Viacnásobné elektronické podpisy v praxi

DIPLOMOVÁ PRÁCA

Daniel Chromek 2006

Fakulta matematiky, fyziky a informatiky
Univerzity Komenského v Bratislave
Katedra informatiky

Viacnásobné elektronické podpisy v praxi

DIPLOMOVÁ PRÁCA

Diplomant: **Daniel Chromek**
Diplomový vedúci: **Mgr. Juraj Vaško**
Názov študijného odboru: **Informatika**
Bratislava 2006

Zadanie diplomovej práce

Hlavným cieľom diplomovej práce je preskúmať rôzne metódy ukladania viacnásobných elektronických podpisov pre štandardne používané ako i navrhované formáty elektronických podpisov, interakciu ich vzájomnej platnosti a platnosti dokumentu vyžadujúceho podpísanie viacerými osobami ako celku z právneho i technologického hľadiska.
Cieľom práce je rovnako preveriť, aké sú možnosti automatizácie overovania viacnásobných podpisov potrebných v mene právnických osôb.
Výsledkom diplomovej práce by mala byť i analýza premietnutia dosiahnutých výsledkov do podpisových politík a návrh štruktúry podpisovej politiky pre viacnásobné podpisy.

Čestne vyhlasujem, že som diplomovú prácu vypracoval samostatne, s použitím literatúry uvedenej v závere práce.

V Bratislave dňa 20.4.2006

Ďakujem svojmu diplomovému vedúcemu Mgr. Jurajovi Vaškovi za odborné vedenie práce, cenné rady a pripomienky.
Ďakujem doc. JUDr. Daniele Gregušovej, CSc. za rady v oblasti slovenskej legislatívy a taktiež za to, že mi pomohla zorientovať sa v oblasti zákonov.
Ďakujem Mgr. Vladimírovi Mikoláškovi za konzultácie v oblasti uzatvárania zmlúv a osvedčovania podpisov týchto zmlúv.
Ďakujem Petrovi Vargovi za vysvetlenie fungovania procesu legalizácie a pripomienky k návrhu jeho elektronickej verzie.
Ďakuje svojej rodine a priateľom za to, že pri men stáli a podporovali ma pri písaní diplomovej práce.

Abstrakt

Diplomová práca sa zaoberá rozšírením pojmu elektronického podpisu (EP) na viacnásobný EP a problémami, vyplývajúcimi z tejto problematiky. Je rozobratá slovenská legislatíva, týkajúca sa viacnásobných EP, pričom sú navrhnuté úpravy v postupe vytvárania viacnásobných EP tak, aby bolo možné použiť ETSI formáty EP. K formátu ZEP (ZIP), schválenému NBÚ sú dané alternatívne prístupy, ktoré zaručujú rovnaké vlastnosti, ale sú použité konštrukcie z medzinárodných štandardov. Sú rozlíšené viaceré druhy viacnásobných podpisov a pre každý sa uvádza jeho možné využitie v reálnej praxi. Jedným z využití viacnásobných EP je aj notárske osvedčenie podpisu. Po analýze procesu vydania notárskeho osvedčenia podpisu je navrhnutý jeho elektronický ekvivalent.

Predhovor

Elektronické podpisy a počítačová technika vo všeobecnosti vstupujú čoraz viac do nášho života. Prostredníctvom tejto diplomovej práce sa snažím priblížiť problematiku viacnásobných EP aj čitateľovi, ktorý nemá s EP vo všeobecnosti žiadne skúsenosti. Na čítanie nie sú potrebné žiadne vedomosti z bezpečnosti alebo kryptológie, pre pochopenie niektorých konštrukcií je však potrebná znalosť matematiky.

V prvej kapitole je čitateľ uvedený do základov kryptológie a počítačovej bezpečnosti. Sú v nej vysvetlené všetky pojmy z tejto oblasti, ktoré sa budú v texte ďalej vyskytovať.

V druhej kapitole sú rozobraté dva hlavné pojmy. Prvým je to infraštruktúra verejných kľúčov (PKI) a v druhým je pojem elektronického podpisu.

V tretej kapitole sa hovorí o tom, na čo by sa dal použiť viacnásobný EP, je spravené rozdelenie viacnásobných EP a uvedené príklady pre každý druh viacnásobných EP. Tak isto sú uvedené zákony a vyhlášky, týkajúce sa viacnásobných EP.

Vo štvrtej kapitole je načrtnutá abstrakcia viacnásobných EP, sú vymenované konštrukcie pre všetky druhy EP a popísané, akú časť abstrakcie pokrývajú. Sú tu vymenované problémy, týkajúce sa viacnásobných EP. Analyzuje sa postup vytvárania viacnásobných EP podľa slovenskej legislatívy a uvedené sú alternatívy k formátu ZEP (ZIP).

V poslednej kapitole je zanalyzovaný proces legalizácie a tiež je stanovený formát notárskeho osvedčenia podpisu.

Obsah

Zoznam obrázkov a tabuliek

Zoznam obrázkov

Zoznam tabuliek

Úvod

S masovým rozšírením počítačov sa presúva čoraz informácií do digitálnej podoby. Nasadenie informačných systémov (IS) v organizácii zefektívňuje prácu s informáciami, ktorých množstvo z roka na rok rastie. IS umožňujú ukladať a pracovať s veľkým množstvom dát, čo umožňuje ľahšie zneužitie týchto informácii. Preto sa kladú požiadavky na ochranu informácií a bezpečnú manipuláciu s nimi. Tieto požiadavky vyplývajú buď z legislatívy, alebo zo snahy jednotlivých firiem zabezpečiť dôvernosť svojich údajov, ako aj údajov svojich klientov. S týmito požiadavkami stúpajú nároky na zabezpečenie informácií či už fyzicky alebo elektronicky. Elektronickým zabezpečenie informácií je snaha o naplnenie viacerých odlišných cieľov, ako napr. zaručenie dôvernosti, integrity a nepopierateľnosti autorstva informácií.

Elektronický podpis umožňuje zachovanie integrity, autentickosti a nepopierateľnosti autorstva, preto sa jeho miera využitia v rámci IS neustále zvyšuje. V roku 1999 bola schválená direktíva EÚ o elektronických podpisoch, ktorá vytvorila podmienky na rozširovanie EP v Európe. Odporúčania direktívy sa odrazili aj v Slovenskom zákone o elektronickom podpise, ktorý bol schválený v roku 2001. Tým sa otvorila cesta pre využívanie EP na Slovensku. Používanie EP bolo podporené vydaním dokumentov ETSI, ktoré stanovili formáty EP.

O EP bolo sa už popísalo množstvo odborných článkov aj diplomových prác. Logickým krokom sa javí zvýšenie zložitosti z jediného podpisu na viacero podpisov a analýza, aké problémy je potrebné riešiť. Po vyriešení týchto problémov prinesú viacnásobné EP svoje "ovocie" vo forme širšieho využitia oproti jednoduchým EP. V "papierovom" svete existuje široké využitie viacnásobných podpisov.

Cieľom tejto diplomovej práce je priblížiť čitateľovi viacnásobné podpisy vo všeobecnosti, t. j. rozobrať druhy viacnásobných EP a ako je možné tieto druhy viacnásobných EP využiť v praxi, formáty, do ktorých je možné ukladať viacnásobné EP a podpisové politiky pre viacnásobné EP. Pretože by bolo vhodné, aby sa dali viacnásobné EP aplikovať aj v Slovenských podmienkach, chcel by som rozobrať aj niektoré vybrané časti Slovenskej legislatívy. Keďže nie som odborník na právo, budem sa sústreďovať na technické riešenia, ktoré vyplývajú zo Slovenskej legislatívy a prediskutovať alternatívy k nim. Na záver som si vybral aplikáciu viacnásobných EP v notárskom osvedčení podpisu. Chcel by som analyzovať proces vydania notárskeho osvedčenia podpisu a navrhnúť formát osvedčenia.

Kapitola 1

Úvod do kryptológie

Elektronický podpis je úzko spätý s kryptológiou. Kryptológia je veda, ktorá sa zaoberá ochranou informácie v čase a priestore. Delí sa na *kryptografiu* a *kryptoanalýzu*. Úlohou kryptografie je ochrana informácie a úlohou kryptoanalýzy je skúmanie možností útokov na chránenú informáciu. Kryptológia je postavená na štyroch pilieroch, ktorými sú:

- Symetrické šifrovanie
- Asymetrické šifrovanie
- Jednosmerné hašovacie funkcie (one-way hash functions)
- Generátory náhodných čísel

1.1 Základné pojmy

Abecedou Σ sa nazýva konečná, neprázdna množina symbolov. Konečné postupnosti prvkov zo Σ sa nazývajú slová nad abecedou Σ. Generátor postupností znakov zo Σ sa nazýva *zdroj informácie*. Na to, aby sa dala informácia prenášať, resp. uchovávať, je potrebné ju *transformovať* do vhodného tvaru. Pod transformáciou sa myslí nejaký algoritmus, ktorý prevedie slovo zo vstupnej abecedy na slovo z výstupnej abecedy. Formálnejšie:

Definícia 1 *Transformácia informácie je relácia* $R \subseteq \Sigma_1^+ \times \Sigma_2^+$*, kde* Σ_1 *je vstupná abeceda a* Σ_2 *je výstupná abeceda a platí* $\Sigma^+ = \bigcup_{i=0}^{\infty} \Sigma^i$*, pričom* $\Sigma^n = \{a_1, \ldots, a_n | \forall 1 \leq i \leq n : a_i \in \Sigma\}$.

Väčšinou je žiaduce, aby bolo možné dostať informáciu z transformovanej podoby, preto je možné zaviesť transformáciu ako prostú funkciu $F : \Sigma_1^+ \rightarrow \Sigma_2^+$. Je zbytočné komplikovať situáciu a rozlišovať dve abecedy, pretože je možné položiť $\Sigma = \Sigma_1 \cup \Sigma_2$ a dostaneme $F : \Sigma \rightarrow \Sigma$. Transformovaný tvar informácie musí vyhovovať kritériám, ktoré sa naň kladú. Príkladom kritérií, ktoré sa budú vyskytovať v tejto práci, sú:

- *Dôvernosť* znamená, že k informácii sa vie dostať len určitá skupina ľudí, ktorá vie, ako ju dostať z transformovanej podoby. Dôvernosť sa dosahuje *šifrovaním*. V praxi je väčšinou funkcia (algoritmus) F verejná a do výpočtu vstupuje tajná informácia, ktorá je potrebná na zašifrovanie a dešifrovanie informácie. Táto tajná informácia sa nazýva *kľúč*. Podľa toho, či sa používa rovnaký kľúč na šifrovanie aj dešifrovanie alebo nie, delíme šifrovanie na *symetrické* a *asymetrické*.

- *Integrita* znamená, že informácia môže byť nezistiteľne zmenená len so zanedbateľnou pravdepodobnosťou. Integrita sa dosahuje kontrolnými súčtami a jednosmernými hašovacími funkciami.

- *Autentickosť* informácie znamená, že nikto nemôže preukázať, že je autorom informácie, ktorú vygeneroval niekto iný a autor nemôže poprieť svoje autorstvo informácie. Autentickosť sa dosahuje použitím elektronických podpisov.

Podľa námahy, ktorú treba vynaložiť na prelomenie zvoleného kritéria transformácie, hovoríme o tom, že:

- Kritérium je nepodmienené, to znamená, že ľubovoľná výpočtová sila nepostačuje na prelomenie kritéria, ak je použitá daná transformácia. Príkladom je nepodmienená dôvernosť vo Vernamovej šifre[1].

- Kritérium je výpočtovo podmienené, to znamená, že pri použití transformácie je možné prelomiť kritérium, ak sa použije dostatočná výpočtová sila. V tomto prípade je pre útočníka dôležité, v akom čase sa mu podarí prelomenie kritéria danej transformácie. Príkladom je výpočtová podmienenosť, dôvernosť asymetrických šifrovacích systémov.

Informáciu je potrebné prenášať (uchovávať) v čase alebo priestore, ale z kryptologického hľadiska nie je medzi nimi veľký rozdiel. Pri prenose v priestore stúpa pravdepodobnosť, že sa potenciálny útočník dostane k zašifrovanej informácii, pri uchovávaní v čase stúpa pravdepodobnosť, že útočník zašifrovanú informáciu dešifruje. Podľa toho koľko informácii má útočník k dispozícii sa rozlišuje viacero typov útokov na šifrovacie algoritmy:

- Útok len so šifrovým textom (COA - ciphertext only attack); útočník má k dispozícii len šifrový text.

- Útok so známym otvoreným textom (KPA - known plaintext attack); útočník pozná otvorený (nezašifrovaný text) a snaží sa dešifrovať zašifrovaný text, získať kľúč alebo neznámy šifrovací a dešifrovací algoritmus.

[1] Šifrovanie (aj dešifrovanie) vo Vernamovej šifre (alebo one-time pad) znamená, že sa bity správy xorujú s náhodnými bitmi. Operácia *xor* na bitoch je definovaná nasledovne: $0 \oplus 0 = 0$, $1 \oplus 0 = 0 \oplus 1 = 1$ a $1 \oplus 1 = 0$.

- Útok so zvoleným otvoreným textom (CPA - chosen plaintext attack); útočník si zvolí otvorený text a presvedčí majiteľa kľúča, aby spravil šifrovaciu transformáciu, čo využije na dešifrovanie (iného) šifrového textu.
- Útok so zvoleným šifrovým textom (CCA - chosen ciphertext attack); útočník presvedčí majiteľa kľúča, aby spravil dešifrovaciu transformáciu, čo využije na dešifrovanie (iného) šifrového textu.

Pri prenose v priestore uvažujeme prenos prostredníctvom *prenosového kanála*.

Prenosový kanál

Prenosovým kanálom sa myslí ľubovoľné zariadenie schopné prenášať informáciu v priestore. Pod *komunikáciou* sa myslí posielanie správ (reťazcov znakov) podľa určitých pravidiel. Súhrn pravidiel, podľa ktorých si komunikujúca dvojica vymieňa informácie, sa nazýva *protokolom*. Prenosový kanál nie je ideálny, a teda na ňom vznikajú chyby (šum), tie sa však dajú odhaliť a opraviť vhodnou transformáciou (*kódovaním*), preto je možné abstrahovať od chybného prenosu informácie.
Druhým problémom prenosového kanála, je jeho *nespoľahlivosť*. To znamená, že môže byť odpočúvaný (pasívny útočník) alebo dáta na ňom úmyselne modifikované (aktívny útočník). Preto je potrebné informáciu chrániť vhodnou transformáciou (šifrovaním). V kryptológii sa označuje komunikujúca dvojica ako Alica a Bob, pričom útočníkom je Eva (z anglického eavesdrop - odpočúvať).

1.2 Symetrické šifrovanie

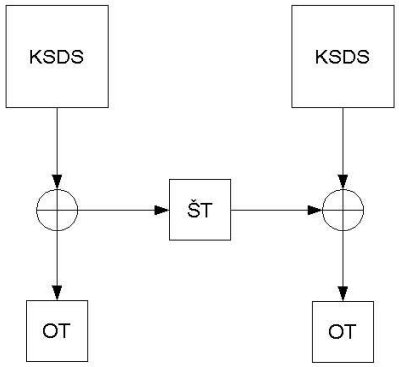
Na symetrické šifrovanie používame dvojicu funkcií. Funkcia $E : M \times K \rightarrow C$ je určená na šifrovanie a funkcia $D : C \times K \rightarrow M$ na dešifrovanie, pričom platí:

$$\forall m \in M \; \forall k \in K : D_k(E_k(m)) = m \tag{1.1}$$

Množina M sa nazýva množinou otvorených textov (plaintext), K množinou kľúčov a C množinou šifrových textov (ciphertext), pričom $M \cup K \cup C \subseteq \Sigma^+$. Z rovnice 1.1 je zrejmé, že pri šifrovaní aj dešifrovaní sa používa rovnaký kľúč $k \in K$. Na zabezpečenie dôvernosti m je nevyhnutné, aby bola množina K dostatočne veľká, aby sa predišlo útoku úplným preberaním, t. j. postupným skúšaním všetkých možných kľúčov a testovanie zmysluplnosti dešifrovaného textu.
V závislosti od spôsobu spracovania otvoreného textu sa symetrické šifry delia na:

- Blokové šifry
- Prúdové šifry

Blokové šifry spracuvávajú otvorený text po reťazcoch pevnej dĺžky a prúdové šifry spracuvávajú otvorený text po bitoch a výsledkom je opäť prúd (stream) bitov. Príkladom využitia prúdových šifier môže byť mobilná komunikácia (šifra A5)



Obrázok 1.1: Princíp fungovania prúdových šifier

alebo bezpečná komunikácia prostredníctvom SSL[2] (RC4).
Príkladom využitia blokových šifier je šifrovanie údajov na pevnom disku v rámci filesystému NTFS (DES, AES, TrippleDES) alebo bezpečná komunikácia prostredníctvom SSL (DES).
Prúdové šifry využívajú na svoju činnosť konečnostavové deterministické zariadenie (KSDZ), ktoré na základe kľúča generuje prúd bitov. Tento prúd bitov sa xoruje s bitmi otvoreného textu. Dešifrovanie prebieha rovnako ako je nakreslené na obrázku 1.2. Keďže KSDZ je deterministické, prúd bitov je pri rovnakých kľúčoch vždy rovnaký, čo umožňuje následné dešifrovanie.

Blokové šifry šifrujú blok textu pevnej dĺžky v závislosti od použitého módu. Blokové šifry môžu mať viacero kôl (iterované šifry), pričom v každom kole je text šiforvaný iným podkľúčom. Podkľúče sa deterministicky odvodzujú od pôvodného kľúča, napr. pomocou jednosmerných hašovacích funkcí. Tento proces sa nazýva *plánovanie kľúča* (key scheduling). Väčšinou je dĺžka otvoreného textu rôzna od dĺžky bloku, preto sa musí otvorený text rozdeliť na viacero blokov (prípadne doplniť na dĺžku bloku), ktoré sa šifrujú v závislosti od toho, aký mód činnosti je použitý.

1.2.1 AES

AES[1] (Advanced Encryption Standard) je štandard, ktorý nahradil štandard DES (Data Encryption Standard) vo verejnej správe USA. Problémom DES-u je krátka dĺžka kľúča (56 bitov), ktorá umožňuje útok úplným preberaním a jeho nepostačujúca rýchlosť. Výberové konanie na algoritmus bolo ukončené v roku 2000 a

[2]SSL (secure socket layer) je protokol, ktorý po splnení určitých podmienok zaručuje zachovanie dôvernosti a integrity správy. Jednou z jeho výhod je variabilnosť, pri výbere použitých šifrovacích a hašovacích algoritmov (cipher suite)

víťazom algoritmus *Rijndael*. Autori algoritmu sú Vincent Rijmen a Joan Deamen. Algoritmus má variabilnú dĺžku bloku aj kľúča. Kľúč môže mať 128, 192 alebo 256 bitov, blok môže mať tiež 128, 192 alebo 256. So zvyšujúcou sa dĺžkou bloku sa zvyšuje počet kôl (pre 128 b je to 10, pre 192 b je to 12 a pre 256 b je to 14 kôl). Algoritmus na svoju činnosť používa nasledujúce operácie:

1. (Subyte transformation) Je invertibilná nelineárna substitúcia bajtov, ktorá operuje nezávisle na každom bajte, pričom sa využíva substitučná tabuľka (S-Box).

2. (Shift Rows) Operácia robí cyklický posun doľava na riadkoch matice. Nultý riadok sa neposúva vôbec, prvý o jeden bit, druhý o dva bity a tretí o tri bity.

3. (Mix Columns) Stĺpcová transformácia.

4. (Add Round Key) Pri tejto operácii sa ku bloku pripočíta po bitoch podkľúč pre dané kolo.

V každom kole algoritmu okrem posledného sa vykonávajú operácie v poradí Subyte, Shift Rows, Mix Columns a Add Round Key. V poslednom kole sa vynecháva operácia Mix Columns. Navyše sa pred prvým kolom spraví operácia Add Round Key s nultým podkľúčom.

1.3 Asymetrické šifrovanie

Prepokladajme, že n účastníkov sa snaží komunikovať tak, aby okrem odosielateľa a príjemcu nemohol správu nikto dešifrovať. Pri použití symetrického šifrovania s rastúcim n neúmerne narastá počet kľúčov, pretože n účastínkov potrebuje $\binom{n}{2}$ (čo je $n(n-1)$, čiže kvadraticky veľa $O(n^2)$) kľúčov, čo prináša zvýšené nároky na manažment kľúčov. Tento problém odstraňuje použitie asymetrického šifrovania, kde postačuje n kľúčov.
Základná myšlienka kryptografie s verejnými kľúčmi (public key kryptography) bola publikovaná v roku 1976. Jej autormi boli Diffie a Hellman. Odvtedy našla široké uplatnenie predovšetkým v protokoloch na výmenu kľúča, v asymetrickom šifrovaní a digitálnych podpisoch. Základnou myšlienkou je použitie dvoch kľúčov, resp. dvoch funkcí (šifrovacej E a dešifrovacej D) namiesto jedného. Jeden kľúč je verejný a druhý privátny. Verejný kľúč používateľ zverejní a je prístupný pre každého. Privátny kľúč je tajný a pri jeho odhalení dochádza k rozbitiu systému. Asymetrické šifrovacie systémy sa delia na:

- Deterministické
- Nedeterministické

Rozdiel medzi nimi je v tom, že v deterministických asymetrických systémoch prislúcha jednému otvorenému textu jeden šifrový text a v nedeterministických asymetrických systémoch viacero šifrových textoch. Podľa učebnice Základy kryptológie[2] je splnenie nasledujúcich podmienok nevyhnutné, aby mohol asymetrický systém fungovať.

1. (Korektnosť) Po dešifrovaní šifrového textu vznikne pôvodný šifrový text. Formálnejšie:
$$\forall m \in M,\ D(E(m)) = e$$

2. (Realizovateľnosť) Funkcie E a D sú realizovateľné v polynomiálnom čase a dajú sa skonštruovať pravdepodobnostným polynomiálnym algoritmom.

3. (Bezpečnosť) Zo znalosti E je nemožné skonštruovať funkciu D^* tak, že $D^*(c) = D(c)$ pre nezanedbateľne veľa $c \in C$.

Problémom asymetrického šifrovania je jeho nepostačujúca rýchlosť. Aj keď s rastúcim výkonom hardwaru sa môže zdať, že tento problém nie je neodstrániteľný, je potrebné si uvedomiť, že zároveň stúpajú nároky na bezpečnosť kvôli použitiu výkonnejších počítačov na kryptoanalýzu. A rastúce nároky na bezpečnosť sa nutne musia odraziť vo zvýšenej náročnosti šifrovacích operácií, čo sa následne odrazí na rýchlosti. Ďalším problémom asymetrického šifrovania je, že nezaručuje *nepodmienenú* dôvernosť. Ak je nejaká vlastnosť nepodmienená, potom ľubovoľná výpočtová sila nepomôže na prelomenie tejto vlastnosti. V tomto prípade stačí preberať priestor správ, šifrovať ich a porovnávať so zašifrovaným textom. Tento problém je obzvlášť závažný, ak je priestor správ malý, a preto sa na jeho zväčšenie používa *vypchávka* (padding)[3], čo je náhodná postupnosť znakov zo Σ.
Z hľadiska tejto diplomovej práce je dôležité, že otočený princíp asymetrického šifrovania sa používa v elektronických podpisoch. Pri asymetrickom šifrovaní odosielateľ zašifruje správu verejným kľúčom adresáta, čím je zaručené, že len adresát vie dešifrovať správu. V elektronických podpisoch podpisovateľ zašifruje tajnú informáciu, ktorá sa viaže na dokument a propojí ju k dokumentu. Keďže dešifrovať vie každý (lebo verejný kľúč je dešifrovací), vie si každý overiť, či tajná informácia pre dokument je platná.

1.3.1 RSA

Jedným z najznámejších asymetrických šifrovacích systémov je RSA. Názov je odvodený od mien autorov Ronalda Rivesta, Adiho Shamira a Leonarda Adlemana, ktorí ho zverejnili v roku 1978.

[3]Padding je náhodná postupnosť znakov pevne stanovenej dĺžky, pridaná (zväčša) na koniec otvoreného textu. Padding umožňuje zväčšiť priestor otvorených textov a navyše každému otvorenému textu bude zodpovedať viacero šifrových textov: ak je použitý padding dĺžky n-bitov, tak je možné k jednému otvorenému textu dostať 2^n šifrových textov.

Inicializácia

Inicializácia sa skladá z nasledujúcich troch krokov:

1. Zvolia sa dve dostatočne veľké rôzne prvočísla p, q a položí sa $n = p \cdot q$
2. Zvolí sa prirodzené číslo e také, že $1 < e < \phi(n)$ a $nsd(e, \phi(n)) = 1$, kde $\phi(n) = (p-1) \cdot (q-1)$ je Eulerova funkcia a nsd je najväčší spoločný deliteľ, e bude verejný kľúč.
3. Vypočíta sa d také, že $e \cdot d \equiv 1(mod\ \phi(n))$, d bude privátny kľúč.

Šifrovanie a dešifrovanie

Nech $m \in \mathbb{Z}_n$ je otvorený text. Potom šifrovanie a dešifrovanie prebieha nasledovne:

$$\begin{aligned} E_e(m) &= m^e mod\, n \\ D_d(c) &= c^d mod\, n \end{aligned}$$

Overenie všetkých troch vlastností je možné nájsť v učebnici Základy kryptológie [2]. Jedna z vlastností RSA, ktorá umožňuje pri vhodnom sociálnom inžinierstve[4] uskutočniť CCA útok (Chosen Ciphertext Attack - útok so zvoleným šifrovým textom), ale zároveň je veľmi želaná pri zaslepených (blind) podpisoch, vyzerá nasledovne: zvoľme náhodne $x \in \mathbb{Z}_n$, dostávame náhodné $x^e \cdot c$,kde c je šifrový text. Potom platí:

$$(x^e \cdot c)^d = x \cdot c^d \ \Rightarrow\ (x^e \cdot c)^d \cdot x^{-1} = c^d$$

Teda ak adresáta odchytenej správy presvedčíme, aby spravil jednu dešifrovaciu transformáciu na pozmenenom šifrovom texte, tak vieme odchytenú správu dešifrovať.

1.4 Jednosmerné hašovacie funkcie (One-way hash functions)

Nech $f : X \to Y$ je funkcia a Y je konečná. Potom f nazývame kryptografickou hašovacou funkciou, ak spĺňa nasledovné vlastnosti:

1. (Jednosmernosť, one-way, preimage resistant) Pre dané $y \in Y$ je ťažké nájsť $x \in X$ také, že $h(x) = y$

[4]Sociálne inžinierstvo je spôsob získavania dvôverných informácií pomocou manipulovania legálnym používateľom. Sociálny inžinier väčšinou použije telefón alebo internet, aby oklamal ľudí a vymámil od nich citlivé informácie alebo ich prinútil konať proti (zabehanému) správaniu. Príkladom môže byť telefonát v mene banky, pričom inžinier sa snaží vymámiť v mene banky PIN ku kreditnej karte, napr. z dôvodu zlyhania databázy.

2. (Slabá odolnosť voči kolíziám) Pre dané $x \in X$ je ťažké nájsť $\overline{x} \neq x$ také, že $h(\overline{x}) = h(x)$

3. (Silná odolnosť voči kolíziám) Je ťažké nájsť $x, \overline{x}$ také, že $h(x) = h(\overline{x})$

Čo znamená "je ťažké", závisí od nárokov, ktoré sa kladú na hašovaciu funkciu. Napr. neexistuje pravdepodobnostný polynomiálny algoritmus, ktorý rieši úlohu. Z definície je jasné, že ak je hašovacia funkcia f silne odolná voči kolíziám, je aj slabo odolná voči kolíziám. Keďže ide o hašovaciu funkciu, je typicky $|X| > |Y|$. Problémom jednosmerných hašovacých funkcí je, ani jedna z vlastností nie je *nepodmienená*. Stačí si uvedomiť, že prebratie všetkých možností určite nájde kolíziu, ak je $|X| > |Y|$.
Ďalším problémom, s ktorým je spojená konštrukcia jednosmernej hašovacej funkcie, je fakt, že množina $X \subseteq \{0,1\}^m$ je konečná a je potrebné, aby fungovala na $X \subseteq \{0,1\}^*$, teda je potrebné rozšíriť ju na nekonečnú množinu, aby sa dali robiť haše ľubovoľne dlhých reťazcov. Tento problém sa dá odstrániť pomocou konštrukcie uvedenej v učebnici Základy kryptológie [2]. Pozitívnym faktom je, že ak pôvodná hašovacia funkcia (na konečnej množine) bola odolná voči kolíziám, tak aj vzniknutá hašovacia funkcia je odolná voči kolíziám.
Dôležitou množinou hašovacích funkcí sú hašovacie funkcie s kľúčom (hash based message authentification code - HMAC). Problémom hašovacích funkcií je, že síce zaručujú integritu správy, ale nie jej autentickosť, pretože útočník môže nahradiť správu aj haš. Preto je potrebné haš ochrániť či už šifrovaním (ako napr. v digitálnych podpisoch), alebo tým, že pri ich výpočte sa využie tajná informácia, ktorá je známa len obom komunikujúcim stranám. Práve druhý princíp využívajú hašovacie funkcie s kľúčom. HMAC funkcia vyzerá nasledovne:

$$h_k(m) = h(k \oplus opad \| \overline{h}(k \oplus ipad \| m))$$

kde $opad$ a $ipad$ sú reťazce deterministicky odvodené z kľúča k, $\|$ je operácia zreťazenia reťazcov a $\overline{h}$ je nejaká obyčajná hašovacia funkcia.

1.5 Digitálne podpisy

Digitálny podpis je súčasťou elektronického podpisu. Viac o rozdieloch medzi elektronickým a digitálnym podpisom sa uvádza v kapitole 2. Ako už bolo uvedené v kapitole o asymetrickom šifrovaní, digitálne podpisy využívajú pri podpisovaní a overovaní prevrátený princíp asymetrického šifrovania.

- (Podpisovanie) Šifruje sa hašovací odtlačok (výsledok hašovacej funkcie, ktorej vstup bol dokument) podpísaného dokumentu[5]. Výsledok šifrovania je digitálny podpis, ktorý sa pripojí k dokumentu.

[5]Dôvodom, prečo sa šifruje len odtlačok a nie celý dokument, je veľkosť podpisu. Odtlačok dokumentu má spravidla zanedbateľnú veľkosť (niekoľko desiatok až stoviek bitov) oproti celému dokumentu (potenciálne megabajty).

- (Overovanie) Overovateľ spraví haš $\overline{h}$ dokumentu. Dešifruje odtlačok h a porovná ho so získaným odtlačkom $\overline{h}$. Podpis je platný, vtedy a len vtedy, ak $h = \overline{h}$.

Transformácia asymetrického šifrovania na schému pre digitálny podpis nemusí byť vždy jednoduchá ako v prípade RSA. Príkladom schémy, ktorá nie je triviálne transformovateľná na schému pre digitálne podpisy, môže byť napr. El Gamalov asymtrický šifrovací systém.
Schéma pre digitálny podpis by mala zabezpečovať nasledujúce vlastnosti.

- Zabrániť zmene obsahu dokumentu bez toho, aby zostal podpis na dokumente platný, čo zabezpečuje použitie jednosmerných hašovacích funkcií.
- Nemožnosť vygenerovania falošného podpisu niekoho iného, čo je zabezpečené nemožnosťou odvodiť súkromný kľúč z verejného a dôvernosťou súkromného kľúča.

Žiaľ, žiadna z týchto dvoch vlastností nie je nepodmienená. Zmena obsahu závisí na podmienenosti nájdenia kolízie v jednosmernej hašovacej funkcii, o čom bolo uvedené, že nie je nepodmienená. Je preto možné vygenerovať náhodnú správu $\overline{m}$, ktorá koliduje s podpísanou správou m, t. j. $h(\overline{m}) = h(m)$. Digitálny podpis k $\overline{m}$ je platný. Tento proces sa nazýva *random message forgery*. Podmienenosť vygenerovania falošného podpisu je daná podmienenosťou asymetrického šifrovania. Ďalšiu vlastnosť, ktorú často požadujeme, aby podpisovateľ nemohol *poprieť* svoj podpis. Žiaľ, digitálny podpis nedisponuje mechanizmom, ktorý by zaručoval nepopierateľnosť podpisov a na jej dosiahnutie je potrebné využiť *časové pečiatky* [3].

1.5.1 RSA

Asymetrický šifrovací systém RSA má vlastnosť, že sa dá priamočiaro použiť na digitálne podpisy. Podpisovací algoritmus zodpovedá dešifrovaciemu algoritmu a overovací algoritmus šifrovaciemu. Dôvodom, prečo sa dá RSA použiť, je nasledovná vlastnosť:

$$\forall x \in M : x \equiv (x^e)^d (mod\ n) \equiv (x^d)^e (mod\ n)$$

Teda šifrovaciu a dešifrovaciu transformáciu možno vymeniť a výsledok bude rovnaký. Táto vlastnosť je záležitosťou RSA a v iných asymetrických systémoch nemusí platiť!

1.5.2 DSA

DSA (digital signature algorithm) je jedným z algoritmov použitých v štandarde DSS (digital signature standard) vlády USA. Ďalšie algoritmy v tomto štadarde sú RSA a EC DSA (DSA nad eliptickými krivkami). Predtým, ako je možné DSA používať na overovanie digitálnych podpisov a podpisovanie, je potrebné urobiť inicializáciu.

- (Inicializácia)

  1. Zvolí sa prvočíslo p, ktoré má aspoň 1024 b a existuje prvočíslo q dĺžky 160 b s vlastnosťou $q \mid p-1$. V praxi sa najskôr vygeneruje prvočíslo q, a potom sa k nemu hľadá prvočíslo p.
  2. Zvolí sa náhodné $h \in Z_p$ a vypočíta sa $g = h^{\frac{p-1}{q}}$. Ak nie je $g > 1$, tak sa volí nové h.
  3. Zvolí sa náhodné $x \in Z_q$, ktoré bude súkromným kľúčom. Položí sa $y = g^x \ mod \ p$ a verejný kľúč je (y, p, q, g).

- (Podpisovanie)

  1. Zvolí sa náhodné $k \in Z_p$.
  2. Vypočíta sa $r = g^k \ mod \ p \ mod \ q$.
  3. Vypočíta sa $s = k^{-1}(h(m) + xr) \ mod \ q$, kde $h(m)$ je haš dokumentu. Ak $r \cdot s = 0$, tak sa prejde na krok 1, inak je dvojica $< r, s >$ digitálnym podpisom dokumentu m.

- (Overovanie)

  1. Vypočíta sa $u_1 = h(m) \cdot s^{-1} \ mod \ q$ a $u_2 = r \cdot s^{-1} \ mod \ q$.
  2. Podpis je platný vtedy ak $g^{u_1} \cdot g^{u_2} \ mod \ p \ mod \ q = r$.

Dôkaz korektnosti je možné nájsť v knihe [2].

1.6 Kryptografické protokoly

Kryptografické protokoly je množina protokolov, ktorých úlohou je

1. Bezpečná komunikácia účastníkov Alice a Boba.
2. Výmena kľúča na bezpečnú komunikáciu.

V praxi sa na šifrovanie komunikácie využíva kvôli rýchlosti symetrické šifrovanie, zatiaľ čo asymetrické šifrovanie sa využíva najmä v protokoloch na výmenu kľúčov. Ako sme spomenuli v časti o prenosovom kanále, tento nie je bezpečný a môže na ňom pôsobiť aktívny alebo pasívny útočník (Eva). Útoky na kryptografické protokoly sa delia na tieto dve základné triedy:

1. (Útok opakovaním - replay attack); útočník využíva opakovanie správ, ktoré sa mu podarilo odchytiť v minulosti. Ochranou pred ním sú časové pečiatky a príležitostné slová (nonces), ktoré zaručujú "čerstvosť" prijatej správy.
2. (Útočník uprostred - man in the middle attack); útočník predstiera niektorú komunikujúcu stranu (alebo viac strán). Ochranou pred ním sú elektronické podpisy.

1.7 Informačná bezpečnosť

Pod informačným systémom je možné predstaviť si množinu hardwaru a softwaru, ktorá je vzájomne nejako prepojená. Zvyčajne je IT systém príliš zložitý na to, aby sa mohla zhodnotiť jeho bezpečnosť, preto sa z neho vyberie len určitá časť, pre ktorú sa spraví podrobná analýza. V rámci analýzy sa najskôr určia *aktíva*, teda časi IT systému, pre ktoré má zmysel uvažovať o bezpečnosti oddelene od iných častí systému [4]. Následne sa určia hrozby, teda elementy, ktoré spôsobia, že systém nebude pracovať alebo bude pracovať v neštandardnom stave. Hrozby sa roztriedia podľa druhu (výpadok prúdu, neoprávnený vstup zvonka...) a určí sa aspoň približne, aká je pravdepodobnosť, že sa hrozba prejaví a aká škoda vznikne, ak sa hrozba prejaví, čím sa určí celková závažnosť hrozby. Je zrejmé, že nie je potrebné brániť sa proti hrozbám, pred ktorými je potrebné vynaložiť viac prostriedkov na ochranu ako je "cena" hrozby. Následne sa určí protiopatrenie voči hrozbe.

1. *Potieranie hrozieb* je protiopatrenie (countermeassure), ktoré zabraňuje, aby sa prejavila.

2. *Odhaľovanie hrozieb* znamená existenciu systému monitorov (watchdogs), ktorý hlási správanie sa časti IT systému.

3. *Minimalizácia prejavov hrozieb*. Niektorým hrozbám sa nedá zabrániť, pretože protiopatrenia by boli nákladnejšie ako škoda, ktorá vznikne ich prejavom. Dá sa však minimalizovať ich dopad tým, že sú vypracované plány v prípade prejavenia hrozby.

4. *Akceptácia hrozieb* znamená, že hrozba je príliš nepravdepodobná, alebo jej prejavy sú zanedbateľné, takže sa ignoruje.

Týmto postupom je možné odhaliť a zvýšiť bezpečnosť IT systému, avšak absolútna bezpečnosť neexistuje, je len ideálny stav časti IT systému. Žiaľ, nie vždy je jednoduché určiť cenu hrozby, prípadne pravdepodobnosť jej dopadu. Navyše je vhodné, aby bezpečnostné opatrenia boli v súlade s niektorou bezpečnostnou normou, tých je však veľa (Common Criteria, Orange book...). Starostlivosť, aby IT systém bol bezpečný, nie je jednorázová záležitosť, ale nikdy nekončiaci proces.

Kapitola 2

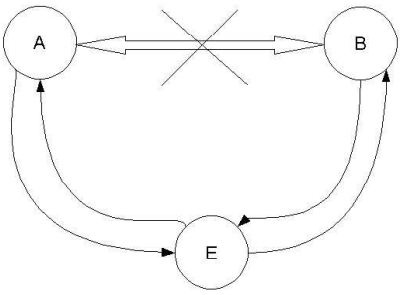
Elektronický podpis a PKI

Elektronický podpis je založený na digitálnom podpise[1]. Digitálny podpis je len haš dokumentu "zašifrovaný"súkromným kľúčom podpisovateľa. Elektronický podpis sa skladá z digitálneho podpisu, ktorý je vytvorený z podpísaných atribútov. Naviac môže obsahovať aj nepodpísané atribúty. To, aké atribúty podpis obsahuje, závisí od formátu elektronického podpisu. Na to, aby bolo možné elektronický podpis overiť, je potrebné porovnať hodnoty hašov, ako bolo uvedené v kapitole 1.5 o digitálnych podpisoch. Pokiaľ porovnanie dopadne úspešne, je potrebné overiť si, či verejný kľúč patrí naozaj podpisovateľovi. Na to slúži certifikát verejného kľúča a infraštruktúra verejných kľúčov, ktorá má vytvoriť dôveru medzi podpisovateľom a overovateľom.

2.1 PKI

Majme používateľov A, B, ktorí chcú medzi sebou komunikovať, pričom požiadavkou je zaručenie dôvernosti komunikácie. Keďže sú od seba vzdialení a nie je možná bezpečná výmena kľúča pre symetrické šifrovanie (napr. na diskete), predpokladajme, že sa rozhodli pre asymetrický šifrovací systém a majú vygenerované kľúčové páry. Dôvernosť ich komunikácie však môže byť narušená aj keď útočník E nevie rozbiť šifrovací systém, ani nepozná ich súkromné kľúče. Na začiatku komunikácie totiž E podvrhne obom používateľom svoj verejný kľúč. A aj B si potom budú myslieť, že komunikujú navzájom, ale celá komunikácia ide cez E, ktorý ju vie pohodlne dešifrovať svojím súkromným kľúčom. Celá situácia je znázornená na obrázku 2.1. Ide o klasický útok s útočníkom uprostred (man-in-the-middle), pričom problémom nie je bezpečnosť samotného kryptosystému, ale preukázanie identity. Tento problém sa rieši prítomnoťou dôveryhodnej tretej strany, ktorá dosvedčí identitu jednotlivých používateľov (prípadne im vygeneruje symetrický kľúč na komunikáciu). Žiaľ, problémov je viacero:

[1]Direktíva EC 1999/93 špecifikuje EP ako dáta pripojené alebo logicky spojené s dokumentom, ktoré slúžia na účel autentifikácie. To znamená, že pod EP je možné predstaviť si napr. aj hašovaciu funkciu s kľúčom.



Obrázok 2.1: Man in the middle útok

1. *Preukázanie identity*; každý používateľ musí pri komunikácii dôveryhodne preukázať svoju identitu, inak hrozí útok s útočníkom uprostred.

2. *Kompromitácia súkromného kľúča*; pri kompromitácii súkromného kľúča musia byť o tom informovaní všetci používatelia, aby nemohol útočník predstierať cudziu identitu používateľovým certifikátom a používať jeho kompromitovaný kľúč.

3. *Vytvorenie nového kľúčového páru*; môže byť problémom, pretože nový verejný kľúč nikto nepozná, a preto používateľ nie je schopný s nikým komunikovať, preto je potrebné kľúč zverejniť.

Riešenie týchto troch problémov ponúka *infraštruktúra verejných kľúčov* (public key infrastructure - PKI). PKI prideľuje verejné kľúče entitám, umožňuje ostatným entitám overiť pridelenie verejného kľúča a poskytuje ďalšie služby, súvisiace s manažmentom kľúčov v distribuovanom systéme [5]. Príkladom PKI môže byť napríklad riešenie Pretty Good Privacy - PGP (a jej open-source verzia GNU Privacy Guard - GPG) alebo hierarchia certifikačných autorít, ktorá funguje aj na Slovensku.
V ďalšom texte sa bude pod označením PKI myslieť typ infraštruktúry verejných kľúčov na báze certifikačných autorít. Na to, aby PKI fungovala, je nevyhnutné, aby v rámci PKI pôsobili dôveryhodní poskytovatelia služieb (trusted service providers - TSP). Tí musia na svoju činnosť splniť kritériá definované zákonom. V rámci PKI môžu pôsobiť tieto typy TSP:

- Certifikačná autorita (CA); vydáva entitám *certifikáty verejných kľúčov*, ďalej len certifikáty, a vydáva informácie o ich stave.
  Certifikát je informácia, ktorá spája entitu s jej verejným kľúčom. Ak má entita podozrenie na kompromitáciu súkromného kľúča (alebo má iný dôvod, pre ktorý už nemôže alebo nechce používať svoj certifikát, napr. zomrie),

oznámi to certifikačnej autorite, ktorá certifikát vydala a tá ho zaradí do *zoznamu zrušených certifikátov* (certificate revocation list - CRL). Podľa slovenskej legislatívy je akreditovaná CA povinná vydávať CRL pre kvalifikované certifikáty[2] tak, aby časový interval medzi dvoma po sebe idúcimi CRL bol menší ako 24 hodín.

- Registračná autorita (RA); identifikuje entitu pred vydaním certifikátu certifikačnou autoritou, aby sa zabránilo vydaniu certifikátu na falošnú identitu.
- Repository authority (RepA); zverejňuje informácie o certifikátoch a zoznamy zrušených certifikátov[3].
- Autorita časových pečiatok (Timestamp authority - TSA); vydáva *časové pečiatky*.
  Časová pečiatka je informácia, ktorá potvrdzuje, že dáta existovali pred konkrétnym časom.
- Vydavateľ podpisových politík (Signature policy issuer -SPI); vydáva *podpisové politiky*, pod ktorými je možné podpisovať údaje. Podpisová politika je množina pravidiel, ktoré musia zabezpečiť, aby ľubovoľní dvaja overovatelia dospeli pri overovaní toho istého elektronického podpisu k rovnakému výsledku.
- Atribútová autorita (AA); vydáva *atribútové certifikáty*, ktoré spájajú entitu so stanoveným atribútom, napr. s rolou v rámci IT systému.

V rámci PKI nie sú certifikáty vydávané len koncovým používateľom, ale každý z TSP má svoj certifikát vydaný nejakou certifikačnou autoritou. Ak sa graficky znázorní v PKI vzťah kto komu vydal certifikát, vznikne orientovaný graf[4], ktorý znázorňuje štruktúru PKI. Formálnejšie:

Definícia 2 *Štruktúrou PKI nazývame orientovaný graf* $G = (V, E)$ *definovaný nasledovne:*

1. *Každý kľúčový pár v PKI je vrchol grafu, teda*

$$V = \{k |\ k \text{ je kľúčový pár v PKI}\}$$

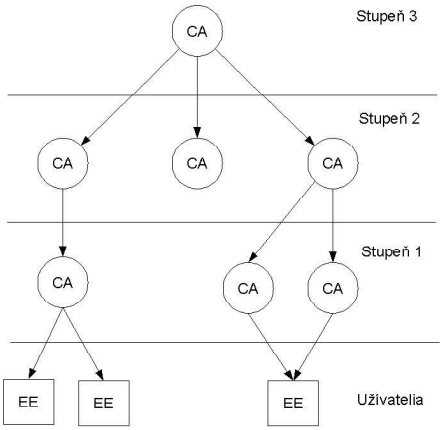
2. *Ak je certifikát na verejný kľúč* $k_{1,V}$ *podpísaný kľúčom* $k_{2,P}$*, potom existuje hrana z* k_1 *do* k_2*, teda*

$$E = \{k_1k_2 |\ \text{kľúčom } k_{2,P} \text{ je podpísaný certifikát na kľúč } k_{1,V}\}$$

[2] Viac o kvalifikovaných certifikátoch v kapitole 2.2

[3] Podľa slovenskej legislatívy je tieto informácie povinná vydávať CA, preto v rámci slovenskej PKI nie je explicitne odlíšená. Rovnako musí na Slovensku CA fungovať aj ako RA a TSA.

[4] Graf je množina vrcholov, ktoré sú pospájané hranami. Formálnejšie: Usporiadanú dvojicu $G = (V, E)$ nazývame grafom, neprázdnu množinu V množinou jeho vrcholov a množinu E, pre ktorú platí $E \subseteq V \times V$, množinou jeho hrán. Orientáciu hrán vyjadruje zápis vrcholu v hrane, t. j. v prípade (x, y), ide o hranu, smerujúcu z vrcholu x do vrcholu y.



Obrázok 2.2: Hierarchia certifikačných autorít

Jednoducho povedané: každý kľúčový pár je uzol a každý certifikát je hranou v grafe, ktorý predstavuje štruktúru PKI. Vo všeobecnosti sa rozlišujú dva druhy štruktúr PKI, a to:

- Hierarchická; graf G je súvislý a acyklický. Existuje pevne stanovená hierarchia certifikačných autorít. Na vrchole je koreňová certifikačná autorita. Príklad hierarchie certifikačných autorít je napr. na obrázku 2.2.
- Zmiešaná (Mesh); graf G je ľubovoľný a nie je pevne stanovená hierarchia certifikačných autorít. Vybrané CA sú koreňové certifikačné autority.

Pri rozlišovaní štruktúry PKI bol uvedený pojem *koreňovej certifikačnej autority* (root CA). Koreňová CA vydáva špeciálny certifikát na svoj verejný kľúč, ktorý je podpísaný privátnym kľúčom, ktorý tomuto verejnému kľúču prislúcha. Takýto druh certifikátu sa nazýva *self-signed* certifikát a v grafe PKI mu zodpovedá *slučka*[5]. Keď sa pri overovaní elektronického podpisu konštruuje certifikačná cesta[6], je nutné, aby sa cesta končila self-signed certifikátom koreňovej CA, inak cesta nemôže byť platná.

[5] Slučka je hrana, ktorá začína aj končí v tom istom vrchole
[6] Viac v kapitole 2.2.4 alebo RFC 3280 [6]

2.2 Elektronický podpis

Elektronický podpis je podľa zákona 215/2002 [7] §3 definovaný ako

> "informácia pripojená alebo logicky spätá s elektronickým dokumentom, ktorá musí pĺňať tieto požiadavky:
>
> a) nemožno ju efektívne vyhotoviť bez znalosti súkromného kľúča a elektronického dokumentu,
>
> b) na základe znalosti tejto informácie a verejného kľúča patriaceho k súkromnému kľúču použitému pri jej vyhotovení možno overiť, že elektronický dokument, ku ktorému je pripojená alebo inak logicky spätá je zhodný s elektronickým dokumentom použitým na jej vyhotovenie."

Znalosť elektronického dokumentu je potrebná, aby bolo možné získať jeho haš. Z bezpečnosti asymetrického systému (ktorý je použitý pri elektronickom podpise), taktiež vyplýva, že nemožno vyhotoviť informáciu podpísanú súkromným kľúčom, takže podmienka a) je splnená. Keďže verejný kľúč je každému známy a je súčasťou certifikátu, je každý schopný digitálny podpis dešifrovať a porovnať s hašom dokumentu. Tým je splnená podmienka b).
V zákone 215/2002 sa rozlišujú dva druhy elektronických podpisov:

- *Elektronický podpis*; je bežným elektronickým podpisom. Na jeho overenie stačí obyčajný certifikát vydaný CA.
- *Zaručený elektronický podpis*; od obyčajného elektronického podpisu sa líši tým, že podpisovanie je pomocou bezpečného zariadenia (čipová karta, USB token... ich zoznam vydáva NBÚ). Na jeho overovanie je potrebný *kvalifikovaný certifikát*. Kvalifikované certifikáty môže vydávať len *akreditovaná CA*. Rozdiel medzi akreditovanou a "obyčajnou" CA je v tom, že akreditovaná CA musí na svoju prevádzku splniť oveľa prísnejšie kritériá (stanovené NBÚ), takže jej certifikátom je možné viac dôverovať.

Pri tvorbe zákona o elektronickom podpise bola smerodajná Európska direktíva o elektronickom podpise z roku 1999 (1999/93 EC). Táto direktíva zaväzuje členské krajiny Európskej únie používať elektronický podpis vo verejnej správe (článok 19), ale hlavne zrovnoprávňuje "advanced" elektronický podpis založený na certifikátoch s vlastnoručným podpisom (článok 20). Zrovnoprávnením zaručeného elektronického podpisu s vlastnoručným podpisom sa v slovenskej legislatíve zaoberajú články II. až VII. §28 zákona 215/2002. Žiaľ, v zákone sa nehovorí nič o vzťahu s notársky overeným podpisom, takže napr. elektronicky podpísaný závet nemôže byť platný.
Každý z používateľov elektronického podpisu k nemu pristupuje z iného uhla pohľadu. Každej entite je pridelená rola v závislosti od toho, aký pohľad na elektronický podpis entita má. Podľa dokumentu ETSI 101 733 [3] ide o tieto roly:

- *Podpisovateľ*; vytvára elektronický podpis pre elektronický dokument na základe podpisovej politiky. Určuje formát a hodnoty atribútov EP.
- *Overovateľ*; Zisťuje platnosť EP. Môže to byť jedna entita alebo viacero entít.
- *TSP*; pomáhajú vytvoriť dôveru medzi podpisovateľom a overovateľom. Zoznam TSP je v kapitole 2.1.
- *Arbitor*; v prípade, že je spor mezdi podpisovateľom a overovateľom ohľadom platnosti EP, arbitor rozhodne o jeho platnosti.

Podľa dokumentu "Multiple Electronic Signatures on Multiple Documents" [8] vzhľadom na to, v akom sú vzťahu elektronický dokument a elektronický podpis, rozlišujeme tri druhy elektronických podpisov:

- *Obalený elektronický podpis*; elektronický dokument má vo svojom formáte uvedenú položku pre elektronický podpis. Príkladom je formát PDF.
- *Obaľujúci elektronický podpis*; elektronický podpis má vo svojom formáte položku pre dáta, ktoré sú podpísané.
- *Externý elektronický podpis*; elektronický podpis je uložený v inom súbore ako dokument.

Obalený a obaľujúci podpis sa nazýva spoločným názvom *interný* elektronický podpis.

2.2.1 Formáty elektronického podpisu

Elektronický podpis je možné používať na viacero účelov, a preto boli vytvorené viaceré formáty EP pre najbežnejšie z nich. Každý EP nezávisle od formátu však musí obsahovať niektoré základné používateľské dáta, nad ktorými je urobený digitálny podpis. Medzi ne patrí najmä haš používateľských dát, podpisová politika (v súlade s ktorou bol podpis urobený) a ďalšie podpísané atribúty, ktoré poskytujú overovateľovi doplnkovú informáciu, ktorá by mala byť podpísaná na základe podpisovej politiky, ako sa o tom zmieňuje dokument ETSI 101 733 [3]. Tento dokument popisuje nasledovné druhy EP:

1. *EP* (Basic Electronic Signature - BES); obyčajný elektronický podpis, ktorý zahŕňa digitálny podpis a ďalšie informácie poskytnuté podpisovateľom.
2. *EP s časovou pečiatkou* (ES-T); je ES rozšíreným o časovú pečiatku
3. *EP s úplnou informáciou* (ES-C); je ES-T, ku ktorému sú pridené referencie pre všetky údaje (certifikáty, CRL...) potrebné na overenie EP.

Ďalej sa v dokumente spomínajú aj formáty EP, ktoré sú dôležité vtedy, keď overovateľ nemá prístup k službám TSP, a tým pádom nevie on-line skontrolovať platnosť podpisu. Medzi tieto formáty patria formáty:

1. *Rozšírený EP* (ES-X long); obsahuje všetky dáta potrebné na konštrukciu certifikačnej cesty. Viac o certifikačnej ceste v kapitole 2.2.4

2. *Rozšírený EP s časovou pečiatkou* (ES-X Timestamp); je ES-X long rozšírený o časovú pečiatku buď nad celým ES-X long (Typ 1), alebo len nad dátami potrebnými na overenenie (Typ 2).

3. *Archívny EP* (ES-A); je ES-X Timestamp, ku ktorému je pridaná archívna časová pečiatka, ktorej parametre (dĺžky kľúčov, hašov...) zaručujú, že ak nenastane prevratný prelom v kryptoanalýze, tak časová pečiatka zostane nerozbitá dĺhý čas. Paramtre zohľadňujú nárast výpočtovej sily v priebehu času a proces opečiatkovania môže byť zopakovaný v prípade oslabenia algoritmov použitých na vytvorenie starej archívnej časovej pečiatky.

Každý z formátov je podrobne popísaný pomocou syntaxe ASN.1. ASN.1 predstavuje spôsob, ako popísať nejakú dátovú štruktúru (v tomto prípade elektronický podpis), pričom sa abstrahuje od spôsobu reprezentácie dátovej štruktúry v pamäti alebo na pevnom disku. Aby bolo možné s dátami pracovať, je stanovené kódovanie, ktoré jednoznačne mapuje ASN.1 štruktúru do binárneho reťazca. Príkladom kódovaní môžu byť napr. DER alebo BER kódovania. Pretože každý z formátov elektronického podpisu obsahuje iné informácie, ktoré musia byť zachytené pomocou ASN.1, bude popísaný len formát BES.

2.2.2 Popis formátu BES pomocou ASN.1

Syntax formátu BES je popísaná v dokumente RFC 2630 Cryptographic Message Syntax (CMS)[9]. Syntax každej správy v CMS sa začína nasledovne:

```
ContentInfo ::= SEQUENCE{
    contentType ContentType,
    content [0] EXPLICIT ANY DEFINED BY contentType }
```

contentType je jednoznačný identifikátor (Object Identifier - OID), ktorý určuje, aké dáta sú v správe uvedené, v tomto prípade elektronický podpis[7]; *content* obsahuje samotné dáta, v prípade elektronického podpisu content = *signedData*.

```
SignedData ::= SEQUENCE{
    version CMSVersion,
    digestAlgorithms DigestAlgorithmIdentifiers,
    encapContentInfo EncapsulatedContentInfo,
    certificates [0] IMPLICIT CertificateSet OPTIONAL,
    crls [1] IMPLICIT CertificateRevocationLists OPTIONAL,
    signerInfos SignerInfos }
```

[7] Ak ide o podpísané dáta, potom OID = { iso(1) member-body(2) us(840) rsadsi(113549) pkcs(1) pkcs7(7) 2}.

DigestAlgorithmIdentifiers ::= SET OF DigestAlgorithmIdentifier

SignerInfos ::= SET OF SignerInfo

EncapsulatedContentInfo ::= SEQUENCE {
eContentType ContentType,
eContent [0] EXPLICIT OCTET STRING OPTIONAL }

Význam jednotlivých polí je nasledovný:

- *version* je použitá verzia CMS, v závislosti od ostatných polí nadobúda hodnotu 1 alebo 3,
- *digestAlgorithms* obsahuje zoznam použitých podpisovacích algoritmov, ktorými boli dáta podpísané,
- *encapContentInfo* obsahuje podpísané dáta. Typ dát je identifikovaný OID-om eContentType a samotné dáta sú v položke eContent. Zaujímavosťou je, že vynechaním poľa eContent sa zostavujú externé podpisy,
- *certificates* a *crls* sú voliteľné polia, ktoré obsahujú certifikáty a CRL, ktoré sa použijú pri overovaní platnosti elektronického podpisu. Úmyslom bolo poskytnúť všetky certifikáty a CRL potrebné na zostavenie a overenie certifikačnej cesty (pozri kap. 2.2.4).
- *signerInfos* obsahuje informácie o jednotlivých podpisovateľoch.

Bolo by zbytočné opisovať štandard, preto sústredíme v rámci tejto diplomovej práce na najdôležitejšie pole SignerInfo.

SignerInfo ::= SEQUENCE{
version CMSVersion,
sid SignerIdentifier,
digestAlgorithm DigestAlgorithmIdentifier,
signedAttrs [0] IMPLICIT SignedAttributes OPTIONAL,
signatureAlgorithm SignatureAlgorithmIdentifier,
signature SignatureValue,
unsignedAttrs [1] IMPLICIT UnsignedAttributes OPTIONAL }

SignerIdentifier ::= CHOICE {
issuerAndSerialNumber IssuerAndSerialNumber,
subjectKeyIdentifier [0] SubjectKeyIdentifier }

SignedAttributes ::= SET SIZE (1..MAX) OF Attribute

UnsignedAttributes ::= SET SIZE (1..MAX) OF Attribute

```
Attribute ::= SEQUENCE {
   attrType OBJECT IDENTIFIER,
   attrValues SET OF AttributeValue }

AttributeValue ::= ANY

SignatureValue ::= OCTET STRING
```

Jednotlivé položky SignerInfo majú nasledovný význam:

- *version* je opäť číslo verzie. Jeho hodnota závisí od sid. Ak je sid issuerAndSerialNumber, potom má version hodnotu 1, inak hodnotu 3,
- *sid* určuje spôsob akým je zadaný verejný kľúč, potrebný na overenie platnosti podpisu. Existujú dve možnosti:
  1. *issuerAndSerialNumber* určuje vydavateľa a sériové číslo certifikátu, ktorý bol vydaný pre daný verejný kľúč. Pretože CA nemôže vydať dva certifikáty s rovnakým seriovým číslom, je certifikát určený jednoznačne.
  2. *subjecKeyIdentifier* určuje certifikát podpisovateľa podľa rozšírenia (extension) subjectKeyIdentifier.
- *digestAlgorithm* je OID hašovacieho algoritmu,
- *signedAttrs* je množinou podpísaných atribútov. Aj keď je toto pole voliteľné, musí byť prítomné, ak je contentType rôzne od id-data. Ak je pole prítomné, musí obsahovať aspoň atribúty *content-type*, ktorý má rovnakú hodnotu ako eContetType v EncapsulatedContentInfo a *message-digest*, ktorý obsahuje odtlačok podpísaných dát,
- *signatureAlgorithm* je OID podpisovacieho algoritmu,
- *signature* je výsledný digitálny podpis získaný pomocou odtlačku dokumentu a podpisovacieho algoritmu,
- *unsignedAttrs* je množina nepodpísaných atribútov.

Každý z atribútov obsahuje svoj OID v poli *attrType* a hodnoty v poli *attrValues*. Medzi najdôležitejšie atribúty patria:

1. *content-type* iso (1) member-body (2) us (840) rsadsi (113549) pkcs (1) pkcs9 (9) 3, OID je teda 1.2.840.113549.1.9.3.
2. *message-digest* iso (1) member-body (2) us (840) rsadsi (113549) pkcs (1) pkcs9 (9) 4, OID je teda 1.2.840.113549.1.9.4.
3. *signing-time* iso (1) member-body (2) us (840) rsadsi (113549) pkcs (1) pkcs9 (9) 5, OID je teda 1.2.840.113549.1.9.5.

4. *countersignature* iso (1) member-body (2) us (840) rsadsi (113549) pkcs (1) pkcs9 (9) 6, OID je teda 1.2.840.113549.1.9.6.

Prvé dva atribúty boli už spomínané. Atribút signing-time obsahuje informáciu o čase, kedy bol podpis vytvorený. Táto informácia však nie je zaručená, pretože podpisovateľ nemusí mať prístup k dôveryhodnému zdroju času (ale napr. len systémovému času počítača, na ktorom podpisuje). Atribút countersignature (kontrasignatúra) špecifikuje jeden alebo viac podpisov nad oktetmi[8] DER kódovaného poľa signatureValue uvedeného v poli SignerInfo v signed-data. Teda kontrasignatúra podpisuje existujúci podpis (vstupom do podpisujúceho algoritmu je digitálny podpis podpísaného EP a podpisovateľov privátny kľúč), čo je dôležité z hľadiska viacnásobných podpisov.

Countersignature ::= SignerInfo

Polia kontrasignatúry majú rovnaký význam ako polia SignerInfo uvedené hore s nasledujúcimi výnimkami:

1. Pole signedAttrs musí obsahovať položku message-digest, ak obsahuje iné atribúty, ale nemusí obsahovať položku content-type.

2. Vstupom pre výpočet odtlačku sú DER kódované oktety signatureValue poľa SignerInfo, na ktoré je kontrasignatúra viazaná.

Kontrasignatúra je príkladom atribútu, ktorý môže mať viacero hodnôt (ale aspoň jednu) vďaka tomu, že AttributeValue je definované ako množina. Pretože unsignedAttrs je tiež definované ako množina, môže obsahovať viacero kontrasignatúr. A pretože kontrasignatúra sama osebe je SignerInfo, môže obsahovať v sebe ďalšiu kontrasignatúru.

2.2.3 Elektronický podpis založený na XML

Formát elektronického podpisu založený na XML predstavuje alternatívu k EP založenému na CMS. Výhodou XML je jednoduchá čitateľnosť vzhľadom na to, že XML súbory sú textové na rozdiel od binárneho podpisu zalažoného na CMS. Ďalšou výhodou je napr. to, že ľubovoľný formulár sa dá jednoducho reprezentovať ako XML súbor, ktorý sa dá zobraziť jediným spôsobom, čo je dôležité z hľadiska slovenskej legislatívy. Podpísanie XML formulára elektronickým podpisom založeným na XML je oveľa logickejšie ako podpisom založeným na CMS.
XML (eXtended Markup Language) predstavuje spôsob, ktorým sa dajú popísať ľubovoľné dáta. Štruktúra dát je popísaná pomocou *document type definition*, v skratke DTD, ktorý špecifikuje, čo má spĺňať systém do seba vnorených tagov, ich atribútov a textových hodnôt medzi tagmi, ktoré predstavujú samotné dáta, aby bol platný (valid). XML dokument môže vyzerať nasledovne.

[8] Oktetom je osmica bitov, t. j. jeden bajt

```
<xml-dokument>
        <autor>
                <meno>Ján Novák</meno>
                <Adresa>
                        <Ulica cislo="1">Bernolákova</ulica>
                        <Mesto>Bratislava</Mesto>
                </Adresa>
        </autor>
</xml-dokument>
```

Počet výskytov elementu	Označenie
1	žiadne označenie
0,1	?
0 a viac	*
1 a viac	+

DTD špecifikuje, ako musí vyzerať XML dokument na to, aby bol platný. Pre každý element, či už ide o tag, alebo atribút, je v DTD popísané koľkokrát sa môže vyskytovať, ako je to zobrazené v tabuľke.
DTD pre XML dokument v príklade by vyzeral nasledovne:

```
<xml-dokument>
        (<autor>)+
                <meno></meno>
                <Adresa>
                        <Ulica cislo></ulica>
                        <Mesto></Mesto>
                </Adresa>
        </autor>
</xml-dokument>
```

To že dokument môže mať viacero autorov je naznačené + pri tagu *autor*. EP založený na XML je popísaný v dokumentoch ETSI 101 903 XML Advanced Electronic Signatures (XAdES) [10] a XML Signature Syntax and Processing [11]. Dokument ETSI popisuje nasledovné formáty XML EP:

1. *XAdES* je analógiou BES-u,
2. *XAdES-T* je analógiou ES-T,
3. *XAdES-C* je analógiou ES-C,
4. *XAdES-X* je analógiou ES-X,

5. *XAdES-X-L* je formát medzi ES-X a ES-A. Oproti XAdES-X obsahuje navyše všetky údaje potrebné na overenie certifikačnej cesty ale neobsahuje časové pečiatky ako v ES-X Typ 1 alebo 2.

6. *XAdES-A* rozširuje XAdES-X-L o archívnu časovú pečiatku.

DTD pre XAdES vyzerá nasledovne:

```
<ds:Signature ID?>
   <ds:SignedInfo>
      <ds:CanonicalizationMethod/>
      <ds:SignatureMethod/>
      (<ds:Reference URI? >
         (<ds:Transforms>)?
         <ds:DigestMethod>
         <ds:DigestValue>
      </ds:Reference>)+
   </ds:SignedInfo>
   <ds:SignatureValue>
   (<ds:KeyInfo>)?
   <ds:Object>
      <QualifyingProperties>
         <SignedProperties>
            <SignedSignatureProperties>
               (SigningTime)
               (SigningCertificate)
               (SignaturePolicyIdentifier)
               (SignatureProductionPlace)?
               (SignerRole)?
            </SignedSignatureProperties>
            <SignedDataObjectProperties>
               (DataObjectFormat)*
               (CommitmentTypeIndication)*
               (AllDataObjectsTimeStamp)*
               (IndividualDataObjectsTimeStamp)*
            </SignedDataObjectProperties>
         </SignedProperties>
         <UnsignedProperties>
            <UnsignedSignatureProperties>
               (CounterSignature)*
            </UnsignedSignatureProperties>
         </UnsignedProperties>
      </QualifyingProperties>
   </ds:Object>
</ds:Signature>
```

Všetky elementy definované v dokumente XML Syntax and Processing [11] sa začínajú prefixom "ds". Význam jednotlivých položiek, ktorý nie je zrejmý z ich názvu alebo nie je zhodný z významom poľa v CMS podpise s rovnakým názvom, je nasledovný:

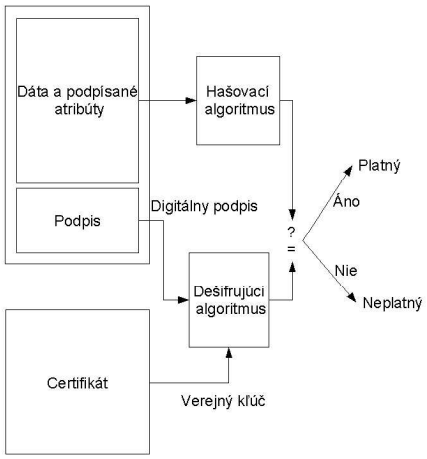
- *CanonicalizationMethod*; definuje postupnosť transformácii, ktorá sa nazýva *kanonizáciou*, a aplikuje sa na SignerInfo pred výpočtom odtlačku, ak sú dáta oktety, aby bol výpočet odltačku jednoznačný. Niektoré kanonizácie môžu napr. vyžadovať konverziu na formát UTF-8.
- *ds:Reference URI*; element Reference špecifikuje hašovací algoritmus a odtlačok a ďalšie atribúty, ktoré sa vzťahujú k odtlačku. Atribút URI identifikuje dátový objekt použitím URI-referencie (ktorá je špecifikovaná v RFC 2369).
- *ds:Transforms*; je postupnosť transformácií URI referencie, ktoré sú medzi sebou prepojené. Výstup z poslednej transformácie je vstupom pre algoritmus špecifikovaný v DigestMethod.
- *QualifyingProperties* je kontainerom pre všetky informácie, ktoré sa týkajú zaručeného EP.
- *SignedProperties* je zoznam podpísaných atribútov.
- *UnsignedProperties* je zoznam neporpísaných atribútov.

2.2.4 Overovanie platnosti zaručeného elektronického podpisu

Jednou z najdôležitejších činností, ktoré sa spájajú s elektronickým podpisom je overovanie platnosti elektronického podpisu. Celé overovanie platnosti dokumentu môžeme rozdeliť na dve časti.

1. *Overenie odtlačku* znamená overenie podpisu v súlade s RFC 2630 [9], t. j. dešifrovanie odtlačku z digitálneho podpisu pomocou verejného kľúča a overovacieho algoritmu a jeho porovnanie s odtlačkom získaným z podpísaných dát pomocou hašovacieho algoritmu. Celá situácia je znázornená na obrázku 2.3.
2. *Zostavenie a kontrola certifikačnej cesty* znamená overenie pravosti verejného kľúča podľa RFC 3280 [6].

Overovanie pravosti verejného kľúča je zložitým algoritmom, v ktorom je zostavená a overená jedna alebo viac *certifikačných ciest*.
Certifikačná cesta je pole certifikátov, pre ktoré platí viacero podmienok, najmä však:



Obrázok 2.3: Overovanie digitálneho podpisu

1. $\forall i, 1 <= i < n$: Vydavateľ (Issuer) certifikátu C_{i+1} = držiteľovi (Subject) certifikátu C_i.

2. Certifikát C_1 je selfSigned certifikát.

3. Certifikát C_n je certifikát koncového používateľa (end entity certificate). Verejný kľúč uvedený v certifikáte koncového používateľa sa používa na overenie odtlačku buď podpísaného dokumentu, alebo časovej pečiatky.

Príčinou vzniku viacerých certifikačných ciest môže byť napr. to, že entita si nechá vydať na svoj verejný kľúč certifikát od viacerých CA. Príkladom môže byť vydanie certifikátov koncovému užívateľovi CA-mi, ktoré patria do dvoch odlišných PKI, napr. jedna je slovenská akreditovaná CA a druhá je rakúska akreditovaná CA. Celý algoritmus zostavenia a overenia certifikačnej cesty je uvedený v RFC 3280 [6] a dokumente Certificate Path Validation [12], preto budú uvedené len hlavné myšlienky algoritmu.

1. Pri kontrole certifikátu sa musí overiť či doba, v ktorej certifikát overujeme, je v čase platnosti certifikátu. To znamená, že pre čas overovania platnosti platí: $notBefore <=$ čas overovania $<= notAfter$.

2. Je nutné skontrolovať, či certifikát nebol zrušený, tzn. či jeho platnosť bola predčasne ukončená. Na toto slúži CRL alebo OCSP[9]. V prípade, že máme

[9]OCSP - online certificate status protocol je protokolom určeným na zisťovanie platnosti certifikátu. Systém vyžaduje OSCP server, ktorý odpovedá na žiadosti overovateľov o zistenie platnosti certifikátu. Výhodou je zistenie okamžitého stavu platnosti certifikátu bez potreby zháňať aktuálne CRL.

k dispozícii CRL, je potrebné od jeho vydania čakať stanovený čas *cautionPeriod*. Podľa Certificate Path Validation [12] je cautionPeriod časová perióda, ktorá umožní, aby sa informácie o zneplatnení certifikátu spropagovali zneplatňovacím procesom a zverejnili pre overovateľov. Pokiaľ platí, že čas overenia $<=$ čas vydania CRL + cautionPeriod, potom je výsledkom overovania EP *neúplné overenie*.

3. Je potrebné overiť platnosť CRL a OCSP odpovedí, tzn. je potrebné overiť ich podpisy. Je potrebné si uvedomiť, že CA, ktorá CRL vydala môže ale nemusí mať rovnaký kľúč na podpisovanie certifikátov a CRL[10].

4. Ak podpis obsahuje jednu alebo viac časových pečiatok je potrebné overiť aj jej (ich) podpisy.

Z procesu overovania platnosti EP vyplýva, že pri overovaní platnosti EP môžu nastať tri prípady:

1. *Platný*. Tento prípad nastane vtedy, ak sú haše rovnaké a existuje aspoň jedna platná certifikačná cesta.

2. *Neplatný*. Tento prípad nastane vtedy, ak sa haše nerovnajú alebo ani jedna zo zostavených ciest nie je platná.

3. *Nemožno overiť (Neúplné overenie)*. Tento prípad nastane vtedy ak sa haše rovnajú, ale nie je k dispozícii dostatok informácií na zostavenie a overenie certifikačnej cesty.

[10]To, či sa jedná o kľúč určený na podpisovanie certifikátov a(lebo) CRL je uvedené v certifikáte v rozšírení (extension) *key usage*. Či je kľúč použitý na podpisovanie OCSP odpovedí a(lebo) časových pečiatok, je uvedené v rozšírení *extended key usage*.

Kapitola 3

Využitie viacnásobných EP v praxi

Táto kapitola bude hovoriť o legislatíve, týkajúcej sa elektronického podpisu a najmä viacnásobných podpisov. Bude sa snažiť ukázať súčasný stav slovenskej legislatívy, týkajúcej sa EP. Budú uvedené príklady využitia viacnásobných EP.
Prvou otázkou, ktorú je potrebné vyriešiť, je "Čo všetko je možné považovať za viacnásobný elektronický podpis?"
V predchádzajúcej kapitole bolo spravené rozdelenie elektronických podpisov na interné a externé. Je možné považovať za viacnásobný podpis viacero externých "jednoduchých" podpisov? Na to, aby bolo možné zodpovedať túto otázku, je potrebné stanoviť, za akých podmienok sa stáva elektronický dokument podpísaný viacerými podpismi.

Definícia 3 *Elektronický podpis sa vzťahuje na elektronický dokument, ak je s ním* previazaný alebo logicky spojený. *To znamená, že zmena elektronického dokumentu zneplatní elektronický podpis, vzťahujúci sa na dokument s pravdepodobnosťou blížiacou sa k 1*[1].

Je dôležité uvedomiť si, že podpis podpisu elektronického dokumentu sa tiež vzťahuje na pôvodný dokument, pretože zmenou elektronického dokumentu je zneplatnený. Toto nie je zabezpečené použitou technológiou, logicky to však platí v kontexte daného dokumentu a malo by to byť podchytené v podpisovej politike. Napr. nech je kontrasignatúrou napr. nejaké osvedčenie, že podpis na dokumente je právoplatný. Potom zmenou dokumentu bude zneplatnený podpis, osvedčenie samo osebe zostáva platné, ale v kontexte daného dokumentu platné nie je, pretože osvedčuje niečo, čo nie je pravda.

Definícia 4 *Elektronický dokument je podpísaný viacerými elektronickými podpismi, ak sa naň vzťahujú* apoň dva *elektronické podpisy. Týmto podpisom sa hovorí* viacnásobný elektronický podpis dokumentu.

[1]Pri použití SHA-256 by to malo byť $p = 1 - 2^{-256}$.

3.1 Slovenská legislatíva týkajúca sa EP

Pojem EP je zadefinovaný v zákone 215/2002 [7], ako už bolo spomínané v kapitole 2.2. Dôležitou pre aplikácie viacnásobných EP v praxi sa však ukazuje najmä vyhláška 542/2002 "o spôsobe a postupe používania elektronického podpisu v obchodnom a administratívnom styku" [13]. Táto vyhláška definuje spôsob vytvárania viacnásobných EP. Pretože spôsob vytvárania viacnásobných EP ovplyvňuje formáty, ktoré je možné použiť, je príloha tejto vyhlášky odcitovaná spolu s dôsledkami v kapitole 4. NBÚ tiež v dokumente "Schválené formáty zaručených elektronických podpisov" špecifikuje formáty ZEP-ov, tieto však tiež budú rozobrané v kapitole 4, takže teraz bude podrobnejšie rozobratá vyhláška 542/2002 (ďalej vyhláška).
V §3 vyhlášky sa definuje používanie EP a ZEP nasledovne:

> Používanie elektronického podpisu
>
> (1) V administratívnom styku možno na podpisovanie elektronického dokumentu používať len zaručený elektronický podpis.
>
> (2) V obchodnom styku možno na podpisovanie elektronického dokumentu používať elektronický podpis alebo zaručený elektronický podpis.

Vyhláška teda obmedzuje použitie EP na obchodný styk, čo však nie je v rozpore s direktívou EÚ. Vyhláška ďalej stanovuje v §6 až §8 ako prebieha spracovanie elektronického dokumentu na strane administratívy.
Z hľadiska tejto diplomovej práce je dôležité, že pri spracovaní podávaných elektronických dokumentov podpísaných ZEP-om, sa používa elektronická podateľňa, ktorá sa správa ako automatický podpisovač a overovač. Podania môžu byť tiež podpísané viacerými osobami, preto je možné uvažovať o uplatnení viacnásobných EP. V §9 je špecifikované, aké formáty elektronických dokumentov sa môžu používať v obchodnom styku a v styku s verejnou mocou (tieto sú vymenované v prílohe 3 vyhlášky). Z hľadiska tejto diplomovej práce je najdôležitejšia príloha 2 vyhlášky, ktorá hovorí, ako má prebiehať podpisovanie elektronického dokumentu, pričom rozoberá tri prípady:

1. Jediný podpisovateľ,
2. Viacerí podpisovatelia na jednom mieste,
3. Viacerí podpisovatelia na viacerých miestach.

Ako už bolo spomínané, príloha 2 je veľmi dôležitá pre návrh formátu EP, preto je celá odcitovaná v kapitole 4.4. Pri návrhu formátu viacnásobného EP je potrebné si uvedomiť, aké vlastnosti má mať viacnásobný EP. Pri používaní viacnásobných EP môžu byť relevantné nasledovné požiadavky:

1. *Jednotlivé podpisy vo viacnásobnom EP môžu byť v rôznych formátoch*. Táto požiadavka je veľmi ťažko realizovateľná a aj keď by bolo veľmi pekné, aby mohol každý podpisovateľ zvoliť formát, je nemožné skombinovať podpisy založené na CMS a XML. Napr. podpis by bol v jednom z formátov CMS a kontrasignatúra na tento podpis by bola v XML. Formáty uvedené v kapitole 2 však s takouto možnosťou nepočítajú, a preto by toto riešenie nebolo kompatibilné so svetovými štandardmi.

2. Jednotliví podpisovatelia nemusia byť na jednom mieste, takže je žiaduce, aby *bolo možné dopĺňať podpisy jednotlivých podpisovateľov vo viacnásobnom podpise*, pričom nie je podstatné, v akom poradí elektronický dokument jednotliví podpisovatelia podpisovali.

3. *Nie je žiaduce, aby boli podpisovatelia schopní odoberať podpisy iných podpisovateľov*. Táto vlastnosť nie je potrebná v prípade, že je počet podpisovateľov vopred daný, pretože podpisovatelia na konci procesu podpisovania dostanú (aspoň by mali) dokument so všetkými podpismi a vedia si skontrolovať, či nejaký podpis nechýba. Príkladom, kde je možné túto vlastnosť vyžadovať, sú napr. petície v elektronickej forme, ktoré sú typickým prípadom dokumentu s viacerými podpismi. Zrejme je nežiaduce, aby podpisovatelia boli schopní odoberať podpisy iných podpisovateľov a týmto spôsobom sabotovať petíciu. Celý problém sa dá vyriešiť pomocou externých elektronických podpisov založených na XML, kde by v rámci podpisu bola referencia na dokument (petíciu) uložený na webe. Problém ale ostáva pri CMS.[2]

4. *Ak už elektronický dokument podpísali všetci podpisovatelia, je žiaduce zabrániť ďalším potenciálnym podpisovateľom podpisovať dokument*. Táto vlastnosť má oporu vo vyhláške 542/2002, ale v princípe nie je potrebná. Stačí, ak je v dokumente uvedené, kto ju podpisuje a všetkým podpisovateľom, ktorým je doručená finálna verzia so všetkými podpismi, stačí overiť, či podpisovatelia zodpovedajú osobám uvedeným v dokumente. V prípade petície, kde počet podpisovateľov nie je známy, je potrebné postupovať špecifickým spôsobom. Napr. môže sa obmedziť doba, kedy je možné podpisovať pomocou EP a potom odfiltrovať všetky podpisy s časovou pečiatkou, ktorých čas vzniku spadá mimo tohoto intervalu.

Druhá požiadavka bola zohľadnená pri tvorbe štandardov k viacnásobným EP. Naviac, posledná požiadavka bola zohľadnená pri tvorbe vyhlášky 542/2002, výsledkom čoho je nekompatibilita slovenského viacnásobného EP s rozšírenými štandardmi, ako je to uvedené v kapitole 4.

[2] Spôsob, ako organizovať petíciu v elektronickej forme vyžaduje splnenie viacerých predpokladov, ktoré sa prejavujú aj v požiadavkách kladených na používaný EP. Bohužiaľ, táto diplomová práca neposkytuje dostatok priestoru, aby sa dala pokryť aj táto špecifická oblasť.

3.2 Využitie viacnásobných EP v praxi

Viacnásobné EP možno využiť v obchodnom ako aj v administratívnom styku. Reálne sa viacnásobné podpisy vyskytujú napr. v zmluvách, kde podľa zákona musia byť podpisy účastníkov na jednej listine (najmä zmluvy okolo nehnuteľností), ale aj v podaniach a pri osvedčovaní podpisu (pred) notárom. Je potrebné rozlíšiť jednotlivé prípady viacnásobných EP. Ak to nie je uvedené v dokumente, v prípade zmlúv alebo podaní nie je dôležité, v akom poradí podpisovatelia dokument podpíšu. Naopak, v prípade notárskeho osvedčenia (úradného overenia na matrike) je podstatné, že notár (matrika) osvedčuje *existujúci podpis* v dokumente, a teda je jeho podpis s týmto podpisom nejakým spôsobom zviazaný. V zásade môžeme rozlíšiť viacero druhov viacnásobných EP:

- Nezávislé podpisy, pri ktorých poradie nie je relevantné. Príkladom sú už spomínané zmluvy o nehnutelnostiach, napr. kúpno-predajná zmluva.
- Nezávislé podpisy, pri ktorých je poradie relevantné.
- Previazané podpisy. Príkladom je elektronické notárske osvedčenie podpisu podľa zákona 323/1992 o notároch a notárskej činnosti.

3.2.1 Nezávislé EP

Problémom pri nezávislých podpisoch je ako zaručiť, že všetci podpisovatelia majú verziu elektronického dokumentu spolu so všetkými podpismi. Spôsobov, ako túto požiadavku zabezpečiť, je viacero.
Prvý spôsob vychádza síce z reality, ale jeho uplatnenie v elektronickej forme ruší všetky výhody, vyplývajúce z nej. Princíp je ten, že sa všetci podpisovatelia stretnú na jednom mieste, kde dokument podpíšu a zoberú si každý svoju kópiu. Nevýhody sú zrejmé. Používatelia musia ísť na miesto, kde dokument podpíšu, nemôžu tak urobiť z pohodlia svojho domova a ak nepotrebujú ďalej s elektronickým dokumentom pracovať, je to pre nich skôr na príťaž.
Druhý spôsob je ten, že všetci podpisovatelia sú na rôznych miestach a chcú podpísať elektronický dokument tak, aby na konci procesu podpisovania mali všetci k dispozícii ten istý elektronický dokument podpísaný všetkými podpisovatelmi. Predpokladom je, že všetci používatelia si môžu medzi sebou zasielať správy v elektronickej forme, alebo vedia, kde je dokument k dispozícii. V druhom prípade je potrebné, aby sa k dokumentu dostali len podpisovatelia, čo implikuje ďalšie požiadavky na autentifikáciu a dôvernosť dokumentu a spojenia. Tak isto je kľúčové zabezpečiť integritu prenášaných dát. Veľmi vhodné by bolo, aby sa dal s procesom podpisovania používať niektorý (ideálne viaceré) z formátov CMS a(lebo) XML, aby boli podpisy kompatibilné s medzinárodnými štandardmi.
Ako už bolo spomínané, nezávislé EP sa môžu využívať v:

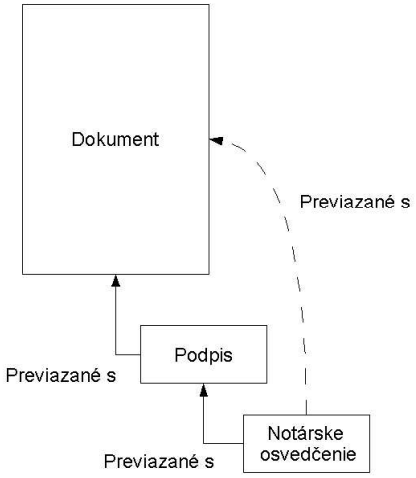
- Typoch zmlúv, v ktorých zákon vyžaduje, aby boli podpisy účastníkov na jednej listine.

- Podaniach, ktoré robí viac ako jedna osoba. Pretože podanie je administratívnym stykom, musia podľa vyhlášky 542/2002 podpisovatelia podpisovať ZEP-om.
- Petíciách.

V štandardoch ETSI a W3C nie je možné nájsť popis formátu, ktorý by umožňoval rozlišovať poradie podpisovateľov. Rozšírenie CMS/XML nezávislých podpisov na viacnásobné podpisy, ktoré túto vlastnosť umožňujú, však nie je problematické a je popísané v kapitole 4.

3.2.2 Previazané EP

V praxi sa s previzanými podpismi stretávame, najmä pri osvedčovaní podpisov, či už notárom, alebo iným úradom. Osvedčenie sú dáta pripojené k podpísanému dokumentu, ktoré potvrdzujú nejakú vlastnosť dokumentu alebo podpisu. Napr. pri osvedčovaní podpisu notár svojím osvedčením zaručuje, že daná osoba pred ním daný dokument v daný čas podpísala alebo uznala podpis za svoj. Celá situácia je zobrazená na obrázku 3.1.
Teoreticky je však možné využiť previazané EP aj pri elektronických podateľniach.



Obrázok 3.1: Notárske osvedčenie pripojené k podpisu

Elektronická podateľňa pri prijatí pripojí na podpísaný dokument vlastné dáta, ktoré obsahujú údaje o tom, kedy a ktorou elektronickou podateľňou bol dokument prijatý. Teda podpis podateľne sa viaže na celý dokument. V elektronickom svete však nie je potrebné robiť podpis nad podpísaným dokumentom, ale stačí podpísať podpis, pretože tento je previazaný s dokumentom a pri podpísaní podateľňou sa

podpis podateľne stane previazaným s podpisom pôvodného podpisovateľa. Tieto dve previazania zaručujú, že podpis podateľne je previazaný aj s pôvodným elektronickým dokumentom, takže sa zachová princíp, ktorý platí v "papierovej" podobe. V súhrne je možné previazané viacnásobé EP využívať v:

- Osvedčeniach.
- Elektronickej podateľni.

Kapitola 4

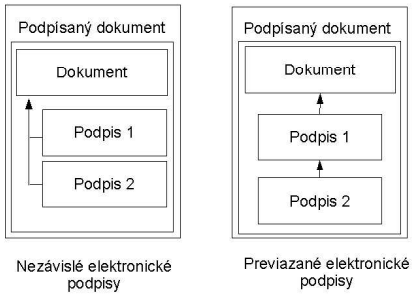
Viacnásobné elektronické podpisy

V predchádzajúcich dvoch kapitolách bol popísaný elektronický podpis z pohľadu technického, legislatívneho i využitia v praxi. V tejto kapitole bude rozšírený pojem jednoduchého EP na viacnásobné EP, pričom bude zohľadnené špecifikum slovenskej legislatívy. V predchádzajúcej kapitole boli rozobraté požiadavky, ktoré kladie prax na viacnásobné podpisy. Tieto požiadavky sú reflektované v použitých technológiách a formátoch EP, ktoré sa používajú na viacnásobné podpisy. Slovenská legislatíva je výnimočná v tom, že sa pri používaní *zaručených* viacnásobných EP neriadi medzinárodnými štadardmi, ale v dokumente "Schválené formáty zaručených elektronických podpisov" navrhuje vlastný formát ZEP (ZIP), ktorý nie je kompatibilný s formátmi ETSI, aj keď ich využíva. Dôvodom na tento krok je vyhláška 542/2002, ktorá stanovuje okrem iného aj postup pri používaní zaručených viacnásobných EP v obchodnom styku.
Ako už bolo povedané, pri tvorbe štandardov sa prihliadalo na potreby praxe, a preto je možné vo všeobecnosti rozlíšiť dva druhy viacnásobných EP:

1. *Nezávislé* (independent) podpisy; jednotlivé "jednoduché" EP nie sú medzi sebou žiadnym spôsobom prepojené a ich poradie nie je relevantné. Celá situácia je znázornená na obrázku 4.1.

2. *Previazané* (embedded, chained) podpisy; jednotlivé "jednoduché" EP sú medzi sebou nejakým spôsobom prepojené a ich poradie je preto dôležité (nie je možné previazať podpis s neexistujúcim podpisom). Príklad, keď je podpis previazaný s iným podpisom, je znázornený na obrázku 4.1.

Z pohľadu medzinárodných štandardov sa viacnásobnými podpismi zaoberajú napríklad dokumenty ETSI 101 733 a ETSI 101 903 a najmä príslušné normy, z ktorých tieto dva dokumenty vychádzajú, t. j. RFC 2630 "CMS Syntax and processing" a W3C odporúčanie "XML Signature Syntax and processing".
V prípade viacnásobných previazaných podpisov je možné rozlišovať rôzne previazania medzi podpismi. EP je spravený nad dátami, v tomto prípade inými EP. Vo všeobecnosti môže byť týchto podpisov jeden alebo viac. Ak je implemento-



Obrázok 4.1: Viacnásobné elektronické podpisy

vané previazanie dvoch podpisov, ako je to urobené v ETSI podpisoch pomocou kontrasignatúr, je možné rozšíriť toto previazanie na viac podpisov nasledovne:

1. *Spraviť kontrasignatúru nad každým s podpisov* je fungujúcim riešením, ktoré však prináša redundanciu v podobe viacerých kontrasignatúr[1]. Tento problém je odstránený v XML podpisoch pomocou referencí.

2. *Spraviť "kontrasignatúru" nad všetkými podpismi naraz* je riešenie, ktoré presadzuje slovenská legislatíva v prípade takzvaného uzatváracieho podpisu.

Previazanie podpisov je možné znázorniť pomocou orientovaného grafu, ktorý neobsahuje cykly[2]. V tomto grafe sú jednotlivé podpisy vrcholmi a hranou je reprezentované previazanie, t. j. ak existuje hrana (x, y), potom je podpis x urobený nad podpisom y. Podpisy, z ktorých nevedie žiadna hrana, sú urobené nad dokumentom, ktorý je jednoznačne daný k danému grafu podpisov.
Otázkou je, či by nebolo vhodné zlúčiť dva takéto grafy do jednej štruktúry, čo by umožňovalo zachytiť napr. podpisy nad viacerými dokumentmi. Takéto rozšírenie je možné spraviť veľmi jednoducho:

Definícia 5 *Graf $G = (V, E)$ sa nazýva* grafom podpisov*, ak spĺňa nasledovné podmienky:*

1. *V je konečná množina.*

2. *Neobsahuje cykly ani slučky,*

[1] Pre n podpisov je potrebné spraviť n kontrasignatúr.

[2] Cyklus v grafe je definovaný nasledovne: Nech G = (V,E) je graf a $v, v_1, \ldots, v_n \in V$ také, že $(v, v_1) \in E, (v_n, v) \in V$ a $(v_i, v_{i+1}) \in V \; \forall i \in \{1, \ldots, n-1\}$ ani slučky. Cykly nemôže obsahovať preto, lebo by nebolo možné takéto podpisy vôbec vytvoriť, pretože napr. v triviálnom prípade dvoch zacyklených podpisov by na vytvorenie prvého bolo potrebné mať hotový druhý podpis a naopak. Podobná úvaha sa dá použiť pri cykle ľubovoľnej dĺžky. Rovnako pri slučkách by podpis musel byť urobený nad sebou samým.

3. *Existuje množina* $V' \subseteq V$*, taká že* $\forall(x,y) \in E : x \notin V'$ *a nazýva sa množinou dokumentov.*

4. *Existuje funkcia* $f : E \to N$*, ktorá je injektívna*[3].

Prvá podmienka znamená, že nemôžeme mať nekonečný počet dokumentov alebo podpisov. Druhá podmienka je jasná už z predchádzajúcich úvah. Tretia podmienka vyčleňuje špeciálnu množinu vrcholov, ktoré budú reprezentovať dokumenty. Dokumenty nemôžu byť podpismi, takže ak sa nachádzajú v hrane, tak musia byť jedine na druhom mieste. Posledná podmienka zavádza nejaké ohodnotenie hrán, v ktorom dve hrany nemajú rovnaké hodnoty. Toto ohodnotenie je dobré pri stanovovaní poradia, v ktorom budú dáta vstupovať do hašovacej funkcie pri vytváraní a overovaní podpisu. Je nutné stanoviť nejaké poradie napr. od najnižšej hodnoty po najvyššiu, pretože hrany v obyčajnom orientovanom grafe sú si rovnocenné a keby overovateľ pomiešal poradie dokumentov pri výpočte odtlačku, s pravdepodobnosťou blížiacou sa k istote, by mu vyšiel iný odtlačok.
V skutočnosti je podmienka injektívnosti príliš silná, postačuje podmienka $\forall x, y, z \in V : (x,y) \in E \wedge (x,z) \in E \Rightarrow f(x,y) \neq f(x,z)$.
Teda nemusia mať všetky hrany rôzne hodnoty, ale postačí, ak majú rôzne hodnoty hrany vedúce z jedného vrchola.
Ideálne by bolo, keby formát viacnásobných podpisov vedel zachytiť ľubovoľný graf podpisov.

4.1 Viacnásobné ETSI podpisy

Pretože aj v kapitole 2.2 sa spomína najprv podpis založený na CMS, a potom na XML, bude zachované toto poradie pri uvedení, akým spôsobom pristupujú oba formáty podpisov k problematike viacnásobných EP.

4.1.1 Viacnásobný podpis založený na CMS

Dokument ETSI 101 733 [3] (a RFC 2630) ukazuje ako zachytiť viacnásobné EP pomocou formátov založených na CMS, pričom sa explicitne rozlišuje medzi previazanými a nezávislými EP.

1. Nezávislé EP sú podporované pomocou viacerých nezávislých polí *signerInfo*. Každé signerInfo musí obsahovať vštky potrebné atribúty a musí byť overovateľom spracovávané osobitne.

2. Previazané EP sú podporované pomocou nepodpísaného atribútu *countersignature*. Na každý podpis je možné vyhotoviť viacero kontrasignatúr a každá z kontrasignatúr môže mať vlastné kontrasignatúry.

[3]Funkcia je injektívna vtedy, ak dvom rôznym vzorom zodpovedajú dva rôzne obrazy, t. j. $\forall x, y \in D(f) : x \neq y \Rightarrow f(x) \neq f(y)$

V prípade viacnásobných EP založených na CMS nie je možné vytvoriť ľubovoľný graf previazaných podpisov, len niektoré jeho špeciálne prípady. Ide konkrétne o orientovaný graf, v ktorom je počet hrán, *odchádzajúcich* z vrchola obmedzený na 1. Dvôvodom je obmedzenie, vyplývajúce z definície kontrasignatúry a nemožnosti podpisovať viacero dokumentov jediným podpisom.

4.1.2 Viacnásobný podpis založený na XML

Pretože XML EP nie je kópiou CMS EP, existujú aj rozdiely medzi podporovaním viacnásobných EP, pretože kopírovanie syntaxe ASN do XML by nedovoľovalo využívať niektoré výhodné vlastnosti, ako sú referencie.

1. Nezávislé EP sú podporované viacerými elementmi *Signature*. Ich zmysel je rovnaký ako nezávislé signerInfo.

2. Previazané EP sú podporované pomocou elementu Countersignature, ktorý má rovnaký význam ako pole counter-signature v CMS.

Zmenou v XML oproti CMS je zavedenie *manifestu*. Každý element Signature obsahuje element Reference na podpísané dáta. Element reference môže ukazovať priamo na dáta alebo na ľubovoľný element v XML podpise. Význam elementu Manifest je v tom, že v sebe obsahuje viacero referencí. Takže v prípade, že viacero podpisovateľov podpisuje množinu dokumentov, nemusia mať v elemente Signature vymenované všetky referencie, čo je redundantné, ale majú tam referenciu na manifest, ktorý obsahuje všetky potrebné referencie na jednom mieste.
Problémom odporúčania W3C je, že neobsahuje explicitný postup ako vytvoriť nezávislé podpisy, preto postup, ktorý je opísaný vyššie, môže byť vhodný, ale nemusí byť jediný ani najlepší.
Pomocou referencí je možné realizovať *ľubovoľný* acyklický orientovaný graf. Pokiaľ manifest chápeme tiež ako uzol v grafe, potom manifest umožňuje znížiť počet hrán, t. j. referencí.

4.1.3 Nezávislé podpisy so stanoveným poradím

Ak je pri vzniku viacnásobného EP kladená požiadavka na to, aby overovateľ vedel z podpisu zistiť poradie jednotlivých podpisovateľov, nie je možné priamočiaro aplikovať nezávislé ETSI podpisy, pretože tieto v sebe nenesú informáciu o poradí podpisovateľov.
Prvým nápadom, akým by bolo možné určiť poradie podpisov bez toho, aby bolo nutné ETSI podpisy rozširovať, je utriediť podpisy vzostupne podľa času z časových pečiatok. Tento postup má však nasledovné nevýhody:

1. Nedá sa aplikovať, pokiaľ formát EP v sebe nenesie časovú pečiatku. Aj keď je časová pečiatka veľmi vhodná, pretože poskytuje dôveryhodnú informáciu

o čase vzniku podpisu, nemusí v podpise figurovať, pretože z kryptografického hľadiska ide o "luxusný" objekt. To znamená, že na jeho vytvorenie je potrebný celý rad požiadaviek, ktoré nie sú triviálne[4].

2. Nedá sa aplikovať, pokiaľ sa časy na časových pečiatkach jednotlivých podpisov od seba líšia len veľmi málo (rádovo sekundy). Ak boli totiž jednotlivé podpisy vytvorené zhruba v rovnakom čase, neznamená to, že žiadosti o časovú pečiatku prídu na server časových pečiatok v rovnakom poradí v akom boli odoslané. To znamená, že pri vkladaní časových pečiatok sa môže "pomiešať" poradie jednotlivých podpisov.

3. Autorita časových pečiatok nie je povinná okamžite vydať časovú pečiatku a ani vydávať časové pečiatky v tom istom poradí, v akom prišli požiadavky na server TSA.

Druhá a najmä tretia nevýhoda si vynúti zavrhnutie tohoto spôsobu na určovanie poradia jednotlivých podpisov, pretože poradie *nie je* garantované.
Druhý spôsob ako zabezpečiť poradie podpisov je nasledovné použitie kontrasignatúr: Prvý podpisovateľ podpíše dokument. Každý nasledujúci podpisovateľ vytvorí kontrasignatúru na podpis predchádzajúceho podpisovateľa. Tento postup je použiteľný, ale jeho nevýhodou je, že ak sa ukáže neplatnosť niektorého z podpisov, tak všetky jeho kontrasignatúry sú neplatné. Výhodou tohto podpisu je, že nevyžaduje žiadne zásahy do štruktúry elektronického podpisu, ktorý podpisovatelia používajú.
Tretí spôsob, ktorý by bol okopírovaním "papierovej" praxe, znamená napísať v dokumente poradie podpisovateľov. Tento spôsob tiež nevyžaduje zásahy do štruktúry EP, ale vyžaduje zásahy do štruktúry dokumnetu. Problémom môže byť platnosť podpísaného dokumentu, ak sa podarí dokázať, že podpisy neboli pripojené k dokumentu v tom poradí, v akom je to napísané v dokumente.
Posledný spôsob si vyžaduje malý zásah do EP, ale zabraňuje pomiešaniu podpisov za predpokladu, že podpisovatelia nebudú vedome sabotovať proces podpisovania. Pri podpisoch založených na CMS spočíva technika v zavedení *podpísaného* atribútu, ktorý by obsahoval v sebe poradové číslo podpisu. Pri podpisoch založených na XML je potrebné vytvoriť referenciu na poradové číslo podpisu a pridať ju k ostatným referenciám podpisu. Toto riešenie nesie so sebou aj problémy súvisiace s medzinárodnou akceptáciou takýchto podpisov a Slovenskou ale aj zahraničnou legislatívou. Na to, aby sa toto riešenie presadilo aj mimo územia Slovenska je totiž potrebné presadiť ho do medzinárodného štandardu, pretože Slovensko nemá takú silnú organizáciu (akou je napr. Microsoft), ktorá by mohla presadiť vlastný štandard.

Atribút poradia v CMS

Atribút v CMS je definovaný nasledovne:

[4]Už len vybudovanie fungujúcej autority časových pečiatok je drahá záležitosť.

```
Attribute ::= SEQUENCE{
   attrType OBJECT IDENTIFIER,
   attrValues SET OF AttributeValue}

AttributeValue ::= ANY
```

Pole *attrType* by v prípade poradového atribútu obsahoval jednoznačný identifikátor tohto atribútu zakotvený v zozname OIDov[5]. Pole *attrValues* by bolo typu *INTEGER* a obsahovalo by samotné poradové číslo podpisu.

Poradie podpisu v XML

Element SignedInfo má v XML nasledovnú schému:

```
<element name="SignedInfo" type="ds:SignedInfoType"/>
   <complexType name="SignedInfoType">
      <sequence>
         <element ref="ds:CanonicalizationMethod"/>
         <element ref="ds:SignatureMethod"/>
         <element ref="ds:Reference" maxOccurs="unbounded"/>
      </sequence>
      <attribute name="Id" type="ID" use="optional"/>
   </complexType>
```

Element *Reference* obsahuje referenciu na objekt, v ktorom bude uložené poradové číslo podpisu. Zostáva otvorené, kde sa bude tento objek nachádzať. Pretože sa poradové číslo vzťahuje na podpis samotný, myslím si, že by malo byť súčasťou samotného elementu *Signature*. V rámci elementu Signature je jediným vhodným elementom, do ktorého by sa poradové číslo podpisu dalo uložiť element Object, ktorý môže obsahovať ľubovoľný objekt. Nech sa tento objekt nazýva *SignatureNumber*. Problémom ostáva ako reprezentovať dáta uložené v elemente SignatureNumber. Možností je viacero, buď bude poradové číslo uložené ako atribút elementu SignatureNumber, alebo sú uzavreté medzi vstupným a výstupným tagom. DTD elementu, ktorý by v sebe obsahoval poradové číslo ako atribút, by vyzeralo nasledovne:

```
<!ELEMENT SignatureNumber( ) >
<!ATTLIST SignatureNumber
            Id ID #IMPLIED
            Value INTEGER #IMPLIED >
```

A DTD elementu, ktorý by v sebe obsahoval poradové číslo uzvareté otvárajúcim a ukončujúcim tagom, by vyzeralo nasledovne:

[5] OID je skratka pre Object IDentifier

<!ELEMENT SignatureNumber(#CDATA) >
<!ATTLIST SignatureNumber
Id ID #IMPLIED

Výhodou prvého prístupu je, že nepotrebuje kanonizáciu, pretože element neobsahuje žiadne dáta. Kanonizáciou je však možné docieliť transformáciu jednej formy na druhú, len si treba zvoliť, ktorá z foriem bude zobratá za kanonickú. Jeho nevýhodou je malá rozšíriteľnosť v prípade, že by boli kladené na číslovanie ďalšie požiadavky a súčasne je požiadavkou nezmeniť štruktúru elementu SignatureNumber. Pri druhom prístupe je kanonizácia, naopak, nevyhnutná, pretože zápis čísla môže byť v rôznom formáte. Rozširovanie elementu SignatureNumber ponúka oproti prvej možnosti zakomponovať informácie aj do samotných dát aj keď to nie je správne, pretože to mieša logiku dát reprezentovanú v tagoch so samotnými dátami.
Atribút *Id* musí byť zachovaný v oboch možnostiach, aby naň mohli referencie odkazovať.
Riešenie atribútom je technicky jednoduché, ale prináša so sebou nasledovné problémy:

1. Je potrebné stanoviť identifikátor (OID), ktorý bude identifikovať atribút.
2. Niekto musí zaregistrovať OID.
3. Nie je zaručené, že ďalšie štáty budú akceptovať výber OID-u. Tým pádom nie je zaručená medzinárodná kompatibilita.

Prvé dva body by znamenali hľadanie silnej medzinárodnej podpory, aby bol nový OID akceptovaný a dostal sa do medzinárodných štandardov. Bez toho nie je možné docieliť medzinárodnú kompatibilitu, a tým pádom jeho zavedenie na Slovensku nemá význam.

4.2 Overovanie viacnásobných EP

Overovanie viacnásobných EP prináša oproti overovaniu jedného EP nasledovné problémy:

- Zvýšenie počtu podpisov, ktoré je nutné overiť, t. j. zvýšené nároky na výkon vyplývajúce najmä z overovania rôznych certifikačných ciest. Pri vhodnej implementácii je možné v niektorých prípadoch zefektívniť overovanie podpisov.
- Prítomnosť všetkých potrebných podpisov.
- Prítomnosť nejakých nadbytočných podpisov.
- Prípad, keď niektoré z prítomných podpisov nie sú platné.

Niektoré úvahy, ktoré budú uvedené ďalej, sa dajú použiť aj pri jednom EP. V takomto prípade bude o tom spravená zmienka.

4.2.1 Overovanie digitálnych podpisov

Overovanie digitálnych podpisov sa veľmi nelíši od overovania digitálneho podpisu pri jedinom EP. Jednotlivé dáta sa spracujú hašovacou funkciou do digitálnych odtlačkov, ktoré sa porovnajú s odtlačkami získanými dešifrovaním digitálnych podpisov. Úspora času sa dá dosiahnuť napr. pri paralelných podpisoch uložením hašu dokumentu, čím sa pri n paralelných podpisoch nad jedným dokumentom ušetrí $n - 1$ hašovacích transformácii, za predpokladu, že podpisy neobsahujú žiadne podpísané atribúty. Teda pri hašovaní klesne úspora času z lineárnej na konštantnú[6] vzhľadom na počet hašovacích transformácií.
Žiaľ, pri dešifrovacej transformácii nie je takéto zefektívnenie možné.

4.2.2 Overovanie certifikačnej cesty

Pri overovaní certifikačnej cesty je možné využiť fakt, že pri overovaní niektorého z EP boli už niektoré časti grafu PKI spracované overovacím algoritmom. Problémom je, že podpisy sú vo všeobecnosti vytvorené v rôznom čase, a teda môže dvôjsť k revokáciám, ktorými sa overovač pri overovaní predchádzajúcich podpisov nezaoberal. Tento problém sa dá odstrániť jednoducho: ak pri overovaní overovač narazí na certifikát, ktorý sa už v priebehu predchádzajúceho overovania vyskytol a čas, ku ktorému sa platnosť certifikátu overuje je menší, ako čas, ku ktorému sa platnosť certifikátu overovala v predchádzajúcom prípade, potom je možné použiť výsledok predchádzajúceho overenia pre terajšie overenie. Negatívnym faktom je, že pri takomto potenciálnom zefektívnení časovej náročnosti rastie pamäťová náročnosť, pretože je nutné pamätať si identifikátory certifikátov a ich platnosť, resp. neplatnosť v danom čase[7].
Úspora času pri tom závisí od štruktúry grafu, ktorý predstavuje PKI a môže veľmi kolísať. Extrémnym prípadom, kedy je časová úspora nulová, je PKI, ktorú reprezentuje nesúvislý graf, pričom certifikáty podpisovateľov sú každý v inom komponente. Opačný prípad je, keď všetkým podpisovateľom vydala certifikáty tá istá CA, potom stačí overiť cestu ku koreňovej CA v ideálnom prípade len raz.
Hore uvedené zrýchlenie overovania pravosti certifikátu je tým pravdepodobnejšie, čím je graf PKI väčší čo do počtu vrcholov a hrán. Samozrejme, aj v veľkých a "hustých" grafoch s veľkým počtom hrán sa nájdu prípady, keď toto urýchlenie nepomôže, ale ich počet je malý oproti prípadom, kedy pomôže. Môj hrubý odhad, ktorý nie je podložený výpočtami, je, že je to možné použiť v 50 - 75% prípadov.

4.2.3 Prítomnosť potrebných EP a neprítomnosť nadbytočných EP

Tieto dva prípady boli zlúčené do jedného, pretože spolu úzko súvisia. Zároveň je však otázka, ktoré podpisy v dokumente majú, a ktoré nemajú byť aktuálna

[6] Z $O(n)$ na $O(1)$, pričom na vstupe je dokument a n digitálnych podpisov.

[7] Aj keď pamäťové nároky v tomto prípade nebudú nejako kritické, niekoľko kilobajtov na certifikát

aj v prípade jedného podpisu. Schopnosť overovať prítomnosť potrebných, a ak to podpisová politika vyžaduje aj neprítomnosť nadbytočných podpisov na úrovni overovača, by veľmi prispela k automatizácii procesu overovania, pretože by nebolo nutné overovať tieto skutočnosti ručne. Fakt, kto má všetko byť podpísaný v dokumente, vyplýva veľmi často z povahy dokumentu. Napríklad, pri zmluve môže byť jasne určené, kto všetko túto zmluvu uzatvára, a v akej roli. Naopak, pri petícii toto nemá zmysel uvádzať, pretože počet podpisovateľov nie je vopred známy. Preto by bolo možné vyhodnocovať, kto má dokument podpísať, len ak by sa z formátu dokumentu dal tento fakt zistiť. Zároveň však nie je nutné rozširovať túto funkcionalitu na všetky typy dokumentov[8], pretože by to zbytočne limitovalo použitie ostatných formátov.
Veľmi dobrým príkladom, kde by bolo takéto rozšírenie overovača vítané, je elektronická podateľňa. Elektronická podateľňa musí spracovať veľké množstvo podpísaných dokumentov a nie je v ľudských silách kontrolovať, či podpisy v nich sú správne.
Ako spraviť formát tak, aby bolo z neho jasné, ktoré podpisy musia na dokumente byť, aby bol platný? Sú možné dva prístupy:

1. Ad-hoc prístup: organizácia, ktorá v elektronickej podateľni túto funkcionalitu potrebuje, si ju tam zakomponuje. Napr., ak potrebuje spracovávať formuláre založené na XML, tak ich upraví tak, aby z dát v týchto formulároch bolo jasné, kto ich má podpísať, a potom overí, či ich podpísali naozaj stanovené osoby.

2. Systematický prístup: stanoví sa enkapsulácia dokumentov tak, aby dodané dáta informovali overovač, kto má dokument podpísať.

Ad-hoc prístup

Tento prístup je z dlhodobého hľadiska zlý a nesystematický, ale z krátkodobého hľadiska môže byť vhodným riešením, pretože upravenie používaného formátu (napr. už spomínaných XML formulárov) a overovača môže byť otázkou niekoľkých človekodní až človekotýždňov a nie je potrebné čakať na dlhé štandardizačné a legislatívne procesy. Zároveň je možné vytvoriť si formát a overovač urobený na mieru a nie je potrebné prispôsobovať sa všetkým možným situáciám, na ktoré musí pamätať štandard.
Samozrejmou nevýhodou je nekompatibilita so zvyškom sveta a možné chyby v návrhnu formátu, ktoré sa môžu odraziť v zníženej bezpečnosti ale napr. aj nemožnosti ďalšieho rozširovania formátu.

Prístup cez medzinárodné štandardy

V tomto prípade je to presne naopak ako pri ad-hoc prístupe. Výhodou je pravdepodobne dobrý návrh, pretože sa predpokladá, že štandard budú robiť odborníci.

[8] Napr. človek, ktorý do roka uzavrie 10 zmlúv, nepotrebuje mať overovač s touto schopnosťou.

Štandard by mal byť akceptovaný, takže odpadajú problémy s nekompatibilitou. Problémom je časový horizont, do ktorého je štandard navrhnutý a schválený a až na jeho základe je možné vykonať implementáciu do overovacej aplikácie.

4.3 Podpisové politiky pre viacnásobné EP

Podpisová politika je súbor pravidiel, ktorý musí dodržať podpisovateľ pri vytváraní EP a overovateľ pri jeho overovaní. Ak vytvorenie aj overenie podpisu prebehlo v súlade s podpisovou politikou, potom ľubovoľní dvaja overovatelia musia dospieť pri overovaní platnosti podpisu k tomu istému výsledku. Podpisová politika by mala byť dostupná v ľudsky čitateľnej forme, ale môže byť dostupná aj vo forme spracovateľnej podpisovačom, resp. overovačom. Podpisové politiky pre jednoduché podpisy vo forme ASN.1 sú popísané v RFC 3125 "Electronic Signature Policies" [14].
Podpisové politiky pre viacnásobné EP musia okrem požiadaviek na samotný podpis uvažovať aj s požiadavkami na vzťahy medzi podpismi. Politikami pre viacnásobné podpisy sa zaoberá dokument ETSI TR 102 045 "Electronic Structures and Infrastructures (ESI); Signature policy for extended bussiness model" [15]. V ňom sa konštatuje, že nie je možné napísať politiku pre všetky možnosti, aké môžu nadobúdať vzťahy medzi podpismi. Toto je dané tým, že podpisov na dokumente môže byť neobmedzené množstvo a s množstvom podpisov rastie aj zložitosť grafu podpisov. Preto sa v tomto dokumente vypichujú len niektoré vybrané prípady viacnásobných podpisov, tj. nezávislé podpisy, sekvenčné podpisy (čo sú nezávislé podpisy so stanoveným poradím) a kontrasignatúry. Pre tieto druhy viacnásobných EP sa uvádzajú aj príklady[9] a podpisové politiky, ktoré by mohli byť vhodné pre tieto prípady.
Na základe týchto príkladov a odporúčaní je možné napísať podpisovú politiku pre viacnásobné EP v ľudsky čitateľnej forme, problémom je ale podpisová politika v XML alebo ASN.1. Vydavatelia podpisových politík môžu vydať podpisové politiky pre niektoré bežné situácie, ale nemá zmysel vydávať všeobecnú podpisovú politiku pre viacnásobné EP.
V podpisovej politike vo formáte ASN.1 pre "jednoduché" podpisy sú špecifikované nasledovné podmienky kladené na podpis:

- *signerAndVeriferRules*; špecifikuje podmienky kladené na formát EP, ktoré podpísané a nepodpísané atribúty musia byť prítomné v podpise. Ďalej sú v ňom špecifikované, či musí alebo nemusí byť v podpise certifikát podpisovateľa alebo len jeho referencia alebo musia byť prítomné (referencie) všetkých certifikátov v certifikačnej ceste.

- *signingCertTrustCondition*; špecifikuje okrem iného, ktoré self-signed certifikáty sú dôveryhodné, aká je maximálna dĺžka certifikačnej cesty a ktoré

[9] Je zaujímavé, že pre kontrasignatúru sa uvádza notársky podpis, i keď chápanie notára a jeho osvedčenia podpisu je trochu iné ako je to v 5. kapitole.

informácie o revokácii musia byť prítomné v podpise.

- *timeStampTrustCondition*; špecifikuje pomienky kladené na overenie certifikačnej cesty TSA.
- *attributeTrustCondition*; špecifikuje podmienky kladené na overenie certifikačnej cesty AA, ktorá vydala atribútový certifikát na niektorý z certifikovaných atribútov prítomných v podpise.
- *algorithmConstraintSet*; špecifikuje množinu podpisovacích a hašovacích algoritmov ako aj dĺžky kľúčov, ktoré je možné použiť na vytvorenie podpisu a podpísanie certifikátov, CRL, OCSP odpovedí, atribútových certifikátov a časových pečiatok.

Tieto podmienky sú nepovinné a sú zapísané v poli *SignatureValidationPolicy* podpisovej politiky rozdelené medzi polia *CommonRules* a *CommitmentRules*, pričom platí: ak je podmienka prítomná v CommonRules, nesmie byť v CommitmentRules a ak nie je prítomná v CommonRules, musí byť v CommitmentRules.
Podpisové politiky pre viacnásobné podpisy by mali zachovávať tieto podmienky, ale navyše by mali určovať podmienky pre vzťahy medzi podpismi a ktoré podpisy, resp. podpisy ktorých rolí, musia byť prítomné a podmienky kladené na podpis jednotlivých rolí. Rozšírenie podpisovej politiky na podpisovú politiku pre viacnásobné podpisy je možné opäť spraviť pomocou poľa *signPolExtensions*, ktoré je súčasťou poľa SignatureValidationPolicy. To má jednoduchú syntax:

```
SignPolExtensions ::= SEQUENCE OF SignPolExtn

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

4.3.1 Vzťahy medzi rolami

Bolo by vhodné, aby podpisová politika vedela definovať obmedzenie podpisov na ľubovoľnú grafovú štruktúru. Jednotlivé uzly ako podpisovatelia zrejme nemajú význam, zaujímavejšie sú podpisy z hľadiska rolí, ktoré títo používatelia majú. Celá štruktúra by mohla vyzerať nasledovne:

```
extnValue ::= SEQUENCE OF Relationship

Relationship ::= SEQUENCE {
                 multiplicity Multiplicity,
                 signerRole Role,
                 relatedElements SET OF Element }

Element ::= SEQUENCE OF {
```

```
            elementCount INTEGER,
            elementType ElementType }

ElementType ::= CHOICE OF {
                  any INTEGER,
                  document OBJECT IDENTIFIER,
                  roleID OBJECT IDENTIFIER }

Role ::= CHOICE OF {
         any INTEGER,
         roleID OBJECT IDENTIFIER }

Multiplicity ::= CHOICE OF {
                   standard OBJECT IDENTIFIER,
                   defined SEQUENCE OF INTEGER }
```

Čiže definovaný atribút zachytáva postupnosť vzťahov, ktoré musia byť splnené, t. j. prítomné v grafe podpisov. Vzťah, *Relationship*, je popísaný pomocou role podpisu, ktorý je urobený nad nejakými elementami grafu, ktoré môžu byť dokumentom, podpisom nejakej role alebo hocijakej role. Podpisovateľ musí mať rolu špecifikovanú v poli *signerRole* a táto je byť rola definovaná cez OID, alebo hocijaká rola. Každý vzťah sa môže vyskytovať viacero krát. To, koľko krát sa vzťah vyskytuje je naznačené poľom *multiplicy*. Štandardne existuje niekoľko výskytov, ako napr. vôbec (0-krát), práve raz, aspoň raz a ľubovoľne veľa (0 a viackrát). Tieto by mohli byť zadefinované cez OID-y v poli *stadnard*. Pre prípad, že by bolo potrebné použiť iné násobnosti, je určené pole *defined*, ktoré obsahuje neprázdnu postupnosť nezáporných čísel. Možnosti aspoň raz a ľubovoľne veľa by mohli byť identifikované pomocou záporných čísel, napr. -1 a -2.
Množina elementov, nad ktorými je podpis urobený, je uložená v poli *relatedElement* a platí, že sa v ňom nenachádzajú dva Elementy s rovnakým *ElementType*. Element je identifikovaný svojím typom, ktorý je zachytený v poli ElementType, a výskytom, ktorý je v poli *elementCount*. ElementType môže byť, ako už bolo spomínané, jednou z troch hodnôt:

1. *any*; podpis hocijakej role,
2. *roleID*; podpis role špecifikovanej OID-om,
3. *document*; podpis je spravený nad dokumentom, ktorý je špecifikovaný OID-om. Tento OID musí popisovať typ dokumentu, napr. splnomocnenie, pretože podpisová politika nemôže vedieť, čo bude v dokumentoch, ktoré budú podpisovatelia podpisovať.

4.3.2 Podpisy (rolí) prítomné na dokumente

Toto rozšírenie by malo popisovať, podpisy ktorých rolí a v akom množstve musia byť prítomné na dokumente. Nie je však potrebné, pretože tieto role sa dajú vytiahnuť z jednotlivých vzťahov predchádzajúceho atribútu.

4.3.3 Podmienky na podpisy rolí

Podpisová politika by mala ponúkať možnosť definovať podmienky ako pre "jednoduchý" podpis pre každú z rolí. Je potrebné spraviť mapovanie medzi rolou a "jednoduchou" podpisovou politikou. Toto je možné riešiť dvoma spôsobmi:

- Vytvoriť štandardné podpisové politiky pre vybrané role a na ne sa odvolávať, pričom referencia by mala obsahovať OID a haš politiky. Pre rozlíšenie som ju pomenoval externá politika (external policy).
- Enkapsulovať podpisovú politiku. Pre rozlíšenie som ju pomenoval interná politika (internal policy).

Riešenie, ktoré pokrýva obe možnosti, by potom mohlo vyzerať takto:

```
extnValue ::= SEQUENCE OF RoleToPolicyMapping

RoleToPolicyMapping ::= SEQUENCE {
                            role OBJECT IDENTIFIER,
                            policy PolicyMapping }

PolicyMapping ::= CHOICE OF {
                     externalPolicy ExternalPolicy,
                     internalPolicy SignaturePolicy }

ExternalPolicy ::= SEQUENCE {
                     idenifier OBJECT IDENTIFIER,
                     hash OCTET STRING }
```

4.4 Viacnásobné podpisy z hľadiska slovenskej legislatívy

V zákone 215/2002 sa explicitne nespomína pojem viacnásobného podpisu. Viacnásobné podpisovanie zaručeným EP však upravuje vyhláška NBÚ 542/2002 o spôsobe a postupe používania EP v obchodnom a administratívnom styku [13]. V prílohe 1 "používanie zaručeného EP v obchodnom styku" je doslova uvedený spôsob podpisovania elektronického dokumentu či už jediným alebo viacerými podpisovateľmi:

> mať extnValue s rovnakou syntaxou, ako majú pravidlá v rámci CommonRules

I. *Podpisovateľom je jednotlivá fyzická osoba v mieste vytvorenia elektronického dokumentu*
...

Pre tože táto časť zákona sa netýka témy tejto diplomovej práce, bola pre stručnosť vypustená a nebude ani komentovaná.

II. *Podpisovateľmi sú viaceré fyzické osoby v mieste vytvorenia elektronického dokumentu*
Na podpisovanie elektronického dokumentu v obchodnom styku možno použiť iba bezpečné zariadenie na vyhotovenie EP a bezpečné zariadenie na vyhotovenie časovej pečiatky. Na vlastné podpisovanie je ustanovený tento postup:

a) vytvorí sa charakteristický odtlačok ("hash") elektronického dokumentu podľa prijatej podpisovej schémy,

b) každá z fyzických osôb vytvorí z charakteristického odtlačku ("hash") elektronického dokumentu podľa písmena a) zaručený elektronický podpis na základe svojho súkromného kľúča, ku ktorému vlastní platný kvalifikovaný certifikát,

c) každý zaručený elektronický podpis vytvorený podľa písmena b) sa opatrí časovou pečiatkou,

d) všetky zaručené EP podľa písmena c) chápané ako postupnosť znakov sa zreťazia do súhrnnej postupnosti, z ktorej sa opäť vytvorí charakteristický odtlačok ("hash") podľa prijatej podpisovej schémy,

e) určia sa fyzické osoby reprezentujúce podpisujúce strany elektronického dokumentu,

f) ak je podpisujúcich strán viac ako dve, označí sa každá z nich identifikátorom, napríklad A, B, C,

g) každá z fyzických osôb reprezentujúca podpisujúcu stranu vytvorí na základe svojho súkromného kľúča, ku ktorému vlastní platný kvalifikovaný certifikát, zaručený EP z charakteristického odtlačku ("hash") súhrnnej postupnosti podľa písmena d) a tento pripojí alebo logicky spojí s podpisovaným elektronickým dokumentom,

h) elektronický dokument podpísaný podľa písmena g) zašle každá z fyzických osôb reprezentujúcich podpisujúcu stranu každej z fyzických osôb reprezentujúcej ostatné podpisové strany.

Táto časť sa zaoberá postupom vytvárania viacnásobného zaručeného EP, v prípade, že sú všetci podpisovatelia na jednom mieste. Pretože sa v procese podpisovania robia podpisy nad zreťazenými podpismi, tento proces neoumožňuje využitie klasických ETSI podpisov kvôli bodom d) až g). Z týchto bodov totiž vyplýva,

že jednotlivé podpisy sa medzi sebou preväzujú, avšak spôsob ich previazania *neumožňuje* použitie poľa (elementu pri XML EP) counter-signature (Contersignature pri XML EP), pretože kontrasignatúra zväzuje len dva podpisy a v tomto prípade môže ísť vo všeobecnosti o viac podpisov, nad ktorými by mala byť vytvorená kontrasignatúra. Navyše, aj keby boli podpisovatelia len dvaja nedala by sa kontrasignatúra použiť, pretože je vytváraná nad digitálnym podpisom v rámci SignerInfa (v CMS, resp. elementu Signature v XML) a nie nad celým podpisom t. j. celým SignerInfo aj s podpísanými a nepodpísanými atribútmi. Zabezpečujú tieto body nejakú vlastnosť, ktorá sa nedá zaručiť pomocou ETSI zaručených podpisom? Ak áno, je táto vlastnosť potrebná? Odpovede na tieto dve otázky sa pokúsim zodpovedať v nasledujúcom texte.

V prvom rade sa pokúsim zodpovedať otázku vlastností podpisu, ktorý sa vytvára v bodoch $d)$ až $g)$. Pretože všetci podpisovatelia sú na jednom mieste, všetci si vedia overiť totožnosť ostatných podpisovateľov a vedia, ktoré podpisy v dokumente majú a ktoré nemajú byť. Nech proces podpisovania dospel do bodu $c)$. Dokument je podpísaný všetkými podpisovateľmi, ktorých počet sa môže, ale nemusí zhodovať s počtom zúčastnených strán. Podpisy jednotlivých strán znamenajú, že každá zo strán potvrdzuje integritu "celých" podpisov. Keby jednotlivé strany použili namiestno podpisov nad zreťazenými podpismi kontrasignatúry pričlenené k jednotlivým podpisom, výsledok by nebol veľmi vzdialený výsledku dosiahnutému v bodoch $d)$ až $g)$. Jediným problémom by bolo, že pre informácie obsiahnuté v nepodpísaných atribútoch ako aj pre informácie pomocné pri vytváraní certifikačnej cesty, by nebola zaručená integrita. V prípade nepodpísaných atribútov je situácia jasná. Ak je informácia v atribúte taká dôležitá, nie je problém zaradiť ju do podpísaných atribútov. V prípade certifikátov a CRL je situácia problematickejšia, ale je potrebné si uvedomiť, že nikde v postupe sa neuvádza, že podpisovatelia musia pripojiť informácie potrebné na overenie certifikačnej cesty. Každopádne v prípade CRL a certifikátov CA sú tieto informácie v zmysle zákona 215/2002 dostupné online a ich dostupnosť musia zabezpečiť CA. Problém môže mať overovateľ, ktorý si ich musí zháňať sám, ale vie sa k nim stále dostať. Takže posledným nevyriešeným problémom zostávajú certifikáty podpisovateľov. Tieto si môžu jednotlivý podpisovatelia vymeniť, pretože sú sústredení na jednom mieste, prípadne sa dohodnúť na forme, akou si ich budú odovzdávať[10]. Ak by certifikáty podpisovateľov potreboval ďalší overovateľ, ktorý nie je jedným z podpisovateľov, môže si ich vypýtať od samotných podpisovateľov. Vo všeobecnosti je jeho problém, odkiaľ tieto certifikáty získa. Kontrasignatúra vyrieši aj logické spojenie s dokumentom.

Zaujímavý je posledný bod. Zdá sa, že nie je dôležitý, ale je potrebné si uvedomiť, že jednotliví podpisovatelia sa môžu na jednom mieste podpisovať na viacerých strojoch, prípadne, nemusia mať všetci k dispozícii média, aby si podpísaný dokument mohli zobrať so sebou. Z postupu nevýplýva, či podpisy zástupcov strán

[10] Ale je ich problém, ako sa k nim dostanú. Keď sú uložené v podpise, je to pre overovateľa pohodlné, ale pohodlie by nemalo brániť používaniu medzinárodných formátov a obmedzovať funkčnosť ZEPu.

sú voči sebe viacnásobné nezávislé podpisy, a tak je možné ho interpretovať aj tak, že po podpisovaní zástupcov vznikne toľko podpísaných dokumentov, koľko je zástupcov.
Zhrnutie: je možné nahradiť podpis nad zreťazenými podpismi pomocou kontrasignatúr.
Pri odpovedi na druhú otázku vzniká nasledovný problém: je možné, aby medzi podpismi vytvorenými v bodoch a) až c) neboli podpisy zástupcov jednotlivých strán?
Ak áno, svojimi podpismi vyjadrujú súhlas nad dokumentom a dobrovoľne ho podpisujú. Ak nie, potom podpisujú fakt, že jednotliví podpisovatelia podpísali (pred nimi, pretože sú na jednom mieste) daný dokument. Tento fakt je však už obsiahnutý v ZEP-e, pretože CA pri vydávaní kvalifikovaného certifikátu zabezpečí jednoznačnú identifikáciu osoby a pravdepodobnosť, že by niekto zistil ich súkromný kľúč z bezpečného zariadenia, je blízka nule. Takže nie je potrebné dodatočné potvrdenie, že daný dokument podpisovatelia naozaj podpísali (ak ho explicitne nevyžaduje zákon).
Ak nie je v zákone uvedené inak, nie je táto vlastnosť potrebná a postačuje, ak sú na dokumente nezávislé viacnásobné podpisy.

III. *Podpisovateľmi sú viaceré fyzické osoby*
V prípade, ak sa na podpisovanie elektronického dokumentu v obchodnom styku použijú dve bezpečnostné zariadenia alebo viac bezpečných zariadení na vyhotovenie a overenie elektronického podpisu a bezpečných zariadení na vyhotovenie časovej pečiatky, je postup takýto:

a) každá z podpisujúcich fyzických osôb sa označí identifikátorom, napríklad A, B, C, pričom fyzická osoba s identifikátorom A je podpisovateľom vlastniacim nepodpísaný elektronický dokument (ďalej len "primárny podpisovateľ"),

b) primárny podpisovateľ vyhotoví na základe vlastného súkromného kľúča, ku ktorému vlastní platný kvalifikovaný certifikát, zaručený EP elektronického dokumentu opatrený časovou pečiatkou,

c) vytvorený zaručený EP pripojí alebo logicky spojí s podpisovaným elektronickým dokumentom a zašle fyzickej osobe s identifikátorom B,

d) fyzická osoba s identifikátorom B overí integritu prijatého elektronického dokumentu a platnosť podpisu primárneho podpisovateľa a na dôkaz súhlasu podpíše elektronický dokument vlastným zaručeným EP, tento opatrí časovou pečiatkou a pripojí alebo logicky spojí s prijatým elektronickým dokumentom podpísaným primárnym podpisovateľom,

$e)$ elektronický dokument podpísaný podľa písmena $d)$ zašle fyzickej osobe s nasledujúcim identifikátorom, ktorá vykoná v svojom mene analogické úkony opísané v písmene $d)$,

$f)$ posledná z podpisujúcich fyzických osôb zašle podpísaný elektronický dokument doplnený o podpisy jednotlivých predchádzajúcich podpisujúcich osôb primárnemu podpisovateľovi,

$g)$ primárny podpisovateľ všetky zaručené EP pripojené k podpisovanému elektronickému dokumentu a chápané ako postupnosť znakov zreťazí do súhrnnej postupnosti, z ktorej opäť vytvorí charakteristický odtlačok ("hash") podľa prijatej podpisovej schémy, tento podpíše svojím zaručeným EP, opatrí časovou pečiatkou a pripojí alebo logicky spojí s podpisovaným elektronickým dokumentom,

$h)$ elektronický dokument s podpismi podľa písmena $g)$ zašle primárny podpisovateľ všetkým ostatným podpisujúcim fyzickým osobám.

Táto časť sa zaoberá postupom vytvárania zaručeného viacnásobného EP, v prípade že podpisovatelia nie sú na jednom mieste. Opäť kroky $g)$ a $h)$ zabraňujú použitiu ETSI podpisov. Dôvodom na toto konštatovanie je fakt, že primárny podpisovateľ preväzuje svoj podpis so všetkými podpismi, čo nie je umožnené kontrasignatúrami.

Je však primárny podpisovateľ potrebný? To, že primárny podpisovateľ podpíše všetky podpisy garantuje ich integritu pri rozposielaní podpísaného elektronického dokumentu všetkým podpisovateľom a nemožnosť pridať ďalšie podpisy do dokumentu. Ako už bolo spomínané v kapitole 3.1, táto vlastnosť nie je potrebná v prípade, že je počet podpisovateľov známy. Integritu tiež nie je potrebné kontrolovať týmto spôsobom, stačí spôsob, ktorým sa kontrolovala integrita v kroku $d)$[11]. Všetci podpisovatelia si v priebehu podpisovania overujú integritu podpísaného dokumentu, ktorý majú podpísať. Využitím podobných argumentov ako v II. časti je možné ešte viac poukázať na nepotrebnosť primárneho podpisovateľa a uzatváracieho podpisu spraveného nad všetkými podpismi.

Zhrnutie: touto prílohou sa teda upravuje používanie *zaručeného* EP a nie "obyčajného" EP, ktorý je tiež možné využiť v obchodnom styku, ako už bolo spomínané v kapitole 3. Problém s používaním zaručeného EP vyplýva z uvedeného postupu na podpisovanie či už v prípade, že sú všetci podpisovatelia prítomní v mieste vytvorenia elektronického dokumentu alebo nie. Pretože sa nedá použiť ETSI EP založený na CMS ani na XML, vznikol formát *ZEP (ZIP)*, ktorý v sebe obsahuje viacero "jednoduchých" ETSI podpisov a spĺňa vyhlášku 542/2002 [13]

[11] Nie je tam explicitne napísaný spôsob, akým sa zabezpečuje integrita prenášaného dokumentu, ale predpokladám, že keby bola integrita riešená podpísaním celého podpísaného dokumentu, tak by to bolo napísané.

4.4.1 Formát ZEP (ZIP)

Formát ZEP (ZIP) je popísaný v dokumente "Schválené formáty zaručených elektronických podpisov" [16]. Formát definuje adresárovú štruktúru, v ktorej sú uložené podpísané dokumenty, ich zaručené EP a informácie na overenie jednotlivých podpisov (certifikáty, CRL). ZEP (ZIP) teda nie je formátom elektronického podpisu, ale formátom na uloženie *viacnásobného podpisu*. Súbory sú uložené do prehľadnej štruktúry, pre každý typ údajov potrebných na overenie (certifikáty, informácie o revokácii, politiky) existuje (ale nemusí) adresár, v ktorom sú súbory uložené. Celá štruktúra je skomprimovaná pomocou algoritmu ZIP. Názov každého súboru je odvodený od času vloženia súboru do podpisu v tvare *Generalized Time* nasledovne:

<prefix>YYYYMMDDhhmmssZ.<koncovka>

Význam jednotlivých znakov v názve súboru je nasledovný: YYYY je štvorciferným vyjadrením roku, MM je dvojciferným vyadrením poradového čísla mesiaca a DD je dvojciferným vyjadrením poradového čísla dňa v roku. Nasledujúca šestica vyjadruje čas, kde hh sú hodiny, mm sú minúty a ss sú sekundy. Teda ak bol súbor vytvorený 23. januára 1983 o 6:20 a 34 sekúnd, potom bude jeho názov vyzerať nasledovne:

<prefix>19830123062034Z.<koncovka>

Zoznam prefixov je v tabuľke.

Prefix	Význam
D	Hlavný adresár, v ktorom sú uložené všetky súbory podpisu (nepovinné)
S	Externý podpis
F	Uzatvárací podpis pre viacnásobný podpis
A	Archívny podpis
P	Podpisová politika
C	Certifikát
CRL	Zoznam zrušených certifikátov
OCSP	Odpoveď OCSP servera

Koncovka je štandardná podľa typu súboru (der, p7m, p7s...). V uzatváracom a archívnom podpise sú v poli encapContentInfo informácie, na ktoré sa vzťahuje uzatvárací podpis.

Výhody a nevýhody formátu ZEP (ZIP)

Problémom formátu ZEP je, že nie je podchytený v žiadnom medzinárodnom štandarde (ako sú napr. formáty CAdES a XAdES), napriek tomu, že formáty jednotlivých čiaskových podpisov sú na nich založené. Preto nie je možné použiť

ho na podpisovanie dokumentov tam, kde je predpoklad, že overovateľ nebude postupovať v súlade so slovenskou legislatívou a nebude mať k dispozícii overovač, podporujúci formát ZEP. Táto skutočnosť obmedzuje jeho používanie a je málo pravdepodobné, že sa ho podarí presadiť medzinárodne.
Výhodou formátu ZEP je:

- Enkapsulácia podpisov a údajov potrebných na overenie do jedného formátu, takže ich overovateľ nemusí hľadať na celom pevnom disku, resp. internete, i keď certifikáty a CRL je možné pridávať aj do ETSI podpisov, takže prínos nie je až taký výrazný. Výhodou je odstránenie potenciálne redundantných certifikátov CA a CRL, ak budú mať certifikačné cesty spoločné časti.
- Miešanie externých ETSI podpisov založených na XML a CMS do jediného formátu viacnásobného podpisu.

Kompatibilitu so slovenskou legislatívou nepovažujem za výhodu. Slovenskú legislatívu v tomto smere nepovažujem za najvhodnejšie riešenú, pretože vynucuje použitie formátu, ktorý nemôže byť implementovaný pomocou ETSI podpisov.

4.4.2 Alternatívy k ZEP (ZIP)-u

Pretože hlavnou nevýhodou formátu ZEP (ZIP) je nesúlad s medzinárodnými štandardami, bolo by vhodné nájsť nejaký akceptovateľný formát, ktorý by mohol zastúpiť hlavné výhody ZEP (ZIP)u, t. j. enkapsuláciu všetkých dát do jedného formátu a možnosť kombinácie CMS a XML podpisov. Je teda potrebné nájsť formát, ktorý vie kombinovať textové a binárne dáta a spĺňa podmienku medzinárodného štandardu. Možností je viacero:

1. XML,
2. MIME správy.

XML

XML môže byť alternatívou k ZEP (ZIP) formátu, pretože v rámci XML dokumentu, čo sú vlastne textové dáta, môžu byť dáta v niektorých elementoch binárne. Problémom je, že žiadny z medzinárodných štandardov sa touto cestou nevybral, preto by bola situácia len o trochu lepšia ako v prípade ZEP (ZIP)-u. Výhodou je, že XML súbor je možné prezerať v prípade núdze aj ručne, pretože sú to textové dáta. V prípade ZEP súboru ručné prezeranie nie je možné, pokiaľ overovateľ nevie, že ide o nejakú adresárovú štruktúru skomprimovanú algoritmom ZIP, pretože je to binárny súbor[12]. Nevýhodou riešenia založeného na XML bude pravdepodobne väčšia veľkosť, ktorá je spôsobená použitím textového formátu a nepoužitím kompresie nad celým súborom. Na zmiernenie tejto nevýhody je možné špecifikovať kompresiu vnútri súboru, t. j. napr. opakovať samotné podpisy alebo certifikáty.

[12] Samozrejme, dá sa na to prísť z hlavičky súboru, ale je veľmi málo ľudí, ktorí vedia vytiahnuť typ súboru z binárnej hlavičky.

MIME správy

Syntax MIME správ je uvedená v RFC 2045 až 2049. Z RFC 2046 [17] vyplýva, že MIME správy môžu v sebe obsahovať textovú aj binárnu informáciu a môžu byť zložené z viacerých častí. Každá z týchto častí môže obsahovať iný druh informácie, napr. jedna časť môže obsahovať mailovú správu a druhá časť obrázok k nej priložný. To, o akú informáciu ide je popísané v poli *Content-type* v hlavičke MIME správy alebo jej časti. Pre textovú informáciu je vyhradený typ *text* a pre binárnu informáciu neznámeho obsahu, v tomto prípade elektronický podpis a dáta potrebné na jeho overenie, typ *application* a podtyp *octet-stream*[13], skrátene application/octet-stream.
Jednotlivé EP a dáta potrebné na overenie je možné rozhodiť do viacerých častí MIME správy tak, aby každému certfikátu, CRL a podpisu pripadala jedna časť. Toto riešenie je výhodné, pretože podľa RFC 2046 [17] ak aplikácia nevie pracovať s dátami uloženými v časti správy, ktoré sú typu application/octet-stream, je štandardnou akciou uloženie do súboru. Takže v prípade, že MIME správa obsahuje podpisy, CRL a certifikáty a overovateľova aplikácia na zobrazovanie MIME správ ju nevie spracovať, tak všetky podpisy, certifikáty a CRL uloží každý do iného súboru. Samozrejme, pretože nevie, o aké dáta ide, nevie im priradiť koncovku súboru, čiže pre certifikáty a CRL, napr. *cer* alebo *der*, ale to už bude vedieť priradiť overovateľ, ktorý by už mal vedieť, že ide o enkapsuláciu viacnásobných EP spolu s dátami potrebnými na ich overenie, prípadne si to vie odkonzultovať s podpisovateľom.
Celá MIME správa by mala mať v hlavičke uvedené Content-type: *multipart/mixed*, aby bolo jasné, že ide o správu, pozostávajúcu z viacerých častí. Podtyp mixed musí byť uvedený, aby MIME správa spĺňala štadard[14]. Ďalej je nutné v poli *boundary* nastaviť reťazec, ktorý bude oddeľovať jednotlivé časti, ako aj ukončovať správu. Jednotlivé časti by mali content-type nastavený na application/octet-stream. Tento content-type je nutné nastaviť tento content-type aj pri XML podpisoch, čo sú vlastne textové súbory. Dôvod je ten, že pri nastavaní content-type na text sa táto časť MIME správy štandardne zobrazí a neuloží do súboru a navyše je nutné špecifikovať pre ňu ďalšie, z hľadiska overovania nepodstatné informácie, ako je napr. znaková sada.
Pretože MIME správy vedia enkapsulovať viacero rôznych typov dát, vedia miešať binárny a textový vstup a sú zadefinované v medzinárodných štandardoch, sú vhodným kandidátom alternatívneho formátu k formátu ZEP (ZIP).

[13] Podtypy pre binárne informácie sú len dva, druhý je postscript, ktorý sa v prípade enkapsulácie EP do MIME správ použiť nedá.

[14] Podľa RFC 2046 [17] musia byť všetky správy, pozostávajúce z viacerých častí, ktoré implementácia nepozná posudzované ako multipart/mixed. Navyše podtyp mixed je "určený pre správy, pozostávajúce z viacerých častí, kde časti sú nezávislé a musia byť zabalené dokopy v stanovenom poradí" [17]. V tomto prípade však poradie jednotlivých podpisov, CRL a certifikátov nie je relevantné.

Kapitola 5

Notárske osvedčenie ako EP

V kapitole 3 bolo spomínané, že notárske osvedčenie podpisu (ďalej len osvedčenie) môže byť považované za kontrasignatúru. Aby bolo možné určiť, či toto tvrdenie je pravdivé, je nutné zodpovedať otázku, čo je to vlastne osvedčenie. Zákon 323/1992 "O notároch a notárskej činnosti" [18] definuje pre tento účel pojem *legalizácie* v §58:

§ 58
Legalizácia

(1) Legalizáciou notár alebo ním poverený zamestnanec osvedčuje, že osoba, ktorej podpis má byť osvedčený v jeho prítomnosti, listinu vlastnoručne podpísal alebo podpis na listine uznala pred ním za vlastný. Pre zistenie totožnosti tejto osoby platí § 48.

(2) Osvedčenie pravosti podpisu sa vyznačí na listine vo forme osvedčovacej doložky, ktorá obsahuje

   a) poradové číslo knihy osvedčenia pravosti podpisov (osvedčovacia kniha),
   b) meno, priezvisko, trvalé bydlisko, prípadne miesto pobytu účastníka, jeho rodné číslo, dátum narodenia,
   c) údaj, ako bola zistená totožnosť účastníka,
   d) konštatovanie, že účastník listinu vlastnoručne podpísal pred notárom alebo že uznal podpis na listine za svoj vlastný,
   e) miesto a deň osvedčenia,
   f) podpis notára a odtlačok úradnej pečiatky.

(3) Ak sa skladá listina, na ktorej sa podpis osvedčuje, z niekoľkých listov (hárkov), alebo ak má prílohy, postupuje sa podľa § 44 ods. 3.

(4) Notár legalizáciou neosvedčuje pravdivosť skutočností uvádzaných v listine.

(5) Osvedčenie pravosti podpisu na listinách so vzťahom k cudzine vykonáva výlučne notár.

Z hľadiska technického je možné zadefinovať na základe zákona osvedčenie ako:

Definícia 6 *Notárske osvedčenie podpisu sú dáta pripojené k podpisu osoby podpísané notárom, ktoré dosvedčujú, že daná osoba dokument podpísala pred daným notárom v danom čase alebo uznala podpis za svoj.*

Pretože kontrasignatúra v CMS je SignerInfo, ktoré obsahuje v sebe aj podpísané atribúty (t. j. dáta), *je možné* ju použiť ako osvedčenie. V tejto kapitole sa budem snažiť vyriešiť nasledovné problémy:

1. Je potrebné elektronické osvedčenie alebo jeho vlastnosti pokryté niektorým z elektronických podpisov? Argumenty pre a proti.
2. Je potrebné celý proces legalizácie kopírovať do elektronickej formy, alebo je možné niektoré kroky vynechať?
3. Ak je osvedčenie potrebné, ako má vyzerať kontrasignatúra?
4. Kto by mal vydávať notárom certifikáty? Aké certifikáty by to mali byť a čo všetko by mali obsahovať?

Po vyriešení týchto problémov je možné navrhnúť osvedčenie. Môj návrh sa bude snažiť minimalizovať množstvo potrebných úprav v sloveskej legislatíve, a zároveň nezaostávať po technickej stránke.

5.1 Požiadavky kladené na osvedčenie

V tejto časti bude popísaný proces legalizácie, pričom budú identifikované požiadavky kladené na proces legalizácie v elektronickej forme.

5.1.1 Použitie osvedčenia

V prvom rade je potrebné zamyslieť sa nad tým, čo vlastne osvedčenie zaručuje a kde všade sa používa. Výsledky je možné zhrnúť do nasledovných bodov.

- Používa sa všade, kde jeho použitie stanovuje zákon, ale nič nebráni osvedčovať podpisy na listinách aj v prípadoch, keď to zákon nezakazuje.
- Notár identifikuje osobu, ktorá si chce nechať vydať osvedčenie na základe preukazu totožnosti.
- V prípade osvedčenia notár svojím osvedčením garantuje, že daná osoba podpísala pre ním dokument daného dňa alebo uznala podpis na listine za svoj.

- Notár si vedie nezávislú evidenciu o osvedčeniach, ktoré vydáva. Navyše evidenciu si vedie aj Notárska komora SR (NKSR).

Pri pokuse nahradiť jednotlivé využitia a záruky existujúcimi prostriedkami v rámci PKI, je možné zistiť, že niektoré sa nahradiť dajú a iné nie. V prvom prípade je možné zmeniť zákon tak, aby boli osvedčenému podpisu ekvivalentné ZEP. Proces lagalizácie však má viac výstupov ako len samotné osvedčenie aj keď je práve toto pre podpisovateľa najdôležitejšie, a práve tieto výstupy sú "pridanou hodnotou", ktorú nie je možné získať len z procesu vyhotovenia ZEP-u aj s časovou pečiatkou. Problémom je aj samotná legislatíva, ktorá stanovuje, že ZEP je ekvivalentný vlastnoručnému podpisu.[1]

Druhým problémom je jednoznačná identifikácia na základe preukazu totožnosti. Žiadateľ o cerifikát sa musí pred jeho vydaním identifikovať u registračnej autority, ktorá jeho totožnosť overí a na základe toho mu môže CA vydať certifikát[2]. Osoba, ktorá si nechá vydať certifikát však nie je povinná uviesť v ňom také údaje, aby bolo možné osobu identifikovať pomocou certifikátu. Dokonca nemusí uviesť ani svoje meno a môže si nechať vydať certifikát na pseudonym. Údaje o držiteľovi certifikátu však CA má a v prípade vyšetrovania ich môže poskytnúť rovnako ako notár. Možnosť, že držiteľ certifikátu mohol sfalšovať svoju identitu, napr. falošným občianskym preukazom, je rovnako pravdepodobná u CA ako u notára.

V prípade tretieho bodu vychádzajú garancie zo samotnej technológie ZEP-u. Dátum a čas je možné garantovať pomocou časových pečiatok. To, že osoba podpísala dokument, je zaručené tým, že privátny kľúč podpisovateľa je uložený na bezpečnom zariadení, z ktorého je nemožné dostať ho (aspoň nie skôr než dá podpisovateľ svoj certifikát revokovať) a len podpisovateľ pozná k nemu PIN, takže nikto iný ho nevie používať. To, že podpisovateľ môže "požičať" svoju čipovú kartu a prezradiť PIN niekomu inému je možné, avšak v tomto prípade je CA povinná hneď ako na to príde, zrušiť mu certifikát. Navyše možnosť, že niekto podobným spôsobom vymámi z podpisovateľa privátny kľúč a PIN, je rovnaká, ako že nahovorí podpisovateľa, aby dokument podpísal sám (v elektronickom aj papierovom svete). V tomto prípade by mohol notár pôsobiť ako prekážka ľudskej naivity a nezodpovednosti.

V prípade nezávislej evidencie by bolo potrebné rozšíriť informácie, ktoré si PKI uchováva, aby bolo možné zaznamenať všetky informácie, ktoré v prípade osvedčovania uchováva notár a NKSR. Autorita časových pečiatok si vedie evidenciu o časových pečiatkach, ktoré vydáva a CA si vedie evidenciu o držiteľoch certifikátov. Informácie z týchto dvoch zdrojov však nepokrývajú všetky informácie, ktoré sa uchovávajú v procese legalizácie. To, či sú všetky tieto informácie potrebné, sa pokúsim zanalyzovať pri popisovaní procesu legalizácie v kapitole 5.1.3.

[1] Samozrejme, legislatíva sa dá zmeniť.

[2] Úlohy registračnej autority plní podľa slovenskej legislatívy CA. Samostatná RA ako organizácia na Slovensku neexistuje.

5.1.2 Potreba elektronického osvedčenia

Argumenty proti elektronickému osvedčeniu:

- Osvedčenie zaručuje, že podpis naozaj vykonala osoba uvedená v osvedčení. Pri spojení privátneho kľúča s podpisovateľom však pravdepodobnosť, že nastane opak, je veľmi malá. V prípade, že ide o ZEP, je pravdepodobnosť, že niekto skonštruuje platný podpis podpisovateľa priamo úmerná podpisovateľovej ochote vydať bezpečné zariadenie a prezradiť k nemu PIN. V prípade sporu je možné dopátrať sa k identite podpisovateľa podľa záznamov CA.
- Nezávislá evidencia notára sa dá sčasti nahradiť evidenciami CA a TSA. Niektoré údaje nie sú pri elektronickom osvedčovaní potrebné, niektoré sú. Viac informácii je v kapitole 5.2.2.

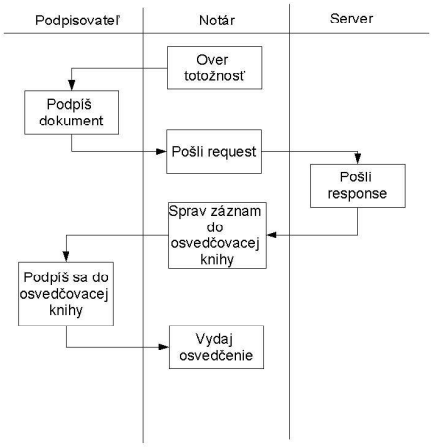
Argumenty pre elektronické osvedčenie:

- Osvedčenia sú vyžadované v niektorých prípadoch zákonom.
- Osvedčenie zvyšuje dôveryhodnosť podpisu, pretože obsahuje údaje, ktoré jednoznačne identifikujú podpisovateľa.
- Osvedčenie zaručuje na rozdiel od "obyčajného" podpisu ďalšie vlastnosti, preto je logické, že budú na zariadenia a algoritmy použité pri konštrukcii osvedčenia a notárskeho podpisu vôbec kladené zvýšené nároky. V takomto prípade urobením osvedčenia nad podpisom je možné ochrániť podpis pred rozbitím podpisovacieho algoritmu. Viac sa tejto problematike venuje kapitola 5.5.2.

Záver: ZEP garantuje, že podpis vykonal podpisovateľ za predpokladu, že nedal k dispozícii svoj privátny kľúč niekomu inému. Overovateľ však vo všeobecnosti nemá ako overiť identitu podpisovateľa, pretože certifikát nemusí obsahovať údaj, ktorý osobu jednoznačne identifikuje (napr. rodné číslo) a k týmto informáciám sa môže dostať len cez CA. CA mu však tieto údaje neposkytne pretože by to bolo v rozpore so zákonom 428/2002 "O ochrane osobných údajov". Riešením by bolo dotazovať sa na CA s otázkou, či je predpokladaná identita držiteľa certifikátu správna alebo nie. Tak isto nie je možné pokryť všetky údaje v osvedčení pomocou údajov z dvôveryhodných tretích strán. Preto *je elektronické notárske osvedčenie potrebné*.

5.1.3 Proces legalizácie

Proces legalizácie je postavený tak, aby sa vylúčilo zneužitie podpisu na dokumente v čo najväčšej miere. Proces nepredpokladá, že notár je dôveryhodná tretia strana a poskytuje isté záruky aj voči nečestnému notárovi. Celý proces je znázornený na obrázku 5.1. Všetky kroky procesu si zaslúžia pozornosť, pretože majú dopad na požiadavky kladené na elektronickú formu osvedčenia, preto uvádzam stručný popis:



Obrázok 5.1: Proces legalizácie

1. *Overenie totožnosti*
   Notár overí totožnosť podpisovateľa podľa preukazu totožnosti (občiansky preukaz, pas...). Tento krok sa vykonáva preto, že údaje o podpisovateľovi sú súčasťou evidencie a aj samotného osvedčenia. Notár musí zistiť totožnosť podpisovateľa jednoznačne. Dáta, ktoré od podpisovateľa potrebuje, sú vymenované ďalej.

2. *Podpis dokumentu podpisovateľom*
   Tento krok je pri legalizácii vynechaný, pretože podpis už v čase osvedčovania existuje. Notár sa presvedčí, že podpisovateľ vykonal akt podpisovania pred ním.

3. *Poslanie požiadavky na server NKSR*
   Notár musí poslať požiadavku o pridelenie poradového čísla osvedčenia na NKSR. Serveru sa posielajú nasledovné údaje:

   - Číslo preukazu totožnosti,
   - Rok osvedčenia,
   - Identifikačné číslo notára,
   - Typ listiny (zmluva, závet...), táto položka je len informatívna.
   - Priezvisko, rodné priezvisko, meno a tituly pred a za menom podpisovateľa,
   - Rodné číslo,
   - Dátum požiadania o osvedčenie,

- Dátum označenia osvedčenia za chybné - v prípade, že sa notár pomýli, je osvedčenie označené za chybné, ale ostáva zaznamenané v databáze.

Všetky tieto údaje sa uložia aj do lokálnej databázy osvedčení na notárskom úrade. Tento krok sa vykonáva kvôli tomu, aby bola zaistená dvojitá evidencia o osvedčeniach, a to na NKSR a u notára.

4. *Poslanie odpovede zo servera NKSR*
   Server požiadavku príjme, priradí jej jednoznačné poradové číslo (identifikátor) a uloží do databázy. Poradové čísla sú rastúce. Server vráti notárovi poradové číslo osvedčenia.

5. *Vytvorenie záznamu v osvedčovacej knihe*
   Notár vytvorí záznam v osvedčovacej knihe, ktorý obsahuje:

   - Dátum,
   - Poradové číslo osvedčenia,
   - Kolónku na podpis podpisovateľa, ktorému sa osvedčuje (legalizuje) podpis.

   Osvedčovacia kniha je ďalšou "papierovou" evidenciou o vydaných osvedčeniach, takže aj v prípade katastrofálneho výpadku, kedy by zlyhali server NKSR aj databáza u notára, je možné zistiť, či notár dané osvedčenie naozaj vydal.

6. *Podpísanie záznamu v osvedčovacej knihe*
   Podpisovateľ, ktorý si nechal legalizovať podpis, sa podpíše do osvedčovacej knihy. Toto sa robí za účelom:

   - Notár si vie porovnať podpisy na dokumente a v osvedčovacej knihe. Aj v prípade, že podpisovateľ sfalšuje svoju identitu, ťažko sfalšuje pri legalizácii podpis cudzej osoby v osvedčovacej knihe. V prípade sporu môže oba podpisy preskúmať súdny znalec, ktorý určí ich pravosť.
   - V prípade, že sa spolčí notár s podpisovateľom, nemali by byť schopní vytvoriť falošný podpis v osvedčovacej knihe. V prípade pochybností opäť rozhoduje súdny znalec.
   - Podpisovateľ nemôže poprieť, že požiadal o osvedčenie alebo legalizáciu, pretože nemôže poprieť svoj podpis.

   Toto sú ďalšie z významov osvedčovacej knihy. Zámerne neboli spomínané v predchádzajúcom kroku, pretože sú umožnené až podpisom podpisovateľa.

7. *Vydanie osvedčenia notárom*
   Notár vydá a nasledovejne podpíše osvedčenie, ktoré obsahuje:

   - Meno a priezvisko podpisovateľa,

- Rodné číslo podpisovateľa,
- Adresu podpisovateľa,
- Poradové číslo osvedčenia pridelené serverom NKSR,
- Dátum a obec, v ktorej bolo osvedčenie vykonané,
- Titul, meno a priezvisko notára a ak vydal osvedčenie poverený pracovník tak aj jeho titul, meno a priezvisko,
- Podpis notára alebo povereného pracovníka,
- Okrúhlu pečiatku notárskeho úradu.

Tento krok je pre podpisovateľa najdôležitejší a je cieľom celého procesu.

V priebehu tohoto procesu by mal byť ešte jeden krok, ktorý je pre notárov veľmi dôležitý, a to je inkasovanie platby za vydanie osvedčenia. Táto problematika sa však už viac dotýka témy "elektronického notára" a príliš odbieha od témy tejto diplomovej práce. Pokúsim sa teraz previesť proces legalizácie do elektronickej formy.

5.1.4 Zhrnutie požiadaviek

Na osvedčenie v elektronickej forme sa budú klásť nasledovné požiadavky:

- Zabezpečenie rovnakých vlastností ako má jeho "papierová" forma,
- Bezpečnosť na úrovni ZEP-u, alebo vyššia,
- Čo najväčší súlad so slovenskou legislatívou,
- Minimalizácia zmien existujúceho stavu pri realizácii elektronického osvedčovania.

Požiadavky sú kladené tak, aby elektronické osvedčenie malo rovnaké vlastnosti ako papierové osvedčenie. Prechod na elektronické osvedčenie a jeho používanie vyžadovalo čo najmenej úsilia a, samozrejme, prínos by mal byť čo najväčší. To znamená, že je možné minimalizovať úpravy v zákonoch a vyhláškach a používať čo najviac komponentov z "papierového" procesu legalizácie.

5.2 Analýza požiadaviek

Pri analýze požiadaviek sa sústredím na analýzu procesu legalizácie a údajov, ktoré vstupujú do tohto procesu. V oboch prípadoch sa sústredím na to, ktoré kroky (údaje) sú potrebné a ktoré treba vynechať.

5.2.1 Analýza procesu legalizácie

Overovanie totožnosti

Overovanie totožnosti je veľmi dôležité kvôli pravdivému vyplneniu údajov na osvedčení a zasielaných na server NKSR. Sú dve možnosti, ako môže notár pristupovať k overovaniu totožnosti:

- Overí totožnosť podpisovateľa alebo použije overenie totožnosti spravené iným notárom uložené v cetrálnej databáze.
- Spoľahne sa na overenie totožnosti CA a použije údaje z certifikátu[3].

Prvý prípad je identický s "papierovou" formou procesu legalizácie. Najväčšou nevýhodou, ktorú by mal proces vydania elektronického notárskeho osvedčenia odbúrať, je nutnosť dostaviť sa kvôli identifikácii do notárskeho úradu. Žiaľ, tento prípad inú možnosť neponúka, pretože podpisovateľ, ktorý si chce nechať overiť podpis musí aspoň raz za obdobie platnosti certifikátu prísť na notársky úrad. Ale je to pravdepodobne najperspektívnejšou cestou, pretože notárska obec by veľmi ťažko akceptovala fakt, že musí osvedčiť podpis niekoho, koho nemá možnosť fyzicky overiť[4] a prevziať údaje o totožnosti od niekoho iného. Zaujímavou otázkou je, či by bolo možné zriadiť pre notárov databázu, v ktorej by boli zaregistrované osoby a ich certifikáty, alebo by mal takúto databázu každý notár. V prvom prípade vidím problémy minimálne s ochranou osobných údajov[5]. Bolo by to však výhodné pre klientov, pretože po jedinom zaregistrovaní by boli schopní požívať výhody elektronického osvedčovania na každom notárskom úrade bez nutnosti fyzickej návštevy až do vypršania platnosti prípadne revokácie certifikátu. Otázkou by bolo, kto by mal prevádzkovať centrálnu databázu. Ako kandidát sa natíska NKSR z viacerých dôvodov. Je už notármi akceptovaná v procese legalizácie. Z procesu legalizácie vyplýva, že už má skúsenosti s prevádzkou podobných databáz, navyše NKSR prevádzkuje aj služby ako sú register listín a pod. Alternatívou by bolo vytvorenie samostantného úradu, ale ten by nemal spomenuté výhody NKSR. Komerčnú prevádzku databázy neberiem do úvahy ako možnosť, pretože:

- Fungovala by na Slovensku len jedna? Ak áno, mala by monopol na poskytovanie služby, čo nie je vhodné.

[3]Teoreticky by mohol použiť údaje z registračnej autority. Ale ako bolo uvedené vyššie, registračná autorita je podľa slovenskej legislatívy súčasťou CA. Iné, dôveryhodné tretie strany, pôsobiace v rámci PKI nemôže notár zobrať ako zdroj informácii o podpisovateľoch, pretože tieto neidentifikujú podpisovateľa.

[4]Podľa vyjadrenia jedného zo zamestnancov NKSR notári nedvôverujú nikomu, preto by bolo veľmi ťažké žiadať, aby akceptovali identitu človeka, ktorú by im zaručili napr. informácie poskytnuté CA.

[5]Súčasťou osvedčenia sú aj rodné čísla, takže tieto by museli byť aj v databáze, pretože nie sú povinnou súčasťou certifikátu.

- Ak by fungovali viaceré, je otázne, ako by bola zabezpečená kompatibilita medzi nimi. Navyše, čo v prípade, že by notár nedôveroval niektorému z prevádzkovateľov[6]?

- Komerčná prevádzka by niečo stála. Od koho by vyberala poplatky? Od notárov, ich klientov alebo oboch?

Problémom centrálnej databázy identít je aktuálnosť. Údaje ako rodné číslo sa, samozrejme, nemenia, avšak napr. adresa sa zmeniť môže. To, aby boli údaje aktuálne, by malo byť na podpisovateľovi, pretože je v jeho záujme, aby údaje na osvedčení, ktoré mu bolo vydané, boli správne. Toto platí najmä v prípade elektronického notára ako servera, pretože server nemá ako fyzicky skontrolovať identitu podpisovateľa.
Druhá možnosť by zase bola výhodná pre notárov, pretože by im umožňovala uchovať si klientelu[7], ale zabrzdila by rozvoj elektronického notárstva.
V prípade, že sa notár spoľahne na identifikáciu podpisovateľa certifikačnou autoritou, je najväčším problémom, že certifikát neidentifikuje osobu jednoznačne. Toto riešenie však prináša výhodu v tom, že nie je potrebné budovať paralelnú databázu identít popri existujúcej, ktorú už spravujú CA.
Certifikát nemusí obsahovať všetky údaje, ktoré sa posielajú na server NKSR a sú uvedené na osvedčení. Otázkou je, či sú všetky tieto údaje pre vydanie osvedčenia nevyhnutné a nedá sa rovnaký cieľ dosiahnuť aj s ich vlastnou podmnožinou. Údaje na osvedčení slúžia na to, aby bolo osvedčenie jednoznačne spojené s podpisovateľom a notárom, časom a miestom vydania a jednoznačnej identifikácii osvedčenia. Sú všetky tieto dáta potrebné? Tieto otázky budú zodpovedané v kapitole 5.2.2. Ukazuje sa, že pre vydanie osvedčenia je potrebné získať viac údajov, ako je povinné uviesť v certifikáte, takže certifikát nie je možné považovať za prostriedok, dokazujúci identitu. Tento problém sa dá riešiť nasledovne:

1. Vydávať certifikáty, ktoré jednoznačne identifikujú osobu a obsahujú všetky údaje, ktoré sú nevyhnutné v procese legalizácie,

2. Pri overovaní totožnosti notári spolupracujú s CA.

V prvom prípade je najväčšiou prekážkou neochota zverejňovať osobné údaje v certifikáte zo strany držiteľov certifikátov. V prípade použitia tejto možnosti by však systém fungoval priamočiaro.
Iná vec by bola, keby CA spolupracovali s NKSR a poskytovali im identitu ľudí, ktorým vydali certifikáty, čo je však v priamom rozpore so zákonom o ochrane osobných údajov. Riešením by bolo vytvorenie interface zo strany CA, ktorý by zadaním certifikátu a údajov identifikujúcich osobu, ktoré vstupujú do procesu legalizácie, vrátil, či certifikát patrí osobe alebo nie. Toto riešenie nie je v rozporom

[6] Toto je otázne aj pri jedinej komerčnej databáze.

[7] Klient, ktorý by chcel osvedčovať svoj podpis u iného notára, by sa musel opäť fyzicky dostaviť na notársky úrad.

so zákonom o ochrane osobných údajov, pretože neposkytuje osobné údaje, len overuje ich platnosť. Tieto osobné údaje však notár musí odniekiaľ získať a jedinou možnosťou je uviesť ich v požiadavke o osvedčenie. Posledným bodom zostáva, ako donútiť CA, aby spolupracovali s NKSR, pretože zatiaľ ich k tomu nič nenúti. V tomto prípade by bola najvhodnejšia zmena zákona.
Záver: Za najjednoduchšiu cestu overovania identity považujem centrálnu databázu identít NKSR, ktoré sú spojené s certifikátmi, ale nalepším riešením z technologického pohľadu by bolo využiť existujúcu databázu identít, ktorú spravujú CA. Podpisovateľovi sa potom stačí preukázať certifikátom a svojimi identifikačnými údajmi a notár si overí jeho identitu z databázy identít CA.

Podpis dokumentu podpisovateľom

Tento krok sa samozrejme vynecháva v prípade uznania podpisu za svoj, pretože v tomto prípade už podpis existuje. Tento krok nie je zaujímavý v prípade, že podpisovateľ podpisuje dokument u seba, a potom ho zašle notárovi (t. j. v prípade legalizácie). V prípade, že podpisovateľ príde k notárovi a podpisuje dokument u neho, je situácia iná, pretože:

- Podpisovateľ nemusí mať so sebou počítač.
- Notár nemusí akceptovať fakt, že by sa mal podpisovať na podpisovateľovom počítači a naopak podpisovateľ nemusí akceptovať, že by sa mal podpisovať na notárovom počítači.
- Podpisovateľ nemusí chcieť, aby dokument opustil jeho počítač, najmä ak sa jedná o citlivý dokument.

V prvom rade: notár by *nikdy* nemal podpisovať *nič* na inom počítači ako na svojom. Ostatné počítače sa považujú za nedôveryhodné. Notár by tak isto nemal poskytovať svoj počítač na podpisovanie niekomu inému, pretože existuje riziko zavlečenia škodlivého kódu do počítača určeného na podpisovanie[8]. Zároveň sú s podpisovaním u notára na počítači určenom notárom, spojené problémy s tým, že podpisovacia aplikácia by mala vedieť pracovať so všetkými typmi bezpečných zariadení, ktoré sú schválené NBÚ, pretože podpisovateľ môže vlastniť akékoľvek bezpečné zariadenie.
Záver: podpis dokumentu podpisovateľom v elektronickej forme je situácia, ktorá nastať môže, aj keď je pravdepodobnejšie, že väčšina ľudí využije skôr uznanie podpisu za svoj bez nutnosti chodiť na notársky úrad. Pretože pre notára by bolo príliš nákladné a potenciálne nebezpečné umožňiť podpisovať sa na notárovom počítači, je najvhodnejšia možnosť, aby sa každý (notár aj podpisovateľ) podpisoval na svojom počítači. Faktom, že citlivý dokument sa dá v tomto prípade odcudziť

[8]Toto riziko však existuje aj pri obyčajnom podpisovaní. Nie je vylúčené, že podpisovacia aplikácia pri podpisovaní podvrhnutého obsahu nespraví pretečenie buffera, a tým nespustí škodlivý kód.

notárom, a to dokonca aj v prípade, že notár by podpisoval údaje na médiu, akým je napr. USB kľúč alebo CD. V rámci PKI sú "dôveryhodné tretie strany" brané idealisticky a takto by sa malo pozerať aj na rolu notára. Aj keď je možné zábrániť niektorým podvodom, ktoré notári môžu svojím nečestným konaním spôsobiť, nie je možné zabrániť všetkým možnostiam. V opačnom prípade by sa bolo nutné pozerať na notára ako na rolu v rámci PKI, ktorá má pridelené zvláštne výsady, ale v princípe jej nie je možné dôverovať o nič (alebo len o trochu) viac ako obyčajnému držiteľovi certifikátu, čo by úplne degradovalo používanie osvedčení v elektronickej forme[9].

Komunikácia so serverom NKSR

Na tejto časti sa nič nemení. Notárska aplikácia pošle dáta na server NKSR, ktorý osvedčeniu priradí jednoznačný identifikátor a ten potom vráti naspäť. To, že je celá komunikácia chránená šifrovaním a obe sa strany sa navzájom autentifikujú certifikátmi je samozrejmosťou už teraz, takže pri elektronickom podpisovaní nie je nutné na tejto časti nič meniť.

Osvedčovacia kniha

Dôvody, prečo podpisovateľ robí podpis do osvedčovacej knihy, sú uvedené pri procese legalizácie. Prvé dva z nich sú pri elektronickej forme bezpredmetné, pretože ak vie niekto vytvoriť podpis v dokumente, tak ho vie vytvoriť aj v osvedčovacej knihe, pretože má prístup k súkromnému kľúču alebo prelomil algoritmy, ktoré sa používajú pri vytvárení podpisu. Jedinou výhodou, kedy by mohol notár požadovať vytvorenie podpisu falošným "podpisovateľom", je v prípade rozbitia hašovacej funkcie. V tomto prípade nie je falošný podpisovateľ schopný vytvoriť podpis nad dokumentom, ktorý mu dá podpísať notár, ak nemá k dispozícii dostatočný počet podpisov pravého podpisovateľa.
Posledný z dvôvodov, t. j. nemožnosť podpisovateľa poprieť, že žiadal o osvedčenie, je nevynutná. Keby toto nebolo zabezpečené, mohlo by dôjsť k nasledujúcemu prípadu: A a B majú medzi sebou uzavretú zmluvu, ktorá vyžaduje použitie notárskeho osvedčenia na podpisy. Obaja si podpisy osvedčia. Hociktorý z účastníkov môže poprieť, že notárske osvedčenie žiadal, pretože notár nemá o tom doklad. Pretože osobné údaje A aj B sú v notárskej databáze identít, mohol notár vydať osvedčenie bez problémov. V prípade súdneho sporu by bola platnosť osvedčenia spochybnená a tým pádom aj platnosť celej zmluvy. Možnosť zobrať žiadosti o osvedčenia z uloženej realizovanej komunikácie medzi notárskym úradom a klientom zavrhujem, pretože:

- Kládlo by to zbytočné nároky na výkonnosť.

[9] Načo by niekto používal notára na osvedčenie, že dokument podpísal, keby mu aj tak nedôveroval?

- Dokumentácia sa dá sfalšovať, podpis sa dá sfalšovať so zanedbateľnou pravdepodobnosťou.

Je teda nevyhnutné pred vydaním osvedčenia zaznamenať, že používateľ o osvedčenie žiadal, alebo potvrdil, že mu osvedčenie bolo vydané. Toto by sa mohlo robiť viacerými spôsobmi:

- Používateľ podpíše náhodné slovo (nonce), ktoré potom bude vystupovať aj v osvedčení. Tento prípad sa dá použiť kedykoľvek v priebehu procesu osvedčovania až do vydania samotného osvedčenia. Nevýhodou je, že vyžaduje ďalší údaj v osvedčení.

- Používateľ podpíše poradové číslo pridelené serverom NKSR. Dá sa použiť len po obdržaní odpovede zo servera NKSR ako pri "papierovom" osvedčovaní. Výhodou je, že túto evidenciu si uchováva len notár a nemieša ju do osvedčenia.

Osvedčovacia kniha vo forme podpísaných nonces kladie zvýšené nároky na notársku aplikáciu spojené s generovaním náhodných čísel, zatiaľ čo forma podpísaných identifikátorov navyše nezaťažuje notárov počítač. Z hľadiska bezpečnosti neprinášajú nonces nič nové, pretože ide len o to, aby používateľ podpísal niečo, čo je spojené s osvedčením, preto je vhodnejšia možnosť podpisov identifikátorov ZEP-om s časovou pečiatkou.
Záver: ako záznam v osvedčovacej knihe môže slúžiť identifikačné číslo osvedčenia podpísané ZEP-om s časovou pečiatkou.

Vydanie osvedčenia notárom

Táto časť sa v ničom nelíši od "papierovej" formy procesu legalizácie. Notár zašle osvedčnie používateľovi, ktorý musí jeho prijatie potvrdiť, napr. podpísaním osvedčenia. Ak by to neurobil, mohol by poprieť, že osvedčenie dostal. V prípade, že sa podpisovateľ podpisuje u notára, mu notár, samozrejme, osvedčenie vydá na prenosnom médiu, ale aj tak musí podpisovateľ osvedčenie podpísať, aby mal o tom notár dôkaz.

Zhrnutie

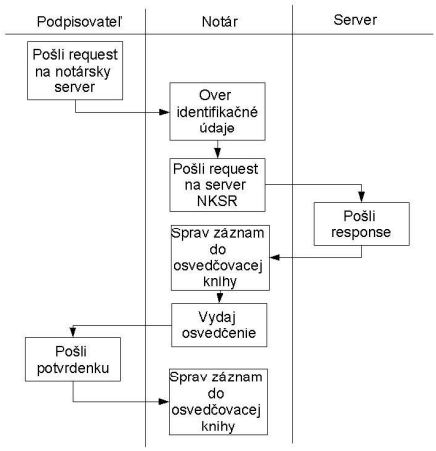
Po zhodnotení všetkých krokov procesu boli navrhnuté jeho úpravy do elektronickej formy. Výsledný proces legalizácie v elektronickej forme sa líši od "papierovej" formy v nasledujúcich bodoch:

- Notár overuje totožnosť na základe certifikátu, ktorý sa je mapovaný na identitu v databáze identít. Ak je podpisovateľ prítomný v notárskom úrade, môže overiť jeho identitu dodatočne aj podľa preukazu totožnosti.

- Všetko sa podpisuje elektronicky.

- Podpisovateľ na konci procesu podpisovania vydá notárovi potvrdenku, že dostal osvedčenie.

Tento proces sa však dá ešte upraviť zlúčením troch krokov, t. j. overenia totožnosti, podpis dokumentu a podpis do osvedčovacej knihy do jedného. Predpokladajme, že elektronické podpisovanie notárom, alebo jeho zamestnancom je len medzistupeň k elektronickému notárovi ako serveru.
Obsahj tejto časti sa priamo netýka tejto diplomovej práce, ale predsa len by bolo vhodné upraviť proces vydania elektronického osvedčenia tak, aby sa síce nelíšil veľmi od "papierovej" formy, ale nebolo ho ťažké upraviť pre elektronického notára. Elektronický notár by mal fungovať ako server, ktorý vydáva (okrem iného) elektronické osvedčenia. To znamená, že pre podpisovateľa sa javí ako objekt, ktorý na základe požiadavky podpisovateľa (request) transparentne[10] vydá osvedčenie (reply) alebo pošle chybovú hlášku. To si vyžaduje nutnosť navrhnúť formáty request a reply tak, aby vedeli zapúzdrovať vydávanie osvedčení, aké si vyžaduje ľubovoľný právny systém. To však znamená rozsiahle štúdium národných zákonov v jednotlivých krajinách. Výsledný proces z pohľadu vysokej úrovne by vyzeral, ako je to znázornené na obrázku 5.2. Celý proces by vyzeral nasledovne:



Obrázok 5.2: Proces elektronickej legalizácie

1. Podpisovateľ pošle notárovi požiadavku, v ktorej je uvedené apoň:
   - Elektronický podpis dokumentu.
   - Identifikátor certifikátu.
   - Čas odoslania.

[10]To znamená, že podpisovateľ nevie, čo sa deje vnútri, len mu príde osvedčenie alebo oznámenie o chybe.

2. Notár (notársky zamestnanec, server) overí platnosť certifikátu, podpisovateľa identifikuje podľa databázy identít a skontroluje, či bol podpis vyhotovený privátnym kľúčom, zodpovedajúcim verejnúmu kľúču uvedenému v certifikáte. Ak je niektorá z podmienok neplatná, pošle podpisovateľovi oznámenie o chybe.

3. Notár pošle požiadavku na server NKSR, údaje v požiadavke vytiahne z databázy identít.

4. Server NKSR požiadavku notára spracuje a vráti mu identifikačné číslo osvedčenia.

5. Notár vyhotoví osvedčenie ako kotrasignatúru na podpisovateľov elektronický podpis a zašle ho späť podpisovateľovi.

6. Podpisovateľ zašle notárovi podpísanú potvrdenku, že dostal osvedčenie s uvedeným identifikačným číslom.

7. Notár podpis odoslaný v kroku 1 a potvrdenku odoslanú v kroku 6 založí do osvedčovacej knihy. Ak mu nebola doručená potvrdenka v stanovenom časovom intervale, notár zneplatní osvedčenie.

5.2.2 Analýza údajov

Do procesu legalizácie vstupuje viacero údajov. Nie všetky údaje z "papierovej" formy procesu sú potrebné aj v jeho elektronickej forme. Zoznam údajov je v tabuľke 5.2.2. Výstupom z analýzy údajov bude zistenie, či jednotlivé údaje sú, alebo nie sú potrebné. Potreba údajov je indikovaná pomocou kľúčových slov REQUIRED, RECOMMENDED a OPTIONAL s súlade s RFC 2119 [19]. Význam použitých kľúčových slov je nasledovný:

- *REQUIRED* (MUST, SHALL) - explicitné vyjadrenie, že údaj (vlastnosť) musí byť prítomný(á).
- *SHALL NOT* - explicitné vyjadrenie, že údaj (vlstnosť) nesmie byť prítomný(á).
- *RECOMMENDED* - údaj (vlastnosť) nemusí byť prítomný(á), ale je nutné podrobne zvážiť, ak sa pri implementácii rozhodne o inej ceste.
- *OPTIONAL* - údaj (vlastnosť) môže ale nemusí byť prítomný(á).

Uvedené pojmy budú analogicky použité aj v ďalšom texte.

Poradové číslo osvedčenia

Identifikácia osvedčenia poradovým číslom je vhodná, pretože v prípade sporu okolo pravosti osvedčenia je možné ho jednoznačne určiť. Pretože poradové číslo určuje server NKSR, nie je možné bez notára skonštruovať falošné osvedčenie s pravým poradovým číslom. Taktiež je vhodné kvôli vyhľadávaniu osvedčenia v databáze osvedčení. Treba si ale uvedomiť, že v minulosti, keď NKSR nemala rozbehnutý server, ktorý spracováva osvedčenia sa osvedčenia museli zaobísť bez tohto údaju. Poradové číslo však nie je ťažké získať, pretože NKSR prevádzkuje zmienený server. Stačilo by prepojiť notársku aplikáciu s NKSR tak, ako je to v súčasnom "papierovom" osvedčovaní.
Záver: REQUIRED.

Údaj	Využitie
Poradové číslo osvedčenia	Osvedčovacia kniha Osvedčenie
Číslo preukazu totožnosti podpisovateľa	Databáza NKSR
Rok osvedčenia	Databáza NKSR
Typ listiny	Databáza NKSR
Meno podpisovateľa	Databáza NKSR Osvedčenie
Priezvisko podpisovateľa	Databáza NKSR Osvedčenie
Tituly pred a za menom	Databáza NKSR
Rodné číslo podpisovateľa	Databáza NKSR
Podpis podpisovateľa	Osvedčovacia kniha Dokument
Adresa podpisovateľa	Osvedčenie
Identifikačné číslo notára	Databáza NKSR
Titul, meno a priezvisko notára	Osvedčenie
Titul, meno a priezvisko notárskeho zamestnanca	Osvedčenie
Podpis notára (not. zamestnanca)	Osvedčenie
Okrúhla pečiatka notárskeho úradu	Osvedčenie
Dátum požiadania o osvedčenie	Databáza NKSR Osvedčovacia kniha
Dátum a obec, kde bolo osvedčenie vystavené	Osvedčenie
Dátum označenia osvedčenia za chybné	Databáza NKSR

5.2.2 Zoznam údajov

Číslo preukazu totožnosti

Je dôležité v prípade identifikácie kvôli tomu, aby bolo možné zistiť, či doklad s daným číslom existuje a je spojený s podpisovateľom, ktorý sa ním identifikoval.
Záver: REQUIRED.

Rok osvedčenia

Dá sa vypočítať z dátumu vydania osvedčenia, vo všeobecnosti sú pri osvedčeniach dvôležité dva časy. Dátum a čas požiadania o osvedčenie a dátum a čas vydania osvedčenia. Prvý čas by mal byť uložený len v databáze NKSR, aby bolo možné aby bolo možné kontrolovať činnosť notára. Ak by napr. bol rozdiel medzi prijatím požiadavky a časom na osvedčení príliš veľký, mohlo by to znamenať, že sa jedná o podvod. Druhý čas je dvôležitý pre podpisovateľa a mal by byť uvedený na osvedčení. Žiadne iné časové údaje na osvedčení nie sú potrebné.
Záver: Je potrebný čas prijatia požiadavky na serveri a čas vydania osvedčenia - REQIURED.

Typ listiny

Táto položka je len informatívna, môže byť uvedená, ale nemusí. Navyše v prípade, že notárovi bude zaslaný len podpis, ako bolo navrhnuté v procese elektronickej legalizácie, notár nemá ako zistiť typ listiny.
Záver: OPTIONAL.

Meno, priezvisko a tituly podpisovateľa

Meno podpisovateľa je len informatíne, pretože meno neidentifikuje podpisovateľa jednoznačne. Je však vhodné, ak overovateľ vie z osvedčenia prečítať, na čí podpis je osvedčenie vydané. Kontrolou by mohlo byť porovnanie údajov o mene, priezvisku a titule s údajmi uvedenými v certifikáte podpisovateľa.
Záver: RECOMMENDED.

Rodné číslo podpisovateľa

Je nevyhnutné na to, aby bolo osvedčenie spojené s podpisom podpisovateľa, pretože jednoznačne identifikuje podpisovateľa.
Záver: REQUIRED.

Podpis podpisovateľa v osvedčovacej knihe

Osvedčovacia kniha je modifikovaná v súlade s procesom elektronického osvedčovania. V nej sa nachádza podpis podpisovateľa v dokumente a potvrdenka. Záver: REQUIRED.

Adresa podpisovateľa

Pretože rodné číslo identifikuje podpisovateľa jednoznačne, je len informatívna, ale rovnako ako v prípade mena a priezviska je vhodné ju uviesť, aby overovateľ mohol z osvedčenia zistiť, pre koho bolo osvedčenie vydané.
Záver: RECOMMENDED.

Identifikačné číslo notára

Identifikačné číslo notára je údaj, ktorý sa skladá z dvoch čísel. Prvým číslom je poradové číslo úradu v rámci Slovenska. Druhé číslo identifikuje osobu na tomto úrade, pričom notár má číslo 0. Jenoznačne identifikuje notára aj všetkých jeho zamestnancov.. Identifikačné číslo notára musí byť uvedené v komunikácii so serverom NKSR. Je však veľmi vhodné uviesť ho aj v osvedčení, pretože identifikuje notára jednoznačne, je však potrebné umožniť napr. prostredníctvom portálu NKSR, aby si hocikto mohol overiť na základe poradového čísla, mena a priezviska notára alebo notárskeho zamestnanca, či je tento notár/zamestnanec identifikovaný stanoveným číslom.
Záver: REQUIRED pre databázu NKSR aj pre osvedčenie.

Titul, meno a priezvisko notára

Je len informatívne, pretože identifikačné číslo notára na osvedčení identifikuje jednoznačne, ale opäť je veľmi vhodné pre overovateľa.
Záver: RECOMMENDED.

Titul, meno a priezvisko notárskeho zamestnanca

Je len povinné, pretože identifikačné číslo notára na osvedčení a meno notárskeho zamestnanca identifikuje zamestnanca jednoznačne. Ak by bola uvedená položka prázdna, osvedčenie vydal priamo notár.
Záver: REQUIRED.

Podpis notára alebo notárskeho zamestnanca

Musí byť, aby mohlo byť osvedčenie vôbec platné.
Záver: REQUIRED.

Okrúhla pečiatka notárskeho úradu

V elektronickej forme zmysel nemá, pretože z osvedčnia je jasné, kto ho vydal. Vhodný je ale identifikátor, že ide o notárske osvedčenie, t. j. nejaký OID. Záver: Ako pečiatka je NOT REQUIRED, ako OID je REQUIRED.

Dátum požiadania o osvedčenie

Je vhodné pamätať si nielen dátum, ale aj čas. Databáza môže prideliť osvedčeniu vlastný čas a v osvedčovacej knihe sa čas vyberie z požiadavky podpisovateľa.
Záver: REQUIRED.

Dátum a obec, kde bolo osvedčenie vystavené

Podpis notára v osvedčení by mal obsahovať časovú pečiatku, aby nebolo pochýb o tom, kedy osvedčenie vzniklo. Miesto je bezpredmetné, pretože notársky server môže byť umiestnený fyzicky hocikde a jeho URL nemá tiež zmysel uvádzať, pretože je podstatné, kto osvedčenie vydal, t. j. ktorý notár, a nie na ktorom počítači podpisoval. V budúcnosti totiž môže nastať situácia, že notársky úrad bude prevádzkovať viacero serverov, ktoré budú príjmať požiadavky a všetky budú vydávať osvedčenia za toho istého notára.
Jediným dôvodom, pre ktorý by bolo možné uviesť miesto osvedčenia, je, aby overovateľ vedel v súlade s ktorou legislatívou je osvedčenie robené. Na toto však nie je potrebné uvádzať miesto vytvorenia osvedčenia, pretože sa tento údaj dá "schovať" do OID-u uvedeného pri okrúhlej pečiatke notárskeho úradu.
Záver: Dátum a čas sú REQIUIRED obsiahnuté v časovej pečiatke. Obec je NOT REQUIRED.

Dátum označenie osvedčenia za chybné

V databáze NKSR musí byť uvedený presný dátum a čas označenia osvedčenia za chybné kvôli poslednému kroku v procese, t. j. keď podpisovateľ nedoručí notárovi potvrdenku.
Záver: REQUIRED.

Potreba údajov	Údaje
REQUIRED	Poradové číslo osvedčenia Číslo preukazu totožnosti Čas požiadania o osvedčenie a čas vydania osvedčenia Rodné číslo podpisovateľa Identifikačné číslo notára (aj pre osvedčenie) Podpis notára (notárskeho zamestnanca) OID osvedčenia
RECOMMENDED	Meno, priezvisko a tituly podpisovateľa Adresa podpisovateľa Titul, meno a priezvisko notára Titul, meno a priezvisko notárskeho zamestnanca
OPTIONAL	Typ listiny
SHALL NOT	Okrúhla pečiatka notárskeho úradu Miesto vystavenia osvedčenia

Zhrnutie

Údaje, ktoré vstupujú do procesu osvedčovania sú rozčlenené podľa dôležitosti a sú uvedené v tabuľke. Na základe analýzy údajov je možné navrhnúť formát notárskeho osvedčenia.

5.3 Formát notárskeho osvedčenia

Spôsoby ako vytvoriť notárkske osvedčenie sú dva:

1. Vytvoriť osvedčenie ako dáta, tieto dáta potom podpísať, pričom do podpisovania vstupuje aj osvedčovaný EP.

2. Vytvoriť osvedčenie ak kotrasignatúru, kde dáta v osvedčení sú súčasťou kontrasignatúry.

Prvá možnosť je síce kópiou "papierového" osvedčenia, ale dá sa prirodzene nahradiť pomocou kontrasignatúr, kde podpísané dáta v osvedčení sú podpísané atribúty, preto sa bude ďalej brať len možnosť osvedčenia pomocou kontrasignatúr. Kontrasignatúra v CMS je SignerInfo a na jeho rozširovanie slúžia atribúty. Dáta v osvedčení môžeme zabaliť do jedného veľkého atribútu alebo rozdeliť do viacerých malých atribútov. Tieto atribúty musia byť podpísané, aby nemohol niekto vyhodiť údaje z osvedčenia bez toho, aby sa to nedalo zistiť.

5.3.1 Jediný atribút notárskeho osvedčenia

Táto možnosť má nižšiu flexibilitu a jej jedinou výhodou je, že je potebné registrovať len jediný OID oproti viacerým pri rozdelení dát do viacerých atribútov. Všetky dáta sú v jednej štruktúre, čo je síce vhodné pri spracovaní osvedčenia človekom, ale pre počítač je to bezpredmetné, takže to nepovažujem za výhodu. Celá štruktúra v ASN.1 by mohla vyzerať nasledovne:

```
attrValue ::= SEQUENCE{
    version INTEGER,
    identNumber INTEGER,
    IDcardNumber STRING,
    birthNumber STRING,
    notaryID INTEGER,
    dateAntTimeOfIssuance GENERALIZED TIME,
    notaryName STRING,
    notaryEmployeeName STRING,
    signerName STRING OPTIONAL,
    signerAddress STRING OPTIONAL,
    documentType STRING OPTIONAL}
```

Dáta nie je potrebné ďalej štrukturovať, takže táto štruktúra postačuje. Na rozlíšenie, ktoré OPTIONAL pole je prítomné sa použije vhodné kódovanie.

5.3.2 Viaceré atribúty notárskeho osvedčenia

Viaceré atribúty prinášaju vyššiu flexibilitu v tom, že v prípade potreby zmeny údajov v osvedčení nie je problém atribút pridať alebo vyhodiť. Overovač potom

môže spracovať atribúty, ktoré pozná a vyhodiť overovateľovi informáciu o tom, že niektoré atribúty sa nepodarilo rozpoznať a nemusí celé osvedčenie zamietnuť, pretože nepozná verziu atribútu. Typy atribútov by boli identické s typmi uvedenými vyššie, akurát nie je potrebné zadefinovať atribút *version*, pretože v tomto prípade nemá zmysel.

5.4 Formát notárskeho certifikátu

Okrem samotného podpisu je nevyhnutné uviesť aj v certifikáte fakt, že osoba, ktorá osvedčenie vydala, je notár alebo notársky zamestnanec. V opačnom prípade by hrozilo falšovanie notárskeho osvedčenia. Na vyvorenie podpisu, samozrejme, nie je potrebný certifikát, a pretože syntax notárskeho osvedčenia je (bude) známa, je možné skonštruovať notársky podpis. Keby bol overovateľ odkázaný len na informácie z podpisu, príp. osvedčenia, jediný spôsob, ako by si mohol overiť, či dané osvedčenie naozaj existuje, je použiť stránku NKSR. Aj tak však existujú nasledovné nevýhody:

1. Veľa ľudí, potenciálnych overovateľov, vôbec nevie, že čosi také funguje a keby to aj vedeli, tak by ich táto komplikácia mohla odradiť od používania elektronickej formy dokumentov.

2. Stránka NKSR neposkytuje dostatok informácii, a tak by si mohol podpisovateľ dať osvedčiť jeden dokument a následne na základe tohto osvedčenia a falošného certifikátu vygenerovať veľa ďalších osvedčení. Falošný certifikát by si nechal vystaviť napr. na menovca notára. Aj keď údaje v osvedčení a v databáze NKSR by zmeniť nemohol, mohol by takto osvedčovať ľubovoľné množstvo elektronických dokumentov, ktoré by mal vopred prichystané[11].

Na základe týchto dvôvodov je nevyhnutné, aby bol notársky status uvedený v certifikáte. Pri abstraktnom pohľade je notár (resp. notársky zamestnanec) rolou v rámci PKI, ktorú môžu mať pridelenú viaceré osoby. Spôsobov, akým možno naznačiť, že daná osoba má štatút notára je viacero:

1. *Atribútové certifikáty*,

2. *Vhodný extension v certifikáte*.

Pri výbere najvhodnejšej možnosti bude zobratá do úvahy náročnosť implementácie v Slovenských podmienkach.

5.4.1 Atribútové certifikáty

Atribútové certifikáty slúžia na preukázanie toho, že nejaká osoba v rámci PKI má stanovenú vlastnosť. Držiteľ atribútového certifikátu je zadefinovaný v poli *Holder* a existujú tri spôsoby, ako môže byť identifikovaný:

[11] Kvôli času vydania osvedčenia.

- *baseCertificateID* - podľa vydaného certifikátu verejného kľúča.
- *entityName* - podľa mena držiteľa certifikátu vo forme distinquished name.
- *objectDigestInfo* - podľa verejného kľúča alebo inej veličiny k nemu jednoznačne priraditeľnej na základe jej hašu.

V prvom rade je potrebné povedať, že pojem atribútového certifikátu sloveská legislatíva nepozná. V §6 zákona 215/2002 sa doslovne uvádza, že:

> (1) Certifikát verejného kľúča (ďalej len "certifikát") je elektronický dokument, ktorým vydavateľ certifikátu potvrdzuje, že v certifikáte uvedený verejný kľúč patrí osobe, ktorej je certifikát vydaný (ďalej len "držiteľ certifikátu").

Inde v zákone ani vo vyhláškach nie je spomínaný iný typ certifikátu ako certifikát verejného kľúča, takže v prípade zvolenia tejto možnosti by bolo nevyhnutné zadefinovať pojem atribútového certifikátu ako aj časť PKI, ktorá je s životným cyklom atribútového certifikátu spojená. Ide najmä o atribútové autority, ktoré vydávajú atribútové certifikáty.
Aj keď sú atribútové certifikáty prirodzenou možnosťou na potvrdzovanie atribútov držiteľa certifikátu verejného kľúča v rámci PKI, ich zavedenie prináša nasledovné ťažkosti:

- Potrebu zmeny zákona a príslušných vyhlášok.
- Vybudovanie siete atribútových autorít.
- Potrebu prepísania existujúcich overovačov, aby akceptovali aj atribútové certifikáty.

Ako najväčšou prekážkou sa javí budovanie siete AA, a to najmä z hľadiska finančných investícií a nízkeho okruhu klienov[12].

5.4.2 Extension

Extension v certifikáte je pole, ktoré rozširuje údaje uvedené v certifikáte. Podľa RFC 3280 [6] je definované nasledovne:

```
Extension ::= SEQUENCE {
    extnID OBJECT IDENTIFIER,
    critical BOOLEAN DEFAULT FALSE,
    extnValue OCTET STRING }
```

Význam polí v Extension je nasledovný:

- *extnID* je extension OID, ktorý identifikuje, o akú extension ide,

[12] Notárov je na Slovensku len niekoľko stoviek.

- *critical* špecifikuje, či je extension kritická alebo nie. Ak systém, ktorý používa certifikáty pri spracovaní certifikátu narazí na extension, ktorá je kritická a nepozná ju, musí certifikát zamietnuť,
- *extnValue* je hodnotou extension.

Výhodou použitia extension oproti atribútovým certifikátom je, že nie je potrebné dopĺňať existujúce PKI o atribútové autority, a tým pádom ani výrazne meniť existujúce podpisovače a overovače. Nevýhodou tohto prístupu je opäť OID, ktorý je potrebné definovať a najlepšie presadiť do medzinárodných štandardov kvôli kompatibilite so zvyškom sveta. Je možných viacero prístupov k extension, preto budú najprv rozobraté prístupy k extension a potom samotná kritickosť notárskemu.
Sú možné dva prístupy k extension:

- Špecifikovať v extnID, že ide o notársky certifikát.
- Špecifikovať v extnID, že ide o nejakú rolu, ktorá bude popísaná v extnValue.

Oba prístupy majú svoje výhody. Výhodou prvého prístupu je jednoduché spracovanie overovačom, stačí ak overovač rozpozná OID. Nevýhodou je malá možnosť rozširovania. To by mohlo byť zaujímavé v prípade, že by bolo potrebné rozšíriť elektronické podpisovanie aj na iné inštitúcie, napr. už spomínanú matriku. V tomto prípade by bola lepšia druhá možnosť, pretože by nebolo nutné pre každú inštitúciu definovať nový OID, len by sa zadefinovala nová extnValue. Z hľadiska overovača je to jedna podmienka navyše, takže zložitosť overovania veľmi nevzrastie.
Čo sa týka kritickosti, je nevyhnutné, aby bola táto extension označená ako *kritická*, čo zabezpečuje, že overovače, ktoré ju nerozpoznajú, zamietnu takýto notársky certifikát.

5.5 Bezpečnosť notárskeho osvedčenia

Notárskym osvedčením a notárskym podpisom notár garantuje určité skutočnosti v závislosti od toho, na čo bol notárov podpis použitý, pričom ich nie je možné zo zákona garantovať "obyčajným" podpisom ani ZEP-om. To znamená, že na bezpečnosť notárskeho podpisu sú kladené vyššie nároky ako na ZEP.
V čom by bolo možné zlepšiť bezpečnosť elektronického podpisu pre notárov? Bezpečnosť ZEP-u ovplyvňujú nasledovné faktory:

1. Bezpečné zariadenie, pomocou ktorého vzniká ZEP.
2. Algoritmy a kľúče použité pri ZEP-e.
3. Požiadavky kladené na akreditovanú CA.
4. Osveta podpisovateľov.

5.5.1 Bezpečné zariadenie

Pri vytváraní notárskeho podpisu je nevyhnutné použiť bezpečné zariadenie, aby bolo zaručené, že k privátnemu kľúču notára sa nedostane útočník. Z bezpečného zariadenia nie je možné privátny kľúč vytiahnuť v reálnom čase, pretože je na ňom uložený v zašifrovanej podobe, pričom šifrovací kľúč je odvodený od PINu. Útočník, samozrejme, môže skúsiť útok hrubou silou a skúšať všetky šifrovacie kľúče, pretože si vie skontrolovať, či sa získaný dešifrovaný kľúč hodí k verejnému kľúču. Efektívnejšie je však použiť nepriame metódy útoku, ako je napr. meranie spotreby elektrického prúdu a týmto obmedzovať množinu šifrovacích kľúčov. Je preto možné klásť zvýšené nároky na bezpečné zariadenie. Samozrejme, tieto kritériá nesmú byť premrštené, pretože cena za bezpečné zariadenie by mohla byť astronomická. Navyše by mala byť zachovaná prenosnosť bezpečných zariadení, aby sa mohol notár podpisovať na ľubovoľnom mieste s použitia notebooku a pripojenia na internet.

5.5.2 Algoritmy a kľúče

Algoritmy použité pri vytváraní EP sú schvaľované a publikované v medzinárodných štandardoch. Takže je stanovaná množina algoritmov, ktoré sú približne rovnako bezpečné[13]. Preto výber algoritmu závisí od odporúčaní, napr. iniciatívy NESSIE[14].
Dĺžka kľúčov zohráva pri bezpečnosti veľkú úlohu. Čím je väčšia, tým sa zväčšuje množina, z ktorej je možné kľúče vyberať, a tým pádom sa zvyšuje odolnosť voči útoku úplným preberaním, aj keď niektoré špeciálne kľúče nie je možné používať kvôli útokom na ne. Samozrejme, s rastúcou dĺžkou sa zvyšujú nároky na výkon čipov bezpečných zariadení, pretože čip musí napr. vykonať matematické operácie nad väčšími číslami. Kľúče použité pri konštrukcii notárskeho podpisu by mali byť dlhšie ako je odporúčané pre obchodný styk, pretože nároky na bezpečnosť notárskeho podpisu sú väčšie ako napr. na podpis kontaľa spoločnosti. Väčšia dĺžka kľúčov umožňuje ochrániť pôvodný podpis pred rozbitím podpisovacieho algoritmu nasledovne:
podpisovateľ podpíše dokument a notár mu pre tento dokument vydá osvedčenie, t. j. spraví kotrasignatúru nad jeho podpisom. V priebehu času sa však podarí rozbiť podpisovateľov podpisovací algoritmus alebo výkon počítačov narastie tak, že je možné odhaliť podpisovateľov privátny kľúč úplným preberaním. Za predpokladu, že algoritmus použitý notárom, alebo dĺžka notárovho kľúča je stále dostatočná, je podpisovateľov podpis chránený a nikto, ani samotný podpisovateľ nemôže spochybniť jeho platnosť na základe útokov na podpisovací algoritmus alebo dĺžku kľúča, pretože notárova kontrasignatúra garantuje, že podpis bol vytvorený a nie je zmenený.

[13] Porovnávanie týchto algoritmov je mimo rozsah tejto diplomovej práce a vyžaduje si hlboké znalosti z kryptoanalýzy.
[14] www.cryptonessie.org

Je potrebné si uvedomiť, že tento prípad sa vzťahuje len na *podpisovací* algoritmus a nie na *hašovací*. Ak totiž niekto nájde dva zmysluplné[15] texty s rovnakým hašom, tak podpis podpisovateľa bude platiť na oboch bez ohľadu na to, koľko kontrasignatúr je spravených nad podpisovateľovým podpisom. Samozrejme, nič nie je trvalé a jedného dňa bude možné rozbiť aj notárov podpis, ale tomu sa nedá zabrániť. Je potrebné si uvedomiť, že ani osvedčenia v "papierovej podobe" nie sú na večné časy, lebo aj papier sa raz znehodnotí, resp. degraduje tak, že osvedčenie nebude čitateľné. Je však vhodné zaistiť, aby osvedčenia boli platné čo najdlhšie a to sa dá napr. extrémne dlhými kľúčmi (so spomínanými nevýhodami ohľadne výkonu) alebo archívnymi časovými pečiatkami nad osvedčeným podpisom. Nikto nezaručí, že deň po vydaní osvedčenia nebude publikovaný útok, ktorý úplne rozbije hašovací alebo podpisovací algoritmus. Odbornosť ľudí, ktorí algoritmy navrhli a posudzovali však môže zaručiť odolnosť algoritmov.

5.5.3 Akreditovaná CA

Otázka toho, kto bude vydávať certifikáty pre notárov, je riešená v kapitole 5.6, preto sa teraz sústredím len na požiadavky na bezpečnosť CA. Z pohľadu bezpečnosti notárskeho EP sa CA pri svojej činnosti sústreďuje na to, aby:

1. Kľúče vygenerované pre klientov boli dostatočne kvalitné, ak im ich generuje.
2. Jej privátny kľúč nebol kompromitovaný.
3. Obdobie medzi oznámením o kompromitácii kľúča a revokáciou certifikátu bolo čo najkratšie.

Kvalita kľúča

Pri generovaní kvality kľúča je možné klásť zvýšené kritéria na náhodnosť kľúča použitím certifikovaného fyzikálneho generátora náhodných čísel a testovaním kľúčov proti známym útokom.

Kompromitácia privátneho kľúča CA

Na privátny kľúč CA sú kladené extrémne bezpečnostné nároky, čo je vyriešené dobre už pri akreditovanej CA.

[15] Zmysluplné preto, že bude ťažké presvedčiť súd o tom, že niekto podpísal náhodne vyzerajúce dáta a nie napr. kúpno-predajnú zmluvu. Takýto prípad "ukončil život" hašovaciemu algoritmu MD4, kedy boli prezentované dve kúpno-predajné zmluvy s rôznymi cenami nehnuteľností s rovnakým hašom.

Proces revokácie klientovho certifikátu

Zo zákona musí byť CA schopná neodkladne revokovať klientov certifikát 24 hodín denne 7 dní v týždni.
Z horeuvedených argumentov vyplýva, že postačuje bezpečnosť na úrovni akreditovanej CA a nie je nutné zavádzať vyšší stupeň bezpečnosti CA.

5.5.4 Osveta podpisovateľov

Žiadna technológia nepomôže, keď notár dá k dispozícii niekomu čipovú kartu a prezradí PIN. Je preto nevyhnutné vyškoliť notárov a ich zamestnancov a naučiť ich správne pracovať s elektronickými dokumentmi, základom bezpečnosti a zdravej nedôvere. Je pre dobro notárov a ich zamestnancov, aby boli obozretní, pretože ručia za svoje výkony a môžu byť podľa zákona postihovaní.

5.6 Vydávanie certifikátov notárom

Ako už bolo spomínané, bolo by vhodné, aby notárske osvedčnie bolo aspoň na úrovni ZEP-u. To však predpokladá vydávanie certifikátov notárom, ktoré sú na úrovni kvalifikovaného certifikátu a tieto sa nedajú vydávať, ak nie je k dispozícii CA, ktorá splnila podmienky na akreditovanú CA.
Kto by mal prevádzkovať akreditovanú CA, ktorá by vydávala certifikáty notárom? Možnosti sú nasledovné:

1. ACA[16] by bola prevádzkovaná štátom.
2. ACA by bola komerčná.

Obe možnosti majú svoje pozitíva aj negatíva.

5.6.1 Štátom prevádzkovaná ACA

Výhodou štátom prevádzkovanej ACA je, že za jej činnosť ručí štát. Do úvahy v tomto prípade prichádzajú dve CA:

1. Koreňová ACA prevádzkovaná NBÚ.
2. CA prevádzkovaná NKSR.

V prvom prípade je výhodou, že koreňová CA je akreditovanou CA a prevádzkovateľom je inštitúcia, ktorá usmerňuje PKI v rámci Slovenskej republiky. Nevýhodou je neznalosť notárskeho prostredia.
V druhom prípade sú výhody a nevýhody presne opačné, t. j. NKSR CA nie je akreditovanou CA, ale má skúsenosti s vydávaním certifikátov pre notárov.
Ideálnym riešením by bolo využiť odborné skúsenosti z odboru bezpečnosti pracovníkov NBÚ, praktické skúsenosti s notármi pracovníkov NKSR a prerobiť NKSR CA na akreditovanú CA.

[16] Akreditovaná CA.

5.6.2 Komerčná ACA

Pre komerčné ACA by bolo vydávanie kvalifikovaných certifikátov notárom určite zaujímavým trhom. Ak by bol špecifikovaný formát kvalifikovaného certifikátu pre notára, nebol by problém vydávať im certifikáty komerčnými ACA.
Problém vidím zo strany akceptácie takéhoto riešenia notármi, pretože sa stráca štátna garancia, ale keby sa táto možnosť legislatívne podchytila, tak by ho museli akceptovať.

Záver

Cieľom tejto diplomovej práce bolo predstavenie niektorých vybraných problémov z oblasti viacnásobných EP. V práci sa mi podarilo rozšíriť pojem EP na viacnásobných EP, pričom bola vykonaná abstrakcia pojmu viacnásobného EP do grafovej štruktúry. Boli predstavené viaceré základné druhy viacnásobných EP a pre ne boli uvedené aplikácie v praxi.
Boli predstavené problémy s podpisovými politikami pre viacnásobné EP a spravené rozšírenie podpisovej politiky na politiku pre viacnásobné podpisy v ASN.1.
Kvôli aplikáciám viacnásobných EP v slovenských podmienkach bola analyzovaná slovenská legislatíva a formát viacnásobného EP, ktorý z nej vyplýva, pričom boli navrhnuté alternatívy k tomuto formátu. V poslednej časti bola spravená analýza procesu notárskeho osvedčenia podpisu, ktorá ukázala reálne využitie kontrasignatúr na Slovensku.
Počas tvorby tejto diplomovej práce sa ukázali ďalšie oblasti, ktoré by bolo zaujímavé preskúmať. Jednak je to problém s vyčíslením efektívnosti zjednodušenia overovania certifikačnej cesty pri viacnásobných EP a potom sú to podpisové politiky pre viacnásobné EP v XML.
Viacnásobné EP majú mnoho aplikácii v elektronickom obchode ako aj v štátnej správe. V oboch prípadoch existujú procesy, ktoré nie je možné pokryť bez implementácie viacnásobných EP.

Literatúra

[1] *Anouncing the Advanced Encryption Standard*. FIPS, 2001.

[2] Martin Stanek. *Základy kryptológie*. 2004.

[3] *Electronic signatures formats*. ETSI, 2000.

[4] *Common Criteria I,II,III*. ISO/IEC, 1999.

[5] *Introduction to Public Key Technology and Federal PKI Infrastructure*. NIST, 2001.

[6] *RFC 3280 - Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List Profile*. The Internet Society, 2002.

[7] *Zákon 215/2002 o elektronickom podpise*. 2002.

[8] Antonio Lioy a Gianluca Ramunno. *Multiple Electronic Signatures on Multiple Documents*. 2004.

[9] *RFC 2630 - Cryptographic Message Syntax*. The Internet Society, 1999.

[10] *XML Advanced Electronic Signatures (XAdES)*. ETSI, 2002.

[11] *XML Signature Syntax and Processing*. W3C, 2001.

[12] Peter Rybár. *Certificate Path Validation*. NBÚ SR, 2005.

[13] *Vyhláška 542/2002*. NBÚ, 2002.

[14] *RFC 3125 Electronic Signature Policies*. The Internet Society, 2001.

[15] *Signature policy for extended bussiness model*. 2003.

[16] *Schválené formáty zaručených elektronických podpisov*. NBÚ, 2004.

[17] *RFC 2046 Multipurpose Internet Mail Extension (MIME) Part Two*. The Internet Society, 1996.

[18] *Zákon o notároch a notárskej činnosti*. 1992.

[19] *RFC 2119 Key words for use in RFCs to Indicate Requirement Levels*. The Internet Society, 1997.