Univerzita Karlova v Praze
Matematicko-fyzikální fakulta

BAKALÁŘSKÁ PRÁCE

Ema Krejčová

Digitální peníze

Katedra algebry

Vedoucí bakalářské práce: Doc. RNDr. Jiří Tůma, DrSc.

Studijní program: Matematika, obecná matematika

2006

Prohlašuji, že jsem svou bakalářskou práci napsala samostatně a výhradně s použitím citovaných pramenů. Souhlasím se zapůjčováním práce.

V Praze dne 8. 8. 2006 Ema Krejčová

Obsah

Název práce: Digitální peníze
Autor: Ema Krejčová
Katedra: Katedra algebry
Vedoucí bakalářské práce: Doc. RNDr. Jiří Tůma, DrSc.
e-mail vedoucího: jiri.tuma@mff.cuni.cz

Abstrakt: Digitální peníze byly vymyšleny jako prostředek k přesouvání peněz po síti. Funkční systém digitálních peněz musí být bezpečný proti nejrůznějším typům útoků (např. dvojí užití téže mince, padělání mincí, spiknutí některých uživatelů proti jiným), ale zároveň musí zachovat anonymitu poctivých uživatelů. V této práci se pokusíme shrnout podmínky, které musí digitální platební systém splňovat, a mechanismy, které tento systém používá. Už byla navržena řada digitálních platebních systémů, některé z nich se zde pokusíme popsat a porovnat.

Klíčová slova: kryptografie, elektronický obchod, digitální peníze

Title: Digital Cash
Author: Ema Krejčová
Department: Department of Algebra
Supervisor: Doc. RNDr. Jiří Tůma, DrSc.
Supervisor's e-mail address: jiri.tuma@mff.cuni.cz

Abstract: Digital cash was invented to enable money transfers through computer networks. A digital cash system has to be secure against various types of attack (e.g. forgery, double-spending, fraud, frame up) but at the same time provide complete anonymity for a honest user. In this paper we summarize criteria which a digital system should meet and procedures which it uses for this purpose. We describe some of the digital systems that were proposed and compare them.

Keywords: cryptography, electronic commerce, digital cash

1. Úvod

K čemu mají být digitální peníze?

Mají usnadnit obchod přes internet, umožnit platby na dálku. Mají si ale také uchovat výhody papírových nebo kovových peněz, které třeba u platebních karet chybějí - zejména anonymitu nakupujícího.

Internet je rozšířen po celém světě a slouží lidem k nejrůznějším účelům. A protože lidská společnost se zakládá především na obchodu, zkoumají a zkoušejí se možnosti, jak obchodovat přes internet. Představme si například servery nabízející ke stažení hudbu — jak mají zájemci za stahování platit? Například digitálními penězi. Informace za informace, bity za bity. Systém placení musí být jednoduchý a nesmí být nákladný, protože se předpokládají přesuny pouze malých částek.

Už byla navržena řada digitálních platebních systémů, zde se pokusíme některé z nich popsat a porovnat. Zejména se pokusíme shrnout metody, které se k vytváření digitálních platebních systémů používají.

Ve druhé kapitole se zmíníme o věcech, které se digitálních peněz přímo netýkají, na které se ale budeme později odkazovat. Ve třetí a čtvrté kapitole popíšeme, co vlastně jsou digitální peníze a co od nich očekáváme. V páté kapitole popíšeme jednotlivé mechanismy a procedury, ze kterých se systémy digitálních peněz skládají. V šesté kapitole uvedeme přehled digitálních platebních systémů, které byly zatím navrženy, a v sedmé kapitole stručně zmíníme o bezpečnosti digitálních peněz.

2. Základy

2.1 Něco z algebry, značení

Definice všech pojmů jako např. okruh či grupa a jejich základní vlastnosti zde uvádět nebudeme, lze je najít např. v [1]. Uvedeme si ale značení, které budeme používat:

Symbol $\mathbb{Z}_n$ bude značit okruh celých čísel $\{0, 1, \ldots, n-1\}$ s operacemi sčítání a násobení modulo n.

Stejným symbolem $\mathbb{Z}_n$ budeme někdy značit i aditivní grupu $\{0, 1, \ldots, n-1\}$, ale k nejasnostem by tu dojít nemělo.

Symbol $\mathbb{Z}_n^*$ bude značit multiplikativní grupu všech invertibilních prvků okruhu $\mathbb{Z}_n$, tj. všech takových $a \in \mathbb{Z}_n$, pro která existuje $a^{-1} \in \mathbb{Z}_n$ takové, že $a \cdot a^{-1} = 1$. Invertibilní prvky okruhu $\mathbb{Z}_n$ jsou právě ta $a \in \{0, 1, \ldots, n-1\}$, která jsou s n nesoudělná.

Je-li n prvočíslo, je $\mathbb{Z}_n^*$ cyklická a má přesně $\varphi(n-1)$ generátorů.

Symbol $\varphi(n)$ značí Eulerovu funkci, která pro každé přirozené $n \geq 2$ udává počet čísel menších než n nesoudělných s n. Pro $n = 1$ se definuje $\varphi(n) = 1$.

Pro a nesoudělná s n platí

$a^{\varphi(n)} \equiv 1 \; mod \; n$ (Eulerova věta, pro n prvočíslo též malá Fermatova věta)

K výpočtu Eulerovy funkce slouží následující vzorce:

$$\varphi(n) = \varphi(p_1^{e_1}) \cdot \ldots \cdot \varphi(p_k^{e_k})$$
$$\varphi(p^k) = p^{k-1}(p-1)$$
$$\varphi(p) = p - 1$$

kde p je prvočíslo a $n = p_1^{e_1} \cdot \ldots \cdot p_k^{e_k}$ je rozklad n na prvočinitele. [1]

Symbolem $\{0,1\}^k$ budeme značit množinu všech bitových řetězců (řetězců 0 a 1) délky k. Symbol $\{0,1\}^*$ pak bude znamenat množinu řetězců libovolné délky.

Symbol $\|$ bude znamenat zřetězení, symbol $\oplus$ binární sčítání.

2.2 Jednosměrné funkce

Kryptografie pracuje zejména s takovými funkcemi, které je poměrně snadné spočítat, ale obtížně se dají invertovat. Na těchto obtížně invertovatelných (jednosměrných) funkcích je postavena bezpečnost kryptografických protokolů (ovšem samotný fakt, že danou funkci lze obtížně invertovat, bezpečnost nezaručí, to uvidíme později).

Jednosměrné funkce použitelné v praxi by měly mít „zadní vrátka“, tj. měla by existovat tajná informace, s jejíž znalostí už by bylo snadné spočítat invers.

Většina používaných jednosměrných funkcí je založena buď na obtížnosti faktorizace velkých čísel (např. RSA funkce) nebo na obtížnosti výpočtu tzv. diskrétního logaritmu.

Diskrétní logaritmus:

Mějme funkci

$$f : \mathbb{Z}_{p-1} \to \mathbb{Z}_p^*$$
$$x \to g^x$$

kde p je prvočíslo a g je generátor grupy $\mathbb{Z}_p^*$. Pokud x probíhá všechny prvky $\mathbb{Z}_{p-1}$, pak g^x probíhá všechny prvky $\mathbb{Z}_p^*$. Obě množiny mají stejně prvků, tedy f je bijekce. Existuje k ní proto také inverzní funkce f^{-1}, kterou nazýváme diskrétní logaritmus (přívlastek „diskrétní“ jej má odlišit od obvyklého logaritmu definovaného pro reálná čísla).

RSA funkce:

$$RSA_e : \mathbb{Z}_n \to \mathbb{Z}_n$$
$$x \to x^e$$

kde n je násobek dvou různých (velkých) prvočísel p a q. Číslo e je nesoudělné s $\varphi(n)$. Vezmeme-li prvek $d = e^{-1} \ mod \ \varphi(n)$ a funkci RSA_d (je definována, protože d je také nesoudělné s $\varphi(n)$), pak platí:

$$RSA_e \circ RSA_d = RSA_d \circ RSA_e = id_{\mathbb{Z}_n}$$

$$x^{ed} = x^{de} = x$$

Abychom toto tvrzení dokázali, musíme rozebrat tři případy:

a) je - li $x \in \mathbb{Z}_n^*$, je $x^{\varphi(n)} = 1$ a tedy $x^{ed} = x^{ed \ mod \ \varphi(n)} = x^{de \ mod \ \varphi(n)} = x$.

[1] Důkazy těchto vztahů lze najít např. v [2], str. 253, 257.

b) pokud p i q dělí x, je $x = 0$ (jsme v $\mathbb{Z}_n$), a tedy $x^{ed} = x^{de} = x$.

c) pokud např. p dělí x a q nedělí x, máme

$$x^{ed} \equiv 0 \ mod \ p$$
$$x^{ed} \equiv x^{ed \ mod \ (q-1)} \equiv x \ mod \ q$$

protože platí $\varphi(n) = (p-1)(q-1)$. Z toho plyne, že $x^{ed} \equiv x \ mod \ n$. Stejně tak pro x^{de}.

Funkce RSA_e udává určitou permutaci množiny $\mathbb{Z}_n$ a je snadné ji spočítat. Se znalostí d je snadné spočítat i její invers, funkci RSA_d. Bez znalosti d je ale invertování obtížné (za předpokladu, že p a q jsou dost velká). Je stejně obtížné jako faktorizace n. Umíme-li totiž faktorizovat $n = pq$, můžeme spočítat $\varphi(n)$ a pak pomocí Euklidova algoritmu i d. Protože $\gcd(e, \varphi(n)) = 1$ (tak jsme si e zvolili), existují čísla c a d taková, že $de + c\varphi(n) = 1$, tedy $ed \equiv 1 \ mod \ \varphi(n)$. Naopak umíme-li z dvojice (n, e) získat d, umíme i faktorizovat n. Tento algoritmus je o něco složitější, bez důkazu je popsán např. v [3], str. 287, další podrobnosti lze nalézt např. v [2], str. 271.

Ještě také není vyloučena možnost, že invertovat RSA funkci lze i jinak než nalezením d. V tom případě by srovnávání s faktorizací nebylo správné. Ekvivalence problému invertování RSA funkce a problému faktorizace nebyla zatím dokázána, nicméně se má za to, že jednodušší způsob invertování RSA funkce není.

Ještě je třeba upřesnit, co znamená „obtížně invertovat“ a „lehce spočítat“.

Definice. *Funkci g nazveme zanedbatelnou, pokud $g(x) < \frac{1}{p(x)}$ pro každý polynom p a pro všechna dostatečně velká x.*

Definice. *Funkci $f : \{0,1\}^* \to \{0,1\}^*$ nazveme jednosměrnou, pokud platí:*

1. *existuje polynomiální algoritmus A (polynomiální v délce vstupu), který tuto funkci počítá,*
2. *a pro jakýkoli pravděpodobnostní polynomiální algoritmus B platí, že pravděpodobnost nalezení inversu f pomocí B je zanedbatelná, tj.*

$$Pr[B(f(x)) = x] < \frac{1}{p(x)}$$

pro všechna x dostatečně dlouhá a pro libovolný polynom p. Pravděpodobnost zde bereme přes všechny náhodné volby x a všechny náhodné bity v pravděpodobnostním algoritmu B.

2.3 Hašovací funkce *(hash function)*

Hašovací funkce h bere jako vstup zprávu z, tedy řetězec bitů libovolné délky, a dává jako výstup řetězec bitů dané délky n, tzv. **otisk**.

$$h : \{0,1\}^* \to \{0,1\}^n$$
$$z \to h(z)$$

Hašovací funkce reprezentují dlouhé zprávy pomocí krátkých, tzv. otisků. Mohou se používat při kontrole, zda byla zpráva z správně vytvořena, např. ke kontrole vstupních hesel — při vytvoření hesla se uloží jeho otisk a při dalším zadávání hesla se ověří, že otisk je stejný.

Nás bude zajímat užití hašovacích funkcí při digitálním podpisu, kdy se místo zprávy z podepisuje pouze její otisk $h(z)$.

Funkce h zobrazuje nekonečnou množinu na konečnou, rozhodně tedy není prostá. Dvojici (z', z), pro kterou platí $h(z') = h(z)$, říkáme **kolize**. Kdyby bylo snadné nacházet kolize, bezpečnost digitálního podpisu by tím dost utrpěla. Řekněme, že Alice podepíše otisk $h(z)$ pro zprávu z. Pokud by záškodník Záviš měl nezanedbatelnou šanci nalézt zprávu $z' \neq z$, že by $h(z') = h(z)$, mohl by předstírat, že Alice podepsala z', a ne z. Chceme tedy, aby k danému obrazu $h(z)$ nebylo možné (s nezanedbatelnou pravděpodobností) najít druhý vzor $z' \neq z$. Chceme dokonce ještě víc, chceme, aby k funkci h nebylo možné najít takovou dvojici (z, z'), že $h(z') = h(z)$. To aby ani Alice nemohla podvádět — podepsat z a pak tvrdit, že podepsala z'. Funkci, u níž je pravděpodobnost nalezení kolize zanedbatelná, říkáme **bezkolizní**. Bezkolizní hašovací funkce je jednosměrná.

2.4 Šifrování s veřejným klíčem

Pod pojmem šifrování si člověk asi nejprve představí takovou komunikaci, kdy si dva nebo více lidí posílají tajné zprávy, které lze přečíst jen pomocí zvláštního klíče, který všichni zúčastnění znají a nikdo zvenčí jej nezná (a nemůže jej ani uhodnout). Pokud opravdu takový klíč mají, je bezpečnost jejich komunikace zaručena. Takto funguje **symetrická kryptografie**, která k zašifrování i rozluštění zprávy používá tentýž klíč.

Jak si ale účastníci tajný klíč předají? Distribuce klíčů se musí provádět na jiném principu. Šifrování zpráv navíc nemusí mít za cíl pouze udržet nějakou informaci v tajnosti, může také zajišťovat ověření totožnosti osob, např. právě u digitálních podpisů. Těmto účelům slouží **asymetrická kryptografie**, ve které se pro zašifrování používá jiný klíč než pro luštění.

Při šifrování s veřejným klíčem je každému účastníkovi přiřazena dvojice klíčů (vk, sk), veřejný klíč vk může znát kdokoli (může být třeba na internetu), soukromý klíč sk je osobním tajemstvím. K každému veřejnému klíči patří funkce F_{vk}, pomocí níž se zprávy šifrují a jíž slouží vk jako parametr. I funkce F_{vk} jsou veřejné.

Mějme účastníka jménem Bob, jehož dvojice klíčů je v_B, s_B. Alice chce Bobovi poslat tajnou zprávu z. Alice zná jeho veřejný klíč v_B i funkci F_{v_B}, tak spočítá $c = F_{v_B}(z)$ a pošle c. Aby zpráva opravdu zůstala v tajnosti, nesmí se z dát odvodit z c. Funkce F_{v_B} tedy musí být jednosměrná. Jediný, kdo umí zprávu rozluštit, je Bob, kterému to umožní jeho soukromý klíč („zadní vrátka“).

Naopak chce-li Bob zveřejnit zprávu x a chce-li, aby bylo všem jasné, že je od něj, zveřejní $F_{v_B}^{-1}(x)$. Díky „zadním vrátkům“ umí funkci F_{v_B} invertovat. Každý si pak může zprávu x přečíst, když na ni znovu použije F_{v_B}. A protože Bob je jediný, kdo F_{v_B} umí invertovat, musel zprávu zveřejnit on. To je princip digitálního podpisu.

V šifrovacím systému (kryptosystému) s veřejným klíčem už si účastníci nemusí předávat informace o klíči, každý si vygeneruje svůj pár klíčů, soukromý si uschová

a veřejný dá ostatním volně k dispozici.

Používá se např. RSA funkce. V tomto případě jako veřejný klíč slouží dvojice (n, e), jako soukromý klíč číslo d. Číslo n je součin dvou velkých prvočísel, e je zvoleno tak, aby bylo nesoudělné s $\varphi(n)$. Pro d platí, že $ed \equiv 1 \ mod \ \varphi(n)$. Zašifrovat zprávu z znamená vypočítat z^e. Rozluštění pak probíhá $(z^e)^d \equiv z \ mod \ n$ (viz kap. 2.2).

3. Uvedení do problému digitálních peněz

Jak už jsme řekli, digitální peníze mají fungovat stejně jako papírové či kovové peníze.

Základní jednotkou pro nás bude (elektronická) **mince** . Mincí v případě digitálních peněz rozumíme řetězec znaků, třeba i prázdný, naslepo podepsaný bankou (viz kap. 5.1).

Při transakcích digitálními penězi budou vystupovat tři strany — **zákazník** (Alice), **obchodník** a **banka** (většinou budeme předpokládat, že zákazník i obchodník mají účet u téže banky). Transakce, stejně jako při placení obyčejnými penězi, probíhá ve třech krocích:

1. Výběr - Zákazník vybere peníze ze svého účtu v bance.
2. Nákup - Zákazník zaplatí penězi v obchodě.
3. Vložení - Obchodník peníze vloží na svůj účet v bance.

Druhý a třetí krok mohou proběhnout najednou - pak mluvíme o systému s okamžitou kontrolou *(online system)*. Jinak se jedná o systém bez okamžité kontroly *(offline system)*, kdy všechny kroky probíhají odděleně.

V některých systémech musí vlastním transakcím předcházet určitá iniciace, analogická otevření bankovního účtu v běžných penězích.

Každý krok při transakci zahrnuje komunikaci mezi dvěma stranami. Záznam této komunikace nazýváme **protokol**.

Každá strana transakce má své požadavky. Zákazník chce mít možnost nakupovat anonymně, zejména aby banka neměla přehled, kde za své peníze nakupuje. Obchodník chce mít jistotu, že peníze, které od zákazníka přijal, od něj přijme banka a připíše mu je na účet. Banka zas chce mít jistotu, že množství peněz, které vydala, bude stejné jako množství peněz, které přijme k uložení, tj. že jednu minci není možné uložit dvakrát.

Zákazníci i obchodníci mají v bance své účty, ty samozřejmě anonymní nejsou. Chceme ale, aby zůstala utajena cesta peněz, které byly vybrány z jednoho účtu a později uloženy na jiný, čili aby u peněz, které ukládáme, nebylo možné zjistit, kdo je předtím vybral.

Pod pojem digitální peníze nespadají všechny systémy, které umožňují přesuny peněz bez fyzické přítomnosti bankovek čí mincí. Ovládání běžného účtu přes internet, které některé banky umožňují, nebo kreditní karty zde zkoumat nebudeme, protože nesplňují základní podmínku — ochranu soukromí zákazníka. Při takových způsobech placení má banka stále přehled o tom, kolik peněz zákazník kde utratil, obchodník zase získá mnoho informací o zákazníkově účtu a vůbec je celý systém špatně chráněn proti zneužití.

Pozn.: Mluvíme-li zde o tom, že některá strana nemůže získat nějakou informaci,

myslíme tím, že žádný algoritmus, kterým ji bude zjišťovat, nedává správnou odpověď s nezanedbatelně větší pravděpodobností než náhodný výběr.

4. Vlastnosti digitálních platebních systémů

Ideální systém digitálních peněz, jak jej formulovali Okamoto a Ohta v [4]:

1. Chrání soukromí uživatelů, nelze zpětně propojit zákazníka s jeho nákupy. (anonymita)
2. Je bezpečný, peníze není možné kopírovat nebo používat víckrát. (bezpečnost)
3. Nezávisí na fyzickém umístění, peníze lze přesouvat přes internet. (nezávislost)
4. Obchod nemusí mít spojení s bankou. (bez okamžité kontroly)
5. Peníze je možné přesouvat mezi uživateli. (směnitelnost)
6. Elektronické mince daných hodnot lze rozměňovat. (rozměnitelnost)

Ne každý systém, o kterém bude řeč, splňuje všechny výše uvedené podmínky. Nicméně podmínky 1 a 2, tj. anonymita a bezpečnost, jsou nutné, jinak se nedá mluvit o systému digitálních peněz.

Anonymita *(privacy)*

Podmínka anonymity znamená, že ani obchodník ani banka (a ani kdyby se spojili) nemají šanci zjistit zákazníkovu totožnost. Tato podmínka je nejčastějším důvodem, proč lidé nemají k digitálním penězům důvěru — anonymní platby dokonale kryjí zločince a podvodníky. Ovšem mechanismy, které zaručují uživatelům anonymitu (jsou-li dobře vymyšleny), fungují pouze tehdy, když uživatel jedná poctivě.

N. Ferguson v [5] uvádí silnější podmínku anonymity, kterou autoři [4] nebrali v úvahu, ale ve skutečnosti je třeba s ní počítat. Je třeba také zaručit, aby banka nemohla rozeznat, že dvě různé platby uskutečnil tentýž zákazník. Může se totiž stát, že při jedné transakci vyjde totožnost zákazníka najevo, a pak by bylo možné vystopovat všechny jeho platby.

Bezpečnost *(security)*

Platební systém je bezpečný, pokud žádná strana transakce nemůže podvádět. Speciálně u digitálních peněz to znamená, že uživatelé nemohou vyrábět falešné mince, případně z jedné vybrané mince jich získat víc nebo použít jednu minci víckrát. Proti falšování mincí se banka chrání svým digitálním podpisem a proti dvojímu užití speciálním zašifrováním zákazníkovy totožnosti na minci. Použije-li zákazník minci jednou, zůstává v anonymitě. Použije-li ji podruhé, už může být identifikován.

Systém musí být také bezpečný proti útokům zvenčí, např. když banka vydává peníze, musí si být jistá, že je dostane ten, kdo by měl, a ne někdo, kdo se za něj vydává. Bezpečnost vlastně znamená, že žádná strana nemusí spoléhat na poctivé jednání jiné strany. Základní heslo je, nevěřit nikomu.

Nezávislost *(hardware independence)*

Nezávislé platební systémy mohou peníze přesouvat bez problémů po internetu. K zajištění bezpečnosti nepotřebují speciální zařízení jako např. čipové karty, používají pouze kryptografické prostředky pro zabezpečení. Závislé systémy mohou

používat i fyzické prostředky, např. různé ochranné prvky na kartách, které brání pozměňování informací na kartě a jejichž poškozením se karta znehodnotí (tzv. *tamper-resistant devices*). Závislé systémy jsou lépe zabezpečeny a v praxi se také užívají častěji.

Okamžitá kontrola *(offline/online payment)*

V systému s okamžitou kontrolou musí každá platba, kterou zákazník provede, projít nejprve přes banku, která zkontroluje, že použitá elektronická mince je v pořádku. Takový systém hned odhalí, když se některý uživatel pokusí podvádět, ale jeho provoz je velmi nákladný, protože vyžaduje neustálé spojení obchodníka s bankou a v praxi je proto téměř nepoužitelný. Výhodnější je systém bez okamžité kontroly, ve kterém se peníze dostanou do banky až ve chvíli, kdy je obchodník chce vložit na svůj účet. Takový systém odhalí případný pokus o podvod až později, a musí tedy v sobě mít mechanismy, které podvodníka najdou. Buď musí existovat pro obchodníka nějaká možnost, jak ověřit pravost mince na místě, anebo musí být banka schopna rozeznat při pokusu o vložení falešné mince, kdo podváděl - zda zákazník nebo obchodník. Banka také musí mít schopnost odhalit totožnost podvodníka, podmínka anonymity má přece jen určité hranice.

Směnitelnost *(transferability)*

V systému s touto vlastností se nemusí mince po každém provedeném obchodu vracet zpět do banky, ale mohou volně měnit majitele. Tím se digitální peníze ještě více přibližují skutečným penězům. Takový systém je teoreticky možný, ale má jedno velké nebezpečí - čím déle bude elektronická mince v oběhu, tím větší šanci bude mít ten, kdo se pokouší prolomit její zabezpečení. Systémy, které umožňují převádění peněz mezi uživateli, ke svým mincím proto často přidávají „datum trvanlivosti“, po kterém se mince musí vrátit do banky.

Rozměnitelnost *(divisibility)*

Rozměnitelnost znamená, že mince vydaná s určitou hodnotou se může dělit na více menších mincí, které v součtu dávají původní hodnotu. Pokud systém tuto vlastnost nemá, musí si zákazník vždy vybrat minci přesně takové hodnoty, jakou má zamýšlený nákup, nebo vybrat více mincí různých hodnot a z nich nasčítat výslednou cenu, tím ovšem velmi naroste počet potřebných transakcí. Mimochodem, papírové a kovové peníze tuto vlastnost také nemají. Mám-li stokorunu, nemůžu z ní polovinu ustřihnout a zaplatit jí 50 Kč.

Další otázka je, kam až se mince může dělit? Bude-li možné elektronicky zaplatit částku v setinách haléřů, co se stane, když zákazník utratí 1 setinu a zbylých 99 si bude chtít uložit v bance (nebo dokonce vybrat v hotovosti)?

Autoři [6] uvádějí ještě další dvě vlastnosti, které mohou být důležité pro uvedení digitálních peněz do praxe. Za prvé je to určitá **pružnost** *(scalability)*, neboli schopnost zvládnout i větší nárůst počtu uživatelů v systému, a tím i větší počet transakcí. Systém, který má být pružný, obvykle nemá jen jeden centrální server (banku), ale běží na více serverech najednou. Systém s okamžitou kontrolou moc pružný být nemůže, protože musí všechny transakce jednu po druhé ověřovat, což jeho kapacitu velmi omezuje. Druhou vlastností je **všeobecná přijatelnost** *(acceptability)*, neboli možnost vkládat peníze i do jiné banky než do té, která je vydala. Systém, který by předpokládal větší množství uživatelů, by se bez této vlastnosti neobešel.

Autoři [7] se zaměřili ještě na jednu vlastnost, která je v praktickém systému velmi důležitá, na **úspornost** *(efficiency)*. Systém, který zaručuje naprostou bezpečnost, ale vyžaduje při transakcích příliš náročné výpočty a příliš velké přenosy dat, je v praxi nepoužitelný. Příliš náročným výpočtem zde rozumíme např. funkce polynomiální v délce vstupu, ale s vysokým řádem polynomu.

5. Metody a procedury

5.1 Podpis naslepo *(blind signature)*

Digitální platební systémy používají podpisy naslepo, aby splnily základní požadavek anonymity uživatelů. Cílem je získat na určitý dokument podpis určité osoby, aniž by ta osoba viděla, co podepisuje. Po zveřejnění dokumentu pak bude schopna potvrdit pravost svého podpisu, ale nebude schopna určit, kdy a pro koho dokument podepsala.

Digitální podpis založený na RSA: [2]

Alice (A) chce od banky (B) podpis zprávy z. Dvojice (n, e) je veřejný klíč B, d je její soukromý klíč. Platí, že $ed \equiv 1 \ mod \ \varphi(n)$.

1. A zvolí náhodné číslo k takové, že $0 \leq k \leq n-1$ a $\gcd(n,k) = 1$ (k je „zamlžovací faktor“).
2. A zamlží zprávu z, tj. vypočítá $z^* = zk^e \ mod \ n$ a z^* pošle B.
3. B zprávu podepíše, tj. vypočítá $p^* = (z^*)^d \ mod \ n$ a p^* pošle A.
4. A teď může zprávu zveřejnit, tj. vypočítat $p = k^{-1}p^* \ mod \ n$. Platí, že $p = z^d$, jinými slovy p je původní dokument s novým podpisem banky.

Výpočtem $p^e \ mod \ n$ se může kdokoli přesvědčit o pravosti podpisu. Pracujeme totiž s RSA funkcí, a tedy $z^{ed} = z$ a $k^{ed} = k$ (předpokládáme, že z i k jsou menší než n).

Digitální podpis užívající diskrétní logaritmus: [3]

Mějme dvě prvočísla p a q taková, že p dělí $q-1$. Vezmeme podgrupu G řádu q v Z_p^*, g bude její generátor. Dále vezmeme bezkolizní hašovací funkci $h : \{0,1\}^* \rightarrow Z_q$. Soukromý klíč podepisující osoby (B) bude náhodně zvolené $s \in \{0, \ldots, q-1\}$, její veřejný klíč bude v, kde $v = g^s$. Veřejně známá jsou i čísla p, q a g. Stejně jako v předchozím případě A chce získat od B podpis zprávy z.

Vyjdeme z tzv. *Schnorrova protokolu*:

1. A pošle B zprávu z.
2. B náhodně zvolí $r \in \{0, \ldots, q-1\}$ a spočítá $a = g^r \ mod \ p$.
3. Dále B spočítá $c = h(z \| a)$ a $b = r - cs \ mod \ q$ a pošle je A. Dvojice (c,b) je podpis zprávy z.

---

[2] podle [3], str. 475

[3] podle [2], str. 93–97

K ověření pravosti podpisu A vypočítá $x = g^b v^c \ mod \ p$ a vyzkouší, zda $c = h(z\|x)$.

Bez použití hašovací funkce h by A musel c volit náhodně, tímto způsobem se komunikace o jeden krok zkrátí.

Jak z tohoto protokolu pro obyčejný digitální podpis vytvořit protokol pro podpis naslepo:

1. B náhodně zvolí $r \in \{0, \ldots, q-1\}$, spočítá $a^* = g^r \ mod \ p$ a pošle jej A.
2. A zvolí $u, w, y \in \{0, \ldots, q-1\}$, $u \neq 0$, spočítá $a = (a^*)^u g^y v^w \ mod \ p$, $c = h(z\|a)$ a $c^* = (c-w)u^{-1}$ a pak c^* pošle B. (zde slouží u, w, y jako „zamlžovací faktory“)
3. B spočítá $b^* = r - c^* s \ mod \ q$ a pošle jej A.
4. A ověří, že $a^* = g^{b^*} v^{c^*} \ mod \ p$, spočítá $b = ub^* + y \ mod \ q$ a získá tak podpis (c, b) zprávy z.

Princip podpisu naslepo tedy spočívá v tom, že zpráva se nejprve zamlží pomocí náhodného faktoru, buď se tímto faktorem násobí nebo se na něj umocní. Zamlžená zpráva je nečitelná pro každého, kdo příslušný faktor nezná, jediný způsob, jak by ji mohl rozluštit, je, že by vyzkoušel všechny možné hodnoty faktoru, až by našel nějakou smysluplnou zprávu. Pokud se faktor vybírá z dostatečně velké množiny, je šance úspěchu tohoto snažení malá.

5.2 Sdílení tajemství *(secret sharing)*

Někdy může být vhodné rozdělit tajný klíč mezi více osob. Představme si třeba truhlu s pokladem zamčenou čtyřmi zámky a od každého zámku má klíč někdo jiný. Nikdo z nich nemůže sám truhlu otevřít, ale všichni dohromady mohou. [4]

Podobnou metodu použijeme u digitálních mincí, aby na nich byla umístěna totožnost zákazníka, ale aby při poctivém užívání mincí zůstala skrytá. Informace o zákazníkovi se totiž rozdělí na dvě části, z nichž žádná sama o sobě nic neříká. Teprv z obou dohromady můžeme zákazníkovi přijít na jméno.

Jedna možnost, jak informaci rozdělit, je pomocí binárního sčítání. Mějme řetězec bitů I, který identifikuje zákazníka. Přičteme-li k němu náhodný řetězec R stejné délky, dostaneme $I \oplus R = P$. Dvojice (P, R) představuje dvojici klíčů. Žádný z řetězců sám o sobě nedává žádnou informaci o I. Sečtou-li se dohromady, dostaneme I.

Jiná možnost je použít I jako parametr přímky. Jeden bod přímky $kx + I$, kde k je tajné pevně dané číslo, opět nedává možnost odhalit I. Dva takové body (s různými hodnotami x) už k určení I stačí.

5.3 Bitový závazek *bit commitment*

Bitový závazek je něco jako pečeť, která zaručí, že došlá zpráva je neporušená a nepozměněná, přesně v té podobě, jak byla napsána. Jako příklad můžeme vzít

[4] podrobnosti ke sdílení tajemství např. v [3], str. 524–526

takovou sázku: Alice chce předpovědět nějaký výsledek, ale nechce jej zatím zveřejnit. Bob ale zase chce mít jistotu, že Alice svou předpověď nezmění podle toho, jak věc dopadne. K tomu jim poslouží bitový závazek. Uvedeme zde dva příklady protokolu pro bitový závazek [5]

Bitový závazek pomocí symetrické kryptografie

1. Bob vygeneruje náhodný řetězec bitů R a pošle jej Alici.
2. Alice k řetězci R připojí svůj závazek b(tj. bit nebo posloupnost bitů, které chce předpovědět), celou tuto zprávu zašifruje náhodným klíčem K a pošle Bobovi.

Bob teď zprávu rozluštit nemůže, protože nezná K, a zatím tedy neví, co Alice předpověděla. Až nadejde vhodná doba, Alice mu pošle K, takže si její závazek bude moci přečíst a zároveň zkontroluje R. Bobův náhodný řetězec R je nutný k tomu, aby Alice nemohla poslat zašifrovanou zprávu s bitovým závazkem b a pak k ní zkoušet klíče tak dlouho, než by našla takový, který dává jinou hodnotu b než tu, ke které se zavázala původně. Takový klíč by jistě našla, ale pokud zpráva musí obsahovat i R, musel by takový klíč R zachovat a šance nalezení takového klíče je nepatrná.

Bitový závazek pomocí jednosměrné funkce

1. Alice vygeneruje dva náhodné řetězce R_1 a R_2.
2. Alice vytvoří zprávu obsahující oba řetězce a její závazek, bit b, a použije na ni jednosměrnou (bezkolizní) funkci h. Výsledek $h(R_1\|R_2\|b)$ spolu s jedním z řetězců, např. R_1, pošle Bobovi.

Bob se teď nemůže dozvědět Alicinu předpověď díky tomu, že h je jednosměrná. Až bude Alice chtít svou předpověď ukázat, pošle Bobovi původní zprávu $(R_1\|R_2\|b)$, Bob na ni vyzkouší h a porovná výsledek s tím, který dostal od Alice v kroku 2. Pokud výsledek i řetězec R_1 souhlasí, bude Alici věřit.

Výhodou tohoto protokolu je, že Bob nemusí nic posílat, dodržení závazku zde zaručuje bezkoliznost funkce h. Alice nenajde zprávu $(R_1\|R_2'\|b')$ takovou, že $h(R_1\|R_2\|b) = h(R_1\|R_2'\|b')$ a $b \neq b'$.

Na řetězci R_2 při komunikaci zdánlivě nezávisí, ale nutný tu je, Alice si musí nechat část zprávy, kterou Bobovi nesdělí. Kdyby Bob znal celý řetězec R, k němuž Alice připojuje závazek, prostě by vyzkoušel $h(R\|0)$ a $h(R\|1)$ a porovnal s tím, co mu poslala Alice.

5.4 Jeden půlí, druhý dělí *(cut and choose)*

Princip je stejný, jako když si děti mají spravedlivě rozdělit koláč — jeden jej rozpůlí a druhý si vybere, kterou polovinu chce. První tedy ve svém zájmu dělí spravedlivě. Nám tato metoda poslouží při vydávání elektronických mincí: [6]

[5] podle [8], str. 86–88

[6] podle [6]

Výběr

1. Alice připraví n mincí stejné hodnoty. Každá bude obsahovat sériové číslo, náhodný řetězec dostatečné délky, který jej odliší od všech ostatních mincí (i na papírových penězích jsou takové řetězce). Dále bude každá obsahovat k identifikačních řetězců, které ponesou informaci o Alicině totožnosti. Alice každý rozdělí na dvě části, jejichž součet bude daný řetězec (viz sdílení tajemství, kap. 5.2), a s použitím bitového závazku (kap. 5.3) zakryje každou polovinu zvlášť.
2. Alice všechny vytvořené mince „zamlží“ (viz protokol na digitální podpis, kap. 5.1) a pošle do banky.
3. Banka vybere $\frac{k}{2}$ mincí a požádá Alici, aby je otevřela, tj. odstranila „zamlžovací faktor“. Banka zkontroluje sériová čísla a požadovanou částku. Také Alici požádá, aby odkryla bitový závazek, a zkontroluje identifikační řetězce.
4. Banka zkontroluje správnost mincí — jejich hodnotu, jednoznačnost sériových čísel, Aliciny identifikační řetězce.
5. Je-li vše v pořádku, banka zbylé mince spojí dohromady, podepíše naslepo jako jedinou minci a odčerpá peníze z Alicina účtu.
6. Alice teď může odstranit zamlžení a minci použít k nákupu.

Nákup

1. Alice pošle obchodníkovi příslušnou minci.
2. Obchodník si ověří podpis banky.
3. Obchodník Alici požádá, aby z každého rozpůleného identifikačního řetězce jednu polovinu odkryla. Dá jí náhodnou posloupnost bitů, kde 1 odpovídá levé polovině a 0 pravé. Alice na příslušné polovině odstraní bitový závazek.

Vložení

1. Obchodník pošle minci do banky, spolu s informací o odkrytých identifikačních řetězcích.
2. Banka ověří podpis a zkontroluje v databázi, zda mince se stejným sériovým číslem už nebyla použita. Pokud ne, uloží si její číslo i informace na ní obsažené do databáze a obchodníkovi na účet přidá příslušnou částku.
3. Pokud ovšem mince už použita byla, banka ji nepřijme. Porovná odkryté poloviny identifikačních řetězců s těmi, které jsou v databázi. Je-li posloupnost naprosto stejná, tj. na obou mincích jsou to tytéž poloviny, padá podezření na obchodníka, který se pokusil minci okopírovat. Liší-li se posloupnost aspoň na jednom místě, je podezřelou Alice. Navíc na místě, kde se oba řetězce liší, lze odhalit Alicinu totožnost.

5.5 Jednoduchá mince *(single term coin)*

Princip jednoduché mince zavedl N. Ferguson v [5]. Je úspornější než dříve užívaný princip „jeden půlí, druhý dělí“, kde bylo potřeba vytvořit velké množství příkazů, z nichž polovina byla zbytečná. Zde se mince vytváří jen jedna.

K zajištení anonymity se používá protokol pro digitální podpis naslepo pomocí RSA, ale s dalším náhodným prvkem (tzv. nahodilý podpis —*randomized blind*

signature). Tento podpis vyžaduje zamlžovací faktory jak pro násobení, tak pro umocňování. Důležité je, že Alice získá podpis určitého čísla, toto číslo si ale nemůže sama vybrat, musí být zcela náhodné a banka, která jej podepisuje, musí být přesvědčena, že je náhodné. Samozřejmě banka nesmí vědět, jaký podpis Alici vydala.

Nahodilý podpis

1. Alice nejprve náhodně zvolí x_1 a zamlžovací faktory α a σ. Vypočítá $\alpha^e x_1 g^\sigma$, kde e je veřejný klíč banky a g je (známý) prvek $\mathbb{Z}_n^*$. Výsledek pošle do banky.
2. Banka náhodně zvolí x_2 a pošle jej Alici.
3. Alice odpoví $f(x_1, x_2) - \sigma$, kde f je jednosměrná funkce $\mathbb{Z}_n^* \to \mathbb{Z}_e$.
4. Banka vynásobí $\alpha^e x_1 g^\sigma$ s x_2 a $g^{f(x_1,x_2)-\sigma}$, aby dostala $\alpha^e x_1 x_2 g^{f(x_1,x_2)}$. Z tohoto čísla vezme e-tou odmocninu a pošle ji Alici.
5. Alice výsledek vydělí faktorem α a dostane dvojici $(x, (xg^f(x))^{\frac{1}{e}})$, kde $x = x_1 x_2$.

Pozn. Exponenty se vždy počítají modulo e. Na konci pak Alice vynásobí výsledný podpis vhodnou mocninou g, aby se zbavila případného přebytečného faktoru g^e.

Základní princip jednoduchých mincí (podrobněji se k němu vrátíme při popisu konkrétních platebních systémů):

Mince je zde reprezentována trojicí čísel X, Y, Z, kde $X = f_x(x)$, $Y = f_y(y)$ a $Z = f_z(z)$, kde f_x, f_y a f_z jsou vhodné jednosměrné funkce. Vstupy x, y, z vzniknou společným úsilím zákaznice Alice a banky, trojnásobným provedením protokolu pro nahodilý podpis. Alice nakonec kromě čísel X, Y, Z dostane od banky dva podpisy $(Z^k X)^{\frac{1}{e}}$ a $(Z^I Y)^{\frac{1}{e}}$, kde k je náhodně zvolený faktor, I identifikuje Alici a e je veřejný klíč banky.

Při placení v obchodě Alice pošle obchodníkovi x, y, z a obchodník pošle Alici náhodné číslo t. Alice vypočítá číslo $r = kt + I \ (mod\ e)$ a podpis $(Z^r X^t Y)^{\frac{1}{e}}$, podpis i r pošle obchodníkovi.

Když bude obchodník chtít peníze uložit, pošle do banky x, y, z, t, r a podpis. Banka ověří pravost mince a uloží ji na obchodníkův účet. Pokud už byla jednou tatáž mince uložena, obdobným způsobem jako předtím se zjistí, kdo podváděl — je-li číslo t stejné, byl to obchodník. Je-li jiné, byla to Alice, a protože dvě různá t dávají dva různé body na přímce $kt + I$, dá se z nich zjistit Alicina totožnost I.

5.6 Binární strom *(binary tree)*

Struktura binárního stromu se dá použít k vytvoření mincí, které jde dělit na menší části, není třeba je utratit celé najednou. Takto jej používají autoři [4].

Celý strom má danou hodnotu n, vrchol v l-té hladině odpovídá hodnotě $\frac{n}{2^{l-1}}$.

Při placení je třeba dodržovat dvě pravidla:

1. jakmile se vrchol použije, už se nesmí použít žádný z jeho předchůdců ani následníků.
2. žádný vrchol nelze použít víckrát.

Minci ve skutečnosti tvoří dva shodné stromy s odpovídajícími vrcholy, v jednom jsou uloženy hodnoty peněz, ve druhém jsou uloženy informace o tom, které vrcholy ještě lze použít.

5.7 Rekurzivní hašování *(continuous hash)*

Tato metoda opět slouží k vytvoření mincí dělitelných na menší části. Poprvé ji v systému digitálních peněz použil Mao v [9]. V jednoduchosti popíšeme, jak tato metoda funguje (podrobněji v kap. 6.3):

Výběr

Alice si vytvoří řetězec k mincí $C_0, C_1, \ldots, C_{k-1}$ pomocí rekurzivního použití hašovací funkce f:

$$C_i = f(C_{i+1}) \text{ pro } i = 0, 1, \ldots, k-1$$

C_k je náhodně zvolené číslo, to ještě není mince. Vrchní minci C_0 nechá naslepo podepsat bankou. Díky jednosměrnosti funkce f jsou tak podepsány všechny mince. Klíč, který banka k podpisu používá, je svázán s hodnotou k, pro jiné hodnoty jsou jiné klíče.

Nákup

Výhoda uvedené struktury mincí je v tom, že Alice nemusí utratit všechny mince najednou u téhož obchodníka, ale může jich část použít v jednom obchodě a část v jiném. Chce-li utratit j mincí $(j < k)$ u jednoho obchodníka, pošle mu C_j a podepsanou C_0. Rekurzivním použitím funkce f na C_j obchodník dostane C_0 a ověří podpis banky. Minci C_j pak podepíše. Chce-li pak Alice nakoupit u dalšího obchodníka za i mincí $(i < k-j)$, pošle mu C_0, C_{j+i} a i. Obchodník po i krocích ověří podpis předchozího obchodníka a po dalších j krocích podpis banky.

6. Příklady digitálních platebních systémů

S prvním digitálním platebním systémem přišli D.Chaum, A.Fiat a M.Naor [10] v roce 1990 (kap. 6.1). Tento systém splňuje základní podmínky anonymity a bezpečnosti, je nezávislý a nevyžaduje okamžitou kontrolu. V roce 1992 předvedli T.Okamoto a K.Ohta [4] jiný systém, který navíc umožňuje směňování mincí mezi uživateli a rozměňování, k tomu mu slouží struktura mincí ve tvaru binárního stromu (za tyto výhody ale platí zeslabením anonymity). Oba tyto systémy používají metodu „jeden půlí, druhý dělí“, proto nejsou příliš efektivní.

Systém M.Franklina a M.Yunga [7] z roku 1993 tuto metodu používá jen v přípravné fázi protokolu, při samotném výběru mincí už ne, z tohoto hlediska je tedy úspornější. Objem přenášených dat je ale stále velký, protože každá mince obsahuje mnoho informací — kvůli zachování anonymity a bezpečnosti obsahuje mnoho identifikačních řetězců zákazníka, které určují směrnice přímek. Při nákupu

se pak z každé přímky odkryje jeden bod a díky tomu lze při dvojím užití téže mince odhalit podvodníka. Na podobném principu funguje i systém tzv. jednoduchých mincí popsaný N.Fergusonem [5], také z roku 1993 (kap. 6.2), ale z hlediska objemu přenášených dat je mnohem úspornější.

W.Mao [9] popsal v roce 1996 nový systém (kap. 6.3) na principu rekurzivního hašování, který kromě toho, že umožňuje vydávání rozměnitelných mincí, navíc obsahuje sám v sobě opatření proti dvojímu užití jedné mince. Tím se liší od předchozích systémů, které mohly podvodníka odhalit, ale musely se spolehnout na autoritu zvenku, tj. soud. V Maově systému se při odhalení podvodníka odhalí přímo jeho soukromý klíč, který pak bude nepoužitelný. Jelikož všechny popsané systémy jsou určeny spíš pro menší platby, nemůže podvodník mnoho získat, a bude-li vydání nového klíče drahé, podvádění se nevyplatí.

Všechny systémy, o kterých jsme zatím mluvili, fungují nezávisle na hardwarovém vybavení, peníze jsou prostě data v počítači jako každá jiná. Zabezpečení takových dat je ale náročné a často nejisté, proto některé digitální platební systémy používají speciální zařízení, tzv. **smartcards**, které fungují na podobném principu jako např. telefonní karty — uživatel si je „nabije“ a pak jimi může platit. Součástí takové karty je mechanismus zabraňující poškození či pozměnění dat, jakási plomba (*tamper-resistant device*), která chrání tajné informace o zákazníkovi a jeho platbách a zabraňuje zneužití. S takovým zařízením pracuje např. systém navržený S. Brandsem [11] roku 1995. Brandsův systém stejně jako Maův používá jako jednosměrnou funkci k utajení informací diskrétní logaritmus, dřívější systémy používaly RSA funkci.

V tomto přehledu jistě nejsou uvedeny všechny digitální platební systémy, které byly kdy vymyšleny, ale snad jsou tu alespoň všechny základní typy, které se v dalších systémech pouze obměňují.

Tři z těchto typů si popíšeme podrobněji:

6.1 Chaum, Fiat, Naor, 1990 *(Untraceable Electronic Cash)*

Vůbec první elektronický platební systém. Tento systém se snaží co nejvíce chránit soukromí uživatelů (*untraceable* = nevystopovatelný, tzn. nelze vystopovat spojení mezi zákazníkem a jeho nákupy). Cílem jeho autorů je, aby ochrana byla nepodmíněná, tedy aby nezávisela pouze na výpočetní složitosti. Pokud uživatel jedná poctivě, neměla by žádným způsobem jít odhalit jeho totožnost.

Peníze jsou zde vytvořeny pomocí digitálního podpisu založeného na RSA (viz kap. 5.1).

Jak tento platební systém funguje:

Banka nejprve zveřejní číslo n, které bude sloužit jako modul pro RSA. Faktorizace n zůstane utajena. Banka také zvolí sudý bezpečnostní parametr k.

Budeme potřebovat dvě bezkolizní funkce $f(x, y)$ a $g(x, y)$ (tj. chceme, aby bylo obtížné nalézt dva vstupy, které dávají stejný výstup). Dále chceme, aby se f „chovala náhodně“, tj. aby odpovědi, které dává, nebylo možné s nezanedbatelnou pravděpodobností uhodnout. Po funkci g chceme, aby při zafixování prvního argumentu byla bijektivní.

Alice má účet s číslem u a k němu přísluší počítadlo v.

Všechny mince mají stejnou danou hodnotu, řekněme 1 Kč. Pro jinou hodnotu bude sloužit jiný modul n.

Výběr

Alice chce vybrat 1 Kč.

1. Alice náhodně zvolí a_i, c_i, d_i a r_i menší než n, kde $i \in \{1, \ldots, k\}$. Vytvoří z nich „semínka“ mincí

$$C_i^* = r_i^3 \cdot f(x_i, y_i) \bmod n$$

   kde $x_i = g(a_i, c_i)$ a $y_i = g(a_i \oplus (u \| (v+i)), d_i)$.
2. Banka vybere náhodně $\frac{k}{2}$ semínek, množinu vybraných indexů označíme R.
3. Alice pro vybraná semínka ukáže příslušné hodnoty a_i, c_i, d_i, r_i, banka je zkontroluje. Čísla u a v banka zná.
4. Banka pošle Alici

$$\Pi_{i \notin R} (C_i^*)^{\frac{1}{3}} \bmod n$$

   a strhne jí z účtu 1 Kč. Zároveň zvýší počítadlo v o hodnotu k.
5. Alice teď může odstranit zamlžení r_i a získá minci

$$C = \Pi_{i \notin R} (f(x_i, y_i))^{\frac{1}{3}} \bmod n$$

   Jednotlivé činitele v minci si přeznačí, očísluje je $1, \ldots, \frac{k}{2}$.

Nákup

Alice platí 1 Kč.

1. Alice pošle C obchodníkovi.
2. Obchodník náhodně zvolí posloupnost bitů $b_1, b_2, \ldots, b_{\frac{k}{2}}$.
3. Alice odpoví pro každé $i \in \{1, \ldots, \frac{k}{2}\}$:
   a) je-li $b_i = 1$, pošle obchodníkovi a_i, c_i a y_i
   b) je-li $b_i = 0$, pošle $x_i, a_i \oplus (u \| (v+i))$ a d_i
4. Obchodník ověří, že C má správný tvar a že Aliciny odpovědi souhlasí s C.

Vložení

Obchodník pošle do banky C a všechny informace, které dostal od Alice. Banka je prověří a pokud jsou v pořádku, připíše mu peníze, a všechny mince spolu s příslušnými informacemi uloží do databáze.

Pokud Alice nepodvádí, může platit zcela anonymně.

Proti dvojímu užití též mince je systém chráněn. Pokud Alice minci okopíruje a použije dvakrát, každý obchodník jí dá jinou posloupnost a aspoň na jednom místě se objeví obě poloviny jejího identifikačního řetězce. Pravděpodobnost, že by dostala dvakrát tutéž posloupnost, je $\frac{1}{2^k}$, a je-li k dost velké, téměř nulová. Alice by se mohla pokusit minci pozměnit, ale pak by byl podpis banky neplatný a obchodník by takovou minci nepřijal.

Obchodník také nemůže kopírovat minci, podruhé by stejnou minci neuložil. Nemůže to ani svést na Alici, protože nemůže odkrýt její identifikační řetězce.

Záškodník Záviš ale podvádět může. Pokud by se mu podařilo zachytit komunikaci mezi Alicí a obchodníkem, mohl by dojít do banky dřív než obchodník

a uložit si peníze sám. Banka nic nepozná a až přijde obchodník, bude vypadat jako podvodník, protože stejné mince už budou uloženy. Nebo kdyby Záviš ukradl (tj. okopíroval) peníze Alici, může je klidně utratit a Alice pak bude vypadat jako podvodník. S touto vadou se nedá nic dělat, pokud chceme anonymní platební systém. Jediná možnost tedy je (jako u obyčejných peněz), dávat si na své peníze pozor.

Pokud by se Alice domluvila se dvěma obchodníky, zaplatila jim oběma stejnou mincí a oba by jí dali stejnou posloupnost b_i, banka by sice poznala, že došlo k podvodu, ale nevěděla by, kdo podváděl. Navíc, oba se mohou bránit, že to byla náhoda, stát se to může, i když je to málo pravděpodobné. Takovému spiknutí by se dalo zabránit, kdyby posloupnost pro každého obchodníka byla daná a od jiného by se lišila na dosti místech. Aby nemohla Alice dvakrát použít tutéž minci u téhož obchodníka, měl by každý obchodník svou databázi použitých mincí nebo by se část posloupnosti b_i dál určovala náhodně.

Aby byla Alice chráněna i proti falešnému obvinění ze strany banky, stačí protokol trochu pozměnit. Alice bude ještě potřebovat svůj digitální podpis. Místo čísla u, které bylo stejné pro všechny mince, bude používat různá čísla $u_i = u\|z_i'\|z_i''$, kde u je číslo Alicina účtu a zbylá dvě čísla jsou volena náhodně. Spolu se semínky C_i^* Alice pošle bance digitálně podepsaný řetězec

$$g(z_1', z_1'')\|g(z_2', z_2'')\| \ldots g(z_k', z_k'')$$

Banka pak ověří, že každé z $\frac{k}{2}$ semínek, které Alice odkryla, obsahuje správné u_i. Bude-li pak banka umět předložit vzory alespoň $\frac{k}{2}+1$ výrazů $g(z_i', z_i'')$, bude mít v ruce důkaz, že Alice použila minci dvakrát.

Alice by ještě mohla podvádět jiným způsobem — podstrčit v bance k podepsání jinou minci (banka prohlíží jen polovinu připravených mincí). Pravděpodobnost, že se jí to podaří, není úplně malá, ale pokud by se o to pokoušela víckrát, pravděpodobnost úspěchu rychle klesá. Pokud je trest za odhalený podvod tak vysoký, aby vyrovnal Alicin zisk při úspěšném podvodu, Alice to ani nebude zkoušet.

6.2 Ferguson, 1993 *(Single Term Coins)*

Systém jednoduchých mincí zachovává podmínku bezpečnosti a silnější podmínku anonymity. Je to systém bez okamžité kontroly a ke svému fungování nepotřebuje speciální hardwarové vybavení. Směňování mincí mezi uživateli nebo rozměňování na menší hodnoty ale neumožňuje.

K vytvoření mincí nepoužívá princip „jeden půlí, druhý dělí“, vytváří minci rovnou pomocí protokolu pro nahodilý podpis, čímž se velmi snižuje objem dat, která je potřeba během transakcí přenést. Mince je tvořena trojicí čísel X, Y, Z a dvojicí podpisů S_x, S_y na principu RSA. Aby se zabránilo dvojímu užití mince, opět je na mincích uvedena totožnost zákazníka. Nemusí tu ale být v mnoha exemplářích, stačí jednou na každé minci. Základní popis jednoduchých mincí jsme uvedli v kap. 5.5, teď je popíšeme podrobně:

Používáme systém RSA s modulem n, veřejný klíč banky je (n, e). Vydává-li banka mince různých hodnot, používá pro každou hodnotu zvláštní pár klíčů.

I značí identifikaci zákazníka (Alice).

Výběr

Budeme potřebovat tři čísla X, Y, Z ve tvaru

$$X = xg_x^{f(x)}$$
$$Y = yg_y^{f(h_y^y)}$$
$$Z = zg_z^{f(h_z^z)}$$

kde f je jednosměrná funkce, g_x, g_y a g_z jsou prvky dost velkého řádu v $\mathbb{Z}_n^*$, veřejně známé. Čísla h_y, h_z jsou prvky řádu n ze $\mathbb{Z}_p$, kde $p-1$ je násobkem n. Těmito čísly se banka jistí, aby Alice nemohla volit své vstupy zcela libovolně, protože tím by mohla výrazně zasahovat do protokolu a není jasné, jestli jí to nepomůže k nějakému podvodu.

1. Alice náhodně zvolí čísla $x_1, y_1, z_1 \in \mathbb{Z}_n^*$ a k nim zamlžovací faktory $\alpha, \beta, \gamma \in \mathbb{Z}_n^*$ (pro násobení) a $\sigma, \tau, \phi \in \mathbb{Z}_e$ (pro umocnění). Vypočítá $\alpha^e x_1 g_x^\sigma$, $\beta^e y_1 g_y^\tau$, $\gamma^e z_1 g_z^\phi$ a pošle je do banky.
2. Banka náhodně zvolí x_2, y_2, z_2, pošle Alici $x_2, h_y^{y_2}$ a $h_z^{z_2}$.
3. Alice náhodně zvolí číslo $k_1 \in \mathbb{Z}_e^*$ a vypočítá exponenty $e_y = f(h_y^{y_1 y_2}) - \tau$ a $e_z = f(h_z^{z_1 z_2}) - \phi$. Dále vypočítá $x = (x_1 x_2 f_0(e_y, e_z))^{k_1}$, kde f_0 je vhodná jednosměrná funkce. Exponent e_x vypočítá trochu jinak, aby po umocnění podpisu na k_1 dostala správnou hodnotu: $e_x = \frac{1}{k_1} f(x) - \sigma$. Všechny tři exponenty e_x, e_y, e_z pošle do banky.

   Pozn.: všechny výpočty se dělají modulo e a výsledný podpis je třeba vynásobit vhodnými mocninami g_x, g_y a g_z.
4. Banka vypočítá zamlžené hodnoty X, Y, Z:

$$Y^* = \beta^e y_1 g_y^\tau \cdot y_2 \cdot g_y^{e_y}$$
$$Z^* = \gamma^e z_1 g_z^\phi \cdot z_2 \cdot g_z^{e_z}$$
$$X^* = \alpha^e x_1 g_x^\sigma \cdot x_2 \cdot g_x^{e_x} \cdot f_0(e_y, e_z)$$

   Platí tyto vztahy:

$$Y^* = \beta^e Y$$
$$Z^* = \gamma^e Z$$
$$X^* = \alpha^e X^{\frac{1}{k_1}}$$

   Banka dále zvolí náhodné $k_2 \in \mathbb{Z}_e^*$ a pošle Alici čísla k_2, z_2, y_2 a zamlžené podpisy $((Z^*)^{k_2} \cdot X^*)^{\frac{1}{e}}, ((Z^*)^I \cdot Y^*)^{\frac{1}{e}}$.
5. Alice vypočítá $y = y_1 y_2$ a $z = z_1 z_2$. Teď už může získat čísla X, Y, Z jako $xg_x^{f(x)}$, $yg_y^{f(h_y^y)}$ a $zg_z^{f(h_z^z)}$. Podpisy získá výpočtem

$$S_x = \left(((Z^*)^{k_2} \cdot X^*)^{\frac{1}{e}} / \gamma^{k_2} \alpha\right)^{k_1}$$
$$S_y = ((Z^*)^I \cdot Y^*)^{\frac{1}{e}} / \gamma^I \beta$$

Umocnění na k_1 je nutné, protože číslo x bylo zvoleno jako $(x_1x_2)^{k_1}$. Díky tomuto umocnění Alice získá e-tou odmocninu ze Z^kX, kde $k = k_1k_2$.

6. Nakonec Alice ověří, že podpisy jsou v pořádku, výpočtem $S_x^e = Z^kX$ a $S_y^e = Z^IY$.

Funkce f_0 se používá proto, aby Alice nemohla volit e_z, e_y jako funkce a. Tím by mohla zjednodušit některé výrazy a získat podpis jen na Y a Z. Platební systém by tak sice neohrozila, ale přesto není radno nechávat jí tak velkou volnost.

Aby Alice nemohla použít starou minci a zkombinovat ji s novou, exponent k musí být opravdu náhodný. Proto jej nevolí Alice libovolně, ale banka k němu ještě přidává svou část (k vzniká jako k_1k_2). Výslednou hodnotu k ale banka nezná, stejně jako hodnoty x, y, z.

Nákup

Alice má tedy čísla x, y, z (základy mincí), k (náhodný parametr), S_x a S_y (podpisy banky). Těchto 6 čísel spolu s Alicinou identifikací I se užije v protokolu při nákupu.

1. Alice pošle obchodníkovi x, y a z.
2. Obchodník pošle Alici náhodně zvolené t.
3. Alice vypočítá číslo $r = kt + I \pmod e$ a podpis $(Z^rX^tY)^{\frac{1}{e}} = (S_x)^tS_y$, obojí pošle obchodníkovi.
4. Obchodník ověří, že odpovědi v kroku 1 a 3 spolu souhlasí.

Protokol lze zkrátit — místo t poslouží výstup hašovací funkce, jejímž vstupem budou mince a určitý identifikační řetězec obchodníka. Ušetří se tak komunikace mezi obchodníkem a zákazníkem, objem přenášených dat tím ale naroste. V tomto případě si také obchodník musí udržovat databázi mincí, které přijal, aby Alice nemohla použít u něj tutéž minci dvakrát.

Vložení na účet

1. Obchodník pošle do banky x, y, z, t, r a $(Z^rX^tY)^{\frac{1}{e}}$.
2. Banka ověří pravost mince a je-li v pořádku, uloží ji na obchodníkův účet. Pokud už stejná mince byla jednou uložena, porovná hodnotu t — je-li stejná, okopíroval minci obchodník, je-li jiná, okopírovala ji Alice, a protože různá t určují dvě různé hodnoty na přímce $kt + I$, dá se spočítat Alicina totožnost I.

Ještě je tu otázka, jak se může Alice bránit proti křivému nařčení, že užila minci dvakrát. V systému, jak jsme ho popsali, předpokládáme, že banka jedná poctivě. Ve skutečnosti toto předpokládat nemůžeme, tak je třeba systém ještě trochu pozměnit. Za prvé, do řetězce I, který identifikuje Alici, se přidá ještě číslo mince, které ji jednoznačně určuje. Za druhé, Alice přidá svůj digitální podpis k I i ke všem informacím, které si s bankou vymění. Když teď bude banka tvrdit, že Alice použila minci dvakrát, musí předložit zápis celého protokolu výběru a také čísla x, y, z. Pokud Alice nepoužila minci dvakrát, banka nemá žádnou informaci o x, y, z, může tedy pouze hádat (což se jí téměř jistě nepodaří). Pokud tedy Alice předloží jinou trojici x, y, z i s příslušnými zamlžovacími faktory, které odpovídají zápisu protokolu, pak ji banka obvinila neprávem, protože (díky jednosměrným funkcím) Alice nemůže nalézt jinou vhodnou trojici než tu, kterou použila pro vytvoření mincí.

6.3 Mao, 1996 *(Lightweight Micro-Cash)*

Tento systém používá digitální podpis naslepo. Navíc používá ještě Schnorrův protokol založený na diskrétním logaritmu pro obyčejný digitální podpis. Každý uživatel v systému má svůj veřejný a soukromý klíč (v, s), tyto klíče si ale nevytváří sám, je určitá certifikační autorita, která klíče rozděluje a zároveň zveřejňuje seznam platných klíčů. Vydává je ale „naslepo“, při poctivém užívání zůstane majitel klíče utajen. Pokusí-li použít dvakrát tutéž minci, bude odhalen a jeho klíč označen za neplatný.

Schnorrův protokol má totiž tu vlastnost, že použije-li se při podepisování dvou různých zpráv stejné číslo r (viz kap. 5.1), dá se odhalit soukromý klíč s. Pro zprávy $z' \neq z$ máme totiž dva podpisy (c', b') a (c, b), téměř jistě je $c' \neq c$, protože používáme bezkolizní funkci. Máme $b = r - cs \ mod \ q$ a $b' = r - c's \ mod \ q$ a tedy

$$s = \frac{b - b'}{c - c'} \ mod \ q$$

Je tedy potřeba jen zajistit, aby Alice pro jednu minci musela použít jedno r.

Výběr

Alice vytvoří pomocí hašovací funkce f řetězec mincí $C_0, C_1, \ldots, C_{k-1}$, kde

$$C_i = f(C_{i+1}) \text{ pro } i = 0, 1, \ldots, k-1$$

C_k je náhodně zvolené číslo, to není mince. Pro podpis náhodně zvolí číslo r_1 a spočítá k němu $a_1 = g^{r_1} \ mod \ p$ (značení viz Schnorrův protokol v kap. 5.1). Vytvoří výraz

$$V = C_0 \| f(a_1 \| v) \| k$$

a nechá jej naslepo podepsat bankou. Protože f je jednosměrná, má tak vlastně podpis pro všechny mince. Symbolem $V(k)$ budeme značit podepsaný řetězec k mincí, tj. $V(k) = V^{\frac{1}{e}} \ mod \ n$, kde (n, e) je veřejný RSA klíč banky.

Nákup

Alice teď může mince postupně utratit u obchodníků, pro identifikaci si je označíme $Ob_1, \ldots, Ob_m$. Každý obchodník jí vrátí nespotřebované mince spolu se svým podpisem, na banku teď tedy můžeme pohlížet jako na prvního obchodníka Ob_0. Každý z obchodníků má svůj certifikát vydaný certifikační autoritou, potvrzující platnost jeho podpisu, označíme je $Cert_0, \ldots, Cert_m$, Alicin certifikát označíme $Cert_A$. Symbolem $(\ldots)_{Ob_i}$ označíme zprávu podepsanou obchodníkem Ob_i.

První nákup:

Alice chce nakoupit u Ob_1 za i mincí.

1. Alice pošle $V(k), Cert_A, Cert_0$ (certifikát náležející bance), minci C_i, číslo i, svůj veřejný klíč v a svůj podpis platby (c, b), kde

$$c = h(V(k) \| Ob_1 \| (\check{c}as \ n\acute{a}kupu) \| a_1)$$
$$b = (r_1 - cs) \ mod \ q$$

2. Obchodník rekurzivním použitím funkce f na C_i získá C_0 a ověří podpis banky. Také ověří, jestli souhlasí v a $Cert_A$. Zároveň ověří, že příslušná certifikační autorita neprohlásila v za neplatný (tady to vyžaduje určité „on-line“ spojení nebo alespoň pravidelně aktualizovanou databázi). Dále ověří podpis platby, tj. spočítá $x = g^b v^c$ a zjistí, zda $h(V(k)\|Ob_1\|\textit{(čas nákupu)}\|x)$ dává c.

Zbylé mince Alici vrátí následujícím způsobem:

3. Alice zvolí další náhodné číslo r_2 a k němu spočte příslušné a_2 (pro Schnorrův protokol) a pošle obchodníkovi $f(a_2\|v)$ (může to poslat už v prvním kroku zároveň s mincemi).
4. Obchodník podepíše zbylé mince jako řetězec $V(k-i)$

$$V(k-i) = (C_i, f(a_2\|v), k-i)_{Ob_1}$$

a $V(k-i)$ spolu se svým certifikátem $Cert_1$ pošle Alici.

Nákup obecně:

Alice má řetězec $V(k)$, z něhož utratila j mincí, $j < k$, postupně u $m-1$ obchodníků. Má teď v ruce $V(k-j) = (C_j, f(a_m\|v), k-j)_{Ob_{m-1}}$ a certifikát $Cert_{m-1}$. Chce-li nakoupit u Ob_m za dalších i mincí, komunikace proběhne stejně jako u Ob_1, jen navíc ještě Alice pošle $Cert_{m-1}$. Obchodník tak po i-násobném použití f najde podpis Ob_{m-1}, ověří, že byl dodržen protokol, a po dalších j krocích ověří i podpis banky. Alice mu pošle podpis platby (c_m, b_m) a obchodník nazpátek Alici vrátí $V(k-j-i)$.

Vložení

Obchodník Ob_m chce uložit mince, které dostal od Alice.

1. Ob_m pošle do banky tyto informace:
   $V(k), V(k-j), V(k-j-i), Cert_m, Cert_{m-1}$
   a podpis $(Ob_m, (\text{čas nákupu}), c_m, b_m, E_m(a_m))_{Ob_m}$.
   Svůj podpis obchodník používá, aby na něj banka nemohla později přijít s falešným obviněním. Symbol E_m znamená, že zpráva je zašifrována obchodníkovým tajným klíčem, šifruje se proto, aby se nedaly propojit dvě platby od téhož zákazníka.
2. Banka ověří, zda mince $C_j, \ldots, C_{j+i}$ už nebyly uloženy, pokud ne, připíše i mincí obchodníkovi na účet a příslušné informace uloží do databáze.

Pokud už druhá taková mince v databázi je, pozná se, kdo podváděl. Liší-li se čas nákupu, podváděla Alice. Ke dvěma platbám ale musela použít dva podpisy (c, b) a (c', b') a z nich lze zjistit její soukromý klíč s i její totožnost. Kromě trestu za podvod tak také přijde o své klíče.

Pokud je čas nákupu stejný, záleží na tom, který obchodník minci do banky přináší. Je-li to ten samý, který uložil první minci, musel si ji sám okopírovat. Je-li to nějaký jiný, znamená to, že některý z obchodníků vybral z řetězce $V(k)$ i mince, které mu nepatřily. Řekněme, že měl dostat mince $C_j, \ldots, C_{j+i}$. K mincím před C_j nemá Alicin podpis platby, budou-li tedy dva obchodníci tvrdit, že mince je jejich, pravdu má ten, kdo má příslušný podpis. Mince za C_{j+i} také nemůže obchodník Ob_m získat, protože další obchodník Ob_{m+1}, kterému měly patřit, se prokáže podpisem Ob_m na $V(k-j-i)$.

Výhodou tohoto systému je nenáročnost výpočtů, které probíhají během nákupu, a relativně malá velikost mince. Také možnost vyloučení podvodníka ze systému je užitečná. Nevýhodou je množství podpisů, které narůstá s množstvím transakcí — pro každou platbu je potřeba speciální podpis.

6.4 Digitální peníze v praxi

V devadesátých letech vznikly na základě některých popsaných systémů i firmy, které nabízely internetový obchod pomocí anonymních digitálních peněz (např. DigiCash Davida Chauma, později eCash a NetCash, aj.). Žádná z nich ale dlouho neprosperovala a v současnosti už snad ani žádná nefunguje. Jediné dvě, které se dnes podařilo objevit, fungují pod názvy Mondex a Octopuscard a obě používají speciální karty (smartcard). Systém fungující nezávisle na hardwarovém vybavení se objevit nepodařilo.

Důvodem, proč se myšlenka digitálních peněz zatím neujala, může být nedůvěra uživatelů. A nedůvěra je v tomto případě oprávněná, protože otázka zabezpečení digitálních peněz je složitá a stále nevyřešená.

7. Otázka bezpečnosti

Systém považujeme za bezpečný, pokud žádná strana transakce (zákazníci, obchodníci nebo banka) nemůže ošidit jinou stranu a pokud také žádný záškodník zvenčí nemůže systém narušit. Situaci navíc komplikuje podmínka anonymity, která znemožňuje přímou kontrolu peněžních transakcí. Snaha o zajištění anonymity a bezpečnosti často způsobí, že je systém tak složitý a nákladný, že se v praxi nemůže vyplatit (viz systémy užívající princip „jeden půlí, druhý dělí“).

Bezpečnost znamená pro každého uživatele systému něco jiného. V každém případě se ale jedná o utajení určité informace, proto se často používají veřejné a soukromé klíče, spojené s nějakou jednosměrnou funkcí. Soukromý klíč má zákazníkům zajistit anonymní nákupy, bance nepadělatelnost mincí, atd. Základní podmínkou existence výše uvedených platebních systémů je tedy existence jednosměrných (obtížně invertovatelných) funkcí. Za obtížně invertovatelné nyní považujeme třeba RSA funkci nebo diskrétní logaritmus, otázkou ale je, jak dlouho ještě budou „obtížné“. Jakmile někdo přijde na efektivní algoritmus, který je invertuje, bude celá na nich postavená kryptografie zbytečná.

Nicméně, i kdybychom měli opravdu jednosměrnou funkci, ještě nemáme vyhráno. Řada autorů se při posuzování bezpečnosti platebních systémů soustředí pouze na to, že funkce f, na které je systém založen, je obtížně invertovatelná. To je tzv „dokazatelná bezpečnost“ (*provable security*) nebo také „redukční bezpečnost“ (*reductionist security*) — dokážeme, že existuje polynomiální algoritmus, který nějakou funkci g, o které se obecně předpokládá, že je obtížně invertovatelná (např. faktorizace), převede (redukuje) na zkoumanou funkci f. Vidíme tedy, že kdo umí v polynomiálním čase invertovat f, umí v polynomiálním čase invertovat i g, a protože u g je to těžké, je to těžké i u f.

Tento argument je pravdivý, ale ve skutečnosti nestačí, jak ukazuje např. [12]. Jednosměrnost funkce nám zaručí, že ze znalosti veřejného klíče žádný záškod-

ník nezíská soukromý klíč. Ale dále je třeba se podívat, jak probíhá protokol při konkrétní transakci. Uživatelé si vyměňují řadu informací a je třeba zjistit, jestli v některém kroku není možné poslat šikovně pozměněnou zprávu tak, aby to druhá strana nepoznala, a vymámit tak nějakou informaci navíc. Někdy ani není třeba nic pozměňovat a z informací, které si strany během transakce vymění, může leccos „prosáknout“. Článek [13] například ukazuje, jak jeden dokazatelně bezpečný platební systém ve skutečnosti není vůbec bezpečný a co všechno se dá poznat z digitálního podpisu naslepo (a jak jsou tedy všechny systémy, které takový podpis používají, napadnutelné). Autoři tohoto článku navrhují podpis během transakcí vůbec neukazovat, místo podpisu naslepo použít obyčejný podpis a při nákupu a ukládání peněz pak použít důkaz s nulovou znalostí (*zero knowledge proof*), tj. protokol, ve kterém zákazník dokazuje, že podpis má, aniž by jej ukázal.

Jak zjistit, že je systém bezpečný? Těžko. Musíme zkoumat bezpečnost z pohledu všech zúčastněných stran, a i když se nám bude zdát vše v pořádku, může se stát, že záškodník najde nějakou díru. Stává se to, a kryptografické protokoly se tak neustále vyvíjejí — vymyslí se protokol, objeví se díra, ta se zalepí, objeví se jiná, ta se také zalepí, … a tak dál. Na konci tohoto vývoje je možná ideální funkční a bezpečný protokol, ale nevíme, jak jsme ještě daleko a jak poznáme, že už jsme na konci. „Dokazatelná“ bezpečnost nám nestačí. Aby se digitální peníze v praxi rozšířily, bude potřeba najít lepší způsob, jak dokázat skutečnou bezpečnost. Systémy, které v současnosti fungují, informace o protokolech a zabezpečení nezveřejňují. To jim ovšem na důvěryhodnosti nepřidává a bez důvěry zákazníků digitální peníze fungovat nebudou.

8. Literatura

[1] Birkhoff G., MacLane S.: Algebra. ALFA, Bratislava, 1973.

[2] Delfs H., Knebl H. : Introduction to Cryptography. Springer-Verlag, Berlin, 2002.

[3] Menezes A.J., Oorschot P.C., Vanstone S.: Handbook of Applied Cryptography. CRC Press, Boca Raton, 1997.

[4] Okamoto T., Ohta K.: *Universal electronic cash.* Advances in Cryptology – CRYPTO '91 (ed. J.Feigenbaum), Lecture Notes in Computer Science 567, Springer -Verlag, Berlin, 1992, 324–337.

[5] Ferguson N.: *Single term off-line coins.* Advances in Cryptology - EUROCRYPT '93 (ed. T.Helleseth), Lecture Notes in Computer Science 765, Springer-Verlag, Berlin, 1994, 318–328.

[6] Foo E., Boyd C., Caelli W., Dawson E. : *A taxonomy of electronic cash systems.* Proceedings of IFIP/SEC '97 13^{th} International Information Security Conference, Chapman and Hall, London, 1997, 337–348.

[7] Franklin M., Yung M.:*Secure and efficient off-line digital money.* Automata, Languages and Programming: 20^{th} International Colloquium, ICALP '93, Lecture Notes in Computer Science 700, Springer-Verlag, Berlin, 1993, 265–276.

[8] Schneier B.: Applied Cryptography. John Wiley & Sons, New York, 1996.

[9] Mao W.: *Lightweight micro-cash for the Internet.* Computer Securtity – ESORICS 96, Lecture Notes in Computer Science 1146, Springer-Verlag, Berlin, 1996, 15–32.

[10] Chaum D., Fiat A., Naor M.: *Untraceable electronic cash.* Advances in Cryptology – CRYPTO '88 (ed. S. Goldwasser), Lecture Notes in Computer Science 403, Springer-Verlag, Berlin, 1990, 319–327.

[11] Brands S.: *Electronic cash for the Internet.* Proceedings of the Internet Society 1995 Symposium on Network and Distributed System Security, San Diego, California, 1995.

[12] Pfitzman B., Waidner M.: *How to break and repair a "provably secure" untraceable payment system.* Advances in Cryptology – Crypto '91 (ed. J.Feigenbaum), Lecture Notes in Computer Science 576, Springer-Verlag, Berlin, 1992, 338–350.

[13] Koblitz N., Menezes A.J.: *Another look at "provable security".* Journal of Cryptology (Online First), 2005.