Crypto-World

Informační sešit GCUCMP

Ročník 7, číslo 3/2005 **15. březen 2005**

3/2005

Připravil: Mgr. Pavel Vondruška
Sešit je přednostně distribuován registrovaným čtenářům.
Starší sešity jsou dostupné na adrese
http://crypto-world.info
(820 registrovaných odběratelů)

Obsah : **str.**

A. Nalézání kolizí MD5 - hračka pro notebook

Vlastimil Klíma [1] , Prague, Czech Republic, 5. března 2005,
http://cryptography.hyperlink.cz , v.klima@volny.cz

Abstrakt

V tomto stručném oznámení shrnujeme výsledky našeho dvou a půl měsíčního výzkumu. Další detaily budou zveřejněny v konferenčním příspěvku.

Jednou z nejvýznamnějších kryptologických událostí posledních let bylo objevení kolizí pro sérii hašovacích funkcí MD4, MD5, HAVAL-128 a RIPEMD čínským týmem v srpnu 2004 [1]. Jejich autoři (Wangová a kol.) však utajili metodu nalézání kolizí a zveřejnili pouze strohá data a informace. V říjnu 2004 se australský tým (Hawkes a kol.) pokusil tuto metodu zrekonstruovat ve skvělé práci [3]. Nejdůležitější "čínský trik" se nepodařilo objevit, ale na základě dat z [1] bylo dobře popsáno diferenční schéma, kterým uveřejněné čínské kolize vyhovují. Naplnění podmínek tohoto schématu bylo však ještě příliš náročné a výpočetně složitější, než ukazovaly výsledky z [1]. V našem výzkumu jsme také analyzovali dostupná data diferenční kryptoanalýzou. Nalezli jsme cestu, jak generovat kolize prvního bloku 1000 - 2000 krát rychleji než čínský tým, což odpovídá nalezení jedné kolize prvního bloku na běžném notebooku za 2 minuty. Čínskému týmu tato fáze trvá jednu hodinu na počítači IBM p690. Naproti tomu byl čínský tým 2 - 80 krát rychlejší při vyhledávání kolizí druhého bloku. Obě metody se proto mohou lišit nejen časově, ale i obsahově. Celkově je naše metoda 3 - 6 krát rychlejší. Konkrétně nalezení první (úplné) kolize nám na notebooku (Intel Pentium 1.6 GHz) trvalo pouze 8 hodin. Poznamenejme, že naše metoda pracuje pro jakoukoli zvolenou inicializační hodnotu. To je velmi zneužitelné pro falšování podpisů SW balíků nebo padělání certifikátů, jak ukazují některé současné práce ([4], [5], [6]). Ukázali jsme, že vyhledávání kolizí hašovací funkce MD5 je možné provádět na domácím počítači. To by mělo být varováním před dalším používáním této hašovací funkce. V příloze uvádíme nové příklady kolizí MD5 pro standardní a zvolenou inicializační hodnotu.

Úvod

Hašovací funkce jsou velmi užitečným kryptografickým nástrojem. Pro zajištění jejich vlastností jednocestnosti a bezkoliznosti musí být hašovací funkce velmi robustní a složité. Proto je vždy velmi vzrušující, když je nalezena nějaká kolize. Jednou z nejvýznamnějších kryptoanalytických prací v minulém roce byla práce čínského týmu [1]. Nejsložitější útok si vyžádala hašovací funkce MD5, proto se budeme dále věnovat jen této funkci.

Připomeňme, že v [1] nebyl zveřejněn postup hledání kolizí, jen strohé údaje, které zde zopakujeme. Kolidující zprávy (M, N) a (M´, N´) se skládají ze dvou bloků, přičemž první bloky zpráv se liší o předem definovaný konstantní vektor C1 (M´= M + C1) a druhé bloky se liší o předem definovaný konstantní vektor C2 = - C1 mod 2^{32} (N´= N + C2), přičemž MD5(M, N) = MD5(M´, N´).

[1] Tento výzkum byl dělán o vánoční dovolené a v lednu - únoru 2005. Autor v této době pracoval pro firmu LEC, s.r.o., Praha, Česká republika, která tento projekt materiálně i finančně podpořila.

Wang a kol. uvedli, že na počítači IBM p690 jim trvá zhruba hodinu, než naleznou blok M. Nalezení bloku N pak trvá od 15 sekund do 5 minut. V první verzi [1] uvedli dva páry kolidujících zpráv. Jimi zvolená hodnota inicializačního vektoru (IV) však neodpovídala popisu MD5, neboť měla obrácené pořadí bajtů (Little vs. Big Endian). V opravené verzi příspěvku den poté uvedli opět dvě dvojice kolidujících zpráv pro MD5, tentokrát se správnou IV. Navíc poznamenali, že jejich útok pracuje pro jakoukoli hodnotu IV.

Po uveřejnění jejich výsledků jsme měli k dispozici pouze čtyři páry kolidujících zpráv. Přesto bylo ukázáno, že dokonce i pouze tato data lze využít ke konstrukci úspěšných útoků [4], [5]. V [4] je ukázáno, že postačí jediná kolize k vytvoření páru různých samorozbalovacích archivů s identickou haší. To může být zneužitelné například při vkládání zadních vrátek do velkých balíků SW při jejich distribuci. Později bylo za účasti jednoho z autorů [1] ukázáno, jak s využitím schopnosti vytvářet kolize pro libovolný inicializační vektor padělat digitální certifikát [6].

V říjnu 2004 byla publikována práce [3] Hawkese a kol., kde se její autoři snaží odhalit "čínskou metodu" hledání kolizí na základě strohých dat a informací uvedených v [1]. V práci vyšetřují vnitřní diference a podmínky pro zprávy, které by měly být splněny, aby došlo ke kolizi čínským postupem [1]. Byla to první analýza a pokus vysvětlit čínskou metodu. Na základě jednoho páru kolidujících zpráv se správnou inicializační hodnotou IV autoři popsali diferenční schéma, které publikovaná kolize splňuje, a které pravděpodobně bylo v pozadí kolize. Nepodařilo se však vysvětlit, jak toto schéma vzniklo. Dále popsali podmínky, které musí splňovat jedna zpráva z kolidujícího páru tak, aby diferenční schéma bylo splněno. Obdrželi dlouhý seznam podmínek, které musí zpráva splňovat. První sada (tzv. ft-podmínky a Tt-podmínky) vzniká u prvního bloku zprávy při průchodu 64 rundami MD5. Pokud se splní ft-podmínky a Tt-podmínky v prvních 16 rundách (přes 200 podmínek) vhodnou volbou bloku M, zbývá ještě naplnit 39 ft-podmínek a "3.2" Tt-podmínek ve zbývajících rundách, které jsou splněny pouze pravděpodobnostně. Celkem je tak potřeba generovat cca $2^{42.2}$ zpráv M, aby jedna z nich splňovala všechny ft-podmínky a všechny Tt-podmínky z rund 17 - 64. Podobně pro splnění ft- a Tt-podmínek druhého bloku zprávy N je nutné podle [3] generovat $2^{42.2}$ zpráv. Složitost celého útoku je pak 2^{43}. Hawkes a kol. se domnívají, že toto je příliš velká složitost, aby se kolize dala vygenerovat za jednu hodinu. Proto dovozují, že Wang a kol. museli použít ještě nějaký další trik. Tento trik je pochopitelně klíčový.

V našem výzkumu jsme vyšli z výsledků [3] a diferenční schéma jsme také zkoumali z hlediska diferencí aditivních (aritmetický rozdíl modulo 2^{32}) i diferencí binárních (XOR, mod 2), stejně jako [3]. Navíc jsme zkoumání podrobili i další kolidující pár, který byl v [1] vytvořen pro špatnou inicializační hodnotu. Potvrdili jsme, že diferenční schéma platí pro oba kolidující páry, neboť více dat nebylo k dispozici. V našem výzkumu se ukázalo, že některé ft- a Tt- podmínky mohou být splněny více cestami, než zvolil Hawkes a kol. To by mohlo teoreticky vést ke snížení výpočetní složitosti. Narostla by však složitost paměťová a složitost příslušného programu na generování kolizí, a tak jsme touto cestou nešli. Nicméně analýza ft- a Tt-podmínek naznačila, že skutečná složitost nalezení kolize by mohla být ve skutečnosti menší než v teoretickém modelu. Dalším výzkumem pak byla nalezena jiná cesta, jak generovat první bloky kolidujících zpráv velmi rychle. Na standardním notebooku jsme obdrželi první blok zprávy během dvou minut, oproti jedné hodině na počítači IBM p690 [1]. Vzhledem ke krátkosti výzkumu jsme nepokročili v urychlení hledání kolizí v druhém bloku tak jako u prvního bloku, i když jsme dosáhli složitosti významně nižší než 2^{42} podle [3]. O tom svědčí i nalezení první kolize na notebooku za 8 hodin. Podle [1] by však hledání kolizí

druhého bloku mělo být 12 - 240 krát rychlejší než u prvního bloku. Pak by kolize byla na notebooku místo za 8 hodin nalezena během dvou minut.

K nalezení kolizí jsme nepoužili žádný superpočítač, pouze běžné domácí počítače. Autor prováděl své experimenty výhradně na notebooku, kde nalezl jak desetitisíce kolizí prvního bloku, tak i úplné kolize MD5 pro platnou inicializační hodnotu i volené inicializační hodnoty. Pro ověření funkčnosti programu jsme také požádali několik přátel o vyzkoušení na jejich domácích počítačích. Za týden experimentování na počátku března tak byly nalezeny desetitisíce kolizí prvních bloků a desítky úplných kolizí.

Výsledek na běžném notebooku (Acer TravelMate 450LMi, Intel Pentium 1.6 GHz) je tento: během 8 hodin bylo nalezeno 331 kolizí prvního bloku a 1 úplná kolize MD5. Vzhledem k tomu, že nalezení 1 kolize prvního bloku trvalo čínskému týmu 1 hodinu na počítači IBM p690, nalezení 331 těchto kolizí by trvalo cca 331 hodin, což je 40 krát více. Výkony notebooku a velkého počítače lze těžko srovnávat z důvodu různých architektur, ale když uvažujeme, že uvedený počítač je 25 - 50 krát rychlejší než notebook (odhad poskytl Ondrej Mikle na základě poměru bogomips), dostáváme velmi hrubý odhad, že naše metoda hledání kolize prvního bloku je 1000 - 2000 krát rychlejší než v [1]. Naproti tomu hledání kolize druhého bloku je 2 - 80 krát pomalejší. Pokud srovnáme celkový čas hledání úplné kolize u čínského týmu (1.0 až 1.08 hodiny) s naším (8 hodin) na 25 - 50 krát pomalejším stroji, je naše metoda celkově 3 - 6 krát rychlejší. Všechna tato srovnání jsou orientační a autor si nečiní žádný nárok na jejich přesnost (přesné jsou pouze časové údaje).

Ukazuje se však, že

- kolizi MD5 lze dnes vyhledat už i na notebooku,
- naše metoda a čínská metoda [1] se zásadně liší v rychlosti a pravděpodobně i v obsahu (v obou částech výpočtů),
- naše metoda je celkově rychlejší,
- metoda pracuje pro jakoukoli zvolenou inicializační hodnotu.

Poděkování

Rád bych poděkoval svým přátelům za jejich pomoc. Milanu Nosálovi (LEC, s.r.o.) za jeho pomoc při ladění programu, Tomáši Rosovi a Ondřeji Pokornému a Milanu Nosálovi za provádění experimentů na jejich domácích počítačích, Tomáši Jabůrkovi za technickou pomoc s experimenty a Ondreji Mikle za pomoc při překladu příspěvku a všem za cenné připomínky.

Poznámky

V posledním experimentu, který dělal Ondřej Pokorný na svém domácím PC (Intel Pentium, 1 GHz), obdržel za 58 hodin a 32 minut 14 kolizí. To dává ještě optimističtější čas pro nalezení kolize (1 kolize za 4 hodiny a 11 minut) než na autorově notebooku.

Domácí stránka projektu

http://cryptography.hyperlink.cz/MD5_collisions.html

Závěr

Příspěvek ukazuje, že kolizi MD5 lze dnes vyhledat i na notebooku. Metoda pracuje pro jakoukoli zvolenou inicializační hodnotu a je celkově rychlejší než původní čínská metoda [1]. Lze očekávat, že po zveřejnění čínské metody dojde k urychlení hledání kolizí druhého bloku v naší metodě, čímž by se celková časová náročnost vyhledání úplné kolize na notebooku mohla snížit až na 2 minuty.

Literatura

[1] Xiaoyun Wang, Dengguo Feng , Xuejia Lai, Hongbo Yu: Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD, rump session, CRYPTO 2004, *Cryptology ePrint Archive*, Report 2004/199, first version (August 16, 2004), second version (August 17, 2004), http://eprint.iacr.org/2004/199.pdf

[2] Ronald Rivest: The MD5 Message Digest Algorithm, RFC1321, April 1992, ftp://ftp.rfc-editor.org/in-notes/rfc1321.txt

[3] Philip Hawkes, Michael Paddon, Gregory G. Rose: Musings on the Wang et al. MD5 Collision, *Cryptology ePrint Archive*, Report 2004/264, 13 October 2004, http://eprint.iacr.org/2004/264.pdf

[4] Ondrej Mikle: Practical Attacks on Digital Signatures Using MD5 Message Digest, *Cryptology ePrint Archive*, Report 2004/356, http://eprint.iacr.org/2004/356, 2nd December 2004

[5] Dan Kaminsky: MD5 To Be Considered Harmful Someday, *Cryptology ePrint Archive*, Report 2004/357, http://eprint.iacr.org/2004/357, 6 December 2004

[6] Arjen Lenstra, Xiaoyun Wang and Benne de Weger: Colliding X.509 Certificates, *Cryptology ePrint Archive*, Report 2005/067, http://eprint.iacr.org/2005/067

[7] Vlastimil Klima: Several observations regarding Chinese collisions of MD5, 3rd International Scientific Conference *Security and Protection of Information*, Brno, Czech Republic, May 3 - 5, 2005, http://www.unob.cz/spi/defaulten.asp, in preparation

Příloha: Příklady

Příklad: Kolize MD5 se standardní inicializační hodnotou IV

IV podle [2]:

```
 context->state[0] = 0x67452301;
 context->state[1] = 0xefcdab89;
 context->state[2] = 0x98badcfe;
 context->state[3] = 0x10325476;
```

První zpráva:

```
0xA6,0x64,0xEA,0xB8,0x89,0x04,0xC2,0xAC,
0x48,0x43,0x41,0x0E,0x0A,0x63,0x42,0x54,
0x16,0x60,0x6C,0x81,0x44,0x2D,0xD6,0x8D,
0x40,0x04,0x58,0x3E,0xB8,0xFB,0x7F,0x89,
0x55,0xAD,0x34,0x06,0x09,0xF4,0xB3,0x02,
0x83,0xE4,0x88,0x83,0x25,0x71,0x41,0x5A,
0x08,0x51,0x25,0xE8,0xF7,0xCD,0xC9,0x9F,
0xD9,0x1D,0xBD,0xF2,0x80,0x37,0x3C,0x5B,
0x97,0x9E,0xBD,0xB4,0x0E,0x2A,0x6E,0x17,
0xA6,0x23,0x57,0x24,0xD1,0xDF,0x41,0xB4,
0x46,0x73,0xF9,0x96,0xF1,0x62,0x4A,0xDD,
0x10,0x29,0x31,0x67,0xD0,0x09,0xB1,0x8F,
0x75,0xA7,0x7F,0x79,0x30,0xD9,0x5C,0xEB,
0x02,0xE8,0xAD,0xBA,0x7A,0xC8,0x55,0x5C,
0xED,0x74,0xCA,0xDD,0x5F,0xC9,0x93,0x6D,
0xB1,0x9B,0x4A,0xD8,0x35,0xCC,0x67,0xE3.
```

Druhá zpráva:

```
0xA6,0x64,0xEA,0xB8,0x89,0x04,0xC2,0xAC,
0x48,0x43,0x41,0x0E,0x0A,0x63,0x42,0x54,
0x16,0x60,0x6C,0x01,0x44,0x2D,0xD6,0x8D,
0x40,0x04,0x58,0x3E,0xB8,0xFB,0x7F,0x89,
0x55,0xAD,0x34,0x06,0x09,0xF4,0xB3,0x02,
0x83,0xE4,0x88,0x83,0x25,0xF1,0x41,0x5A,
0x08,0x51,0x25,0xE8,0xF7,0xCD,0xC9,0x9F,
0xD9,0x1D,0xBD,0x72,0x80,0x37,0x3C,0x5B,
0x97,0x9E,0xBD,0xB4,0x0E,0x2A,0x6E,0x17,
0xA6,0x23,0x57,0x24,0xD1,0xDF,0x41,0xB4,
0x46,0x73,0xF9,0x16,0xF1,0x62,0x4A,0xDD,
0x10,0x29,0x31,0x67,0xD0,0x09,0xB1,0x8F,
0x75,0xA7,0x7F,0x79,0x30,0xD9,0x5C,0xEB,
0x02,0xE8,0xAD,0xBA,0x7A,0x48,0x55,0x5C,
0xED,0x74,0xCA,0xDD,0x5F,0xC9,0x93,0x6D,
0xB1,0x9B,0x4A,0x58,0x35,0xCC,0x67,0xE3.
```

Společná haš:

```
0x2B,0xA3,0xBE,0x5A,0xA5,0x41,0x00,0x6B,
0x62,0x37,0x01,0x11,0x28,0x2D,0x19,0xF5.
```

Příklad: Kolize MD5 se zvolenou inicializační hodnotou IV

```
context->state[0] = 0xabaaaaaa;
context->state[1] = 0xaaacaaaa;
context->state[2] = 0xaaaadaaa;
context->state[3] = 0xaaaaaaea;

První zpráva:
0x9E,0x83,0x2A,0x4C,0x95,0x64,0x5E,0x2B,
0x2E,0x1B,0xB0,0x70,0x47,0x1E,0xBA,0x13,
0x7F,0x1A,0x53,0x43,0x22,0x34,0x25,0xC1,
0x40,0x04,0x58,0x3E,0xB8,0xFB,0x7F,0x89,
0x55,0xAD,0x34,0x06,0x09,0xF4,0xB3,0x02,
0x83,0xE4,0x88,0x83,0x25,0x71,0x41,0x5A,
0x08,0x51,0x25,0xE8,0xF7,0xCD,0xC9,0x9F,
0xD9,0x1D,0xBD,0xF2,0x80,0x37,0x3C,0x5B,
0x89,0x62,0x33,0xEC,0x5B,0x0C,0x8D,0x77,
0x19,0xDE,0x93,0xFA,0xA1,0x44,0xA8,0xCC,
0x56,0x91,0x9E,0x47,0x00,0x0C,0x00,0x4D,
0x40,0x29,0xF1,0x66,0xD1,0x09,0xB1,0x8F,
0x75,0x27,0x7F,0x79,0x30,0xD5,0x5C,0xEB,
0x42,0xE8,0xAD,0xBA,0x78,0xCC,0x55,0x5C,
0xED,0xF4,0xCA,0xDD,0x5F,0xC5,0x93,0x6D,
0xD1,0x9B,0x0A,0xD8,0x35,0xCC,0xE7,0xE3.

Druhá zpráva:
0x9E,0x83,0x2A,0x4C,0x95,0x64,0x5E,0x2B,
0x2E,0x1B,0xB0,0x70,0x47,0x1E,0xBA,0x13,
0x7F,0x1A,0x53,0xC3,0x22,0x34,0x25,0xC1,
0x40,0x04,0x58,0x3E,0xB8,0xFB,0x7F,0x89,
0x55,0xAD,0x34,0x06,0x09,0xF4,0xB3,0x02,
0x83,0xE4,0x88,0x83,0x25,0xF1,0x41,0x5A,
0x08,0x51,0x25,0xE8,0xF7,0xCD,0xC9,0x9F,
0xD9,0x1D,0xBD,0x72,0x80,0x37,0x3C,0x5B,
0x89,0x62,0x33,0xEC,0x5B,0x0C,0x8D,0x77,
0x19,0xDE,0x93,0xFA,0xA1,0x44,0xA8,0xCC,
0x56,0x91,0x9E,0xC7,0x00,0x0C,0x00,0x4D,
0x40,0x29,0xF1,0x66,0xD1,0x09,0xB1,0x8F,
0x75,0x27,0x7F,0x79,0x30,0xD5,0x5C,0xEB,
0x42,0xE8,0xAD,0xBA,0x78,0x4C,0x55,0x5C,
0xED,0xF4,0xCA,0xDD,0x5F,0xC5,0x93,0x6D,
0xD1,0x9B,0x0A,0x58,0x35,0xCC,0xE7,0xE3.

Společná haš:
//hodnota opravena 8.3.2005, díky Janu Kasprzakovi
0xef,0x2e,0xae,0x54,0xe0,0x34,0x70,0x7c,
0xa2,0x6e,0xb0,0x9b,0x45,0xc7,0xe4,0x87.
```

B. Co se stalo s hašovacími funkcemi?

aneb přehled událostí z poslední doby, část 1

Vlastimil Klíma , http://cryptography.hyperlink.cz , v.klima@volny.cz

Abstrakt

Z praktického hlediska se loučíme s hašovací funkcí MD5. Z teoretického, a pro mnohé i z praktického hlediska, se loučíme s hašovací funkcí SHA-1. Jako poslední prakticky bezpečné hašovací funkce zůstávají ty ve třídě SHA-2 (funkce SHA-256/384/512/224). Hledá se nový koncept hašovacích funkcí, neboť ani třída SHA-2 nemá ty teoretické vlastnosti, které bychom si u kvalitní hašovací funkce představovali.

I. Blok, týkající se zejména MD5

Od srpna 2004 do března 2005 se toho v oblasti hašovacích funkcí událo tolik, že stanovisko k jejich bezpečnosti musela řada lidí dvakrát přehodnotit. Události si stručně připomeneme a okomentujeme. Pro hlubší studium uvádíme odkazy na literaturu. Uvádíme pouze práce stěžejní, neboť se zaměřujeme na praktické dopady.

[WFLY04] X. Wang, D. Feng, X. Lai, H. Yu, "**Collisions for Hash Functions MD4, MD5, HAVAL-128 and RIPEMD**", rump session, CRYPTO 2004, *Cryptology ePrint Archive*, Report 2004/199, http://eprint.iacr.org/2004/199

Tým profesorky Wangové prezentoval 16. 8. a 17. 8. 2004 na rump session konference Crypto 2004 datové kolize pro hašovací funkce MD4, MD5, HAVAL-128 a RIPEMD. Oznámil také schopnost generovat kolize pro libovolný inicializační vektor MD5 a kolizi MD5 během hodiny a čtvrt na velkém počítači IBM p690. Kolize pro MD4 dokázali najít se složitostí odpovídající ručnímu výpočtu. To bylo obzvláště frustrující, neboť kolize MD4, získaná Dobbertinem v roce 1996 byla jediná známá "opravdová" kolize, a bylo k ní nábožně vzhlíženo.

Čínský tým ovšem nepublikoval myšlenky, jak kolize získávat, pouze strohá data. Pozn.: Protože funkce MD5 je z uvedených nejdůležitější, budeme práci dále zmiňovat pouze v souvislosti s MD5.

[HPR04] Philip Hawkes, Michael Paddon, Gregory G. Rose: **Musings on the Wang et al. MD5 Collision**, *Cryptology ePrint Archive*, Report 2004/264, 13 October 2004, http://eprint.iacr.org/2004/264.pdf

V této práci se v říjnu 2004 australský tým pokusil čínskou metodu zrekonstruovat pouze na základě zveřejněných kolizí. Nejdůležitější "čínský trik" se nepodařilo objevit, ale na základě dat z [WFLY04] bylo dobře popsáno diferenční schéma, kterým uveřejněné čínské kolize vyhovují. Naplnění podmínek tohoto schématu bylo však ještě příliš náročné a výpočetně složitější, než ukazovaly výsledky z [WFLY04], a tak práce nepřinesla čistě praktické výsledky.

[OM2004] Ondrej Mikle: **Practical Attacks on Digital Signatures Using MD5 Message Digest**, *Cryptology ePrint Archive*, Report 2004/356, http://eprint.iacr.org/2004/356, 2nd December 2004, http://cryptography.hyperlink.cz/2004/collisions.htm

[DK2004] Dan Kaminsky: **MD5 To Be Considered Harmful Someday**, *Cryptology ePrint Archive*, Report 2004/357, http://eprint.iacr.org/2004/357, 6 December 2004

V těchto dvou pracích z prosince 2004 bylo ukázáno, jak lze využít nikoli schopnost generovat kolize, ale pouhou jednu jedinou datovou kolizi MD5, publikovanou výše, ke konstrukci sofistikovaných útoků. Zejména v práci [OM2004] jsou ukázány velké možnosti. Podle ní lze v konečném důsledku určitým postupem docílit toho, že libovolné dva různé, útočníkem volené soubory, se uživatelům jeví jako naprosto shodné, a to prostřednictvím kontroly haše a digitálního podpisu.

[LWW05] Arjen Lenstra, Xiaoyun Wang and Benne de Weger: **Colliding X.509 Certificates**, *Cryptology ePrint Archive*, Report 2005/067, http://eprint.iacr.org/2005/067

Prvního března 2005 bylo uveřejněno další sofistikované využití kolize MD5, tentokrát pro zvolenou inicializační hodnotu. Prof. Wangové stačilo k účasti na tomto projektu jenom jediné. Do svého programu zadat inicializační hodnotu, poskytnutou zbývajícími dvěma autory. Výsledkem jsou dva různé moduly n kryptosystému RSA, které vedou na stejný otisk a tedy po vložení do příslušného pole certifikátu má celý certifikát stejný digitální podpis příslušné certifikační autority. Byla tím ukázána možnost vytvořit k vydanému certifikátu jiný, který příslušná certifikační autorita ve skutečnosti nevydala….

[VK2005] Vlastimil Klima: **Finding MD5 Collisions – a Toy For a Notebook**, *Cryptology ePrint Archive*, Report 2005/075, http://eprint.iacr.org/2005/075, (v češtině "Nalézání kolizí MD5 - hračka pro notebook", http://cryptography.hyperlink.cz/md5/MD5_kolize.pdf.)

Pátého března jsem oznámil, že dokážu generovat kolize MD5 na domácím počítači, a to stejně jako [WFLY04] pro libovolnou inicializační hodnotu. Od této doby je možné nikoli na velkém počítači s 32 procesory, ale i na notebooku generovat libovolné kolize. Metoda, kterou jsem použil (před publikací čínského postupu), se ukazovala jiná v obou fázích postupu, než u [WFLY04]. V první fázi byla 1000 - 2000 krát rychlejší, v druhé 2 - 80 krát pomalejší a celkově 3 - 6 krát rychlejší. Průměrná doba nalezení kolize na notebooku (Pentium 1.6 GHz) je tak 8 hodin. Čínská metoda ještě nebyla v té době publikována. Proto jsem se k podobnému kroku nechystal ani já.

[WY2005] Xiaoyun Wang and Hongbo Yu: **How to Break MD5 and Other Hash Functions**, http://www.infosec.sdu.edu.cn/paper/md5-attack.pdf.

Profesorka Wangová umístila tento příspěvek na svůj web jak se ukázalo (po mírně detektivním zkoumání, které jsme vedli s Pavlem Vondruškou, viz http://www.crypto-world.info/news/index.php?prispevek=1245) velmi v tichosti někdy v období prvního týdne března. Tato událost, na kterou kryptografická komunita čekala půl roku, tak získala další tajemný přídech. Proběhlo to v tichosti a nenápadně. Bylo to zřejmě tak, že jakmile se prof. Wangová dozvěděla, že její příspěvek byl přijat na konferenci Eurocrypt 2005, dala ho k ostatním pracem na své internetové stránky a nechala to osudu. První to našel nějaký člověk, který si přečetl můj článek [VK2005], pak dal vyhledávat "Wang" a to ho dovedlo až k jejímu článku. Byl schován v čínských znacích, ale anglický název prominoval. Tak jsem se to dozvěděl i já (záznam viz newsgroup sci.crypt). Pak jsme se tomu s Pavlem Vondruškou chvíli věnovali, kdy to tam asi dala a výsledek je výše. Informace o tom je také například na http://www.root.cz/zpravicky/cinsti-vedci-promluvili/ s mým komentářem.

Konečně bylo možné vidět onen čínský trik. Ukázalo se, že jsou dva, a to diferenční schéma a tzv. metoda modifikace zpráv. Jak diferenční schéma funguje, bylo v zásadě prozkoumáno Australany (Číňané se liší v několika překvapivých detailech), ale jak bylo vytvořeno, zůstává stále v čínském šuplíčku. Pouze je poznamenáno, že vzniklo tak, aby bylo výhodné pro pozdější fáze schématu. U metody modifikace zpráv dává tento příspěvek jeden příklad. Dále se uvádí, že k hledání jsou použity i jiné modifikace zpráv. Jinými slovy, metoda zůstala velice zahalena do technických detailů, neboť uvedený příklad nelze nijak obecně využít.

Pokračování výzkumu z [VK2005]
V současné době pracujeme na využití některých myšlenek z [WY2005] pro ještě větší urychlení, dosažené v [VK2005]. Potvrdilo se také, že oba přístupy jsou nikoli diametrálně, ale přesto odlišné.

Tento proud novinek týkajících se hašovací funkce MD5 zatím uzavíráme s tím, že nikdo už nepochybuje o zastaralosti této funkce. Bude ale velmi těžké ji nahradit v existujících aplikacích. Pokračujeme v přehledu s SHA-1.

II. Blok, týkající se zejména SHA-1

[WYY05] Wang X., Yin L., Yu H.: **Collision Search Attacks on SHA1**, February 13, 2005, http://theory.lcs.mit.edu/~yiqun/shanote.pdf

V této práci ukazují plnou kolizi SHA-0 a kolizi SHA-1 pro 58 kroků (z 80). Oznamují též, že jsou schopni nalézt kolizi plnohodnotné SHA-1 se složitostí 2^{69} hašovacích operací. SHA-0 by pokořili se složitostí 2^{33} hašovacích operací. Pokud si uvědomíme, že SHA-0 byla určitou dobu standardem, a že se liší od SHA-1 pouze v jedné operaci v základní smyčce, je to ohromný výsledek.

[LWW05b] Arjen Lenstra, Xiaoyun Wang and Benne de Weger: **Colliding X.509 Certificates based on SHA1-collisions**,
http://www.win.tue.nl/~bdeweger/CollidingCertificates/index.html

Tým Lenstra-Wang-Weger připravuje v těchto dnech spuštění experimentu na nalezení kolize certifikátu, podobně jako v [LWW05], tentokrát ale pro hašovací funkci SHA-1 v certifikátu. To už je velmi závažné, protože většina certifikačních autorit tuto funkci používá, a to jako silnější alternativu k MD5. Scénář je stejný jako předtím. Připraví se dva klíče, vypočte se inicializační hodnota pro kolizi SHA-1, Wangová poskytne kolizi a zbytek je stejný jako předtím. Zbývá generovat kolizi SHA-1. Před měsícem oznámená složitost 2^{69} se ale pravděpodobně podaří snížit na 2^{66}. Pracuje se již jen na získání výpočetního výkonu.

Tím přehled pro tentokrát uzavíráme.

C. Popis šifry PlayFair

Pavel Vondruška, ČESKÝ TELECOM, a.s.,
(pavel.vondruska@crypto-world.info)

1. Historie

Šifru PlayFair navrhl v roce 1854 britský všestranný vědec Charles Wheatstone (6. února, 1802 - 19. října, 1875) a to jako vhodnou šifru pro utajení telegrafických zpráv. Jméno však dostala až podle jeho přítele, který byl velkým propagátorem této šifry, skotského barona, poslance britského parlamentu **Lyon Playfaira** (1. května, 1818) - (29. května, 1898).
Bez něj by se použití šifry nepodařilo prosadit. Lord osobně přesvědčoval o jejích kvalitách kolegy poslance a významné britské státní představitele.

Známý je příběh, kdy baron Playfair předvedl šifru britskému ministrovi zahraničí, který sice uznal, že vypadá kvalitně, ale pro její složitost ji nedoporučil. Baron pak šifru předvedl v blízké místní škole, kde žáci její princip velice rychle pochopili a naučili se ji používat bez chyby velice rychle. Po té, co o tom společně s Wheatstonem informoval ministra zahraničí, ten jim odpověděl:
„Ano, džentlmeni, to je možné, ale naši diplomaté se to nikdy nenaučí!“

Šifra se nakonec prosadila především jako vojenská šifra. Britská armáda ji používala během obou Búrských válek (1880-81, 1899-1902) a byla jí používána i za I. světové války. Australská armáda ji dokonce používala i během svých válečných operacích za druhé světové války.

Výhodou této šifry je, že je daleko hůře luštitelná než jiné klasické „ruční šifry“. Její hlavní předností je, že je odolná (na rozdíl např. od jednoduché záměny) proti frekvenční analýze. Z hlediska kryptoanalýzy se vlastně jedná o bigramovou záměnu – tedy záměnu, kdy dvojice písmen otevřeného textu se zamění za jinou dvojici písmen. Je pochopitelné, že frekvenční analýza bigramů vyžaduje záchyt výrazně většího počtu šifrových textů než luštění jednoduché záměny. Vojenským expertům se proto zdál tento systém pro masové použití v armádě vhodný a spolehlivý (zvláště při dodržení dalších bezpečnostních pravidel, jako pravidelná výměna klíčového slova, nezasílání velkého objemu korespondence…) a dále jeho výhodou byla rychlá výuka uživatele, rychlá příprava šifrového textu, rychlá dešifrace textu a celkově malá cena na masové nasazení (srovnej např. náklady na tisk a distribuci kódové knihy…).

2. Popis systému

Uživatel pomocí šifry PlayFair vytvoří z otevřeného textu šifrový tak, že nejdříve otevřený text podle jednoduchých pravidel upraví a potom jej pomocí abecedního čtverce záměny podle pěti prostých pravidel transformuje (zašifruje). Abecední čtverec záleží na dohodnutém hesle - klíčovém slově.

2.1 Pravidla úpravy vstupního otevřeného textu

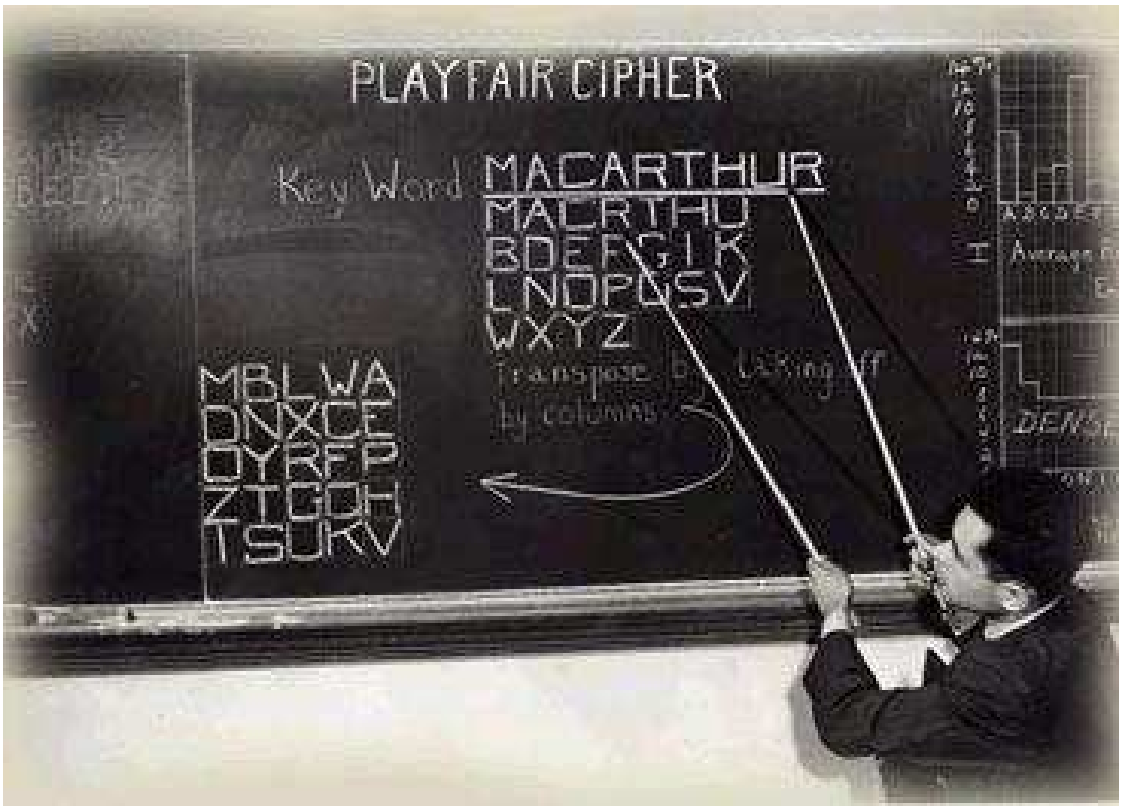
Celý text přepíšeme na text složený pouze z velkých písmen, bez diakritiky a interpunkce a pokud obsahuje text písmeno J, všude ho zaměníme na I (v angličtině se J vyskytuje velmi zřídka, pro češtinu bude vhodnější si pravidla přepsat tak, aby bylo např. nahrazeno písmeno

Q jiným písmenem – řekněme K). Získaný pomocný otevřený text dále rozdělíme do skupin po dvou písmenech (bigramů). Pokud by se v bigramu objevila dvě stejná písmena, musí být oddělena písmenem X a Z. Pokud má původní text lichý počet písmen, doplníme na konec textu opět písmeno X nebo Z.

Příklad – část 1
Postup si předvedeme na tomto otevřeném textu:
Tato šifra je docela jednoduchá!

Odstranění diakritiky a interpunkce, náhrada J za I:
TATO SIFRA IE DOCELA IEDNODUCHA

Rozdělení na bigramy, v případě dvou stejných písmen v bigramu vložení písmen X a Z, doplnění na sudý počet písmen písmenem Z :
TA TO SI FR AI ED OC EL AI ED NO DU CH AZ



Obr. Výuka systému PlayFair
Zdroj : http://library.brooklyn.cuny.edu/about_library/speccoll/Final/Cryptography3c.jpg

2.2 Heslo a abecední čtverec
Nyní se využije domluvené heslo mezi odesílatelem a příjemcem.. Jeho délka by měla být alespoň pět písmen (nedoporučuje se kratší) a každé písmeno v něm může být použito pouze jednou.
Klasický abecední čtverec šifry PlayFair má stranu dlouhou pět písmen. Sestaví se tak, že se nejprve napíše zvolené heslo - klíčové slovo, potom se postupně doplní zbývající písmena mezinárodní abecedy, přičemž písmeno J se vynechá.

Příklad – část 2
Domluvené klíčové slovo zvolíme v našem příkladě slovo: **HESLO**.
Sestavíme abecední čtverec, který bude v tomto případě vypadat takto:

H	**E**	**S**	**L**	**O**
A	**B**	**C**	**D**	**F**
G	**I**	**K**	**M**	**N**
P	**Q**	**R**	**T**	**U**
V	**W**	**X**	**Y**	**Z**

2.3 Šifrování
Šifrování systémem PlayFair je založeno na skutečnosti, že každý bigram v upraveném otevřeném textu se může vyskytnout pouze v jednom ze tří následujících stavů.
Bigram může být společně v jednom řádku, jednom sloupci nebo je každé z písmen bigramu v jiném řádku a sloupci (statisticky nejběžnější situace). Samotné šifrování proto probíhá takto:
2.3.1 Pokud leží obě písmena ve stejném řádku, je každé písmeno bigramu nahrazeno písmenem ležícím v tabulce vpravo od něj. Poslední písmeno v řádku (tedy pokud nemá vpravo od sebe písmeno) se nahradí prvním písmenem téhož řádku.
2.3.2 Pokud leží obě písmena ve stejném sloupci, je každé písmeno bigramu nahrazeno písmenem pod ním. Je-li písmeno v posledním řádku (tedy pokud nemá pod sebou písmeno) je nahrazeno prvním písmenem téhož sloupce.
2.3.3 Pokud je každé z písmen bigramu v jiném řádku a sloupci, je každé písmeno bigramu nahrazeno písmenem nacházejícím se v průsečíku jeho vlastního řádku a sloupce obsahujícího druhé písmeno bigramu. Musí se dodržet pořadí: nejdříve se určí průsečík řádku prvního písmene se sloupcem druhého písmene, potom teprve průsečík řádku druhého písmene se sloupcem prvního písmene. S výhodou se používá představa, že dvě písmena upraveného otevřeného textu vytvářejí uvnitř abecedního čtverce dva vrcholy obdélníka a písmena zašifrovaného textu leží v opačných vrcholech tohoto obdélníka.
2.3.4 Výsledný šifrový text se zapisuje do pětic oddělených jednou mezerou.

Příklad – část 3
Naším prvním bigramem upraveného otevřeného textu je **TA**. Písmena T a A neleží ani ve stejném řádku ani ve stejném sloupci, a proto podle pravidla 2.3.3 jej nahradíme za **PD**.
Druhým bigramem je **TO**. Opět použijeme pravidlo 2.3.3 a dostaneme **UL.**
Na osmý bigram **EL** použijeme pravidlo 2.3.1, neboť písmena E a L leží ve stejném řádku. V tomto případě dostaneme **SO**.
Při převodu jedenáctého bigramu **NO** musíme použít pravidlo 2.3.2, neboť obě písmena N a O leží ve stejném sloupci. Výsledkem je bigram **UF**.

Stejně pokračujeme dále a získáme tento šifrový text:
PD UL EK CU BG LB SF SO BG LB UF FT AS FV

Text rozdělíme do pětic, přičemž poslední skupinu tří písmen doplníme bezvýznamovou skupinou XX .
Výsledný šifrový text : PDULE KCUBG LBSFS OBGLB UFFTA SFVXX

2.4 Dešifrovávání

Opačná transformace (dešifrovávání) probíhá přesně opačně. Nejdříve zašifrovaný text opět rozdělíme na bigramy. Pokud známe heslo, připravíme šifrovací čtverec a podle stejných pravidel 2.3.1-2.3.3 převedeme text na upravený otevřený text. Nakonec doplníme mezery, interpunkci, háčky a čárky, vypustíme přebytečná X a Z a podle smyslu vložíme zpět J, která byla odesílatelem zaměněna za I.

3. Modifikace systému

3.1 Rozšířená šifrovací tabulka

Mimo uvedeného abecedního čtverce o velikosti 5x5 byla rozšířena i verze skládající se ze dvou otevřených abeced a dvou šifrovacích abeced. K jejímu sestavení se používala dvě klíčová slova (dvě smluvená hesla). Použijeme-li např. jako prvé klíčové slovo SECRT (vzniklo ze slova SECRET po vynechání druhého E, neboť jak jsme psali, v hesle se každé písmeno může vyskytnout pouze jednou) a jako druhé klíčové slovo KEYWORD, sestavíme za jejich pomoci následující rozšířenou šifrovací PlayFair tabulku.

A	B	C	D	E		s	e	c	r	t
F	G	H	I	K		a	b	d	f	g
L	M	N	**O**	P		h	i	k	l	**m**
Q	R	S	T	**U**		n	o	p	**q**	u
V	W	X	Y	Z		v	w	x	y	z
k	e	y	w	o		A	B	C	D	E
r	d	a	**b**	c		F	G	H	I	**K**
f	g	h	i	l		L	M	N	O	P
m	n	p	q	**s**		Q	R	S	**T**	U
t	z	v	x	z		V	W	X	Y	Z

Pravidla pro vytváření šifrového textu jsou shodná jako pro čtverec 5x5. Pouze prvé písmeno bigramu se vyhledává v první otevřené abecedě (vlevo nahoře) a druhé písmeno bigramu ve druhé otevřené abecedě (vpravo dole).
Slovo útok (UT OK) se zašifruje pomocí tohoto čtverce takto :
UT se zašifruje na **qs**
OK se zašifruje na **mb**

Poznamenejme, že z hlediska luštění tato modifikace nepřináší vyšší bezpečnost.…

Odkazy:
Šifrování ON-LINE http://www.simonsingh.net/The_Black_Chamber/playfaircipher.htm
Playfair cryptanalysis http://www.umich.edu/~umich/fm-34-40-2/ch7.pdf

D. První rotorové šifrovací stroje …

Pavel Vondruška, ČESKÝ TELECOM, a.s.,
(pavel.vondruska@crypto-world.info **)**

Těsně po skončení první světové války – hned ve čtyřech různých zemích a to zcela nezávisle na sobě – zažádali různí vynálezci o patent na šifrovací stroj, jehož základem byly otočné rotory. Mezi nimi byl také přístroj, který se po několika vylepšeních stal jednoznačně nejznámější šifrovacím strojem na světě – Enigma německého vynálezce Arthura Scherbia. Zájem o vzestup a pád tohoto přístroje, který tak významně ovlivnil průběh druhé světové války, dodnes do řad kryptologů přivádí další mladé adepty o tuto překrásnou vědu.

Seznamme se v této krátké glose se jmény a osudy dalších vynálezců a řekněme si něco o obchodním úspěchu (či spíše neúspěchu) jejich přístrojů.

Mimo již zmíněného německého vynálezce **Arthura Scherbia (1878 -1929),** který svůj patent nahlásil 23. 2. 1918, se zabývali šifrátory na obdobném rotorovém principu jako Enigma další vynálezci a to Americe, Holandsku a Švédsku.

- **Edward Hebern (USA, 1869-1952), patent 1917, 21.3.1921**

Jeden z prvních vynálezců, který se myšlenku rotorových šifrovacích strojů pokusil uplatnit a který ve svůj vynález (ostatně jako všichni zde jmenovaní) hluboce věřil, byl americký vynálezce Edward Hebern (1869 –1952). Během svého života podal více patnetů na stroje založené na uvedeném principu.

Na doprovodném obrázku je jeden ze starších typů jím vyráběných šifrátorů. Přesné datum podání patentu k tomuto zařízení se mi nepodařilo zijistit, v jednom ze zdrojů je uváděn rok 1917. Pozdější patent, na již dokonalejší zařízení, pochází z března roku 1921 a má číslo # 1,510,441.

V polovině 20. let 20. století začal Hebern stavět továrnu nákladem 380 000 dolarů. Prodal však jen dvanáct přístrojů, celkem asi za 1 200 dolarů, a roku 1926 byl nespokojenými akcionáři pohnán k soudu a podle kalifornského obchodního práva shledán vinným.

Singh ve své knize Kniha kódů a šifer uvádí, že příčinou jeho obchodního neúspěchu bylo to, že právě v tomto období se začala nálada americké společnosti měnit. Prezidentem se stal Herbert Hoover, který se pokusil zahájit novou éru mezinárodních vztahů. Jeho ministr zahraničí Henry Stimson vyslovuje svůj proslulý výrok, že „gentleman nečte cizí dopisy“. Stát, který věří, že není správné číst cizí dopisy, časem začne věřit i tomu, že jeho korespondenci také nikdo nečte, takže nevidí důvod pro pořízení kvalitních šifrovacích strojů. Osobně se však domnívám, že rozhodujícím faktorem, proč zařízení Edwarda Heberna neuspělo při prodeji armádě nebo obecněji „státu“ (diplomatické služby, policie atd.), bylo to, že zařízení nemělo pro svůj jednoduchý design z hlediska bezpečnosti velkou odolnost.

Hebern totiž nevěděl, že jeden z velikánů kryptoanalýzy William F. Friedman (24.9, 1891 – 12.11, 1969) podrobil jeho zařízení analýze a ve své práci demonstroval, že zařízení je luštitelné.

- **Hugo Alexander Koch (Holandsko, 1870-1928), patent 7. 10. 1919**

Přístroj na podobném principu (pohyblivé rotory) si v roce 1919 nechal patentovat Hugo Alexander Koch (patent č. 10 700). Ani jemu se však nepodařilo svůj nápad na vývoj a prodej šifrovacích strojů proměnit v obchodní úspěch a roku 1927 (podle jiných zdrojů 1928) svá patentová práva prodává a přebírá je firma Arthura Schrebia. Díky tomu se v návrhu Enigmy objevuje další vylepšení, tzv. Kochův reflektor.

Nebyl však první, kdo se v Holandsku zabýval rotorovými šifrátory. Prvními, kdo je vyvinul a postavil pro využití v holandském námořnictvu, byli dva důstojníci **Theo A. van Hengel (1875 – 1939)** a **R. P. C Spengler (1875 – 1955)** a to již během první světové války – přesněji v roce 1915. Jejich práce však byla utajena a nezachovalo se žádné z jimi navržených zařízení. Jejich jména a práce tak upadla v zapomnění, a proto lze předpokládat, že o těchto šifrátorech H. A. Koch nevěděl a pracoval zcela samostatně. Právem mu tedy patří označení vynálezce.

- **Arvid Damm (Švédsko, ?-1927), patent 10. 10. 1919**

Ve Švédsku získal patent na rotorový šifrátor zakladatel firmy AB Cryptograpf Arvid Damm. Byl však obchodně neúspěšný a v roce 1921 firmu společně s patentem kupují Karl Hagelin a Emanuel Nobel. Syn Karla Hagelina **Boris Hagelin (1892-1983)** se rozvojem původní myšlenky rotorového šifrátoru dále zabýval a neustále jej zdokonaloval. V roce 1948 se stěhuje ze Švédska do Švýcarska. Zde zakládá firmu Crypto AG. Firma v následujících letech získala světovou proslulost prodejem šifrátorů, které koncepčně navázaly na původní modely (jeden z pozdějších modelů je na doprovodném obrázku). Dodnes Crypto AG úspěšně působí na trhu s kryptologií.

Použité zdroje :

[1] Karl de Leeuw: The dutch invention of the rotor machine, 1915 - 1923. Cryptologia 27 (2003), 73 - 94
[2] Simon Singh : Kniha kódů a šifer, DoKořán, Praha 2003, str.137-138
[3] Bengt Beckman: ”Svenska kryptobedrifter”, 1996
[4] http://www.staff.uni-mainz.de/pommeren/Kryptologie/Klassisch/4a_ZylRot/HistRot.html
[5] http://www.meydaonline.com/crypto/history/hebern.htm
[6] http://www.jproc.ca/crypto/hebern_1.html
[7] http://www.geocities.com/ResearchTriangle/Node/3751/cypher.html

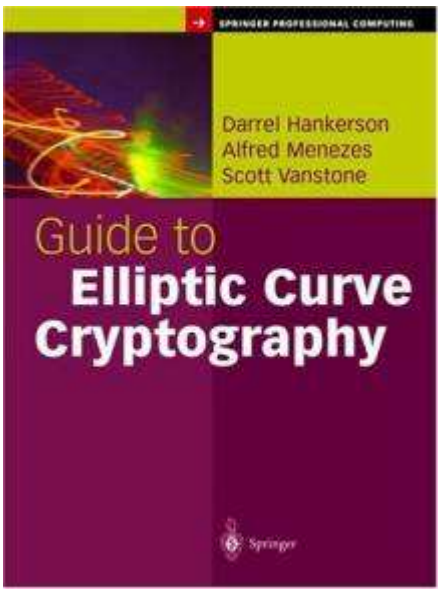
E. Recenze knihy
Guide to Elliptic Curve Cryptography
Darrel Hankerson, Alfred Menezes and Scott Vanstone

Kniha je obsáhlou příručkou praktických aspektů kryptografie eliptických křivek. Její autoři jsou známí profesionálové (dva z autorů jsou také autory knihy Handbook of Applied Cryptography z roku 1996). Je určena všem zájemcům o eliptickou kryptografii (bezpečnostním profesionálům, vývojářům atd.).

Titul: Guide to Elliptic Curve Cryptography
Autoři: Darrel Hankerson, Alfred Menezes and Scott Vanstone
Vydalo: Springer-Verlag, December 2003
ISBN: 0-387-95273-X; 332 pages

Eliptické křivky vstoupily do kryptografie pionýrskými pracemi Neala Koblitze a Victora Millera uprostřed osmdesátých let. Schémata eliptické kryptografie zajišťují tytéž funkcionality jako známý RSA kryptosystém. Jejich bezpečnost je založena na složitosti řešení úlohy diskrétního eliptického algoritmu. V současnosti nejlepší známé algoritmy pro řešení této úlohy mají plně exponenciální charakter (na rozdíl od subexponenciálního charakteru algoritmů pro řešení faktorizace celých čísel) a je tak umožněna konstrukce kryptografických algoritmů s podstatně kratším klíčem než je tomu v případě RSA. Např. bezpečnost eliptického algoritmu s délkou klíče 160 bitů odpovídá (zhruba) bezpečnosti algoritmu RSA s délkou klíče 1024 bitů.

Samotná kniha je orientována prakticky, čtenář zde nenajde vyslovená "top topic" z teorie eliptických křivek, ale kniha především obsahuje veškeré aspekty důležité pro praktické implementace. Kniha sestává celkem z pěti kapitol.



První kapitola je úvodem do problematiky kryptografie veřejného klíče. Obsahuje přehled základních schémat (RSA, diskrétní algoritmus) včetně základních postupů eliptické kryptografie. Obsahuje také stručný přehled známých algoritmů pro faktorizaci celých čísel, řešení diskrétního logaritmu a eliptického diskrétního logaritmu. Na tomto základě je provedeno krátké porovnání požadované délky klíčů v těchto systémech (pro zajištění shodné úrovně bezpečnosti).

Při implementacích eliptických kryptosystémů máme možnost některých voleb. Je třeba vybrat použité konečné těleso (binární, prvočíselné), zvolit reprezentaci prvků tohoto tělesa, algoritmy pro aritmetické výpočty v tomto tělese. Dále je třeba zvolit vhodnou eliptickou křivku včetně reprezentace bodů této eliptické křivky a algoritmy pro výpočty v příslušné eliptické aritmetice. Konečně je třeba se rozhodnout pro použití některého z kryptografických protokolů (a opět zvolit použitou aritmetiku).

Kapitola 2. je úvodem do problematiky konečných těles. Obsahuje mj. přehled používaných algoritmů (násobení n-bitových celých čísel, umocňování, redukce modulo, inverze). Je zde dán přehled vlastností tzv. NIST-prvočísel, což jsou prvočísla obsažená v normě FIPS 186-2 a doporučená pro volbu velikosti těles pro eliptické křivky. Binární tělesa mají specifické

vlastnosti umožňující konstrukce speciálních algoritmů - přehled těchto postupů je dán v odstavci 2.3.

Kapitola 3. je úvodem do problematiky eliptických křivek. Jsou zde ukázány různé postupy pro reprezentaci bodů eliptické křivky a postupy aritmetiky eliptických křivek. Nejprve jsou uvedeny postupy pro zavedení struktury grupy na eliptické křivce a dále důležité využití projektivních souřadnic pro aritmetické výpočty na eliptické křivce. Je zde proveden podrobný přehled těchto algoritmů - optimalizace praktických postupů vedla již ke vzniku obsažné teorie. V odstavci 3.4 jsou analyzovány tzv. Koblitzovy křivky, v odstavci 3.5 pak křivky, pro které lze efektivně počítat endomorfizmy - oboje jsou typy křivek, které umožňují efektivní provádění některých výpočtů. Kapitola obsahuje další analýzy algoritmů včetně přehledu jejich efektivity pro křivky NIST.

Kapitola 4. dává popis eliptických kryptografických protokolů pro řadu situací - digitální podpisy, šifrování veřejným klíčem, ustavení klíčů atd. Jsou zde zmíněna např. následující schémata - ECDSA, EC-KCDSA (podpis); ECIES, PSEC (šifrování); ECMQV (dohoda na klíči). Dále je v této kapitole uveden přehled současného stavu algoritmů pro řešení úlohy eliptického diskrétního logaritmu. Jsou diskutovány následující typy útoků - Pohlig-Hellman, Pollard´s rho s variantami, tzv. index-calculus útok a útoky využívající konstrukce izomorfizmů (Weilovo párování, Tate párování , metoda Weilova spádu).

Poslední kapitola 5 obsahuje vybrané aplikační aspekty eliptické kryptografie - jak v software, tak i v hardware. Je zde poukázáno na útoky z postranních kanálů, které využívají další informace unikající z kryptografických zařízení (vyzařování, informace o spotřebě proudu, chybové hlášky).

Kniha obsahuje tři přílohy. Příloha A obsahuje specifikace vhodných konkrétních parametrů pro volbu eliptických kryptosystémů. Příloha B je přehledem souvisejících norem, které popisují mechanizmy eliptických křivek. Příloha C je seznamem vybraných softwarových nástrojů, které jsou použitelné pro provádění příslušných teoreticko-číselných výpočtů. Knihu doplňuje rozsáhlá bibliografie.

Jak již bylo řečeno výše, kniha je zaměřena především na praxi eliptické kryptografie. Čtenáře, které více zajímá související matematická teorie, lze odkázat na knihu autora *Lawrence C. Washingtona:* **Elliptic Curves: Number Theory and Cryptography** (Chapman & Hall/CRC, May, 2003).
Dále - na červen 2005 je připravováno vydání knihy skupiny autorů *Henri Cohen, Gerhard Frey, Roberto Avanzi, Christophe Doche, Kim Nguyen, Tania Lange*: **Elliptic and Hyperelliptic Curve Cryptography: Theory and Practice** (Chapman & Hall/CRC, June, 2005). Tato kniha bude obsahovat vysoce aktuální přehled problematiky teorie související s problematikou eliptické kryptografie (jeden z autorů - profesor Frey je například tvůrcem metody Weilova spádu, zatím nejefektivnější metody pro řešení úlohy eliptického diskrétního logaritmu - i když tato metoda funguje jen v některých situacích). Teoretické zaměření však není výlučné, kniha bude obsahovat i pohled na řešení celé řady implementačních otázek.
Samozřejmě, pokud čtenáře zajímá ryzí teorie eliptických křivek, lze doporučit celou řadu titulů, zmiňme alespoň některou z knih *J. Silvermana* - např. **Advanced Topics in the Aritmetic of Elliptic curves**, Springer-Verlag 1994.

Jaroslav Pinkava
jaroslav.pinkava@pvt.cz

F. O čem jsme psali v březnu 2000 – 2004

Crypto-World 3/2000

Crypto-World 3/2001

Crypto-World 3/2002

Crypto-World 3/2003

Crypto-World 3/2004

G. Závěrečné informace

1. Sešit

Crypto-World je oficiální informační sešit "Kryptologické sekce Jednoty českých matematiků a fyziků" (GCUCMP). Obsahuje články podepsané autory. Případné chyby a nepřesnosti jsou dílem P.Vondrušky a autorů jednotlivých podepsaných článků, GCUCMP za ně nemá odbornou ani jinou zodpovědnost.

Adresa URL, na níž můžete najít tento sešit (zpravidla 3 týdny po jeho rozeslání) a předchozí sešity GCUCMP, denně aktualizované novinky z kryptologie a informační bezpečnosti, normy, standardy, stránky některých členů a další související materiály:

http://crypto-world.info

2. Registrace / zrušení registrace

Zájemci o e-zin se mohou zaregistrovat pomocí e-mailu na adrese pavel.vondruska@crypto-world.info (předmět: Crypto-World) nebo použít k odeslání žádosti o registraci elektronický formulář na http://crypto-world.info. Při registraci vyžadujeme pouze **jméno a příjmení, titul**, pracoviště (není podmínkou) a **e-mail adresu** určenou k zasílání kódů ke stažení sešitu.

Ke **zrušení registrace** stačí zaslat krátkou zprávu na e-mail pavel.vondruska@crypto-world.info (předmět: ruším odběr Crypto-Worldu!) nebo opět použít formulář na http://crypto-world.info. Ve zprávě prosím uveďte jméno a příjmení a **e-mail adresu**, na kterou byly kódy zasílány.

3. Redakce

E-zin Crypto-World

Redakční práce:	Pavel Vondruška
Stálí přispěvatelé:	Pavel Vondruška
	Jaroslav Pinkava
Jazyková úprava:	Jakub Vrána
Přehled autorů:	http://crypto-world.info/obsah/autori.pdf

NEWS	Vlastimil Klíma
(výběr příspěvků,	Jaroslav Pinkava
komentáře a	Tomáš Rosa
vkládání na web)	Pavel Vondruška

Webmaster Pavel Vondruška, jr.

4. Spojení (abecedně)

redakce e-zinu	ezin@crypto-world.info ,	http://crypto-world.info
Vlastimil Klíma	v.klima@volny.cz ,	http://cryptography.hyperlink.cz/
Jaroslav Pinkava	jaroslav.pinkava@pvt.cz ,	http://crypto-world.info/pinkava/
Tomáš Rosa	t_rosa@volny.cz ,	http://crypto.hyperlink.cz/
Pavel Vondruška	pavel.vondruska@crypto-world.info ,	http://crypto-world.info/vondruska/index.php
Pavel Vondruška,jr.	pavel@crypto-world.info ,	http://webdesign.crypto-world.info
Jakub Vrána	jakub@vrana.cz ,	http://www.vrana.cz/