Vysoká škola ekonomická v Praze

Fakulta managementu v Jindřichově Hradci

Bakalářská práce

2007 **Jana Soukupová**

Vysoká škola ekonomická v Praze

Fakulta managementu v Jindřichově Hradci

Softwarové zabezpečení dat

Vypracovala:

Jana Soukupová

Vedoucí bakalářské práce:

Ing. Jiří Přibil

VŠE

FAKULTA MANAGEMENTU

Vysoká škola ekonomická v Praze
Jarošovská 1117/II, 377 01 Jindřichův Hradec

ZADÁNÍ BAKALÁŘSKÉ PRÁCE

pro akademický rok 2006/2007

Název práce:	**Softwarové zabezpečení dat.**
Zadání práce:	Práce se bude zabývat popisem historie, způsobů a druhů šifrování a ochrany dat. Praktická část bude zaměřena na zjištění rychlosti a efektivity šifrování v závislosti na vstupních datech a zvolených aplikacích/algoritmech. Na základě těchto empirických pokusů bude navržen nejvhodnější způsob ochrany dat.
Jméno studenta:	Jana Soukupová
Ročník:	2.
Obor:	MANAGEMENT
Vedoucí práce:	Ing. Jiří Přibil
Katedra:	Katedra managementu informací
Termín zadání:	23.6.2006
Termín odevzdání:	Dle vyhlášky o průběhu státních závěrečných zkoušek v ak. roce 2006/2007

V Jindřichově Hradci 23.6.2006

Ing. Vladimír Přibyl
proděkan pro pedagogickou činnost

tel.: 0331-361342-4
0331-517911
fax: 0331-361349
IČO 61384399
bank.spoj. KB a.s. J. Hradec
č.ú.19-0304 340207/0100

Prohlášení

Prohlašuji, že bakalářskou práci na téma

»**Softwarové zabezpečení dat**«

jsem vypracovala samostatně.

Použitou literaturu a podkladové materiály

uvádím v přiloženém seznamu literatury.

Jindřichův Hradec, květen 2007

.....

podpis studenta

Anotace

Softwarové zabezpečení dat

Práce se bude zabývat popisem historie, způsobů a druhů šifrování a ochrany dat. Praktická část bude zaměřena na zjištění rychlosti a efektivity šifrování v závislosti na vstupních datech a zvolených aplikacích/algoritmech. Na základě těchto empirických pokusů bude navržen nejvhodnější způsob ochrany dat.

Poděkování

Za cenné rady, náměty a vstřícnost

bych chtěla poděkovat

Ing. Jiřímu Přibilovi,

z Vysoké školy ekonomické v Praze,

Fakulty managementu v Jindřichově Hradci.

Obsah

1 Úvod

Člověk je už ze své podstaty tvor zvídavý až zvědavý, ale ne vše by se mělo dostat k cizím uším. Ať už to jsou jen intimní zážitky sdělované kamarádce, soukromá korespondence, citlivé údaje či dokonce vládní nebo vojenská tajemství. Ale jak toho dosáhnout? Jak tyto údaje ochránit tak, aby se k nim nikdo nepovolaný nedostal a nezneužil je?

Jednou z možností, jak toho docílit, se bude zabývat právě tato práce. Se zrychlujícím se přechodem na používání dat digitálních roste i důležitost jejich zabezpečení, které nám umožňuje právě šifrování. S rozšiřujícími se oblastmi použití a potřebami se zvyšuje i jeho význam a pole působnosti. Z bitevních polí či vládních vyjednávání se přesunulo do našeho každodenního života a my se s ním setkáváme ať chceme či nechceme, ať o tom víme či nevíme.

Tato práce má za úkol nejen poskytnout stručný historický přehled o nejvýznamnějších a nejzajímavějších šifrách od dob starověkého Řecka až do současnosti, ale v další části také některé z nich představit a ukázat v praxi prostřednictvím volně šiřitelného softwaru. Pokusí se doporučit programy vhodné pro běžné používání a odhalit jejich případné slabiny i silné stránky.

Poslední část udělá jakousi tečku za zabezpečením dat a nahlédne do další součásti našeho každodenního života, kterou jsou hesla. Pokusí se seskupit jednoduchá pravidla pro jejich tvorbu i ochranu s příslušným představením a doporučením softwaru.

2 *Trocha teorie na začátek*[1]

Než se dostanu k historii, popisům či přímo pokusům, měla bych zde uvést základní pojmy, se kterými je možno se na poli šifrování setkat. Je to především základní termín kryptologie. **Kryptologie** je věda zabývající se studiem šifer a kódů. Pojem vznikl ze spojení slov kryptós (skrytý) a logos (věda). Dělí se na dvě různé části: zatímco **kryptografie** se zabývá tvorbou šifer, **kryptoanalýza** má naopak za úkol je rozluštit. Dále bude zmiňován pojem **otevřený text**, což je text původní, určený k zašifrování. Aby byla konečná zpráva pro nezasvěceného nečitelná, potřebujeme ještě **šifrovací klíč**. S jeho pomocí, procesem algoritmického převodu, vznikne námi chtěný **šifrový text.** Sled operací sloužících k převodu se nazývá **algoritmus.**

Dále narazíme na různé typy a dělení šifer. Základní rozdělení rozlišuje šifry **substituční** a **transpoziční**. Jak samy názvy napovídají, u prvních budeme něco něčím nahrazovat (substituovat), zatímco u druhých budeme používat přesunutí (transpozici) znaků. Dále se dají rozdělit na **symetrické**, u kterých je jen jeden klíč, který nám slouží jak v procesu šifrování tak dešifrování, a **asymetrické**, kde máme klíče dva: privátní (soukromý) a veřejný, tedy rozdílné klíče pro zašifrování i dešifrování. Jejich podrobnějšímu popisu se budu dále v textu ještě věnovat.

Ještě v úvodu je třeba upozornit na nebezpečí záměny pojmů **šifrování a kódování**. Ačkoli se můžeme setkat s jejich zaměňováním, neznamenají totéž. Společné mají to, že oba převádějí text z jedné formy do druhé, ovšem rozdíl je v tom, že kódování nemá za úkol utajit zprávu původní a klíč je stále stejný a veřejný. Jako příklad můžeme uvést kódování v ASCII (znaku odpovídá předem dané číslo a to vždy, nemění se).

[1] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*

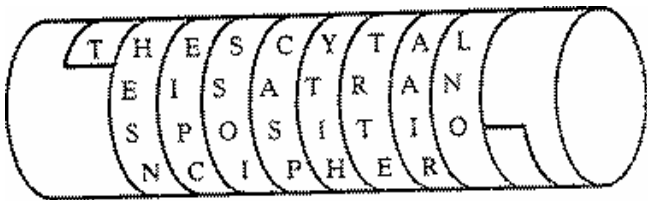
Můžeme se také setkat se **steganografií**. Ta se používá také k tajné komunikaci, ale samotná zpráva se při ní nemění. Neznevidite̊lňujeme obsah, ale samotnou zprávu. Jako příklad můžeme uvést oholení hlavy posla, kdy se zpráva napsala na holou hlavu a posel odjel až když vlasy dorostly a text tedy nebyl vidět. Pro neznalého byla tato zpráva nečitelná, ovšem její velkou nevýhodou byla doba jejího zpracování a odeslání. V současnosti se používají obrázky, filmy nebo například programy. Ideální je při nutnosti vysokého zabezpečení tuto metodu používat spolu s kryptografií.

3 *Historie kryptologie*

Znepřístupnit obsah svých zpráv uměli naši předkové už před mnoha a mnoha lety. Počátky šifrování můžeme nalézt současně s počátky užívání písma. K nejstarším nalezeným důkazům patří egyptské hieroglyfy, které se datují až do roku 1900 př. n. l.. Za počátky šifrování je můžeme považovat především proto, že málokdo toto písmo znal a tudíž text byl určen pouze malému okruhu lidí.

3.1 Starověk

Za první opravdové šifrování můžeme označit systém **Scytale**, který byl používán ve Spartě kolem roku 500 př. n. l. ve vojenské sféře. Jako klíč zde sloužily dvě tyče o určitém, předem daném průměru. Na tyč se navinul proužek kůže, na který se napsal otevřený text. Ten se pak sejmul a poslal druhé straně, která měla tyč stejného průměru. Po navinutí proužku na ni byl text opět čitelný (viz. obr. 1). Jak zjistíme po přečtení textu na obrázku, Scytale je typickým příkladem šifry transpoziční.



Obrázek 1: Scytale[2]

Jako zajímavost lze uvést, že jen o jedno století později se zmiňovala o šifrování i velice známá Kamasútra. V ní bylo mezi dovednostmi, které by měla „smyslná žena“ umět, doporučena i schopnost utajování zpráv.

[2] *Úvod do kryptografie: Hystorické pozadie*. [on-line]. http://www.krypto.szm.sk/hystoria.htm

Jedna z nejznámějších starověkých šifer je bezesporu **Caesarova šifra**. Dokládá to i fakt, že ji často používají děti na své „tajné“ zprávy, i když původce neznají a je jim v podstatě i lhostejný. Je to příklad substitučního šifrování. Každý znak otevřeného textu je nahrazován jiným a to tím, který se nachází o určitý, předem dohodnutý počet pozic směrem kupředu v abecedě (pokud se dojde na konec pokračuje se znovu od začátku). Klíčem k této šifře je tedy číslo určující posunutí. Původně byly klíčem tři znaky a Caesar tuto šifru používal v korespondenci s královnou Kleopatrou. Jako příklad můžu uvést zašifrování slova CAESAR. Pokud použiji jako klíč tři znaky posune se nám první písmeno při použití klasické anglické abecedy z C na F, A zašifrujeme na D atd. až nám vznikne zašifrovaný text FDHVDU.

3.2 Středověk

Pravděpodobně první knihu o šifrování napsal v 16. století benediktinský opat **Johannes Trittheim**[3]. Jeho činnost se však nedočkala ocenění a byl dokonce označen za čarodějníka. V tomto století vznikla také tzv. **Cardanova mřížka**[4] (podle učence Jeronýma Cardana). Jako klíč sloužila mřížka s nepravidelnými otvory, kam se vepsal text. Po sejmutí se do prázdných políček dopsala libovolná písmena. K ještě většímu zabezpečení se používalo různé její otáčení. Dále je možné zmínit i šifru kardinála **Richelieu** (transpozice písmen otevřeného textu podle hesla, vedla ke vzniku transpoziční tabulky) či tzv. **Velkou šifru**[5], kterou si nechal vytvořit Ludvik XIV. a která byla po celá dvě století neprolomena. Byla použita například na dokumenty o muži s železnou maskou (až po rozluštění bylo zjištěno, že to byl právem potrestaný nižší důstojník).

[3] Ondřej Bitto. *Historie kryptologie*.
[4] Viktor Šuman. *Historie kryptografie*
[5] *Kryptologie včera a dnes.* [on-line].

V roce 1832 byla vynalezena **Morseova abeceda**[6]. Není to klasická šifra, spíše bychom ji zařadili mezi výše zmíněné kódování, ale byl to neocenitelný vynález pro komunikaci na dálku bez potřeby třetí osoby a o rozvoj na poli šifrování se určitě zasloužila. Úplně první telegrafický přenos byl uskutečněn v roce 1844 mezi městy Washington a Baltimore.

3.3 Novověk

Opravdu velký rozmach v šifrování nastal počátkem 20. století. Zasloužily se o to především obě světové války. Už dříve byly šifry používány především pro vojenské účely a ve válkách se ukázaly jako zcela nezbytné. Právě informace totiž mohou být tou nejsilnější zbraní.

Jako první důkaz síly informace byl tzv. **Zimmermannův telegram**. Byla to zpráva poslaná německým ministrem zahraničí nabádající mexickou vládu vtrhnout na území USA, čímž by jí zabránila v zásazích na evropském kontinentě. Tento telegram byl ale zadržen a dešifrován britskou organizací, která si říkala Room 40. Otevřený text byl poté oklikou poslán americké vládě, která se na základě toho pustila do bojů a válka se rozšířila z Evropy na celý svět.

Prvním významnějším zástupcem v oblasti kryptografie byl pracovník firmy AT&T Gilbert Vernam, který v roce 1917 objevil proudovou šifru. K telegrafní pásce se přiloží ještě jedna a k šifrovému textu se načítá jak heslo, tak otevřený text pomocí binární operace XOR. Po odeslání a příjmu zprávy se heslo zničí. **Vernamova šifra**[7] je používaná především pro diplomatické účely dodnes, jelikož je považována za absolutně bezpečnou (ve čtyřicátých letech matematicky dokázáno).

[6] Ondřej Bitto. *Historie kryptologie*.
[7] Viktor Šuman. *Historie kryptografie*

Také se začaly používat šifry kombinující jak transpoziční tak substituční šifrování. Příkladem může být německá šifra **ADFGX**[8] vynalezená Fritzem Nebelem. Otevřený text byl nejprve nahrazen dvojicí složenou z písmen ADFGX (z toho vznikl i název, písmena byla úmyslně vybrána podle velkých odlišností při zápisu morseovou abecedou, což eliminovalo chybný přenos) a poté se nově vzniklá zpráva vepsala do tabulky, jejíž záhlaví tvořilo určité slovo (= šifrový klíč). Používat se začala na počátku roku 1918 a němci byli přesvědčeni o její naprosté neprolomitelnosti. To jim ovšem vyvrátil francouzský nadporučík Georges Painvin, který rozluštil jak šifru ADFGX, tak její zdokonalenou podobu ADFGVX.

Obrázek 2: Enigma[9]

Tato práce se sice zabývá převážně šiframi softwarovými, ale byl by hřích nepřipomenout snad nejznámější šifrovací stroj používaný za druhé světové války. Stroj **Enigma**[10] (viz. obr. 2) byl německého původu a sestrojil ho inženýr Artur Scherbius. Na první pohled to byl obyčejný psací stroj. Vlastní princip šifrování byl skryt ve třech válečcích, které byly propojené s písmeny. Při každém stisku se váleček o kousek pootočil a napsal něco jiného. Jako klíč sloužilo nastavení válečku na počátku šifrování či

[8] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
[9] Dirk Rijmenants. *The German Enigma Cipher Machine*
[10] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
Viktor Šuman. *Historie kryptografie*
Ondřej Bitto. *Historie kryptologie*.

dešifrování. Zpočátku byl tento systém v podstatě nerozluštitelný. Bohužel pro německou stranu se to přeci jen povedlo zásluhou Poláků a pracovníků tzv. **Bletchley Parku**. To byla skupina matematiků, lingvistů, šachistů apod. sídlící nedaleko Londýna a usilovně pracující během 2. svět. války na rozluštění především německých šifer, vznikla z původní skupiny Room 40. Stroj, který měl na starosti dekódování textů z Enigmy, se nazýval La Bombe (bomba). Existují dohady, že právě díky úspěšnému rozluštění Enigmy bylo Německo v konečné fázi poraženo. Jako další zástupce strojových šifer můžu uvést například japonský Purpur či po dobu války nerozluštěný americký Sigab nebo anglický TypeX.

Také Japonsko nezahálelo a vytvořilo šifrovací systém, který je známý pod označením **JN-25**[11] (Japanese Navy 25). Princip byl založen na dvojitém šifrování. První spočívalo v nahrazení každého slova odpovídajícím pětimístným číslem z kódové knihy. Ta sloužila i ke druhému zašifrování, kdy klíčem bylo číselné označení strany, sloupce a řádku v knize. Sloupec se četl odshora dolu a čísla, která se zde nacházela byla odečtena od čísel v původní šifrované zprávě. Tato šifra byla po nějaké době také odhalena, ale bohužel pro Američany jen těsně předtím, než Japonci začali používat šifru jinou. I tak jim dešifrování přineslo mnoho zajímavých a opravdu potřebných informací z předchozích komunikací.

Samotné šifrování většinou trvalo nějakou dobu, ale ve válce nepřítel nepočká až mu znepřístupníte obsah zprávy a v pořádku odešlete spojencům. Bylo důležité vymyslet způsob, jak přenos šifrovaných zpráv urychlit. V nouzi se sice dalo používat drmolení či slang, ale nebyly to zrovna bezpečné způsoby. Jako opravdu originální řešení tohoto problému můžeme považovat nápad Philipsa Johnstona. Většinu svého dětství strávil v navažské rezervaci a řeč tamních domorodců zvládal výborně. Ovšem jako jeden z mála, tento jazyk byl

[11] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*

opravdu unikátní jak rozšířením, tak svojí strukturou. Tudíž z navažských domorodců udělal spojaře a po dvou je rozesílal na komunikační spojnice, kde úspěšně fungovali jak šifrovací a dešifrovací zařízení. Snahám japonských kryptoanalytiků odolával **kód Navajo**, jak byl pojmenován, na výbornou. Pro ilustraci příklady tohoto jazyka:[12]

- besh-lo (v navažštině „železná ryba“) – v angličtině ve významu ponorka
- dah-he-tih-hi (v navažštině „kolibřík“) – v angličtině jako stíhací letoun

Posunem dále v historii zjistíme, že vlastně šifrám vděčíme za vynález počítačů. Ty by samozřejmě časem vznikly stejně, ale velká nutnost šifrování a ještě větší potřeba dešifrování tento proces urychlila. **Colossus** byl první programovatelný elektronický počítač a sloužil k rozluštění německé šifry Lorenz, kterou v tajné komunikaci používal Hitler. Následovníkem je mnohem známější **ENIAC**, se kterým se počítačový průmysl začal velice rychle rozvíjet.

[12] Bitto, Ondřej. Šifrování a biometrika aneb tajemné bity a dotyky. Str. 32.

4 Současná kryptologie

4.1 *DES*[13]

Po konci druhé světové války mohl nastat v této oblasti útlum, ale nestalo se tak. Počítačový průmysl začal vzkvétat a nastal problém jak ochránit data, která se nacházejí v počítačích a tečou sítěmi z jednoho do druhého. Původně byly šifry doménou pouze vlády, vojenských a špionážních služeb či významných osob. S příchodem počítačů se to ale změnilo. Šifry začaly potřebovat banky, úřady a později i soukromé podniky. Z tohoto důvodu se rozhodl americký Národní úřad pro standardy (NBS, National Bureau of Standards) vypsat veřejnou soutěž na návrh kryptosystému právě pro tyto instituce. Bohužel napoprvé žádný systém neuspěl a soutěž se v roce 1974, tedy rok po prvním pokusu, konala znovu. To už mohla slavit úspěch firma IBM s **DES (Data Encryption Standard),** jejímž autorem byl Horst Feistel. V roce 1975 si tento systém firma nechala patentovat a rok na to byl prohlášen za standard.

Tehdy bylo uděláno mnoho testů prověřujících bezpečnost systému DES, ale informace o nich byly dosti střežené a Národní bezpečnostní úřad, který testy vedl, upadl v podezření, že našel jakási „zadní vrátka" pro prolomení a nechává si je pro kontrolu. Toto podezření bylo brzy vyvráceno, ale ne tak pochybnosti o bezpečnosti. Snad největšími kritiky DES byli Martin Hellman a Whitfield Diffie, zaměstnanci Stanfordské univerzity. Kritika byla založena především na délce klíče, což bylo 64 bitů (efektivní použitelná délka jen 56 bitů). Tvrdili, že šifru je možno prolomit hrubou silou, tj. zkoušením všech možných kombinací (bylo by jich 2^{56}). Člověk by to samozřejmě nezvládl, ale podle Hellmana a Whitfielda by na to stačil stroj za 20 milionů amerických dolarů a přibližně jeden den času. Národní bezpečnostní úřad to ovšem popřel,

[13] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
Ondřej Bitto. *Historie kryptologie*

dále trval na současné délce klíče a podle jeho propočtů by takový stroj bylo možno postavit nejdříve na konci 80. let, přičemž náklady by vzrostly až na 70 milionů dolarů. Hellman s Witfieldem se však jen tak nevzdávali a stále se pokoušeli dokázat nedostatečnou bezpečnost tohoto systému. NBS si ale stál na svém a k pozdějším pokusům už se ani nevyjadřoval. Program byl sice několikrát vylepšen, ale to nestačilo.

DES byl od svého prohlášení za standard velice rychle rozšířen a oblíben. I přes pochybnosti o jeho bezpečnosti se používal ještě koncem 90. let! Bylo to především kvůli jeho rozšíření, takže i přes značnou kritiku taková plošná výměna nepadala v úvahu. Výpočetní technika se ovšem rozvíjela velice rychle a útoky na tento systém se množily. Roku 1997 vypsala agentura RSA soutěž o prolomení DESu. Trvalo to pouhých 5 měsíců. Úspěšným luštitelem a zároveň vítězem deseti tisíc amerických dolarů byl Rocke Verser. K vyřešení použil internet, resp. jeho 14 tisíc počítačů. Už rok na to byl sestrojen přístroj DES cracker za 250 tis. dolarů, který dokázal rozšifrovat text za necelých 60 hodin.

4.2 *AES*

To už se konečně něco začalo dít a i vysoká místa si začala uvědomovat, že je čas na změnu. Proto byla roku 1997 vyhlášena veřejná soutěž o nalezení náhrady DESu, která by se jmenovala **AES** (Advanced Encryption Standard). Do finále postoupilo pět algoritmů z patnácti: MARS, RC6, Serpent, Twofish a Rijndael. Šifrovací systémy RC6 a Serpent byly vhodnější pro hardwarové použití (byly pomalé), oproti tomu MARS byl spíše pro software a Twofish byl až příliš složitý. Pomyslnou vítěznou korunku si odnesl z tohoto boje Rijndael, algoritmus belgičanů Joana Deamena a Vincenta Rijmena. Tento systém byl přejmenován na AES a roku 2001 byl přijat jako standard. Počítá se, že výměna za nový nebude potřebná dříve než za třicet let.

4.3 *Shamirův algoritmus*[14]

Jedním z největších problémů u symetrických šifer (jen s jedním klíčem) je bezesporu distribuce klíče. Ten se sice dá relativně bezpečně předat osobně, ale ne vždy to jde a pak se musí použít nějaký informační kanál, který ovšem nemusí být bezpečný. Mnoho kryptologů se pokoušelo tento problém vyřešit a jedno z možných řešení navrhl **Adi Shamir**. Odesilatel zašifruje zprávu a pošle příjemci. Ten, jelikož nezná klíč, ji pouze ještě jednou zašifruje a pošle zpátky. Původní odesilatel odšifruje svým klíčem a naposled pošle příjemci. Na zprávě už je jen šifra příjemce, takže pro něj není problém zprávu přečíst. Zní to velice jednoduše, ovšem problém je v potřebě komutativní šifry (nezáleží na pořadí šifrování a dešifrování), jejíž nalezení už tak jednoduché není.

4.4 RSA

Dalšími kdo se o něco podobného pokoušeli byli již zmínění Whitfield Diffie a Martin Hellman (kritika DES). V jejich **Diffie-Hellmanově protokolu**[15] se poprvé objevila idea prvočísel. Společně pak s Ralphem Merklem položili základy **asymetrického šifrování**. Zatímco symetrické používá jen jeden klíč jak k zašifrování tak k dešifrování, u asymetrického používáme klíče dva: soukromý a veřejný. Jak samy názvy napovídají, soukromý může znát pouze někdo, veřejný je volně k dispozici. Princip fungování je založen na tom, že zprávu pomocí veřejného klíče může zašifrovat kdokoli, ale na rozšifrování je třeba soukromý klíč, který zná jen určitá osoba. Jediný problém je v tom, že příjemce zprávy si nikdy nemůže být jist odesilatelem, jelikož zmíněný veřejný klíč může použít kdokoli. Zde se dá opět použít asymetrické šifrování jako prostředek digitálního podpisu, kdy určitá osoba použije svůj soukromý klíč k zašifrování a každý si pak může rozšifrováním veřejným klíčem ověřit

[14] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
[15] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*

totožnost (pokud se to podaří, nemohl zprávu zašifrovat nikdo jiný než majitel soukromého klíče).

Byl to velice významný krok, ale zdaleka ne konečný. Stále tu byl problém jakou matematickou formu zvolit, aby se z veřejného klíče nedal odvodit soukromý a tudíž přenos byl opravdu bezpečný. To se povedlo vyřešit roku 1977 pánům Ronaldovi L. Rivestovi, již zmíněnému Adi Shamirovi a Leonardovi Adlemanovi. Vznikl velice známý algoritmus **RSA**[16] (počáteční písmena jmen vynálezců). Funguje na principu součinu prvočísel a jejich zpětného nalezení (tzv. faktorizace, která je obzvláště u dlouhých čísel v podstatě neřešitelná). Samozřejmě, že byl podán i podnět na vyzkoušení odolnosti. Bylo dáno za úkol faktorizovat číslo o 129 cifrách. Prvočinitelé byli nalezeni až po více jak šestnácti letech roku 1994.

4.5 PGP[17]

Při použití opravdu dlouhého klíče je asymetrické šifrování bezpečné, ale poněkud pomalé. Řešení nalezl v kombinaci symetrické a asymetrické šifry **Phil Zimmermann**. U symetrické byl problém bezpečně předat klíč, u asymetrické čas potřebný k zašifrování. Zimmermann tyto dva problémy vyřešil tím, že zprávu zašifroval symetricky a klíč k ní asymetricky. Tento systém nazval **PGP (Pretty Good Privacy).**

Vždy byl názoru, že utajování informací by nemělo být doménou pouze vládního a vojenského sektoru, ale že by měl mít k šifrování přístup každý. Proto svoje PGP poskytl k volnému stažení z internetu. Tím mu však nastaly veliké problémy. Nejen ze strany společnosti RSA Data Security, které patřil patent na RSA, jenž byl v PGP využíván, ale především ze strany státu. V USA

[16] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
[17] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*

se totiž šifry považují za zbraně a jsou přísně střeženy, což v praxi znamenalo, že byl obviněn z nedovoleného vývozu zbraní. Žaloba byla nakonec stažena.

4.6 HASH[18]

Hashovací funkce se používá převážně u elektronického podpisu. Díky ní nemusíme podepisovat celý dlouhý dokument, ale pouze jeho tzv. otisk. Kvalitní hashovací funkce musí být schopna vytvořit otisk pevné délky z jakéhokoliv dokumentu proměnlivé délky, musí být bezkolizní (nesmí existovat stejný otisk pro dva různé vstupní texty) a jednosměrná (k otisku nelze nalézt odpovídající text).

Mezi nejznámější a nejpoužívanější patří například **MD5**, u kterého je ale od roku 2004 znám postup pro nalezení kolizní dvojice zpráv podobně jako další hash **SHA-1**. Zatím nepokořený zůstává pouze jeho nástupce **SHA-2.**

4.7 Autentizační protokol[19]

V současné době se kryptologie využívá hojně k ověřování identity (autentizaci). Kupříkladu při přihlašování se ke svému bankovnímu účtu přes internet, resp. do jakéhokoliv systému, či pro elektronický podpis. „**Autentizace** je proces, při kterém dochází ke kontrole identity uživatele.“[20] Rozdělujeme tři základní druhy autentizace:

- přihlášením prostřednictvím něčeho, co známe jen my (heslo, PIN)
- pomocí něčeho, co jen my vlastníme (identifikační karty, souhrnně označované jako tokeny)

[18] Martin Fiala. *ABC Linuxu. Hash*
Wikipedie otevřená encyklopedie. *Hašovací funkce*
[19] Bitto, Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*
[20] *Bitto, Ondřej. Šifrování a biometrika aneb tajemné bity a dotyky*

- přes nějakou naší jedinečnou součást, biometricky (hlas, otisk prstu, snímače sítnice)

Při každé takovéto identifikaci posíláme nějakou cestou své údaje, které mohou být snadno zneužitelné. Zde se také uplatňuje kryptografie, která má za úkol zneužití zabránit.

4.7.1 Protokoly „výzva-odpověď“

Tento typ funguje na principu **časově proměnných parametrů**. Princip spočívá v tom, že server odešle klientovi výzvu k identifikaci, klient k ní připojí své heslo a zpět odešle otisk. Server udělá to samé, čímž si, v případě shodnosti, ověří identitu. Bezpečnost spočívá právě ve výzvě, která je jedinečná díky použití časově proměnných parametrů. Ty rozdělujeme do tří základních skupin:

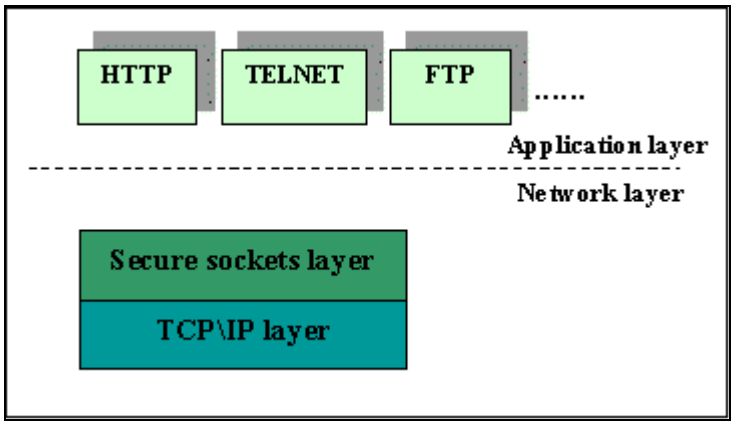
- náhodná čísla
- sekvenční čísla (generování podle předem dohodnuté sekvence)
- časová razítka (hodnota závislá na aktuálním čase, možná kontrola prodlení)

Další na bázi „výzva-odpověď“ je **Needham-Shroeder** protokol. Existují dvě verze, jedna se symetrickými klíči a druhá s asymetrickými. První, využívající symetrické šifrování, počítá se třetí důvěryhodnou stranou, které věří oba komunikující. Tato varianta se stala základem například pro autentizační protokol Kerberos. U druhého není potřeba existence třetí strany, stačí mít jinou možnost ověření.

4.7.2 SSL protokol

Tento protokol je vytvořený firmou Netscape a používá se pro zabezpečení a autentizaci při komunikaci serveru a klienta. V současnosti je SSL (Secure Socket Layer) všeobecně uznávaným standardem.

SSL je jakási přidaná vrstva mezi TCP/IP (přenosový a komunikační protokol) a aplikací (např. HTTP, FTP, TELNET a další). Viz obr.



Obrázek 3: Pozice SSL v modelu TCP/IP[21]

Obsahuje ještě čtyři podprotokoly: Handshake Protocol, Record Layer Protocol, Change Cipher Specification Protocol a Alert Protocol.

Nejpodstatnější je právě **Handshake Protocol**. Zajišťuje autentizaci klienta (není nutné při anonymní komunikaci), autentizaci serveru a výměnu informací nutných pro určení komunikačního šifrovacího klíče. Činnost tohoto protokolu by se dala ve zkratce vysvětlit takto:

1. Klient oslovís server a zašle mu potřebné informace (o podporovaných šifrách, náhodně vygenerovaná data, identifikátor relace…).
2. Server zareaguje posláním svojí verzi SSL, náhodně vygenerovaných dat, identifikátoru, svého certifikátu a žádosti o certifikát klientův.
3. Klient pošle tzv. PreMasterSecret, z něhož se vypočte sdílené tajemství (šifrováno veřejným klíčem serveru), dále svůj certifikát (jen pokud je komunikace neanonymní) a digitálně podepsaná data ke své identifikaci.
4. Ukončením Handshake protokolu začíná šifrování právě dohodnutými klíči.

[21] Petr Odvárka. *SSL Protokol (1) – Princip a přínosy*

V některých případech se používá jen obnovení už jednou použité relace, kde je vypuštěna autentizace a jen se generují nové klíče.

Record Layer Protocol má na starosti následující datové úpravy: fragmentaci (dochází k rozdělování dat do bloků), komprimaci, výpočet MAC (Message Authentication Code, aplikace hashovací funkce) a samotné šifrování algoritmem dohodnutým předem v protokolu Handshake.

Change Cipher Specification Protocol nastavuje parametry šifrování a **Alert Protocol** slouží k upozornění na nastalé chyby.

V textu výše je uvedena výměna certifikátů a tudíž by bylo vhodné připomenout vůbec nejpoužívanější certifikáty v této oblasti a to **certifikát X.509**. Má přesně dané údaje, které musí obsahovat. Je to například verze certifikátu, sériové číslo, doba platnosti, veřejný klíč, informace o certifikační autoritě, informace o majiteli či o algoritmu, který je používán.

4.7.3 Kerberos[22]

Kerberos je autentizační a autorizační systém používaný na nedůvěryhodné a nezabezpečené síti. Vytvořen byl firmou **MIT** (Massachusetts Institute of Technology) a pojmenován podle tříhlavého psa, strážce vchodu do antického podsvětí. Základem je existence tzv. důvěryhodné třetí strany, což je v tomto případě centrální server, který je na rozdíl od zbytku sítě zabezpečen velmi dobře. Není divu, obsahuje totiž všechna přístupová hesla a práva. Skládá se ze dvou částí, autentizačního serveru a serveru poskytujícího lístky. Potřebuje-li klient přístup k nějakému sdílenému zařízení, musí nejprve tento server požádat o pověření, skládající se z lístku a části šifrovacího klíče. Klient nejdříve pošle serveru otevřený text, podle kterého je identifikován a server mu odešle

[22] Pavel Rakovič. *Kerberos, PAM*

zpět tzv. TGT lístek (Ticket-Granting Ticket), na jehož základě dostane klient na požádání přístupový lístek k danému sdílenému zařízení.

4.7.4 SET (Secure Electronic Transaction)[23]

S rozvojem elektronického bankovnictví vznikla i potřeba nějakým způsobem zajistit bezpečnost všech zúčastněných stran. K tomu slouží právě SET, systém pro bezpečné elektronické platby. Má především zabránit tzv. kradení čísel kreditních karet při použití na internetu. Princip funguje následovně:

1. Zákazník si vybere zboží a pošle objednávku, která je šifrovaná veřejným klíčem obchodníka, spolu s informacemi zašifrovanými veřejným klíčem banky (obchodník k nim nemá přístup).
2. Obchodník přesměruje informace ke své bance, která si je ověří kontaktováním banky zákazníka.
3. Zákazníkova banka odečte příslušnou sumu z účtu a dá to na vědomí bance obchodníka.
4. Obchodníkova banka potvrdí platbu a on vyřídí objednávku.

Tento systém má hned několik výhod. Především je to jakási jistota, že zákazník nepodvede obchodníka a naopak (obchodník by si například mohl strhnout více peněz, či zákazník nemusí uvést správné údaje), dále anonymita, kdy obchodník nezná informace o zákazníkovi (tudíž je nemůže zneužít), banka o tom, co si zákazník kupuje a dále také nejsou problémy s nedůvěřivostí banky (v normálním případě by bez jakéhokoliv dokladu či podpisu nemusela věřit obchodníkovi, že má tolik peněz zákazníkovi strhnout). Navíc tento systém zabraňuje i útokům zvenku, při přenosu dat.

[23] Jiří Peterka. *První transakce SET v ČR*

4.7.5 Možné útoky na autentizační protokoly

Utajovaná komunikace slouží jako terč pro různé útoky. Je možné rozlišovat několik druhů útoků na autentizační protokoly:

- man in the middle útok – někdo působící mezi komunikujícími zachytávající jejich rozhovor a klíče, vhodná autentizace
- útok impersonancí – vydávání se za někoho jiného, například zachycením autentifikačního čísla
- útok přehráním – použití zachyceného hesla někoho jiného, jako obranu lze použít časové proměnných parametrů
- útok prokládáním – skládání informací z několika předešlých komunikací
- útok odrazem – vychází z útoku prokládáním, ale některé informace se navrací i samotnému odesilateli

5 Šifrování v praxi

Určitě každý člověk má nějaká soukromá data, která by se neměla z jakéhokoliv důvodu dostat někomu jinému do ruky. Šifrování dat už dávno není doménou jen vládních institucí, armády, bank či teroristů. I běžní uživatelé mohou mít svá data v bezpečí a na výběr mají hned z několika možností.

V předešlé části bylo teoreticky popsáno šifrování od dob starověké Sparty přes Ludvíka XIV. či obě světové války až do současné doby a nyní bude přesunuta pozornost na praktické použití v současnosti se vyskytujících programů. K dispozici je nepřeberné množství jak placených, tak freeware či open source programů. Vyzkoušeny budou ty, které jsou zdarma přístupné a bude představeno použití programu TrueCrypt, dále steganografický software SecurEngine a GPG.

Software bude zkoušen na notebooku s konfigurací Intel Celeron M 430, 1,87 GHz, 512MB RAM s operačním systémem Windows XP Professional.

5.1 TrueCrypt[24]

Tento malý a jednoduchý program je určen jak pro Windows, tak pro Linux a je ho možné stáhnout zdarma z internetu jako open source (například ze serveru Slunečnice[25] nebo přímo z oficiálních stránek). Funguje na principu vytváření virtuálních disků v počítači. Data šifruje za běhu, na pozadí zápisu a čtení, a jeho velkou předností je právě rychlost. Nabízí hned několik druhů šifrování s různou délkou klíče a rychlostí šifrování. Poskytuje také mnoho kombinací zabezpečení od jednoho hesla až po propojení s určitým souborem. Dokáže zašifrovat jak jednotlivé soubory (různé dokumenty, obrázky…), tak i například

[24] Kwolek Jirka. *TrueCrypt - trezor nejen pro porno a nelegální software TrueCrypt - citlivá data v bezpečí*

[25] Slunečnice. Dostupné na: http://www.slunecnice.cz/

USB flashdisky nebo dokonce celé disky. Je možné si vybrat i formátování různými souborovými systémy (FAT32, NTFS). Tyto virtuální disky jsou plnohodnotné, dají se na ně instalovat programy a je možné je i formátovat (obsahují clustery). Program poskytuje možnost vybrat si ze tří druhů vytvoření šifrovaných dat:

1. První z možností je zašifrovat **celý diskový oddíl** (musíme ale počítat se smazáním původních dat).
2. Z jakékoliv souboru můžeme také vytvořit tzv. **kontejner**, do kterého data nahrajeme.
3. Nejbezpečnějším typem je vytvoření **skrytého oddílu** (kontejneru) už v jednou zašifrovaném. I kdyby někdo získal přístupové heslo, k takto vytvořeným datům se nedostane.

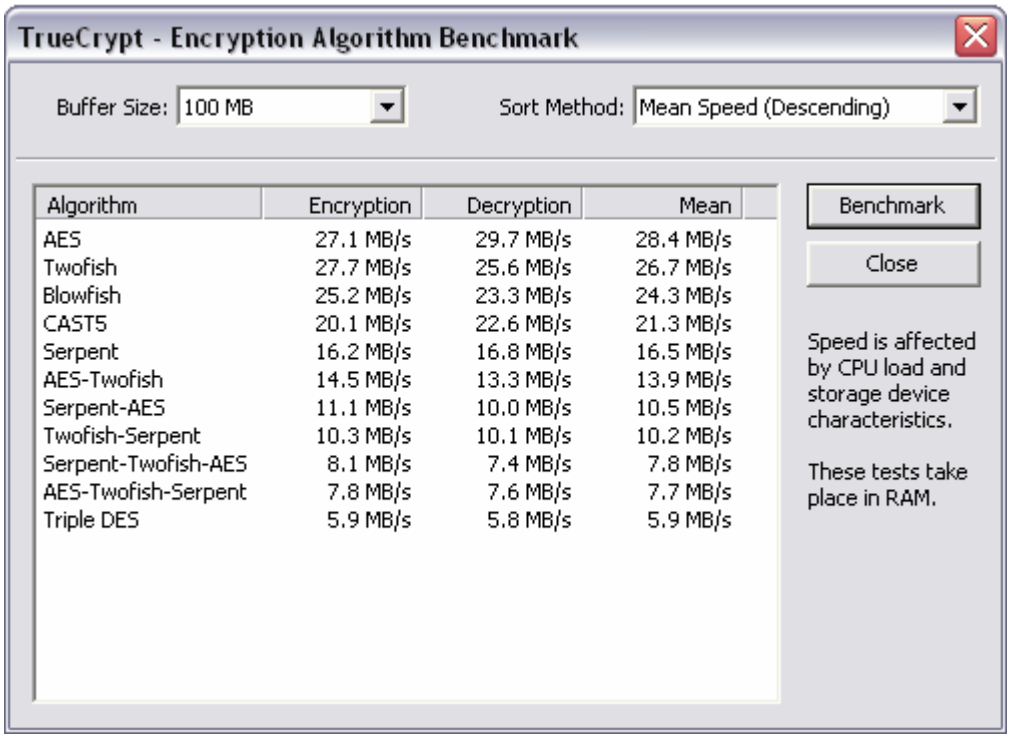
Na vyzkoušení byl použit z internetu volně stažitelný TrueCrypt 4.2a. Nejpodrobněji bude popsána druhá možnost, tvorba klasického kontejneru, ale nebude opomenuto ani několik dalších vlastností tohoto programu.

5.1.1 Vytvoření kontejneru

Po spuštění programu je nejprve nutné vybrat možnost **Create Volume**, která spustí průvodce vytvořením kontejneru. Dále je potřeba určit název a typ souboru, pod kterým má být kontejner vytvořen. Je vhodné vybírat typ podle objemu dat, která jsou ukládána, jelikož soubor bude vykazovat reálnou velikost. To v praxi znamená, že pokud si za název souboru vybereme například data.doc a vložím do něj fotky, bude velice podezřelé, že dokument má velikost 500 MB. Příponu není nutné užívat vůbec (bude to neurčitý soubor, který se při spuštění dotáže na výběr programu pro otevření) a nebo je možné použít příponu .tc, kdy při kliknutí na tento soubor se rovnou spustí TrueCrypt a bude požadováno heslo (je to pohodlnější, ale každý znalý tohoto programu bude vědět, kde se skrývají citlivá data).

Jako další krok je výběr druhu šifrování. Program TrueCrypt nabízí hned několik možností. Buď je možné šifrovat jedním algoritmem, kdy z nabídky je možné si vybrat AES, Blowfish, CAST5, Serpent, Triple DES či Twofish a nebo je k dispozici i možnost kombinovat a to AES-Twofish, AES-Twofish-Serpent, Serpent-AES, Serpent-Twofish-AES a Twofish-Serpent. Ve druhém případě se otevřený text šifruje postupně každým algoritmem.

Dále se vybírá hashovací funkce, kde je možné si vybrat RIPEMD-160, SHA-1 nebo Whirlpool. Jsou zde k dispozici také popisy jednotlivých šifer a po stisku klávesy Benchmark je možné nechat si zobrazit tabulku s rychlostí šifrování u každého algoritmu, kde je pouze nutné vybrat velikost souboru a pak jen vyčkat na zobrazení výsledků (rychlost šifrování se ve dvou testech dělaných po sobě může měnit, záleží na momentálním využití RAM). Také je zde možnost vybrat si, zda chceme výsledky seřadit podle průměrné rychlosti (průměr z rychlosti šifrování a dešifrování) nebo abecedně. Ukázku je možné vidět na obrázku číslo 4:



Obrázek 4: Benchmark

Jak je vidět z uvedené tabulky, nejrychlejší je algoritmus AES a Twofish a nejpomalejší Triple DES (je dokonce pomalejší než šifrování prostřednictvím několika algoritmů). Rychlost by ovšem neměla být jediným rozhodovacím kritériem (přestože v tomto případě převažujícím), je důležité sledovat i délku klíče, která určuje odolnost proti útoku hrubou silou (tipování klíče).

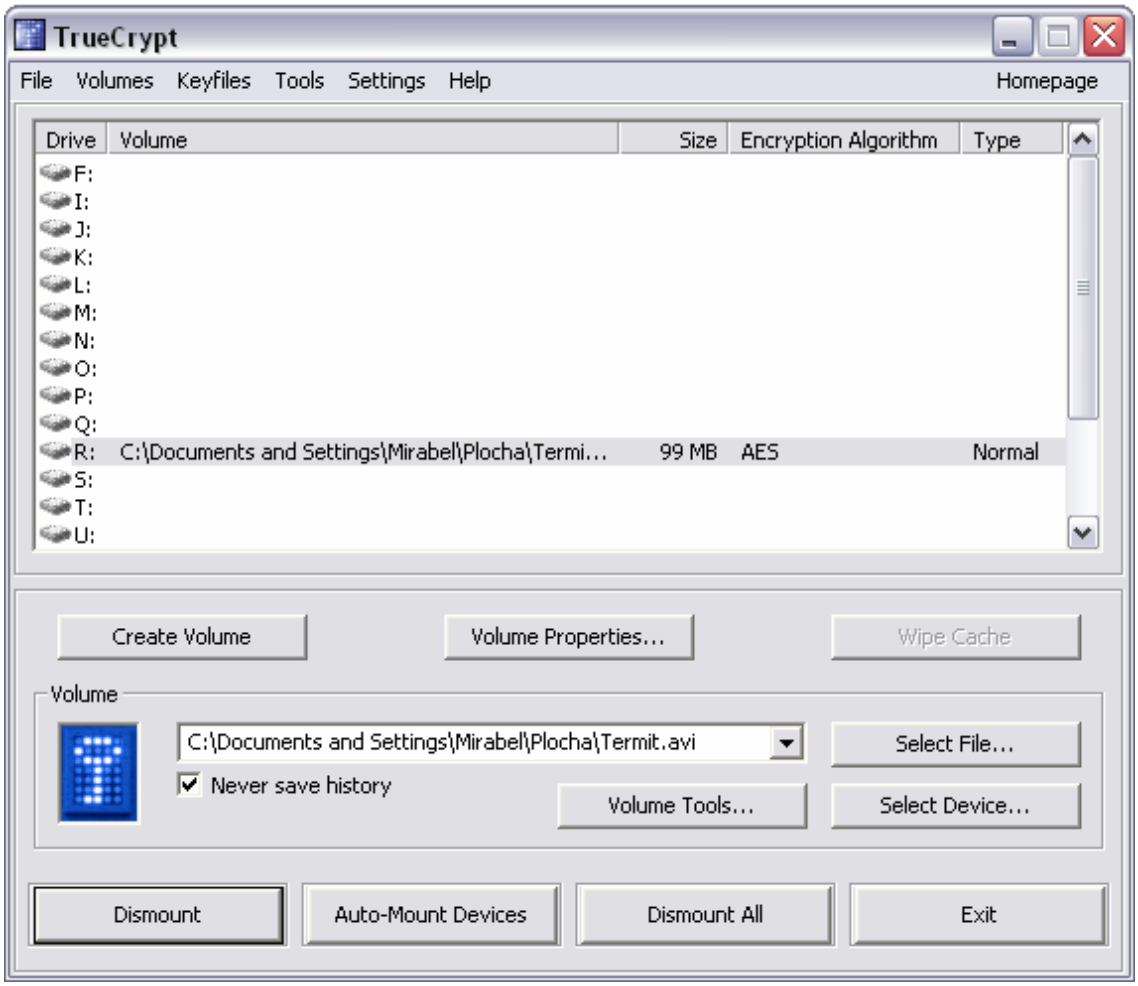
Jako další krok je zvolení velikosti kontejneru (je třeba volit s rezervou, nejde dodatečně zvětšit). Pak už jen stačí zadat přístupové heslo (nebo spíše přímo fráze, doporučuje se zde délka 20 znaků, při menší se objeví upozornění), případně zvolit klíčový soubor, který by byl třeba v případě přístupu ke kontejneru a zformátovat.

Po těchto krocích se vytvoří nový soubor. Pokud byl zvolen například dokument, objeví se v určeném umístění dokument toho jména. Pokud bude snaha ho otevřít jinak než přes TrueCrypt, bude se chovat jako poškozený (nebude ho možné otevřít).

Otevření kontejneru se provádí opět přes spuštění TrueCrypt, kde se tentokrát zvolí Select File, vybere určitý soubor, označí disk, pod kterým se chce spustit a stiskne se Mount.

Po zadání příslušného hesla je získán přístup do kontejneru, se kterým se poté může pracovat jako s jakýmkoliv jiným diskem (tj. například se zobrazí v nabídce disků v libovolném souborovém manažeru). To znamená, že je do něj možnost ukládat data, instalovat programy, mazat apod.

Po ukončení práce se nesmí zapomenout kontejner opět uzavřít a znemožnit tím k němu přístup. To se provádí volbou Dismount.



Obrázek 5: Otevření kontejneru

5.1.2 Skrytý kontejner

Pokud bude potřeba data ještě více zabezpečit a ochránit, je zde možnost vytvořit si v již zhotoveném kontejneru ještě jeden, skrytý. Při přístupu ke kontejneru se dešifruje ten, jehož heslo bylo zadáno. Mohlo by se totiž stát, že se někdo bude pokoušet heslo získat kupříkladu násilím a pak stačí jen prozradit to méně důležité a data s vyšší důležitostí se mu vůbec nezobrazí. Takto schované informace nelze nalézt dokonce ani pomocí forenzní analýzy (analýza elektronických dat používaná například při policejním vyšetřování).

5.1.3 Traveller mode

TrueCrypt dále nabízí také velice užitečné zabezpečení přenosných pamětí, tzv. Traveller mode (cestovní režim). V tomto případě uloží program na přenosný

disk svojí 1MB velkou kopii a po nastavení a vytvoření samospouštěcího souboru *autorun.inf* (který musí být v kořenovém adresáři) je schopen se v jakémkoliv počítači spustit. Při připojení disku se pak vždy automaticky dotáže na heslo.

5.1.4 Závěr

Ovládání tohoto programu je relativně jednoduché a intuitivní. Vytvoření kontejneru ani samotná pozdější manipulace s daty není náročná (obzvláště pokud je vzato v úvahu pozitivum skryté v dobrém zabezpečení). Data tímto způsobem skrytá jsou v bezpečí, jakoukoliv analýzou nelze zjistit jejich povahu a množství, ani jaký šifrovací algoritmus byl použit. Heslo nejde nijak odhalit (kromě lidského selhání vlastníka, které lze ovšem eliminovat vytvořením skrytého kontejneru) a pokud ho člověk zapomene, nemá žádnou šanci se k datům znovu dostat. Umístění kontejneru v počítači lze libovolně měnit a dá se umístit například i na internet či kopírovat a data stále zůstanou v bezpečí.

5.2 Steganografie[26]

Zašifrováním zprávy se bezpečnost dozajista zvýší, ale obecně se ví, že každá šifra je prolomitelná, záleží jen na prostředcích a čase, který je k tomu potřebný. Pokud jsou data opravdu citlivá a jen šifrování nestačí, může pomoci steganografie, která zajišťuje tzv. neviditelnost dat.

Steganografie byla zmíněna již v úvodu práce a v této části bude ještě více rozvedena, jelikož spolu se šifrováním znatelně zvyšuje zabezpečení dat. Steganografie nepoužívá zašifrování zprávy, ale její pečlivé schování. V minulosti se používalo různé ukrývání zprávy například na lidské tělo (viz. v úvodu například oholená hlava otroka) nebo používání tzv. neviditelných

[26] Michal Němec. *Užitečné programy, které jsou zadarmo*
Ondřej Suchý. *Steganografie, aneb jak to dělá Usáma bin-Ládin*
Marek Smetana, Petr Penkala. *Steganografie*

inkoustů (například mléko nebo citron, zpráva se objevila až po zahřátí kupříkladu nad svíčkou).

V současnosti se využívá existence mnoha digitálních formátů, u kterých se dají použít volná místa k přenosu informace, aniž by ji bylo možné zaznamenat. Více se samozřejmě používají nekomprimované formáty, mající větší prostor pro skrytí informace. Zde se naráží právě na jeden z problémů této metody a to je velikost ukryté zprávy. Je celkem logické, že pokud nemá být výrazně a viditelně změněn obrázek, nesmí se do něj otevřeného textu dát příliš, často uváděný poměr je 1:10. Častěji se tedy takto přenáší data textová, než obrázky či zvuky.

Pro přístup k utajené informaci je třeba vědět, že právě v tom jistém obrázku je schovaná, mít příslušný software (ve většině případů ten, který byl použit k ukrytí, nebývají totiž vzájemně kompatibilní) a samozřejmě znát heslo.

V nedávné době se o steganografii začalo více mluvit a to ve spojení s terorismem, když americká zpravodajská služba zjistila, že známý a obávaný terorista Usáma bin Ládin používal právě steganografii k utajované komunikaci. Skrýval souřadnice cílů útoků do pornografického materiálu, který pak zveřejňoval na volně přístupných chatech, kde jej jen tak někdo nemohl odhalit.

5.2.1 SecurEngine[27]

Jako příklad steganografického softwaru byl vybrán program SecurEngine Professional 1.0[28]. Je to relativně malý (na disku zabere necelé 3 MB), ale velice šikovný program. Je pouze anglicky, česká verze není k dispozici stejně jako veškeré materiály, jako například manuál (ten se nepovedlo sehnat ani v angličtině). Umí i klasicky šifrovat a na výběr je asi šest algoritmů, ale

[27] Michal Němec. *Užitečné programy, které jsou zadarmo*
[28] BrotherSoft. *SecurEngine 4.0*

šifrování už bylo popsáno u programu TrueCrypt a tedy u tohoto programu bude nejvíce pozornosti věnováno právě steganografii.

V úvodní obrazovce se nejdříve vybere požadovaná akce, v tomto případě je to Hide Files (skrýt soubory).

Obrázek 6: Úvodní obrazovka

Dále se vybere soubor, který je nutno skrýt (File) a označí ten, do kterého se mají data ukrýt (Carrier). Tento program má na výběr pouze čtyři typy formátů: bmp, gif, htm a png. Po vepsání hesla (požadovány minimálně 4 znaky) a pojmenování konečného souboru je možné přejít ke chtěné akci.

Jak bylo už uvedeno, kapacita uložení je omezená. Pokud je vybrán soubor větší než je možné ukrýt, objeví se upozornění s informací, jak maximálně velký soubor je možné použít. V tomto případě byl ukrýván textový soubor .txt do obrázku typu .bmp velikosti 94,6 kB a maximální kapacita byla uvedena 24 107 B. Program SecurEngine má ale ještě jednu velice šikovnou vlastnost, kterou je možnost komprimace ukrývaného souboru ještě před uložením (je možné ji vypnout). To v praxi znamená, že bylo možné uložit textový soubor až o velikosti 58,5 kB před kompresí (komprese je zde o něco málo efektivnější než například v programu Total Commander, kdy po zkomprimování bylo možné skrýt soubor o původní velikosti 61,3 kB).

Takto vytvořený obrázek má i přes značnou poměrnou velikost zprávy vložené naprosto stejnou velikost jako ten původní. Na dvou následně uvedených obrázcích je možné porovnat vzhled obrázku před skrytím textového souboru a po něm:

Obrázek 7: Před skrytím

Obrázek 8: Po skrytí

Pouhým okem je změna nepostřehnutelná, obrázky vypadají totožné. Steganografie totiž funguje tím způsobem, že v každém pixelu změní barvu o jednotku, tedy o jeden bit. Obrázek se tím viditelně nezmění, ale vznikne dostatek místa pro ukrytí citlivých dat.

Pokud se chceme k původním datům dostat, stačí jen zvolit **UnHide Files** a opět postupovat dle pokynů.

5.2.1.1 Self Decrypt Archive

Určitě je třeba zmínit další vlastnosti, které tento program nabízí. Jednou z nich je **Self Decrypt Archive** (samodešifrovací archiv). Odstraňuje problém s nutností druhé strany mít příslušný software. V tomto případě se vybere soubor, zadá heslo a vznikne aplikace formátu exe., která po kliknutí požaduje heslo. V případě vložení do archivu již dříve používaného obrázku s velikostí 95,6 kB (už s vloženým textovým souborem) je konečná velikost 168 kB.

5.2.1.2 Wipe Files

Určitě se už mnoha lidem stalo, že si omylem smazali data, která odstranit nechtěli. Už se jim určitě i stalo, že se našel někdo, kdo je uměl v počítači znovu nalézt a oni byli zachráněni. Ale co když nastane opačná situace? Co když

právě naopak je třeba smazat data tak, aby je nikdo sebešikovnější ať už z disku, diskety či flash paměti nemohl dostat zpátky? K tomu právě slouží položka **Wipe Files**. Dokáže odstranit data z počítače tak, aby už nikdo nemohl inkriminující materiál objevit. Tato schopnost je k dispozici buď samostatně a nebo jako doplněk při ukrývání dat, kdy se postará o dokonalé smazání skrývaného souboru (s možností výběru mezi klasickým smazáním a dokonalým odstraněním).

5.2.1.3 Závěr

Tento program je relativně intuitivní a poskytuje kompletní ochranu dat. Jediným problémem je nedostatek informací o něm (najít jakýkoliv manuál byl skoro nadlidský výkon, nepodařilo se) a dále se stávalo, že z ničeho nic havaroval (bez jakéhokoliv varování se vypnul, a to především při výběru souborů). Po srovnání s programem TrueCrypt je tento program vhodnější ke steganografii než pro klasické šifrování. Nabízí malé množství algoritmů, citelně chybí benchmark a určitě každý ocení možnost vytváření plnohodnotných virtuálních disků. Jako ocenitelné plus by se dala uvést možnost vytvoření samodešifrovacího archivu, který odstraní podmínku potřeby softwaru při odkrývání.

5.3 PGP (GPG)

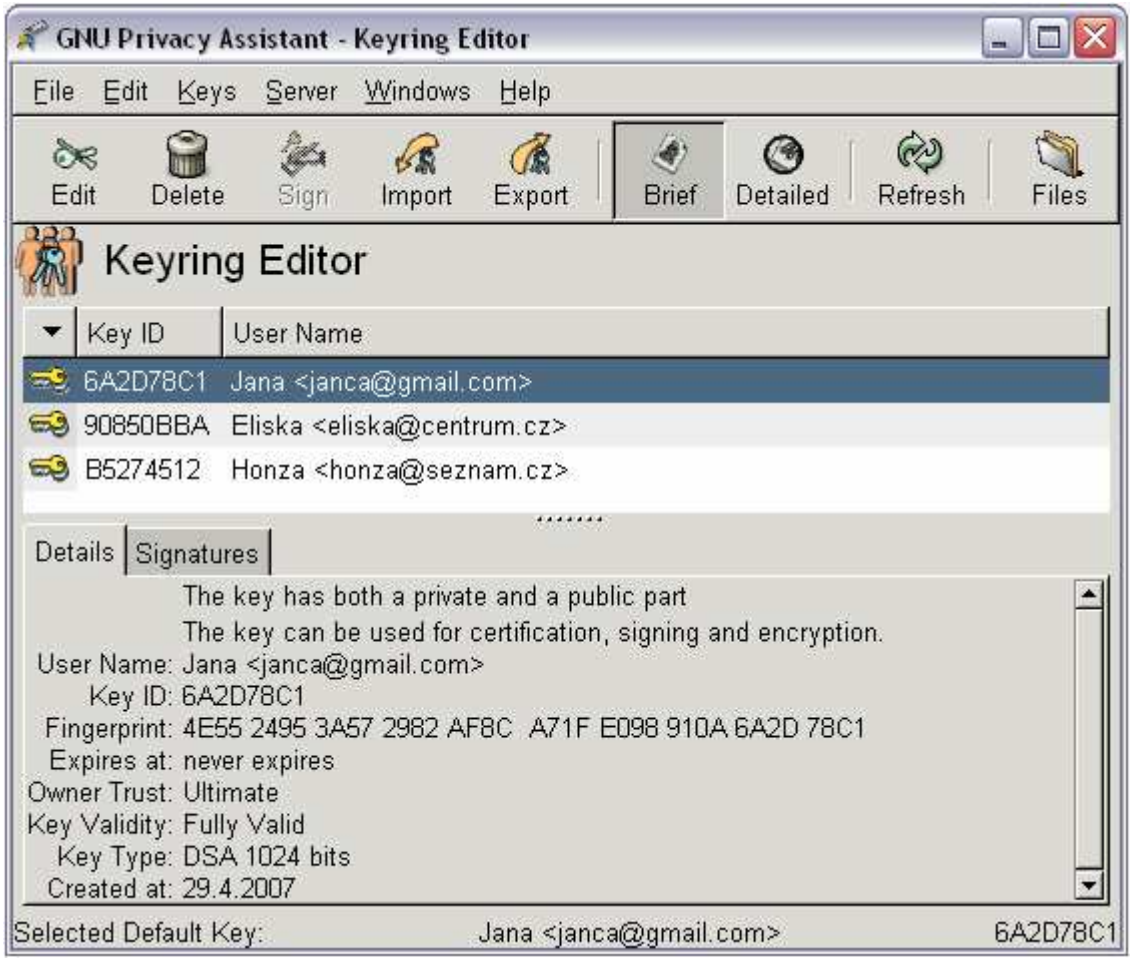
PGP je program určený k podepisování a šifrování dat. Jak již bylo uvedeno výše v textu, je na bázi RSA a jeho autorem je Phil Zimmermann. Nejčastěji je s ním možno se setkat v emailové komunikaci, kde zabezpečuje bezpečný přenos dat. V této kapitole bude popsána jeho volně šiřitelná forma GPG.

Po instalaci programu (byl vybrán GPG4win 1.0.9) je k dispozici balík programů, mezi nimiž je manažer klíčů WinPT, program GPGee, či editor klíčů GPA. Manuály jsou k dispozici v angličtině či němčině.

5.3.1 GPA

Jedním z programů, který se získá nainstalováním balíku GPG4win 1.0.9 je program GPA neboli **GNU Privacy Assistant** – Keyring Editor. Slouží jako jednoduché a přehledné grafické uživatelské prostředí GNU Privacy Guard - správě klíčů. Je zde možné klíče vytvářet, upravovat, mít o nich přehled.

Pro vytvoření nového klíče zadáme volbu *New Key* v menu *Keys*. Objeví se průvodce, do kterého se zadá jméno, emailová adresa (bude součástí klíče k jednodušší identifikaci), heslo (program upozorní, pokud se mu nezdá být bezpečné, v tomto případě je doporučována fráze) a dále je vhodné nechat si udělat záložní kopii. Po vepsání a potvrzení toho všeho je nový klíč vytvořen.

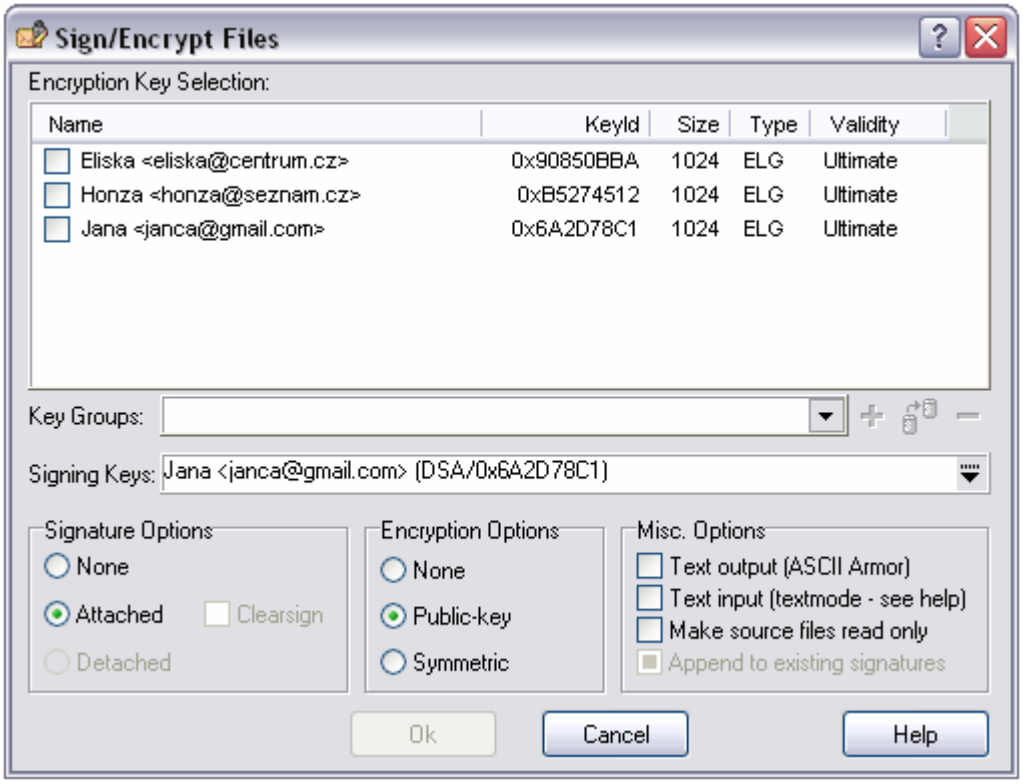


Obrázek 9: Přehled klíčů v GPU

Po kliknutí na určitý klíč se ve spodní části objeví jeho velice podrobný popis. Podle potřeby je možné zde klíče upravovat, mazat či stahovat.

5.3.2 GPGee

Po kliknutí pravým tlačítkem na jakýkoliv soubor je možné najít po instalaci programu novou nabídku *GPGee*, která po rozvinutí nabízí *Sign & Encrypt*, *Sign*, *Encrypt (PK)*, *Encrypt (Symmetric)*, *Configure* a v případě souborů s příponou .asc, .gpg, nebo .sig se objeví položka *Verify/Decrypt*. Podrobněji bude vysvětleno dialogové okno, které se objeví po stisknutí *Sign & Encrypt*, ostatní (kromě *Verify/Decrypt*) jsou velice podobné a změna je pouze v přednastavených parametrech (je tedy možná dodatečně dle přání). V hlavním okně jsou na výběr klíče, které je možné řadit podle různých kritérií, případně slučovat do skupin. Další kolonka umožňuje volbu podpisu.



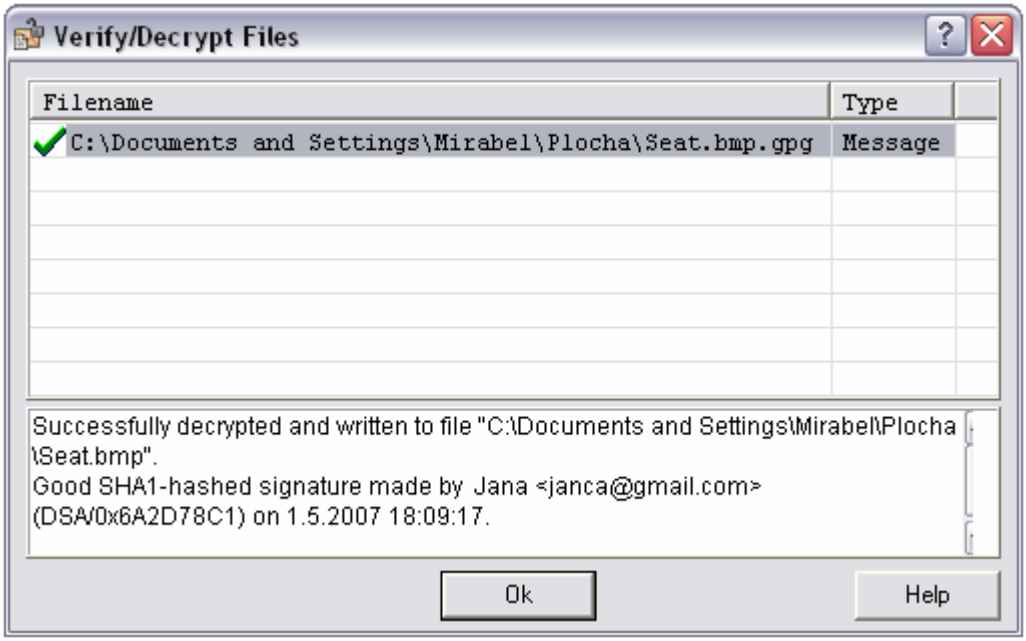
Obrázek 10: GPGee

Ve spodní části je možnost výběru šifrování: žádné (*None*), veřejným klíčem (*Public Key*) či symetrické (*Symmetric*). Hned vedle je možné najít výběr vlastností elektronického podpisu. Na výběr je buď bez podpisu (*None*), nebo zda má být připojený k souboru (*Attached*) či má být oddělený (*Detached*). Další políčka umožní například vytvořit soubor v kódování ASCII nebo soubor

určený jen pro čtení. K dispozici je i možnost nastavení jen textového výstupu nebo nastavení připojení více podpisů (možné jen v některých nabídkách).

Pokud jsou zadána všechna potřebná data, je možné kliknout na tlačítko OK (dříve je neaktivní). Při dotazu na *Passphrase* je třeba vypsat přístupovou frázi ke klíči, kterým je dokument podepisován.

Tímto postupem je vytvořen soubor s příponou .gpg a jak bylo uvedeno výše, po kliknutí pravým tlačítkem na tento soubor se objeví nabídka *Verify/Decrypt*, po jejímž odkliknutí je požadováno heslo pro dešifrování. Po zadání správného hesla se nám objeví původní soubor a tato obrazovka:



Obrázek 11: Ověření/Dešifrování

Zde je možné vidět informace o tom, kam se uložil dešifrovaný soubor, kým byl podepsán a zda nebyl od podepsání změněn (v opačném případě se objeví chybová hláška).

Podobný postup je i u ostatních nabídek (*Sign, Encrypt (PK), Encrypt (Symmetric)*), jen s menšími rozdíly. U pouze podepsaného souboru je přípona .asc a pokud se dešifruje symetricky, zadává se pouze libovolné (smluvené) heslo pro šifrování/dešifrování.

5.3.3 Závěr

Zpočátku bylo velkým problémem se v programu zorientovat a to například i kvůli rozdílu mezi volbami u GPGee jen v nastavení v tom jistém okně. Jinak manuál, nainstalovaný rovnou s programem, je přehledný a srozumitelný. Používání je realitně jednoduché a intuitivní na rozdíl od dřívějších verzí GPG, využívajících příkazový řádek.

6 Bezpečnost hesel[29]

Co potřebuje znát člověk, který si chce vybrat peníze z bankomatu? Heslo. A co přístup k emailové schránce? Opět heslo. Internetové bankovnictví? Zabezpečený počítač? Zaheslovaná data? Je doba informačních technologií a přechodů na data elektronická a kam se člověk podívá (nebo spíš chce podívat), tam musí nejdříve zadat správné heslo, jinak má smůlu.

I v předchozích kapitolách byl často zmiňován výběr hesla. Pokud se zapomene heslo, už nikdy nebude možné se k datům dostat. Pokud se heslo rozšíří do nesprávných rukou, dostane se k datům někdo, kdo nemá. Z toho logicky plyne, že heslo chránící data má v podstatě stejnou důležitost jako data skrytá.

V této kapitole budou shrnuta základní pravidla pro tvoření a ochranu hesel pro vytvoření ještě většího bezpečí jak dat, tak vlastní osoby.

6.1 Základní pravidla

1. **Délka hesla**

   Určitě si každý všiml, že některé servery odmítají hesla kratší než kupříkladu 4 znaky (jako příklad poslouží i výše uvedené šifrovací programy, kdy jeden má podmínku 4 a druhý doporučuje dokonce 20 znaků, i když toleruje kratší, což už není ani heslo, ale spíš přístupová fráze). Není to z důvodu, že by si jejich tvůrci zbytečně vymýšleli, je to úmyslný požadavek, mající za úkol ochránit právě běžného uživatele. Příliš krátké heslo totiž zvyšuje úspěšnost tzv. bruteforce útoku, který využije všechny možné kombinace znaků. To v praxi znamená, že čím méně písmen, tím méně možností a tím kratší doba potřebná

[29] Brbla. *Volba hesla a jeho utajení před světem*
Microsoft Corporation. *Vytváření silnějších hesel*

k rozšifrování. Běžně se tedy uvádí, že heslo by mělo mít délku 8 až 14 znaků.

2. **Speciální znaky a číslice**

   Jak nám může doložit kombinatorika, nejen počet písmen v hesle zvyšuje odolnost, ale důležité je také z jak velkého souboru znaků vybíráme. Proto je velice vhodné používat číslice či speciální znaky jako například: &, @, #, $, %, * apod. Jediný pozor je třeba si dát na servery, kde tyto znaky nejsou přípustné, to by pak mohl být problém. Jako další možnost je střídání malých a velkých písmen, které opět zvyšuje počet kombinačních možností.

3. **Zapamatovatelnost.**

   Ono se řekne, dlouhé heslo se speciálními znaky a číslicemi a úplně nejlepší je nicneříkající vygenerované heslo, ale kdo si to má pamatovat? Jednou z možností je vymyšlení jednoduchého hesla se zaměněním znaků. Například heslo slunicka. Zde je možné využít podobnosti a zaměnit třeba „i“ za 1, či „a“ za @. Výsledkem je heslo $lun1ck@, které je už výrazně bezpečnější než předchozí (především v případě, kdy je k emailové adrese slunicka@seznam.cz).

   Jako velice vhodné je používání různých frází. Například z písniček či básniček. Ze začátku básně Svatební košile „Již jedenáctá odbila a lampa ještě svítila“ je možné vytvořit použitím prvních písmen relativně bezpečné heslo *j11oaljs*. Pro větší jistotu se ještě vloží speciální znaky *j11o@lj$.*

   Toto heslo bude odolné jak proti bruteforce attacu (útok hrubou silou), tak proti tzv. slovníkovému útoku, který využívá faktu, že uživatelé často volí smysluplná slova, která se dají rozluštit při použití speciálního

slovníku obsahujícího i slova daná pozpátku či zdrobněliny. Některé systémy umožňují vepsat i celou frázi, včetně mezer. Pokud na něco takového člověk narazí, určitě by toho měl využít. V případě, že to neumožňují, je ideálním způsobem tato „zkratka“.

4. **Žádná spojitost s majitelem či předmětem**

Tento případ je jednou z největších neřestí uživatelů. Hesla obsahující jméno uživatele, jeho rok narození, jméno jeho psa, přítelkyně či dokonce stejná jako přihlašovací jméno je vůbec to nejhorší, co se dá jako heslo zvolit.

Stejně tak by se neměla používat hesla typu *toneuhodnes, nevim, neznam, mojeheslo, password* či posloupnosti jako *123456, asdfg, 1111* a podobné. Takováto hesla jsou přímo ráj i pro teprve začínající hackery. I pokud se člověk rozhodně vybrat si klasické slovní heslo, měl by si vybrat co nejneutrálnější a co nejméně spojené s jeho osobou a předmětem zaheslování.

5. **Trvanlivost hesla**

Jak se říká, nic není věčné a s heslem by to mělo být také tak. Čím déle heslo člověk má, tím je větší pravděpodobnost, že na něj někdo přijde a zneužije (což v případě PINu ke kreditní kartě není zrovna příjemné). Obecně platí, že čím kratší a méně bezpečné heslo je, tím kratší by měla být i doba jeho používání.

Záleží také na tom, jak často je používáno a k čemu (více si člověk dá pozor na heslo k internetovému bankovnictví než na svůj email). Pokud si dá člověk na výběru hesla opravdu záležet, je ho možné mít třeba i rok a není nijak moc ohrožen. Jestliže ovšem vybírá pouze slova a moc se výběru nevěnuje, trvanlivost by neměla být větší jak měsíc, možní i méně

(obzvláště jde-li o vysoce citlivá data). V případě hesel uvedených v předchozím textu jako nevhodná je výměna hesla nutná okamžitě.

6. Bezpečnost hesla

Člověk si může dát jakkoliv velkou práci s vytvořením hesla a jeho měněním, a může to být naprosto zbytečné, když ho poté nestřeží. Heslo je přístupový klíč k osobním údajům, věcem, právům či penězům a jako takové by se nemělo dávat cizím lidem k dispozici. Čím více lidí heslo zná, tím méně spolehlivé a bezpečné je. A nemusí to být ani úmyslně. Lidé jsou schopní PINy k mobilům nosit na papírku v kabelce hned vedle telefonu, na kartu do bankomatu nalepit štítek s heslem, či si ho napsat nad počítač, aby bylo „po ruce“. To, že bude „po ruce“ i někomu jinému, je nejspíše netrápí.

Také určitě není vhodné, používat pro všechna přihlášení stejné heslo, případně i uživatelské jméno. Když se někdo dostane do jednoho ze zabezpečení, není to taková katastrofa, jako když získá přístup ke všem.

7. Paranoia

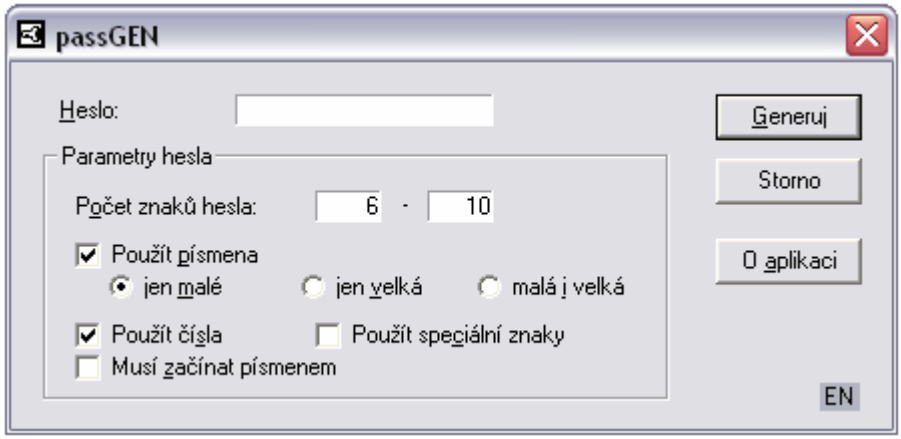
Vybírat pečlivě heslo, měnit ho a hlídat je sice dobrá věc, ale jak se říká, nic se nemá přehánět. Heslo tvořené čtrnácti znaky prokáže skoro stejnou službu jako složené z padesáti (když už nic jiného, tak je lépe zapamatovatelné a psát heslo déle než samotný email je trošku nepraktické) a manželka by asi přístup ke společnému účtu měla mít taky. Pokud není člověk tajný agent či extrémní paranoik od přírody, mělo by mu heslo vytvořené podle návodu uvedeného výše bohatě stačit.

6.2 *Dostupný software*

Jak bylo již řečeno, hesla nás provází každý den na každém kroku a v budoucnosti se to pravděpodobně nezmění (maximálně s rozvojem využívání biometrických dat, což je zatím ve větším měřítku v nedohlednu). Proto je k dispozici velké množství praktického softwaru, který má za úkol nám pomoci. Ať už se to týká generátorů hesel nebo jejich správců.

6.2.1 PassGen

PassGen je jeden z mnoha programů umožňujících vygenerovat náhodné heslo. Tento má výhodu v tom, že je možné si zde zadat parametry hesla. Jaký bude počet znaků, jestli bude obsahovat velká písmena či nikoli a zda v něm budou obsažena čísla či speciální znaky. Po nastavení stačí jen stisknout tlačítko *Generuj*, což je možné dělat do té doby, než se heslo bude uživateli líbit.



Obrázek 12: passGen

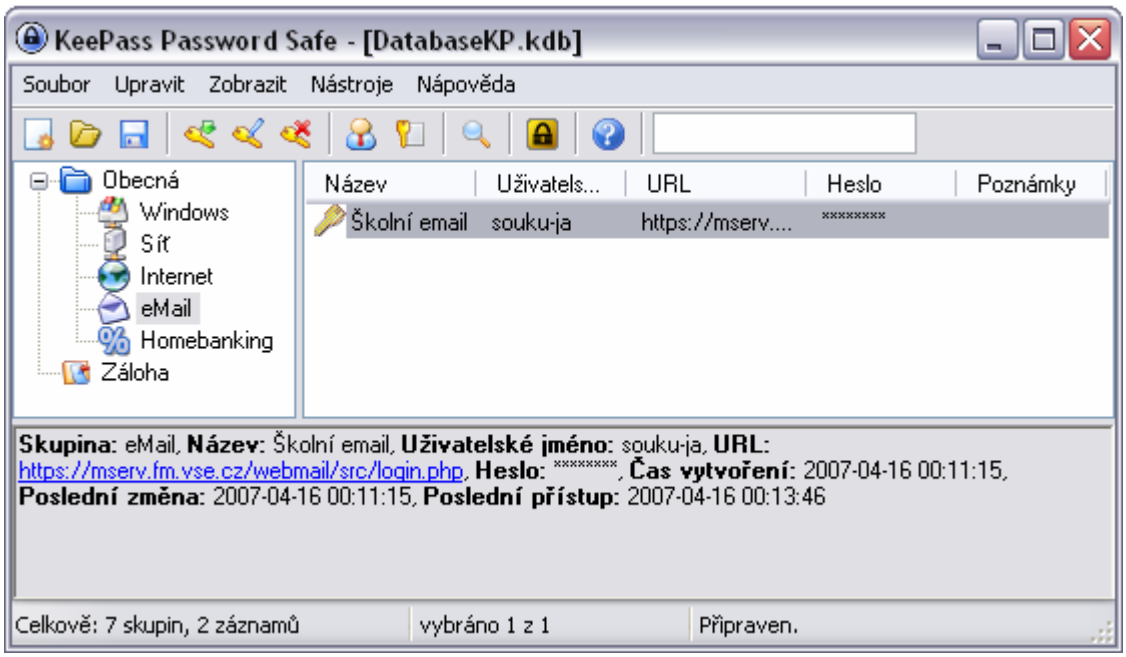
Tento program je freeware a je volně stažitelný z internetu jako komprimovaný soubor, který je třeba pouze dekomprimovat (není potřebná instalace). Ovládání je v češtině s možností přepnutí do angličtiny. Je velice jednoduchý a praktický.

6.2.2 KeePass Password Safe 1.04[30]

Často není vytvořit si bezpečné heslo zas až takový problém, jako si ho pamatovat. Na to je tu program KeePass. Funguje jako správce hesel. Proč si pamatovat mnoho různých hesel, když stačí jen jedno k programu, který prozradí ta ostatní?

Jako první krok je vytvoření si nové databáze. Ta v základu obsahuje pět skupin, do kterých je možno údaje zařadit. Databáze je klasický soubor, který se dá přenášet a funguje všude, kde je tento program (pouze při prvním spuštění bylo třeba nastavit, ve kterém programu se má spouštět).

Po výběru možnosti *Přidat záznam* se otevře dialogové okno, kam se vepíšou všechny potřebné údaje: název záznamu, uživatelské jméno, adresa stránky, heslo (se zobrazovačem jeho kvality v bitech, závislým na délce a různorodosti používaných znaků) a případně poznámky, datum exspirace či přiložení souboru. Náročnější uživatelé mají možnost si vybrat i ikonku záznamu. Konečný vzhled záznamu je na obrázku č. 13:



Obrázek 13: KeePass

[30] Ivo Mareček. *KeePass Password Safe: hesla pěkně pod šifru*

Program nabízí mnoho dalších nastavení jako například další zobrazené sloupce záznamu (kupříkladu přidat zobrazení času vytvoření či exspirace) nebo skrývání jak hesel, tak přihlašovacích jmen pod hvězdičky. Nejocenitelnější vlastností je bezpochyby kopírování jak uživatelského jména, tak hesla do dialogových oken bez potřeby je opisovat.

Dalším nástrojem je také generátor hesel. Je zde možnost zadat kritéria jako délku hesla či obsažené znaky. Ještě obsahuje tzv. Získání náhodných dat, které je vhodné pro vytvoření opravdu silného hesla. V tomto případě není použito pseudonáhodné generace (vždy existuje nějaký systém), ale heslo se generuje přes pohyb myši v určeném poli a vypsanou změť náhodných znaků do pole vedlejšího.

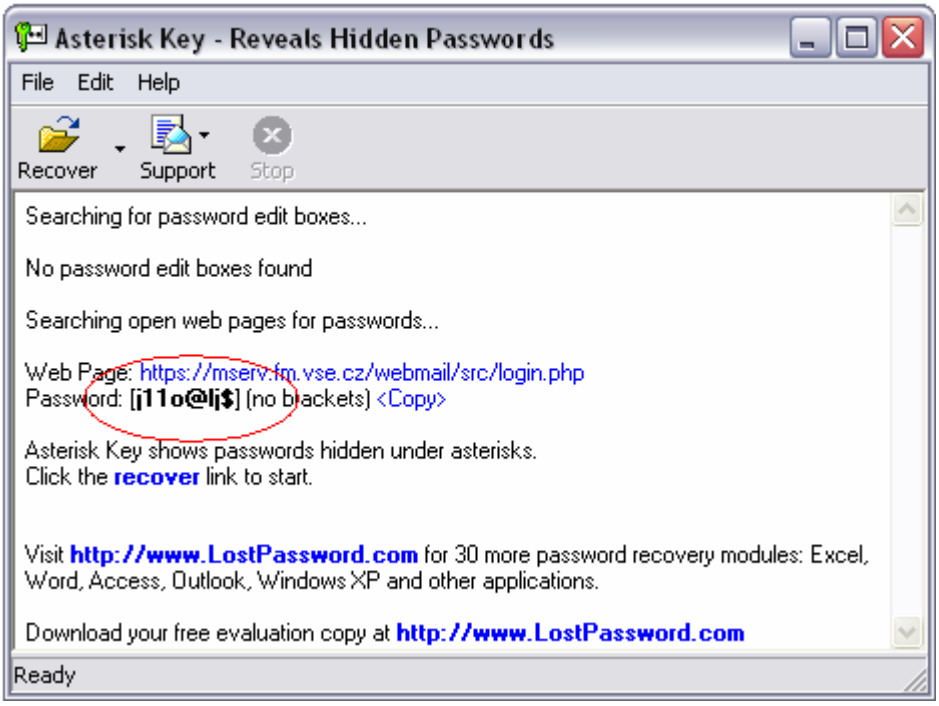
Tento program je také volně stažitelný z internetu jako open source. V základu je anglicky, přičemž k dispozici je možnost stáhnout si balíky různých jazyků (v nabídce jich je přibližně 30). Má příjemné uživatelské prostředí a dobře se s ním pracuje.

6.2.3 Asterisk Key 8.0

Operační systém Windows nabízí při každém novém přihlašování pamatování si hesla. Po přihlášení ke stránce se už objeví přihlašovací jméno a vypíše se hvězdičkami heslo. Může to být velice pohodlné, ale samozřejmě nebezpečné, stačí, aby člověk někoho pustil k počítači a on se může přihlásit aniž by se dostal k samotnému heslu.

Horší případ nastane, pokud má nějaký šikovný program jako je například Asterisk Key. Ten totiž dokáže jen z hvězdičkami vypsaného hesla dostat jeho správné znění.

Jako příklad bylo na stránky přihlášení ke školnímu emailu napsáno heslo již výše vytvořené j11o@lj$ a takto to dopadlo (odhalené heslo je označené červeně):



Obrázek 14: Odhalení hesla

Program heslo odhalil a netrvalo mu to snad ani vteřinu. Výsledek je možné si i uložit či přímo heslo zkopírovat do dotazníkového formuláře.

Tento program podporuje dialogová okna i webové formuláře. U programu Internet Explorer nebyl problém heslo odhalit, ovšem s Mozzilou Firefox si tento program nepovídal a hesla zde vypsaná nenašel (takže kdo bude používat Mozzilu má částečnou jistotu, že jeho hesla tímto programem odhalena nebudou, což ovšem již nelze zaručit u programů jemu podobných).

7 Závěr

První část této práce byla zaměřená na stručný přehled šifer v historii a současnosti, jejich vývoj, použití a funkčnost. Druhá se zaměřila na jejich praktické využití ve vybraných volně dostupných programech a byly zde uvedeny i návody na jejich obsluhu a výsledky vyzkoušení jejich funkcí.

Co se týče programů pro klasické šifrování, je možností výběru opravdu mnoho, přičemž program TrueCrypt představuje pravděpodobně ve své třídě špičku. V oblasti steganografie je situace horší, ale když už se ji někdo rozhodne používat, určitě zkrátka nepřijde a program SecurEngine by se dal jedině doporučit (i když už ne pro klasické šifrování, tam je možno najít programy vhodnější). Nejhorší situace je nejspíše u PGP, respektive GPG. Bylo dosti těžké najít tento program nezastaralý a funkční a i začátky práce s ním nebyly lehké.

Poslední část byla zaměřena na hesla, jejich výběr, ochranu a správu. Byla utříděna pravidla pro jejich utváření a celkovou péči pro co největší bezpečnost. Z programů byl uveden jak jednoduchý a nenáročný generátor hesel passGen, tak komplexní správce KeePass, umožňující jejich pohodlnou editaci a přehlednou správu. Jako důkaz síly druhé strany byl zmíněn i program Asterisk Key, který dokáže odhalit hesla vyjádřená jen pomocí hvězdiček.

Ať už se člověk chrání jakýmkoliv způsobem a s jakkoliv vysoko nastaveným stupněm zabezpečení, měl by mít na paměti to, co kdysi řekl Edgar Allan Poe: „Dokáže-li člověk vymyslet nějakou šifru, dokáže ji také dešifrovat.“[31].

[31] *Kryptologie včera a dnes.* [on-line].

Literatura

1. Bitto Ondřej. *Šifrování a biometrika aneb tajemné bity a dotyky*. Vydání první. Computer Media. 2005. ISBN: 80-86686-48-5.

2. Brbla. *Volba hesla a jeho utajení před světem*. [on-line]. 27.10.2005. Dostupné na: http://www.abowe.brbla.net/1-kapitola-uzivatelske-minimum/bezpecnost-zaklady/volba-hesla-utajeni.php

3. BrotherSoft. *SecurEngine 4.0.* [on-line]. 6.10.2006. Dostupné na: <http://www.brothersoft.com/security/miscellaneous/securengine_23496.html>

4. Dirk Rijmenants. *The German Enigma Cipher Machine.* [on-line]. Poslední aktualizace: 20.2.2007. Dostupné na: <http://users.telenet.be/d.rijmenants/en/enigma.htm>

5. Dobda Luboš. Ochrana dat v informačních systémech. Vydání první. Grada. 1998. ISBN: 80-7169-479-7

6. Ivo Mareček. *KeePass Password Safe: hesla pěkně pod šifru.* [on-line]. 8.2.2006. Dostupné na: http://www.zive.cz/h/Uzivatel/Ar.asp?ARI=127970

7. Jiří Peterka. *První transakce SET v ČR*. [on-line]. Dostupné na: <http://www.earchiv.cz/a98/a822k800.php3>

8. *Kryptologie včera a dnes.* [on-line]. Dostupné na: <http://www.fm.vslib.cz/~ksi/robotika/dokumenty/kodovani/Kryptologie_historie.htm>

9. Kwolek Jirka. *TrueCrypt - trezor nejen pro porno a nelegální software.* [on-line]. 21.09.2006. Dostupné na: <http://www.pctuning.cz/index.php?option=com_content&task=view&id=7562&Itemid=95>

10. Marek Smetana, Petr Penkala. *Steganografie.* Časopis kriminalistika 4/2006. [on-line]. Dostupné na: http://www.mvcr.cz/casopisy/kriminalistika/2006/04/steganografie.pdf

11. Martin Fiala. *ABC Linuxu. Hash.* [on-line]. 21.8.2004. Dostupné na: <http://www.abclinuxu.cz/slovnik/hash>

12. Microsoft Corporation. *Vytváření silnějších hesel.* [on-line]. Copyright 2007. Dostupné na: http://www.microsoft.com/cze/athome/security/privacy/password.mspx

13. Michal Němec. *Užitečné programy, které jsou zadarmo*. [on-line]. 2.2.2004. Dostupné na: <http://www.novinky.cz/internet/uzitecne-programy--ktere-jsou-zadarmo--151-dil-_24752_jse67.html>

14. Ondřej Bitto. *Historie kryptologie*. [on-line]. 2003. Dostupné na: <http://www.fi.muni.cz/usr/jkucera/pv109/2003/xbitto.htm>

15. Ondřej Suchý. *Steganografie, aneb jak to dělá Usáma bin-Ládin*. [on-line]. 27.2.2001. Dostupné na: http://www.lupa.cz/clanky/steganografie-aneb-jak-to-dela-usama-bin-ladin/

16. Pavel Rakovič. Kerberos, PAM. [on-line]. 2005. Dostupné na: <http://www.fi.muni.cz/~kas/p090/referaty/2005-jaro/ct/radkovic_KrbPAM.html>

17. Petr Odvárka. *SSL Protokol (1) – Princip a přínosy.* [on-line]. 25.4.2002. Dostupné na: <http://www.svetsiti.cz/view.asp?rubrika=Tutorialy&temaID=171&clanekID=187>

18. *TrueCrypt - citlivá data v bezpečí.* [on-line]. 25.2.2007. Dostupné na: <http://inetmag.net/truecrypt-citliva-data-v-bezpeci.html>

19. *Úvod do kryptografie: Hystorické pozadie*. [on-line]. Dostupné na: <http://www.krypto.szm.sk/hystoria.htm>

20. Viktor Šuman. *Historie kryptografie*. Referáty studentů AR 2005/2006. [on-line]. 21.10.2005. Dostupné na: <http://akela.mendelu.cz/~lidak/bis/seminar2005/Suman_historie.doc>

21. Wikipedie otevřená encyklopedie. *Hašovací funkce*. [on-line]. Poslední úprava 16.2.2007. Dostupné na: <http://cs.wikipedia.org/wiki/Ha%C5%A1ovac%C3%AD_funkce>