Certificate Policy of Slovakia
QES	Qualified Electronic Signature or Qualified Electronic Seal (kvalifikovaný elektronický podpis alebo kvalifikovaná elektronická pečať)
QSCD	**Q**ualified Electronic **S**ignature/Seal **C**reation **D**evice (zariadenie na vyhotovenie kvalifikovaného elektronického podpisu/pečate)
QTS	Qualified Trust Service (Kvalifikovaná dôveryhodná služba)
TSA	Time-Stamping Authorities
TSP	Time Stamp Protocol
URI	Uniform Resource Identifier
URL	Uniform Resource Locator
XAdES	XML Advanced Electronic Signature
XML	Extensible Markup Language

5 Podpisové politiky

NBÚ schvaľuje podpisové politiky pre QES a pre národné rozšírenia pre dôveryhodnú infraštruktúru a dôveryhodné služby (ďalej len „schválená PP“). Schválené PP sú zverejnené na webovom sídle NBÚ. Podpisová politika je zverejnená v dvoch formátoch:

- v záväznom DER kódovaní podpisovej politiky, ktorej štruktúra v ASN.1 jazyku je uvedená v prílohe A,
- v informatívnom XER kódovaní, ako transformácia z DER kódovania, s textovými komentármi popisujúcimi položky.

Schválené PP a dôveryhodné certifikáty sú zverejnené v textovom dokumente obsahujúcom odkazy na schválené PP a dôveryhodné certifikáty, ktorý je v UTF-8 kódovaní a obsahuje postupnosti riadkov začínajúcich s reťazcami "FILE=", "HASH(SHA256:2 16 840 1 101 3 4 2 1)=" a "NOTICE=", kde v riadku označenom FILE je http adresa na dokument, v HASH riadku je hash odtlačok z DER dokumentu schválenej PP alebo z dôveryhodného certifikátu a v riadku "NOTICE=" je ako prvá hodnota čas konca platnosti schválenej PP alebo dôveryhodného certifikátu, alebo čas predčasného ukončenia platnosti, vo formáte *GeneralizedTime* a medzerou sú oddelené ďalšie nepovinné informácie. Textový dokument odkazov na schválené PP a dôveryhodné certifikáty je zapečatený zdokonalenou elektronickou pečaťou vo formáte CMS AdES.

Ak QES zahrnie jednu z PP, uvedenú v zozname, do podpísaného atribútu alebo podpísaného elementu, aplikácia musí vedieť PP overiť na základe OID PP a

- hash hodnoty uvedenej v poslednej položke PP pri CMS QES teda z DER kódovanej PP *SignaturePolicy* bez hlavičky, teda z položiek *signPolicyHashAlg* a *signPolicyInfo* z PP a
- hash hodnoty z DER kódovanej PP dokumentu pri XML QES a

umožniť aj používateľovi zobraziť OID a hodnotu hash PP.

Proces kontroly:

- aplikácia pre QES, napríklad pri štarte, načíta a overí podpis / pečať http://ep.nbusr.sk/trusted_data/TrustedList.p7m (pomocou KCA NBÚ alebo dôveryhodného zoznamu, v ktorom je certifikát na overenie podpisu / pečate uvedený v <*ServiceTypeIdentifier*> "http://uri.etsi.org/TrstSvc/Svctype/SignaturePolicyAuthority" službe), z ktorého po overení získa zoznam http adries a mien súborov schválených PP, ich hash hodnôt a časov konca platnosti, dokedy sú schválené PP platné,
- podpis / pečať zoznamu schválených PP sa overuje s certifikátom, v ktorom je uvedený OID: 1.3.158.36061701.0.0.1.10.5.0.1 a ktorý vydala KCA NBÚ,
- získaný zoznam hash hodnôt schválených PP a časov konca platnosti schválených PP aplikácia neskôr použije pri overovaní, či overovaný podpis / pečať a ostatné PKI objekty obsahujú bezpečné algoritmy,
- samotný overovaný podpis / pečať a overovaný PKI objekt musí byť vytvorený pred časom konca platnosti schválenej PP. Postup určenia času vytvorenia objektu definuje schéma dohľadu [4] v kapitole 5.3.1 s názvom "SD čl. 32 a 40 nariadenia (EÚ) č. 910/2014".

5.1 Proces výberu podpisovej politiky v aplikácii pre QES

Aplikácia pre vytváranie QES môže mať prednastavenú PP, alebo ponúknuť na výber PP na základe zoznamu PP zverejneného NBÚ na adrese http://ep.nbusr.sk/trusted_data/TrustedList.p7m, pričom sa zobrazí OID PP a text PP z položky *fieldOfApplication*. Na základe výberu PP zo zoznamu sa nastaví PP pre vytváraný podpis alebo pečať.

Podpisovateľ výberom PP určí aj minimálne atribúty podpisu *signerRules*, ktoré musí aplikácia pri podpise vložiť do podpisu alebo pečate. Súčasne výberom PP definuje atribúty vyžadované od overovateľa *verifierRules*, ktoré overovateľ musí doplniť do podpisu, ak PP takéto pravidlá obsahuje. Ak PKI objekt je vytvorený v čase, kedy PP už expirovala, napríklad časová pečiatka pre úroveň A na zabezpečenie dlhodobého overenia a integrity, potom sa použije PP zverejnená NBÚ na adrese http://ep.nbusr.sk/trusted_data/TrustedList.p7m, ktorá bola platná v čase, kedy tento PKI objekt bol preukázateľne vytvorený.

5.2 Proces overenia podpisu v aplikácii pre QES

Aplikácia musí prekontrolovať podpis na základe PP, pod ktorou bol podpis vytvorený. Identifikátor PP je priamo vložený do podpisu podpisovateľom, do podpisovaných atribútov alebo elementov, a tak je podpisom zabezpečené jednoznačné a nepopierateľné identifikovanie vybranej PP. Overenie s inou PP, než akú si zvolil podpisovateľ, je možné ako následné overenie po overení PP zvolenej podpisovateľom alebo tvorcom pečate.

Aplikácia pri overení vypíše informáciu o PP, podľa ktorej overuje daný podpis. Táto informácia musí obsahovať OID, Hash a *fieldOfAplication* PP.

Pri overovaní podpisu na základe PP musia byť splnené všetky požiadavky PP. Napríklad overenie interného podpisu (podpísaný dokument je priamo súčasťou DER podpisu *.p7m) s PP, ktorá je len pre externý podpis <*externalSignedData*> true </*externalSignedData*>, sa označí za neplatné.

V prípade, ak nie sú splnené všetky podmienky PP pre overenie QES, musí byť takto vytvorený QES prehlásený za neplatný.

Podpis musí byť prehlásený za neplatný aj v prípade, že aplikácia pri overovaní nedokáže interpretovať použitú PP, alebo niektoré povinné položky PP.

Pri overovaní QES, musí byť podpis PP platný ku času kontroly overovaného QES.

6 Formáty podpisových politík

6.1 Odporúčania

Uvedené pravidlá PP sú hlavne pre podpisy typu CMS (v ASN.1 kódované v DER) a podpisy XML, kde sa použije mapovanie podpisových atribútov na podpisové elementy.
Kvôli prehľadnosti a jednoduchému zobrazeniu PP v DER je pretransformované DER do XER kódovania.

6.2 Mapovanie podpísaných atribútov pre ASN.1 CMS podpisy z PP do podpísaných elementov pre podpisy vo formáte XML

Pre XML podpisy môžu platiť rovnaké PP ako pre CMS podpisy, kedy sa pri spracovaní DER politiky nahradia atribúty ASN.1 za XML elementy podľa tabuľky č.1.

Tabuľka 1. Niektoré atribúty ASN.1 a XML elementy QES s rovnakým významom obsahu

	ASN.1 atribút	**XML element**
1.	`(id-contentType)`	*(DataObjectFormat)+*
2.	`(id-aa-contentHint)` `.(contentDescription)`	*(DataObjectFormat)+* *.(Description)*
3.	`(id-messageDigest)`	*DigestValue* ako súhrnná hodnota z elementov *Reference*. Viac v štandarde na adrese *https://www.w3.org/TR/xmldsig-core/* .
4.	`(id-signingTime)`	*(SigningTime)*
5.	`(id-aa-ets-signingCertificateV2)` alebo `(id-aa-signingCertificate)`	*(SigningCertificate)*
6.	`(id-aa-ets-sigPolicyId)`	*(SignaturePolicyIdentifier)*
7.	`(id-aa-ets-contentTimestamp)*`	*(AllDataObjectsTimeStamp)** alebo *(IndividualDataObjectsTimeStamp)**
8.	`(id-aa-ets-signerLocation)?`	*(SignatureProductionPlace)?*
9.	`(id-aa-ets-certificateRefs)`	*(CompleteCertificateRefs)*
10.	`(id-aa-ets-revocationRefs)`	*(CompleteRevocationRefs)*
11.	`(id-aa-signatureTimeStampToken)+`	*(SignatureTimeStamp)+*
12.	`((id-aa-ets-escTimeStamp)*` `(id-aa-ets-certCRLTimestamp)*)+`	*((SigAndRefsTimeStamp)* *(RefsOnlyTimeStamp)*)+*
13.	`(id-aa-ets-archiveTimestamp)+`	*(ArchiveTimeStamp)+*
14.	`(id-aa-ets-certValues)`	*(CertificatesValues)*
15.	`(id-aa-ets-revocationValues)`	*(RevocationValues)*
16.	`(id-aa-ets-signerAttr)`	*(SignerRole)*

Výskyt ASN.1 atribútu v CMS podpise a výskyt XML elementu v XML podpise:
( ) - iba jeden raz
( )* - nemusí, ale môže byť aj viackrát
( )? - nemusí, ale môže byť jedenkrát
( )+ - musí byť minimálne jedenkrát

Tabuľka 2. Niektoré algoritmy a atribúty ASN.1 a ich OID

	ASN.1 atribút	OID
1.	id-contentType	1 2 840 113549 1 9 3
2.	id-messageDigest	1 2 840 113549 1 9 4
3.	id-signingTime	1 2 840 113549 1 9 5
4.	Id-aa-ets-otherSigCert	1 2 840 113549 1 9 16 2 19
5.	id-aa-signingCertificate	1 2 840 113549 1 9 16 2 12
6.	id-aa-ets-sigPolicyId	1 2 840 113549 1 9 16 2 15
7.	id-aa-signatureTimeStampToken	1 2 840 113549 1 9 16 2 14
8.	id-aa-ets-certificateRefs	1 2 840 113549 1 9 16 2 21
9.	id-aa-ets-revocationRefs	1 2 840 113549 1 9 16 2 22
10.	id-aa-ets-escTimeStamp	1 2 840 113549 1 9 16 2 25
11.	id-aa-ets-certCRLTimestamp	1 2 840 113549 1 9 16 2 26
12.	id-aa-ets-certValues	1 2 840 113549 1 9 16 2 23
13.	id-aa-ets-revocationValues	1 2 840 113549 1 9 16 2 24
14.	id-aa-ets-archiveTimestamp	1 2 840 113549 1 9 16 2 27
15.	id-aa-ets-contentTimestamp	1 2 840 113549 1 9 16 2 20
16.	id-aa-ets-signerLocation	1 2 840 113549 1 9 16 2 17

6.3 Textový popis základných atribútov podpisovej politiky

Štruktúra ASN.1 DER komponentov PP transformovaná do XER elementov pre jednoduchšie zobrazenie možného základného obsahu PP:

```
<SignaturePolicy>
   <signPolicyHashAlg>
       <algorithm>
              OID algoritmu na výpočet hash hodnoty z položiek podpisovej politiky
              signPolicyHashAlg a signPolicyInfo.
       </algorithm>
   </signPolicyHashAlg>
    <signPolicyInfo>
       <signPolicyIdentifier>
              OID podpisovej politiky.
       </signPolicyIdentifier>
       <dateOfIssue>
              Dátum vydania politiky.
       </dateOfIssue>
       <policyIssuerName>
         <GeneralName>
              <directoryName>
                     Meno vydavateľa politiky.
              </directoryName>
         </GeneralName>
         <GeneralName>
              <uniformResourceIdentifier>
                     Http adresa politiky v DER kódovaní.
              </uniformResourceIdentifier>
         </GeneralName>
       </policyIssuerName>
       <fieldOfApplication>
              Textový popis typu a účelu politiky a pod akým právnym systémom je aplikovateľná.
       </fieldOfApplication>
```

```
<signatureValidationPolicy>
  <signingPeriod>
      <notBefore>
            Čas odkedy je politika použiteľná na podpisovanie.
      </notBefore>
      <notAfter>
            Čas dokedy je politika použiteľná na podpisovanie.
      </notAfter>
  </signingPeriod>
  <commonRules>
      <signerAndVerifierRules>
        <signerRules>
            Požiadavky na podpisovateľa.
            <externalSignedData>
                  true   - externý podpis
                  false  - interný podpis
                         - a ak sa atribút externalSignedData nenachádza, potom
                         je politika určená pre obidva typy podpisov
            </externalSignedData>
            <mandatedSignedAttr>
                  Zoznam povinných podpísaných CMS atribútov.
            </mandatedSignedAttr>
            <mandatedUnsignedAttr>
                  Zoznam povinných nepodpísaných CMS atribútov.
            </mandatedUnsignedAttr>
            <mandatedCertificateRef>
                  <signerOnly/>
                        Do atribútov podpisu Id-aa-ets-otherSigCert, id-aa-
                        signingCertificate alebo XML elementu SigningCertificate
                        je vložený odkaz na certifikát podpisovateľa.
            </mandatedCertificateRef>
            <mandatedCertificateInfo>
                  < fullPath/>
                        Do položky CMS podpisu certificates v bloku SignedData sú
                        vložené certifikáty celej cesty od certifikátu
                        podpisovateľa až po dôveryhodný koreňový certifikát.
            </mandatedCertificateInfo>
        </signerRules>
        <verifierRules>
            <mandatedUnsignedAttr>
                Zoznam povinných nepodpísaných CMS atribútov.
            </mandatedUnsignedAttr>
        </verifierRules>
      </signerAndVerifierRules>
     <signingCertTrustCondition>
        <signerTrustTrees>
            <CertificateTrustPoint>
            Podmienky pre množinu dôveryhodných koreňových certifikátov, na ktorých sa
            musí končiť certifikačná cesta vytvorená od certifikátu podpisovateľa
            alebo certifikátu časovej pečiatky.
                <trustpoint>
                  Množina dôveryhodných koreňových certifikátov, na ktorých musí
                  končiť certifikačná cesta vytvorená od certifikátu podpisovateľa
                  alebo certifikátu časovej pečiatky.
                </trustpoint>
                <acceptablePolicySet>
                    <CertPolicyId>
                    Zoznam OID identifikátorov, ktoré musia byť v každom certifikáte
                    certifikačnej cesty od hĺbky určenej nižšie v
                    requireExplicitPolicy. Kontrolný prienik sa robí zo
                    zjednotenia OID z rozšírení certifikátu CertificatePolicies.
                    </CertPolicyId>
                </acceptablePolicySet>
              <policyConstraints>
                    <requireExplicitPolicy>
```

```
                        Určuje, od akej úrovne certifikačnej cesty smerom k certifikátu
                        podpisovateľa sa začne kontrolovať explicitná podpisová politika
                        (napríklad z acceptablePolicySet).
                        </requireExplicitPolicy>
                     </policyConstraints>
                  </CertificateTrustPoint>
               </signerTrustTrees>
               <signerRevReq>
                  <endCertRevReq>
                        Spôsob kontroly platnosti certifikátu podpisovateľa alebo časovej
                        pečiatky. Napríklad pomocou CRL, OCSP, ...
                  </endCertRevReq>
                  <caCerts>
                        Spôsob kontroly platnosti certifikátov certifikačných autorít
                        pri vytvorenej ceste od certifikátu podpisovateľa alebo časovej
                        pečiatky. Napríklad pomocou CRL, OCSP, ...
                  </caCerts>
               </signerRevReq>
            </signingCertTrustCondition>

            <timeStampTrustCondition>
               <cautionPeriod>
                  Položka obsahuje hodnotu maximálneho čakania, počas ktorého je
                  sprístupnené CRL alebo OCSP odpoveď aktualizovaná (thisUpdate) od času, ku ktorému
                  zisťujeme platnosť certifikátu.
               </cautionPeriod>
            </timeStampTrustCondition>
            <algorithmConstraintSet>
               <signerAlgorithmConstraints>
                  Množina algoritmov a prípadne minimálna veľkosť ich kľúčov použitých na
                  vyhotovenie a overovanie podpisov certifikátov, podpisov časových pečiatok
                  …
               </signerAlgorithmConstraints>
            </algorithmConstraintSet>
         </commonRules>
         <commitmentRules>
            <CommitmentRule>
               <selCommitmentTypes>
                  <CHOICE>
                    <empty></empty>
                  </CHOICE>
               </selCommitmentTypes>
            </CommitmentRule>
         </commitmentRules>
      </signatureValidationPolicy>
   </signPolicyInfo>
   <signPolicyHash>
      Kontrolný Hash integrity podpisovej politiky vypočítaný zo spojených hodnôt zo
      signPolicyHashAlg a signPolicyInfo.
   </signPolicyHash>
</SignaturePolicy>
```

Príloha A (normatívna) ASN.1 pre podpisovú politiku

Vzhľadom na chybne zapísané niektoré štruktúry ASN.1 podpisovej politiky v kapitolách 6 a 11 v ETSI TS 101 733 V1.4.0 alebo v ETSI TR 102 272, sú v tejto prílohe uvedené kompletné ASN.1 moduly pre ASN.1 podpisovú politiku a jej prevod z DER do XER kódovania.

Právny odbor ETSI organizácie povolil uviesť kompletný opravený text ASN.1 PP z ETSI štandardu v NBÚ štandarde na webovom sídle NBÚ.

PP v ASN.1:

```
ETS-ElectronicSignaturePolicies-88syntax { iso(1) member-body(2) us(840)
      rsadsi(113549) pkcs(1) pkcs-9(9) smime(16) id-mod(0) 7 }
DEFINITIONS EXPLICIT TAGS ::=
BEGIN

SignaturePolicy ::= SEQUENCE {
    signPolicyHashAlg    AlgorithmIdentifier,
    signPolicyInfo       SignPolicyInfo,
    signPolicyHash       SignPolicyHash OPTIONAL
}

SignPolicyHash ::= OCTET STRING

SignPolicyInfo ::= SEQUENCE {
    signPolicyIdentifier      SignPolicyId,
    dateOfIssue    GeneralizedTime,
    policyIssuerName     PolicyIssuerName,
    fieldOfApplication   FieldOfApplication,
    signatureValidationPolicy  SignatureValidationPolicy,
    signPolExtensions    SignPolExtensions OPTIONAL
}

SignPolicyId ::= OBJECT IDENTIFIER

PolicyIssuerName ::= GeneralNames

FieldOfApplication ::= DirectoryString

SignatureValidationPolicy ::= SEQUENCE {
    signingPeriod  SigningPeriod,
    commonRules    CommonRules,
    commitmentRules      CommitmentRules,
    signPolExtensions    SignPolExtensions OPTIONAL
}

SigningPeriod ::= SEQUENCE {
    notBefore      GeneralizedTime,
    notAfter       GeneralizedTime OPTIONAL
}

CommonRules ::= SEQUENCE {
    signerAndVeriferRules      [0] SignerAndVerifierRules OPTIONAL,
    signingCertTrustCondition  [1] SigningCertTrustCondition OPTIONAL,
    timeStampTrustCondition    [2] TimestampTrustCondition OPTIONAL,
    attributeTrustCondition    [3] AttributeTrustCondition OPTIONAL,
    algorithmConstraintSet     [4] AlgorithmConstraintSet OPTIONAL,
    signPolExtensions    [5] SignPolExtensions OPTIONAL
}

CommitmentRules ::= SEQUENCE OF CommitmentRule
```

```
CommitmentRule ::= SEQUENCE {
    selCommitmentTypes   SelectedCommitmentTypes,
    signerAndVeriferRules      [0] SignerAndVerifierRules OPTIONAL,
    signingCertTrustCondition  [1] SigningCertTrustCondition OPTIONAL,
    timeStampTrustCondition    [2] TimestampTrustCondition OPTIONAL,
    attributeTrustCondition    [3] AttributeTrustCondition OPTIONAL,
    algorithmConstraintSet     [4] AlgorithmConstraintSet OPTIONAL,
    signPolExtensions    [5] SignPolExtensions OPTIONAL
}

SelectedCommitmentTypes ::= SEQUENCE OF SelectedCommitmentType

SelectedCommitmentType ::= CHOICE {
    empty    NULL,
    recognizedCommitmentType   CommitmentType
}

CommitmentType ::= SEQUENCE {
    identifier     CommitmentTypeIdentifier,
    fieldOfApplication   [0] FieldOfApplication OPTIONAL,
    semantics      [1] DirectoryString OPTIONAL
}

CommitmentTypeIdentifier ::= OBJECT IDENTIFIER

SignerAndVerifierRules ::= SEQUENCE {
    signerRules    SignerRules,
    verifierRules  VerifierRules
}

SignerRules ::= SEQUENCE {
    externalSignedData   BOOLEAN OPTIONAL,
    mandatedSignedAttr   CMSAttrs,
    mandatedUnsignedAttr       CMSAttrs,
    mandatedCertificateRef     [0] CertRefReq DEFAULT signerOnly,
    mandatedCertificateInfo    [1] CertInfoReq DEFAULT none,
    signPolExtensions    [2] SignPolExtensions OPTIONAL
}

CMSAttrs ::= SEQUENCE OF OBJECT IDENTIFIER

CertRefReq ::= ENUMERATED {
    signerOnly(1),
    fullPath(2)
}

CertInfoReq ::= ENUMERATED {
    none(0),
    signerOnly(1),
    fullPath(2)
}

VerifierRules ::= SEQUENCE {
    mandatedUnsignedAttr       MandatedUnsignedAttr,
    signPolExtensions    SignPolExtensions OPTIONAL
}

MandatedUnsignedAttr ::= CMSAttrs

CertificateTrustTrees ::= SEQUENCE OF CertificateTrustPoint

CertificateTrustPoint ::= SEQUENCE {
    trustpoint     Certificate,
```

```
    pathLenConstraint    [0] PathLenConstraint OPTIONAL,
    acceptablePolicySet  [1] AcceptablePolicySet OPTIONAL,
    nameConstraints      [2] NameConstraints OPTIONAL,
    policyConstraints    [3] PolicyConstraints OPTIONAL
}

PathLenConstraint ::= INTEGER (0..MAX)

AcceptablePolicySet ::= SEQUENCE OF CertPolicyId

CertRevReq ::= SEQUENCE {
    endCertRevReq  RevReq,
    caCerts  [0] RevReq
}

RevReq ::= SEQUENCE {
    enuRevReq      EnuRevReq,
    exRevReq       SignPolExtensions OPTIONAL
}

EnuRevReq ::= ENUMERATED {
    clrCheck(0),
    ocspCheck(1),
    bothCheck(2),
    eitherCheck(3),
    noCheck(4),
    other(5)
}

SigningCertTrustCondition ::= SEQUENCE {
    signerTrustTrees     CertificateTrustTrees,
    signerRevReq   CertRevReq
}

TimestampTrustCondition ::= SEQUENCE {
    ttsCertificateTrustTrees   [0] CertificateTrustTrees OPTIONAL,
    ttsRevReq      [1] CertRevReq OPTIONAL,
    ttsNameConstraints   [2] NameConstraints OPTIONAL,
    cautionPeriod  [3] DeltaTime OPTIONAL,
    signatureTimestampDelay    [4] DeltaTime OPTIONAL
}

DeltaTime ::= SEQUENCE {
    deltaSeconds   INTEGER,
    deltaMinutes   INTEGER,
    deltaHours     INTEGER,
    deltaDays      INTEGER
}

AttributeTrustCondition ::= SEQUENCE {
    attributeMandated    BOOLEAN,
    howCertAttribute     HowCertAttribute,
    attrCertificateTrustTrees  [0] CertificateTrustTrees OPTIONAL,
    attrRevReq     [1] CertRevReq OPTIONAL,
    attributeConstraints       [2] AttributeConstraints OPTIONAL
}

HowCertAttribute ::= ENUMERATED {
    claimedAttribute(0),
    certifiedAttribtes(1),
    either(2)
}

AttributeConstraints ::= SEQUENCE {
```

```
    attributeTypeConstraints   [0] AttributeTypeConstraints OPTIONAL,
    attributeValueConstraints  [1] AttributeValueConstraints OPTIONAL
}

AttributeTypeConstraints ::= SEQUENCE OF AttributeType

AttributeValueConstraints ::= SEQUENCE OF AttributeTypeAndValue

AlgorithmConstraintSet ::= SEQUENCE {
    signerAlgorithmConstraints      [0] AlgorithmConstraints OPTIONAL,
    eeCertAlgorithmConstraints      [1] AlgorithmConstraints OPTIONAL,
    caCertAlgorithmConstraints      [2] AlgorithmConstraints OPTIONAL,
    aaCertAlgorithmConstraints      [3] AlgorithmConstraints OPTIONAL,
    tsaCertAlgorithmConstraints     [4] AlgorithmConstraints OPTIONAL
}

AlgorithmConstraints ::= SEQUENCE OF AlgAndLength

AlgAndLength ::= SEQUENCE {
    algID    OBJECT IDENTIFIER,
    minKeyLength   INTEGER OPTIONAL,
    other    SignPolExtensions OPTIONAL
}

SignPolExtensions ::= SEQUENCE OF SignPolExtn

SignPolExtn ::= SEQUENCE {
    extnID   OBJECT IDENTIFIER,
    extnValue      OCTET STRING
}

END

PKIX1Explicit88 { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-explicit(18) }
DEFINITIONS EXPLICIT TAGS ::=
BEGIN

id-pkix OBJECT IDENTIFIER ::= { iso(1) identified-organization(3) dod(6)
internet(1)
                    security(5) mechanisms(5) pkix(7) }

id-pe OBJECT IDENTIFIER ::= { id-pkix 1 }

id-qt OBJECT IDENTIFIER ::= { id-pkix 2 }

id-kp OBJECT IDENTIFIER ::= { id-pkix 3 }

id-ad OBJECT IDENTIFIER ::= { id-pkix 48 }

id-qt-cps OBJECT IDENTIFIER ::= { id-qt 1 }

id-qt-unotice OBJECT IDENTIFIER ::= { id-qt 2 }

id-ad-ocsp OBJECT IDENTIFIER ::= { id-ad 1 }

id-ad-caIssuers OBJECT IDENTIFIER ::= { id-ad 2 }

id-ad-timeStamping OBJECT IDENTIFIER ::= { id-ad 3 }

id-ad-caRepository OBJECT IDENTIFIER ::= { id-ad 5 }

Attribute ::= SEQUENCE {
    type     AttributeType,
```

```
    values   SET OF AttributeValue
}

AttributeType ::= OBJECT IDENTIFIER

AttributeValue ::= ANY

AttributeTypeAndValue ::= SEQUENCE {
    type     AttributeType,
    value    AttributeValue
}

id-at OBJECT IDENTIFIER ::= { joint-iso-ccitt(2) ds(5) 4 }

id-at-name AttributeType ::= { id-at 41 }

id-at-surname AttributeType ::= { id-at 4 }

id-at-givenName AttributeType ::= { id-at 42 }

id-at-initials AttributeType ::= { id-at 43 }

id-at-generationQualifier AttributeType ::= { id-at 44 }

X520name ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-name)),
    printableString      PrintableString (SIZE(1..ub-name)),
    universalString      UniversalString (SIZE(1..ub-name)),
    utf8String     UTF8String (SIZE(1..ub-name)),
    bmpString      BMPString (SIZE(1..ub-name))
}

id-at-commonName AttributeType ::= { id-at 3 }

X520CommonName ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-common-name)),
    printableString      PrintableString (SIZE(1..ub-common-name)),
    universalString      UniversalString (SIZE(1..ub-common-name)),
    utf8String     UTF8String (SIZE(1..ub-common-name)),
    bmpString      BMPString (SIZE(1..ub-common-name))
}

id-at-localityName AttributeType ::= { id-at 7 }

X520LocalityName ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-locality-name)),
    printableString      PrintableString (SIZE(1..ub-locality-name)),
    universalString      UniversalString (SIZE(1..ub-locality-name)),
    utf8String     UTF8String (SIZE(1..ub-locality-name)),
    bmpString      BMPString (SIZE(1..ub-locality-name))
}

id-at-stateOrProvinceName AttributeType ::= { id-at 8 }

X520StateOrProvinceName ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-state-name)),
    printableString      PrintableString (SIZE(1..ub-state-name)),
    universalString      UniversalString (SIZE(1..ub-state-name)),
    utf8String     UTF8String (SIZE(1..ub-state-name)),
    bmpString      BMPString (SIZE(1..ub-state-name))
}

id-at-organizationName AttributeType ::= { id-at 10 }
```

```
X520OrganizationName ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-organization-name)),
    printableString      PrintableString (SIZE(1..ub-organization-name)),
    universalString      UniversalString (SIZE(1..ub-organization-name)),
    utf8String     UTF8String (SIZE(1..ub-organization-name)),
    bmpString      BMPString (SIZE(1..ub-organization-name))
}

id-at-organizationalUnitName AttributeType ::= { id-at 11 }

X520OrganizationalUnitName ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-organizational-unit-name)),
    printableString      PrintableString (SIZE(1..ub-organizational-unit-name)),
    universalString      UniversalString (SIZE(1..ub-organizational-unit-name)),
    utf8String     UTF8String (SIZE(1..ub-organizational-unit-name)),
    bmpString      BMPString (SIZE(1..ub-organizational-unit-name))
}

id-at-title AttributeType ::= { id-at 12 }

X520Title ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-title)),
    printableString      PrintableString (SIZE(1..ub-title)),
    universalString      UniversalString (SIZE(1..ub-title)),
    utf8String     UTF8String (SIZE(1..ub-title)),
    bmpString      BMPString (SIZE(1..ub-title))
}

id-at-dnQualifier AttributeType ::= { id-at 46 }

X520dnQualifier ::= PrintableString

id-at-countryName AttributeType ::= { id-at 6 }

X520countryName ::= PrintableString (SIZE(2))

id-at-serialNumber AttributeType ::= { id-at 5 }

X520SerialNumber ::= PrintableString (SIZE(1..ub-serial-number))

id-at-pseudonym AttributeType ::= { id-at 65 }

X520Pseudonym ::= CHOICE {
    teletexString  TeletexString (SIZE(1..ub-pseudonym)),
    printableString      PrintableString (SIZE(1..ub-pseudonym)),
    universalString      UniversalString (SIZE(1..ub-pseudonym)),
    utf8String     UTF8String (SIZE(1..ub-pseudonym)),
    bmpString      BMPString (SIZE(1..ub-pseudonym))
}

id-domainComponent AttributeType ::= { 0 9 2342 19200300 100 1 25 }

DomainComponent ::= IA5String

pkcs-9 OBJECT IDENTIFIER ::= { iso(1) member-body(2) us(840) rsadsi(113549)
pkcs(1) 9 }

id-emailAddress AttributeType ::= { pkcs-9 1 }

EmailAddress ::= IA5String (SIZE(1..ub-emailaddress-length))

Name ::= CHOICE {
    rdnSequence    RDNSequence
}
```

```
RDNSequence ::= SEQUENCE OF RelativeDistinguishedName

DistinguishedName ::= RDNSequence

RelativeDistinguishedName ::= SET SIZE(1..MAX) OF AttributeTypeAndValue

DirectoryString ::= CHOICE {
    teletexString  TeletexString (SIZE(1..MAX)),
    printableString      PrintableString (SIZE(1..MAX)),
    universalString      UniversalString (SIZE(1..MAX)),
    utf8String     UTF8String (SIZE(1..MAX)),
    bmpString      BMPString (SIZE(1..MAX))
}

Certificate ::= SEQUENCE {
    tbsCertificate       TBSCertificate,
    signatureAlgorithm   AlgorithmIdentifier,
    signature      BIT STRING
}

TBSCertificate ::= SEQUENCE {
    version  [0] Version DEFAULT v1,
    serialNumber   CertificateSerialNumber,
    signature      AlgorithmIdentifier,
    issuer   Name,
    validity       Validity,
    subject  Name,
    subjectPublicKeyInfo       SubjectPublicKeyInfo,
    issuerUniqueID       [1] IMPLICIT UniqueIdentifier OPTIONAL,
    subjectUniqueID      [2] IMPLICIT UniqueIdentifier OPTIONAL,
    extensions     [3] Extensions OPTIONAL
}

Version ::= INTEGER {
    v1(0),
    v2(1),
    v3(2)
}

CertificateSerialNumber ::= INTEGER

Validity ::= SEQUENCE {
    notBefore      Time,
    notAfter       Time
}

Time ::= CHOICE {
    utcTime  UTCTime,
    generalTime    GeneralizedTime
}

UniqueIdentifier ::= BIT STRING

SubjectPublicKeyInfo ::= SEQUENCE {
    algorithm      AlgorithmIdentifier,
    subjectPublicKey     BIT STRING
}

Extensions ::= SEQUENCE SIZE(1..MAX) OF Extension

Extension ::= SEQUENCE {
    extnID   OBJECT IDENTIFIER,
    critical       BOOLEAN DEFAULT FALSE,
```

```
    extnValue      OCTET STRING
}

CertificateList ::= SEQUENCE {
    tbsCertList    TBSCertList,
    signatureAlgorithm   AlgorithmIdentifier,
    signature      BIT STRING
}

TBSCertList ::= SEQUENCE {
    version  Version OPTIONAL,
    signature      AlgorithmIdentifier,
    issuer   Name,
    thisUpdate     Time,
    nextUpdate     Time OPTIONAL,
    revokedCertificates  SEQUENCE OF SEQUENCE {
            userCertificate    CertificateSerialNumber,
            revocationDate     Time,
            crlEntryExtensions      Extensions OPTIONAL
        } OPTIONAL,
    crlExtensions  [0] Extensions OPTIONAL
}

AlgorithmIdentifier ::= SEQUENCE {
    algorithm      OBJECT IDENTIFIER,
    parameters     ANY algorithm OPTIONAL
}

ORAddress ::= SEQUENCE {
    built-in-standard-attributes    BuiltInStandardAttributes,
    built-in-domain-defined-attributes    BuiltInDomainDefinedAttributes
OPTIONAL,
    extension-attributes       ExtensionAttributes OPTIONAL
}

BuiltInStandardAttributes ::= SEQUENCE {
    country-name   CountryName OPTIONAL,
    administration-domain-name      AdministrationDomainName OPTIONAL,
    network-address      [0] IMPLICIT NetworkAddress OPTIONAL,
    terminal-identifier  [1] IMPLICIT TerminalIdentifier OPTIONAL,
    private-domain-name  [2] PrivateDomainName OPTIONAL,
    organization-name    [3] IMPLICIT OrganizationName OPTIONAL,
    numeric-user-identifier    [4] IMPLICIT NumericUserIdentifier OPTIONAL,
    personal-name  [5] IMPLICIT PersonalName OPTIONAL,
    organizational-unit-names  [6] IMPLICIT OrganizationalUnitNames OPTIONAL
}

CountryName ::= [APPLICATION 1] CHOICE {
    x121-dcc-code  NumericString (SIZE(ub-country-name-numeric-length)),
    iso-3166-alpha2-code       PrintableString (SIZE(ub-country-name-alpha-
length))
}

AdministrationDomainName ::= [APPLICATION 2] CHOICE {
    numeric  NumericString (SIZE(0..ub-domain-name-length)),
    printable      PrintableString (SIZE(0..ub-domain-name-length))
}

NetworkAddress ::= X121Address

X121Address ::= NumericString (SIZE(1..ub-x121-address-length))

TerminalIdentifier ::= PrintableString (SIZE(1..ub-terminal-id-length))
```

```
PrivateDomainName ::= CHOICE {
    numeric  NumericString (SIZE(1..ub-domain-name-length)),
    printable      PrintableString (SIZE(1..ub-domain-name-length))
}

OrganizationName ::= PrintableString (SIZE(1..ub-organization-name-length))

NumericUserIdentifier ::= NumericString (SIZE(1..ub-numeric-user-id-length))

PersonalName ::= SET {
    surname  [0] IMPLICIT PrintableString (SIZE(1..ub-surname-length)),
    given-name     [1] IMPLICIT PrintableString (SIZE(1..ub-given-name-length))
OPTIONAL,
    initials       [2] IMPLICIT PrintableString (SIZE(1..ub-initials-length))
OPTIONAL,
    generation-qualifier       [3] IMPLICIT PrintableString (SIZE(1..ub-
generation-qualifier-length)) OPTIONAL
}

OrganizationalUnitNames ::= SEQUENCE SIZE(1..ub-organizational-units) OF
OrganizationalUnitName

OrganizationalUnitName ::= PrintableString (SIZE(1..ub-organizational-unit-name-
length))

BuiltInDomainDefinedAttributes ::= SEQUENCE SIZE(1..ub-domain-defined-
attributes) OF BuiltInDomainDefinedAttribute

BuiltInDomainDefinedAttribute ::= SEQUENCE {
    type     PrintableString (SIZE(1..ub-domain-defined-attribute-type-length)),
    value    PrintableString (SIZE(1..ub-domain-defined-attribute-value-length))
}

ExtensionAttributes ::= SET SIZE(1..ub-extension-attributes) OF
ExtensionAttribute

ExtensionAttribute ::= SEQUENCE {
    extension-attribute-type   [0] IMPLICIT INTEGER (0..ub-extension-
attributes),
    extension-attribute-value  [1] ANY extension-attribute-type
}

common-name INTEGER ::= 1

CommonName ::= PrintableString (SIZE(1..ub-common-name-length))

teletex-common-name INTEGER ::= 2

TeletexCommonName ::= TeletexString (SIZE(1..ub-common-name-length))

teletex-organization-name INTEGER ::= 3

TeletexOrganizationName ::= TeletexString (SIZE(1..ub-organization-name-length))

teletex-personal-name INTEGER ::= 4

TeletexPersonalName ::= SET {
    surname  [0] IMPLICIT TeletexString (SIZE(1..ub-surname-length)),
    given-name     [1] IMPLICIT TeletexString (SIZE(1..ub-given-name-length))
OPTIONAL,
    initials       [2] IMPLICIT TeletexString (SIZE(1..ub-initials-length))
OPTIONAL,
    generation-qualifier       [3] IMPLICIT TeletexString (SIZE(1..ub-
generation-qualifier-length)) OPTIONAL
```

```
}

teletex-organizational-unit-names INTEGER ::= 5

TeletexOrganizationalUnitNames ::= SEQUENCE SIZE(1..ub-organizational-units) OF
TeletexOrganizationalUnitName

TeletexOrganizationalUnitName ::= TeletexString (SIZE(1..ub-organizational-unit-
name-length))

pds-name INTEGER ::= 7

PDSName ::= PrintableString (SIZE(1..ub-pds-name-length))

physical-delivery-country-name INTEGER ::= 8

PhysicalDeliveryCountryName ::= CHOICE {
    x121-dcc-code  NumericString (SIZE(ub-country-name-numeric-length)),
    iso-3166-alpha2-code      PrintableString (SIZE(ub-country-name-alpha-
length))
}

postal-code INTEGER ::= 9

PostalCode ::= CHOICE {
    numeric-code   NumericString (SIZE(1..ub-postal-code-length)),
    printable-code       PrintableString (SIZE(1..ub-postal-code-length))
}

physical-delivery-office-name INTEGER ::= 10

PhysicalDeliveryOfficeName ::= PDSParameter

physical-delivery-office-number INTEGER ::= 11

PhysicalDeliveryOfficeNumber ::= PDSParameter

extension-OR-address-components INTEGER ::= 12

ExtensionORAddressComponents ::= PDSParameter

physical-delivery-personal-name INTEGER ::= 13

PhysicalDeliveryPersonalName ::= PDSParameter

physical-delivery-organization-name INTEGER ::= 14

PhysicalDeliveryOrganizationName ::= PDSParameter

extension-physical-delivery-address-components INTEGER ::= 15

ExtensionPhysicalDeliveryAddressComponents ::= PDSParameter

unformatted-postal-address INTEGER ::= 16

UnformattedPostalAddress ::= SET {
    printable-address    SEQUENCE SIZE(1..ub-pds-physical-address-lines) OF
PrintableString (SIZE(1..ub-pds-parameter-length)) OPTIONAL,
    teletex-string       TeletexString (SIZE(1..ub-unformatted-address-length))
OPTIONAL
}

street-address INTEGER ::= 17
```

```
StreetAddress ::= PDSParameter

post-office-box-address INTEGER ::= 18

PostOfficeBoxAddress ::= PDSParameter

poste-restante-address INTEGER ::= 19

PosteRestanteAddress ::= PDSParameter

unique-postal-name INTEGER ::= 20

UniquePostalName ::= PDSParameter

local-postal-attributes INTEGER ::= 21

LocalPostalAttributes ::= PDSParameter

PDSParameter ::= SET {
    printable-string     PrintableString (SIZE(1..ub-pds-parameter-length))
OPTIONAL,
    teletex-string       TeletexString (SIZE(1..ub-pds-parameter-length))
OPTIONAL
}

extended-network-address INTEGER ::= 22

ExtendedNetworkAddress ::= CHOICE {
    e163-4-address       SEQUENCE {
        number     [0] IMPLICIT NumericString (SIZE(1..ub-e163-4-number-
length)),
        sub-address      [1] IMPLICIT NumericString (SIZE(1..ub-e163-4-sub-
address-length)) OPTIONAL
    },
    psap-address   [0] IMPLICIT PresentationAddress
}

PresentationAddress ::= SEQUENCE {
    pSelector      [0] EXPLICIT OCTET STRING OPTIONAL,
    sSelector      [1] EXPLICIT OCTET STRING OPTIONAL,
    tSelector      [2] EXPLICIT OCTET STRING OPTIONAL,
    nAddresses     [3] EXPLICIT SET SIZE(1..MAX) OF OCTET STRING
}

terminal-type INTEGER ::= 23

TerminalType ::= INTEGER {
    telex(3),
    teletex(4),
    g3-facsimile(5),
    g4-facsimile(6),
    ia5-terminal(7),
    videotex(8)
} (0..ub-integer-options)

teletex-domain-defined-attributes INTEGER ::= 6

TeletexDomainDefinedAttributes ::= SEQUENCE SIZE(1..ub-domain-defined-
attributes) OF TeletexDomainDefinedAttribute

TeletexDomainDefinedAttribute ::= SEQUENCE {
    type     TeletexString (SIZE(1..ub-domain-defined-attribute-type-length)),
    value    TeletexString (SIZE(1..ub-domain-defined-attribute-value-length))
}
```

```
ub-name INTEGER ::= 32768

ub-common-name INTEGER ::= 64

ub-locality-name INTEGER ::= 128

ub-state-name INTEGER ::= 128

ub-organization-name INTEGER ::= 64

ub-organizational-unit-name INTEGER ::= 64

ub-title INTEGER ::= 64

ub-serial-number INTEGER ::= 64

ub-match INTEGER ::= 128

ub-emailaddress-length INTEGER ::= 128

ub-common-name-length INTEGER ::= 64

ub-country-name-alpha-length INTEGER ::= 2

ub-country-name-numeric-length INTEGER ::= 3

ub-domain-defined-attributes INTEGER ::= 4

ub-domain-defined-attribute-type-length INTEGER ::= 8

ub-domain-defined-attribute-value-length INTEGER ::= 128

ub-domain-name-length INTEGER ::= 16

ub-extension-attributes INTEGER ::= 256

ub-e163-4-number-length INTEGER ::= 15

ub-e163-4-sub-address-length INTEGER ::= 40

ub-generation-qualifier-length INTEGER ::= 3

ub-given-name-length INTEGER ::= 16

ub-initials-length INTEGER ::= 5

ub-integer-options INTEGER ::= 256

ub-numeric-user-id-length INTEGER ::= 32

ub-organization-name-length INTEGER ::= 64

ub-organizational-unit-name-length INTEGER ::= 32

ub-organizational-units INTEGER ::= 4

ub-pds-name-length INTEGER ::= 16

ub-pds-parameter-length INTEGER ::= 30

ub-pds-physical-address-lines INTEGER ::= 6

ub-postal-code-length INTEGER ::= 16
```

```
ub-pseudonym INTEGER ::= 128

ub-surname-length INTEGER ::= 40

ub-terminal-id-length INTEGER ::= 24

ub-unformatted-address-length INTEGER ::= 180

ub-x121-address-length INTEGER ::= 16

END

PKIX1Implicit88 { iso(1) identified-organization(3) dod(6) internet(1)
      security(5) mechanisms(5) pkix(7) id-mod(0) id-pkix1-implicit(19) }
DEFINITIONS IMPLICIT TAGS ::=
BEGIN

id-ce OBJECT IDENTIFIER ::= {joint-iso-ccitt(2) ds(5) 29}

id-ce-authorityKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 35 }

AuthorityKeyIdentifier ::= SEQUENCE {
    keyIdentifier  [0] KeyIdentifier OPTIONAL,
    authorityCertIssuer  [1] GeneralNames OPTIONAL,
    authorityCertSerialNumber  [2] CertificateSerialNumber OPTIONAL
}

KeyIdentifier ::= OCTET STRING

id-ce-subjectKeyIdentifier OBJECT IDENTIFIER ::= { id-ce 14 }

SubjectKeyIdentifier ::= KeyIdentifier

id-ce-keyUsage OBJECT IDENTIFIER ::= { id-ce 15 }

KeyUsage ::= BIT STRING {
    digitalSignature(0),
    nonRepudiation(1),
    keyEncipherment(2),
    dataEncipherment(3),
    keyAgreement(4),
    keyCertSign(5),
    cRLSign(6),
    encipherOnly(7),
    decipherOnly(8)
}

id-ce-privateKeyUsagePeriod OBJECT IDENTIFIER ::= { id-ce 16 }

PrivateKeyUsagePeriod ::= SEQUENCE {
    notBefore      [0] GeneralizedTime OPTIONAL,
    notAfter       [1] GeneralizedTime OPTIONAL
}

id-ce-certificatePolicies OBJECT IDENTIFIER ::= { id-ce 32 }

anyPolicy OBJECT IDENTIFIER ::= { id-ce-certificatePolicies 0 }

CertificatePolicies ::= SEQUENCE SIZE(1..MAX) OF PolicyInformation

PolicyInformation ::= SEQUENCE {
    policyIdentifier     CertPolicyId,
    policyQualifiers     SEQUENCE SIZE(1..MAX) OF PolicyQualifierInfo OPTIONAL
```

```
}

CertPolicyId ::= OBJECT IDENTIFIER

PolicyQualifierInfo ::= SEQUENCE {
    policyQualifierId    PolicyQualifierId,
    qualifier      ANY policyQualifierId
}

PolicyQualifierId ::= OBJECT IDENTIFIER (id-qt-cps | id-qt-unotice)

CPSuri ::= IA5String

UserNotice ::= SEQUENCE {
    noticeRef      NoticeReference OPTIONAL,
    explicitText   DisplayText OPTIONAL
}

NoticeReference ::= SEQUENCE {
    organization   DisplayText,
    noticeNumbers  SEQUENCE OF INTEGER
}

DisplayText ::= CHOICE {
    ia5String      IA5String (SIZE(1..200)),
    visibleString  VisibleString (SIZE(1..200)),
    bmpString      BMPString (SIZE(1..200)),
    utf8String     UTF8String (SIZE(1..200))
}

id-ce-policyMappings OBJECT IDENTIFIER ::= { id-ce 33 }

PolicyMappings ::= SEQUENCE SIZE(1..MAX) OF SEQUENCE {
        issuerDomainPolicy    CertPolicyId,
        subjectDomainPolicy   CertPolicyId
    }

id-ce-subjectAltName OBJECT IDENTIFIER ::= { id-ce 17 }

SubjectAltName ::= GeneralNames

GeneralNames ::= SEQUENCE SIZE(1..MAX) OF GeneralName

GeneralName ::= CHOICE {
    otherName      [0] AnotherName,
    rfc822Name     [1] IA5String,
    dNSName  [2] IA5String,
    x400Address    [3] ORAddress,
    directoryName  [4] Name,
    ediPartyName   [5] EDIPartyName,
    uniformResourceIdentifier  [6] IA5String,
    iPAddress      [7] OCTET STRING,
    registeredID   [8] OBJECT IDENTIFIER
}

AnotherName ::= SEQUENCE {
    type-id  OBJECT IDENTIFIER,
    value    [0] EXPLICIT ANY type-id
}

EDIPartyName ::= SEQUENCE {
    nameAssigner   [0] DirectoryString OPTIONAL,
    partyName      [1] DirectoryString
}
```

```
id-ce-issuerAltName OBJECT IDENTIFIER ::= { id-ce 18 }

IssuerAltName ::= GeneralNames

id-ce-subjectDirectoryAttributes OBJECT IDENTIFIER ::= { id-ce 9 }

SubjectDirectoryAttributes ::= SEQUENCE SIZE(1..MAX) OF Attribute

id-ce-basicConstraints OBJECT IDENTIFIER ::= { id-ce 19 }

BasicConstraints ::= SEQUENCE {
    cA       BOOLEAN DEFAULT FALSE,
    pathLenConstraint    INTEGER (0..MAX) OPTIONAL
}

id-ce-nameConstraints OBJECT IDENTIFIER ::= { id-ce 30 }

NameConstraints ::= SEQUENCE {
    permittedSubtrees    [0] GeneralSubtrees OPTIONAL,
    excludedSubtrees     [1] GeneralSubtrees OPTIONAL
}

GeneralSubtrees ::= SEQUENCE SIZE(1..MAX) OF GeneralSubtree

GeneralSubtree ::= SEQUENCE {
    base     GeneralName,
    minimum  [0] BaseDistance DEFAULT 0,
    maximum  [1] BaseDistance OPTIONAL
}

BaseDistance ::= INTEGER (0..MAX)

id-ce-policyConstraints OBJECT IDENTIFIER ::= { id-ce 36 }

PolicyConstraints ::= SEQUENCE {
    requireExplicitPolicy     [0] SkipCerts OPTIONAL,
    inhibitPolicyMapping      [1] SkipCerts OPTIONAL
}

SkipCerts ::= INTEGER (0..MAX)

id-ce-cRLDistributionPoints OBJECT IDENTIFIER ::= {id-ce 31}

CRLDistributionPoints ::= SEQUENCE SIZE(1..MAX) OF DistributionPoint

DistributionPoint ::= SEQUENCE {
    distributionPoint    [0] DistributionPointName OPTIONAL,
    reasons  [1] ReasonFlags OPTIONAL,
    cRLIssuer      [2] GeneralNames OPTIONAL
}

DistributionPointName ::= CHOICE {
    fullName       [0] GeneralNames,
    nameRelativeToCRLIssuer    [1] RelativeDistinguishedName
}

ReasonFlags ::= BIT STRING {
    unused(0),
    keyCompromise(1),
    cACompromise(2),
    affiliationChanged(3),
    superseded(4),
    cessationOfOperation(5),
```

```
    certificateHold(6),
    privilegeWithdrawn(7),
    aACompromise(8)
}

id-ce-extKeyUsage OBJECT IDENTIFIER ::= {id-ce 37}

ExtKeyUsageSyntax ::= SEQUENCE SIZE(1..MAX) OF KeyPurposeId

KeyPurposeId ::= OBJECT IDENTIFIER

anyExtendedKeyUsage OBJECT IDENTIFIER ::= { id-ce-extKeyUsage 0 }

id-kp-serverAuth OBJECT IDENTIFIER ::= { id-kp 1 }

id-kp-clientAuth OBJECT IDENTIFIER ::= { id-kp 2 }

id-kp-codeSigning OBJECT IDENTIFIER ::= { id-kp 3 }

id-kp-emailProtection OBJECT IDENTIFIER ::= { id-kp 4 }

id-kp-timeStamping OBJECT IDENTIFIER ::= { id-kp 8 }

id-kp-OCSPSigning OBJECT IDENTIFIER ::= { id-kp 9 }

id-ce-inhibitAnyPolicy OBJECT IDENTIFIER ::= { id-ce 54 }

InhibitAnyPolicy ::= SkipCerts

id-ce-freshestCRL OBJECT IDENTIFIER ::= { id-ce 46 }

FreshestCRL ::= CRLDistributionPoints

id-pe-authorityInfoAccess OBJECT IDENTIFIER ::= { id-pe 1 }

AuthorityInfoAccessSyntax ::= SEQUENCE SIZE(1..MAX) OF AccessDescription

AccessDescription ::= SEQUENCE {
    accessMethod   OBJECT IDENTIFIER,
    accessLocation       GeneralName
}

id-pe-subjectInfoAccess OBJECT IDENTIFIER ::= { id-pe 11 }

SubjectInfoAccessSyntax ::= SEQUENCE SIZE(1..MAX) OF AccessDescription

id-ce-cRLNumber OBJECT IDENTIFIER ::= { id-ce 20 }

CRLNumber ::= INTEGER (0..MAX)

id-ce-issuingDistributionPoint OBJECT IDENTIFIER ::= { id-ce 28 }

IssuingDistributionPoint ::= SEQUENCE {
    distributionPoint    [0] DistributionPointName OPTIONAL,
    onlyContainsUserCerts      [1] BOOLEAN DEFAULT FALSE,
    onlyContainsCACerts  [2] BOOLEAN DEFAULT FALSE,
    onlySomeReasons      [3] ReasonFlags OPTIONAL,
    indirectCRL    [4] BOOLEAN DEFAULT FALSE,
    onlyContainsAttributeCerts      [5] BOOLEAN DEFAULT FALSE
}

id-ce-deltaCRLIndicator OBJECT IDENTIFIER ::= { id-ce 27 }

BaseCRLNumber ::= CRLNumber
```

```
id-ce-cRLReasons OBJECT IDENTIFIER ::= { id-ce 21 }

CRLReason ::= ENUMERATED {
    unspecified(0),
    keyCompromise(1),
    cACompromise(2),
    affiliationChanged(3),
    superseded(4),
    cessationOfOperation(5),
    certificateHold(6),
    removeFromCRL(8),
    privilegeWithdrawn(9),
    aACompromise(10)
}

id-ce-certificateIssuer OBJECT IDENTIFIER ::= { id-ce 29 }

CertificateIssuer ::= GeneralNames

id-ce-holdInstructionCode OBJECT IDENTIFIER ::= { id-ce 23 }

HoldInstructionCode ::= OBJECT IDENTIFIER

holdInstruction OBJECT IDENTIFIER ::= {joint-iso-itu-t(2) member-body(2) us(840)
x9cm(10040) 2}

id-holdinstruction-none OBJECT IDENTIFIER ::= {holdInstruction 1}

id-holdinstruction-callissuer OBJECT IDENTIFIER ::= {holdInstruction 2}

id-holdinstruction-reject OBJECT IDENTIFIER ::= {holdInstruction 3}

id-ce-invalidityDate OBJECT IDENTIFIER ::= { id-ce 24 }

InvalidityDate ::= GeneralizedTime

END
```

Príloha B (informatívna) Zoznam použitej literatúry

Základná legislatíva Slovenskej republiky a EÚ pre elektronický podpis:
http://www.nbu.gov.sk/urad/pravne-predpisy/doveryhodne-sluzby/index.html

Štandardy NBÚ:
http://www.nbu.gov.sk/doveryhodne-sluzby/standardy/index.html

Zostavenie certifikačnej cesty a overenie platnosti certifikátov:
http://www.nbu.gov.sk/doveryhodne-sluzby/dohlad/index.html

Príloha C História

Verzia	Dátum	Poznámka	Vypracoval
V 1.0	24.8.2005	Prvé vydanie	Ing. Peter Rybár, NBÚ RNDr. Július Šiška, PhD., KPMG
Verzia 1.1	6.11.2005	Jednotný formát NBÚ dokumentov	Ing. Peter Rybár, NBÚ
Verzia 2.0 7561/2016/IBEP/OA-001	19.12.2016	Zmena podľa zákona č. 272/2016 Z. z.	Ing. Peter Rybár, NBÚ