

Crypto-World

Informační sešit GCUCMP

ISSN 1801-2140

Ročník 11, číslo 2/2009

15.únor 2009

2/2009

Připravil: Mgr. Pavel Vondruška

Sešit je přednostně distribuován registrovaným čtenářům.

Starší sešity jsou dostupné na adrese

<http://crypto-world.info>

(1310 registrovaných odběratelů)



Obsah :	str.
A. Blue Midnight Wish, kandidát na SHA-3 aneb poněkud privátně o tom, jak jsem k BMW přišel (V. Klíma)	2-12
B. Nastal čas změn (nejde o Obamův citát, ale o používání nových kryptografický algoritmy) (P. Vondruška)	13-17
C. Pozvánka na konferenci IT-Právo	18-19
D. O čem jsme psali v únoru 1999-2008	20-21
E. Závěrečné informace	22

Příloha: ---

A. Blue Midnight Wish, kandidát na SHA-3 aneb poněkud privátně o tom, jak jsem k BMW přišel

Vlastimil Klíma, nezávislý konzultant a kryptolog, Praha,
(<http://cryptography.hyperlink.cz>, v.klima@volny.cz)

K čemu nám bude další hašovací funkce?

Ve světě existuje stovky kryptografických norem, nástrojů, které jsou standardizovány. A přesto se neustále vyvíjejí nové. Místo toho, aby toto úsilí skončilo a byla konečně vynalezena rychlá a kvalitní šifra, hašovací funkce, podpisové schéma, atd. tak tisíce výzkumníků maří čas přípravou nových kryptografických technik. Velké elektronické a SW firmy staví své profesionální kryptologické týmy, které jim mohou závidět mnohé státní služby, a vyvíjejí nové a nové kryptografické algoritmy, nástroje, techniky, protokoly atd. Avšak my víme, že chtít po někom, aby konečně vyvinul kvalitní šifru (a ti kryptologové dali už pokoj), je stejné jako chtít, aby se už konečně vyvinul kvalitní počítač. Tento proces neskončí.

Kryptografie prorůstá elektroniku

Pro nové flash disky, hard disky, mikroprocesory, paměti, ale hlavně pro narůstající objemy komunikací je potřeba nové šifry. Protože ty staré jsou pomalé. To je jeden rozměr problému, dejme tomu hloubka, odpovídající prohlubujícím se nárokům na daný kryptografický nástroj. Druhým rozměrem je šířka problému. Kryptografie nachází použití stále v nových oblastech a její pole působnosti se rozšiřuje. Nedávno jsme například řešili zajištění ochrany komunikace domácích elektroměrů s centrem jejich řízení v ČR. A požadavky přibývají. Staré šifry, haše a podpisy jsou zapomenuty a požadují se nové. Jenže elektrotechnika svoje meze a věda také. Kromě toho se zapomělo na fakt, že dnešní prakticky využitelná kryptografie je postavena na nedokazatelných principech. Existují výjimky, ale jsou drahé a nepraktické. Ostatní "masová" kryptografie používá nástroje, které mohou být prolomeny. Opakuji a podtrhuji, že prakticky všechny dnešní kryptografické nástroje mohou být prolomeny. To, že se to neděje ze dne na den, je zásluha kryptologů, kteří je navrhovali. Že předvídali nepředvídatelné a maximálně tento v samém základu prohraný proces zpomalili a znesnadnili. Nicméně čas od času se stane a může stát, že nějaká technika je doporučena ke stažení. To je příklad standardů MD4, MD5, SHA-0, SHA-1 (platí jen do konce příštího roku). Místo nich však lze používat třídu hašovacích funkcí SHA-2 a jak jsme zmínili výše, z hlediska rostoucích požadavků elektronického průmyslu a komunikací, se už nyní připravuje SHA-3.

Mezinárodní soutěž na standard

Aby nový standard SHA-3 mohl obstát v nových zařízeních a komunikacích, jsou na něj kladena kritéria, která v prvním čtení vypadají jako zcela nesmyslná: Musí být bezpečnější, flexibilnější na různých platformách a rychlejší než SHA-2 (dosud neprolomený). Mezi bezpečností a rychlostí je jak víme triviální a odvěký rozpor - čím bezpečnější algoritmus, tím je pomalejší. Teď tomu má být naopak. Je to nesmysl? Kupodivu není, protože i elektronika je tlačena do stejného problému, například u disků a pamětí se požaduje, aby byly "větší", ale zároveň "menší" (větší kapacitně, menší rozměrově). Co to znamená? Teoretici musí přijít s novými myšlenkami, novou "kryptografickou technologií", která umožní, aby konkrétně hašovací funkce byla bezpečnější a přitom rychlejší. Když si projdete oficiální požadavky na nový standard, stále se zde hovoří o (vyšší) výkonnosti, (vyšší) flexibilitě a (vyšší) bezpečnosti. Explicitně je to pak vyjádřeno v citaci viz dále. SHA-3 má vzniknout podobně jako AES na základě vypsání mezinárodní veřejné soutěže. Je to skutečně monstrózní akce, která začala v roce 2006 a skončí 31.12.2012, vše pod veřejnou mezinárodní kontrolou.

Detaily naleznete na domácí stránce projektu [1], kterou jistě znáte a aktualizované novinky se snažím přinášet na [2].

In Federal Register / Vol. 72, No. 212 / Friday, November 2, 2007 / Notices

Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA-3) Family

on page 3, NIST states:

NIST expects SHA-3 to have a security strength that is at least as good as the hash algorithms currently specified in FIPS 180-2, and that this security strength will be achieved with significantly improved efficiency.

Obr.: výňatek z požadavků na SHA-3

Česká účast

Do soutěže se přihlásilo 64 algoritmů a do prvního kola jich bylo NISTem propuštěno 51. Vítěz musí projít několika koly, a než bude vybrán, bude vše kontrolováno, kritizováno a přemíláno na několika speciálních mezinárodních konferencích. Soutěž prostřednictvím svých kryptologů obeslaly firmy jako STMicroelectronics, Microsoft, Sony, IBM, RSA, MIT, PGP, Gemalto, Intel, Hitachi, Hifn a známá jména jako Rivest, Schneier a další. Češi jsou podepsáni pod dvěma kandidáty. Pod algoritmem EDON-R je to prof. Aleš Drápal z MFF UK a Vlastimil Klíma, jeho vlastníkem a vynálezcem je Makedonec prof. Danilo Gligoroski, působící nyní na technické univerzitě v Norsku. Norský tým také pomáhal Danilovi s druhým algoritmem s poetickým názvem Blue Midnight Wish (BMW). Jeho vynálezcem a vlastníkem je dvojice Gligoroski-Klíma. Shodou okolností (nebo spíše výsledkem trpělivé práce Gligoroského) jsou oba zmíněné algoritmy v čele rychlostního pelotonu. S určitým odstupem za nimi následuje skupina algoritmů Skein, TIB3 a SHAMATA.

Rychlost

Jak je rychlost důležitá a sledovaná, ukazují měření, provedená pro různé délky zpráv na více než 50 platformách, celkem se jedná o stovky srovnávacích grafů [8]. Proto je také těžké uvádět nějaká čísla. Dalšími faktory jsou délka zprávy, paměť, programovací jazyk, kompilátor, typ a šířka slova procesoru, operační systém a další. Nicméně velmi orientačně lze srovnání vidět z tabulky 1, kde je uveden počet cyklů procesoru, který je potřeba na zpracování jednoho bajtu zprávy. Hašovací funkce nabízí navíc 4 velikosti výstupního kódu (224, 256, 384, 512), v následujících tabulkách je v řádku uveden počet cyklů pro 256/512-bitový výstup a ve sloupci je 64 a 32 bitová architektura CPU a v posledním sloupci požadavky algoritmu na paměť v bajtech. Počet cyklů je NISTem uznávané měřítko rychlosti (i když nemůže být univerzálně objektivní). Jedná se o komplikovanou věc, protože některé hašovací funkce mají tzv. inicializační část, která spotřebuje vždy týž konstantní čas. U některých přihlášených funkcí také záleží na délce zprávy (než se "rozběhnou", jsou třeba řádově pomalejší než na dlouhých zprávách). Nicméně tyto nevýhody se všechny poměrně dobře promítají do jedné měřitelné veličiny, kterou je počet cyklů, které jsou spotřebovány NISTem definovaným procesorem v definovaném referenčním prostředí (průměrně) na jeden bajt zprávy.

Docela dobrá přehledná následující tabulka je převzata z [11] (komentáře viz tamtéž). Tabulka uvažuje některé algoritmy za prolomené. Podle normálních kryptografických

zvyklostí by tomu tak skutečně bylo, ale NIST to může ještě změnit, pokud se mu některý algoritmus bude zdát nadějný s nějakou opravou.

Algoritmus	rychlost na 64-bit CPU	rychlost na 32-bit CPU	Paměť (B)	Poznámky
"Dobré" algoritmy, rychlejší než 10 cyklů na bajt				
Edon-R	4.30/2.29	6.46/10.0	256/512	
Blue Mid- night Wish	7.85/4.06	8.63/13.4	264/528	
TIB3	7.68/6.24	13/4.95		64bitový kód může být pravděpodobně rychlejší
Skein	7.6/6.1	32.8/32.5	100/200	
SHAMATA	8/11	15/22		
"Dobré" algoritmy, pomalejší než 10 cyklů na bajt				
SIMD	11/85	12/118		512-bit nepoužívá SSE optimalizaci
Arirang	15/11.3	20.1/55.2		
BLAKE	16.7/12.3	61.7/28.3		
Shabal	12.3	16.2		
LUX	14.9/12.5	16.7/28.2		
Keccak	12.5/25	53.3/106		
Luffa	13.4/23.2	13.9/25.5		
Aurora	15.7/27.4	25.4/46.9		
Twister	15.8/17.5	35.8/39.6		
CHI	24/16	49/78		

Algoritmus	rychlost na 64-bit CPU	rychlost na 32-bit CPU	Paměť (B)	Poznámky
JH	16.8	21.3		
Grøstl	22.4/30.1	22.9/37.5		
LANE	25.7/145	40.5/152		
SHAvite-3	26.7/38.2	35.3/55		
ESSENSE	39/27			
Echo	28.5/53.5	32.5/61		
Hamsi	?	?		
"Pěkné" algoritmy				
MD6	28/44	68/106	> 700	Vyžadovaná paměť je příliš velká pro čipové karty
SANDstorm	37/95	62/297		Paměť > 650 bajtů není praktická pro čipové karty. Používá se tabulka.
Lesamnta	54.5/59.2	54.5/59.2		Pomalé
SWIFFTX	57	57(?)		Příliš pomalé
Fugue	61/133	36/75		Příliš pomalé
"Pomalé" algoritmy				
CubeHash	160	200		Příliš pomalé
Crunch	161/446			Příliš pomalé
ECOH	>1000			Příliš pomalé
FSB		324/507		Příliš pomalé
"Prolomené" algoritmy				

Algoritmus	rychlost na 64-bit CPU	rychlost na 32-bit CPU	Paměť (B)	Poznámky
Abacus	37.6	37.6		Prolomené
Blender				Prolomené
Cheetah	10.5/15.6	15.3/83.8		Útok prodloužením zprávy
DCH				Prolomené
Dynamic SHA	27.9/47.2	27.9/47.2		Útok prodloužením zprávy
Dynamic SHA2	21.9/67.2	21.9/67.3		Útok prodloužením zprávy
EnRUPT				Prolomené
Khichidi-1				Prolomené
MCSSHA-3		60		Prolomené
MeshHash	13.7/18.5	42.5/67.3		
NaSHA	28.4/29.4	39		
Sarmal	9.4/10.9	19.2/23.3		
Sgàil	61			Prolomené
Spectral Ha- sh				Prolomené (near collisions)
StreamHash				Prolomené
Tangle				Prolomené
Vortex	46.3/56.1	69.4/90.1		Korelace výstupních bitů, rychlost < 3 cykly na bajt s použitím budoucích čipů Intel !
"Vyřazené" algoritmy, které už nejsou součástí soutěže				

Algoritmus	rychlost na 64-bit CPU	rychlost na 32-bit CPU	Paměť (B)	Poznámky
Boole	7.68/7.68	21.5/21.5		Withdrawn
HASH 2X				Prolomené
Maraca	5.3			Pomalé pro krátké zprávy (například pro IPsec autentizační paket o 40 bajtech potřebuje > 6kB paměti (nemožné pro čipové karty))
NKS2D				Prolomené
Ponic	3000	7000		Prolomené, pomalé
WaMM	360	360		Odstoupil
Waterfall	16.3	16.3		Odstoupil

Tabulka 1 [11]: Rychlost je uvedena v počtu cyklů CPU na bajt, a to před lomítkem pro 256 bitový výstup hašovací funkce a za lomítkem pro 512bitovou haš, požadovaná paměť je také uvedena pro 256/512 bitový výstup

Konkurence

Někteří autoři (včetně BMW a EDON-R) publikovali přihlášené kandidáty dříve. Týmy kryptologů se pak bavily tím, jak rychle vyřadí své konkurenty. Byla to napínavá podívaná a soutěž, kde "podražení konkurenta" je přímo smyslem a součástí pravidel hry. Za uplynulé dva měsíce tak bylo "prolomeno" 17 kandidátů. Některé zcela (například přímo nalezení kolizí), jiné byly jen "škrábnuty". Neoficiálně, nicméně fakticky, zbylo 34 algoritmů, které budou prezentovány za několik dní (25. - 28. 2.) na konferenci NIST v Leuvenu [7]. Jsou to ARIRANG, AURORA, BLAKE, Blue Midnight Wish, CHI, CRUNCH, CubeHash, ECHO, ECOH, Edon-R, ESSENCE, FSB, Fugue, Grøstl, Hamsi, JH, Keccak, Lane, Lesamnta, Luffa, LUX, MD6, SANDstorm, Shabal, SHAMATA, SHAvite-3, SIMD, Skein, Spectral Hash, SWIFFTX, TIB3, Twister a Vortex a tři, které jsou považovány za prolomené, ale ještě se samy nevzdaly (Cheetah, EnRUPT, NaSHA, Sarmal). Některí autoři doufají, že nalezené chyby jim NIST odpustí, a umožní tzv. opravy, ale pravděpodobně budou zklamáni. Kdyby NIST připustil takové opravy, soutěž by nikdy neskončila. To je jistě škoda (třeba u algoritmu Cheetah, kde byla nalezena vážná chyba, ale s velmi jednoduchou nápravou), ale NIST má problém opačný, a to jak ze hry vyřadit i ty dobré algoritmy. Plánuje zúžit portfolio na 15 kandidátů, na něž by se mohli kryptoanalytici soustředit a aby se tím toto úsilí mohlo koncentrovat, namísto rozptylování. Oficiálně totiž soutěž vzdalo jen 9 algoritmů, a tak ze

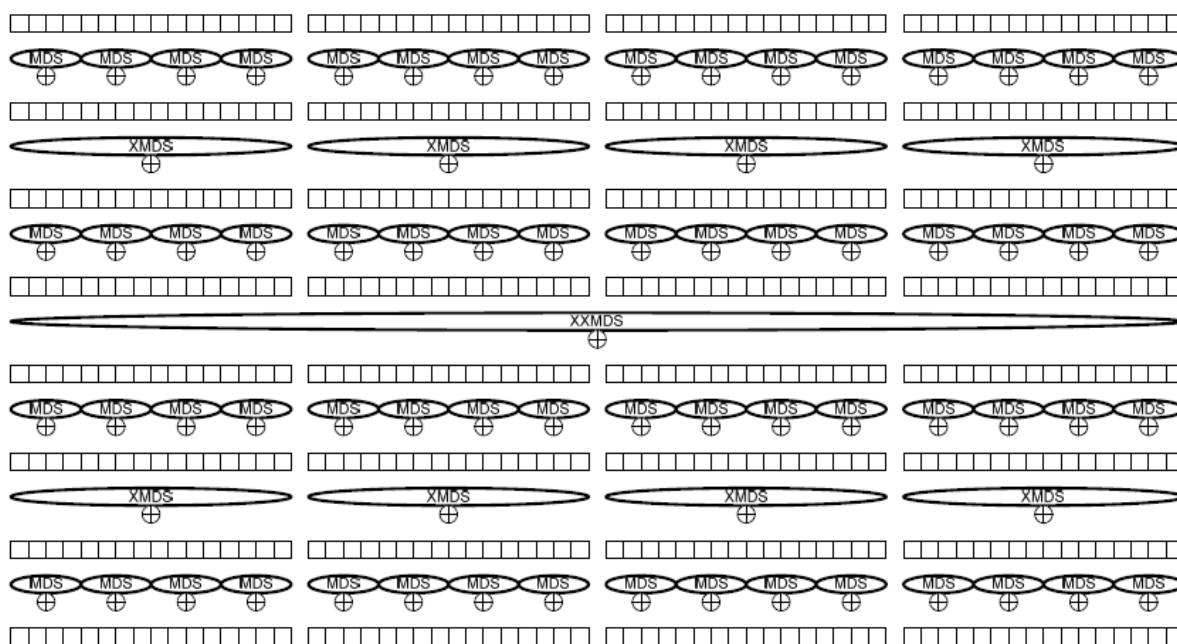
zbývajících 42 jich zbývá vyřadit 25. Jakým způsobem je vyřazovat, k tomu se má také konat velká diskuse na konferenci v Leuvenu.

Vnitřní problém kryptologů

Začneme od konce, tj. od toho, co je vidět nyní, když soutěž všechny kandidáty odhalila. Je vidět, že některé týmy nerespektovaly jakoby "nesmyslné" požadavky na současné zvýšení rychlosti i bezpečnosti, a přidržely se klasických postupů. Pochopitelně nikdo neviděl do karet konkurenci, takže si musel sám vybrat cestu jak překonat tento rozpor.

HDN

Když jsem ještě nějaký čas před zahájením soutěže uvažoval o případném návrhu, v první řadě jsem pomýšlel na odpovědnost za návrh, na bezpečnost, na ostudu, kdyby ta funkce měla slabinu. V druhé řadě na rychlost a použitelnost. Vznikl nový koncept hašovací funkce HDN [12], založené na dvojité síti DN [13], o němž jsem se bláhově domníval, že by mohl být základem nového hašovacího standardu. Podařilo se ukázat, že HDN má některé velmi dobré bezpečnostní vlastnosti, které lze dokázat, a tak mít větší jistotu, že útočník tyto bariéry nepřekoná. Rychlost HDN byla myslím jen 3x menší než rychlost SHA-2. Považoval jsem to za vynikající výsledek (promiňte mi tuhle drzost), uznejte sami, když se podíváte na obrázek, co to HDN je, že je to překvapení. Později jsem pochopil, že tenhle koncept (vyvinul jsem k němu teorii dvojité sítě DN a teorii SNMAC[14]), může být základem nových hašovacích funkcí, ale nikoliv těch komerčních, nikoli mezinárodních standardů, použitelných pro všechno. Málo platné, i když vyvinete krásný a odolný hard disk, kterým lze také štípat dříví (pokud ho dobře nabrousíte 😊), není to nic platné, protože se nevejde do mobilního telefonu.



Obr.: Tohle je základní stavební prvek HDN [12], která jich obsahuje cca 8, každý čtvereček reprezentuje substituční box typu 8x8 a MDS označuje lineární matici. Je to "bytelné" schéma a dokonce rychlejší než řada kandidátů SHA-3.

Stále tatáž chyba

Hodně týmů postupovalo tak jako já před několika lety, když jsem navrhoval hašovací funkci HDN a jen ubralo na bezpečnosti tak, aby rychlost nebyla tolik snížena. Tyto návrhy většinou

používají osvědčené solidní stavební prvky, semtam nějaké vylepšení, a proto se v nich budou špatně hledat nedostatky. Pravděpodobně ale nemají v soutěži šanci! Proč by NIST vybíral pomalejší návrhy, které jsou (potenciálně) bezpečnější, když má na výběr rychlejší algoritmy, které jsou potenciálně bezpečné? A bezpečné budou do té doby, než se u nich neobjeví chyba. Takže se domnívám, že ze soutěže vypadnou ty návrhy, které jsou pomalejší než SHA-2, byť jen o málo. Opět je to škoda, neboť mezi takovými algoritmy je řada návrhů zvučných týmů a jmen. Mají ale smůlu, protože nepochopili, že komunikační a informační průmysl nebude mít žádnou motivaci k tomu použít nový hašovací algoritmus SHA-3 místo rychlejších a stále platných SHA-2. A navíc, tyto algoritmy nebudou brzo rychlostně stačit. A tohle NIST ví. NIST má ovšem odpovědnost i za bezpečnost. Přesto si myslím, že NIST uvedenou cestou půjde, což dedukuji z jeho prohlášení, že v případě nalezení slabín vybraného standardu (kdykoliv v následujících letech) nebude vyhlašovat soutěž novou, ale použije kandidáta, který se umístí na druhé příčce.

Turbo-SHA

Na soutěž SHA-3 jsem zapomněl, protože žádný ohlas ze světa na nový koncept HDN nepřišel. O vánocích roku 2007 jsem si procházel známý archiv <http://eprint.iacr.org/> a udělal jsem si radost, když jsem se "pustil" do návrhu hašovací funkce Turbo-SHA Danila Gligoroského a Sveina Knapskoga [15]. Pak jsme s Danilem začali diskutovat o bezpečnosti a jak Turbo vylepšit. Dělal jsem si poznámky do jednoho dokumentu, algoritmus prodělal několik desítek základních verzí a řadu variant u některých. Nebýt Danila a jeho mistrovství v počítačové vědě, matematice a kryptografii, nebylo by žádné BMW. Opět se potvrdilo, že na pomezí oborů vznikají snadněji nové myšlenky. Takže po roce práce jsme mohli BMW poslat do soutěže NIST.



Obr.: Ze strany <http://bluwishsilver.net/index.html>

Název BMW

Nejtěžší úlohou manažerů bývá vymyslet název nového produktu. Musí to znít dobře, nesmí to narušit chráněné značky, dobře pamatovatelné, melodické, je to skutečně pěkná věda a manažeři si jí užívají. My jsme také potřebovali nějaký název. Trochu jsme se tím bavili jako manažeři a skončili jsme u Blue Wish. Používali jsme to nějakou dobu jako pracovní název, ale pak Danilo zapátral na internetu a dobře udělal, byla to registrovaná značka! Nechali jsme toho a pracovali dál. Když jsme se blížili ke konečné variantě algoritmu, byla zrovna půlnoc

jako už potisícátépáté, tak jsem si řekl, že to je ve skutečnosti takový půlnoční algoritmus, čili Blue Midnight Wish. A tak nám to zůstalo 😊.

Nové technologie u návrhů kandidátů

Jak a co ty nejrychlejší algoritmy použily, aby docíli-li vysokou rychlost? Co je tím úžasným objevem, který umožňuje kvalitu i rychlost? Co je tím zázrakem, který se dá realizovat i v nejmenších mikroprocesorech i v 64bitových instrukčních souborech rychle a kryptograficky kvalitně?

Tím technologickým zázrakem jsou polynomiální rovnice. Je známo, že soustavy rovnic, které jsou tvořeny polynomy o mnoha neznámých (s booleovskými proměnnými, tj. s hodnotou 0 a 1), nedovedeme řešit v polynomiálním čase. Jinými slovy polynomy jsou dobrými kandidáty na stavební prvky kryptografických schémat. Ovšem nějaký náhodný polynom 32. stupně s proměnnými a_0, \dots, a_{31} a b_0, \dots, b_{31} , třeba $a_0 * a_1 * a_2 * a_3 * \dots * a_{31} \text{ xor } a_0 * b_0 * b_2 * b_3 * a_{12} \text{ xor } \dots \text{ xor } \dots$, kde $*$ je logické AND, je docela náročný na výkon i plochu procesoru a také na čas. V softwaru zabere hodně času - cca tolik, kolik je v polynomu operací $*$ a xor. V hardware zabere zase nějakou plochu a spotřebuje čas na součet neparalelizovatelných výpočtů. Existuje však jedna operace moderních procesorů, která je velmi rychlá (proběhne na jeden takt procesoru, tj. nejrychleji jak je možné), a přesto poskytuje 32 (64) polynomů vysokých řádů najednou. Jedná se o operaci ADD, tj. "obyčejné" plus. Technologickým nástrojem většiny rychlých algoritmů soutěže jsou pouze operace ADD a XOR moderních procesorů! Jaké polynomy poskytuje obyčejná operace $a + b$ dvou 32bitových slov a a b , je vidět v tabulce.

Označme bity slov a a b jako $a = (a_{31}, a_{30}, \dots, a_1, a_0)$ a $b = (b_{31}, b_{30}, \dots, b_1, b_0)$ a začněme je sčítat jako ve škole. postupně při přechodu doleva nám vznikají bity přenosu c_1, c_2, \dots, c_{31} . Součet $a + b$ označme $s = a + b$ a jeho bity ($s_{31}, s_{30}, \dots, s_1, s_0$). Pokud máme bity přenosu, je to jednoduché vyjádření

$s = (a_{31} \text{ xor } b_{31} \text{ xor } c_{31}, a_{30} \text{ xor } b_{30} \text{ xor } c_{30}, \dots, a_1 \text{ xor } b_1 \text{ xor } c_1, a_0 \text{ xor } b_0)$. Teď budeme muset ty bity přenosu dopočítat. Takže postupně:

$$c_1 = a_0 * b_0,$$

$$c_2 = a_1 * b_1 \text{ xor } a_1 * c_1 \text{ xor } b_1 * c_1$$

$$c_3 = a_2 * b_2 \text{ xor } a_2 * c_2 \text{ xor } b_2 * c_2$$

$$c_4 = a_3 * b_3 \text{ xor } a_3 * c_3 \text{ xor } b_3 * c_3$$

.....

$$c_{30} = a_{29} * b_{29} \text{ xor } a_{29} * c_{29} \text{ xor } b_{29} * c_{29}$$

$$c_{31} = a_{30} * b_{30} \text{ xor } a_{30} * c_{30} \text{ xor } b_{30} * c_{30}$$

Pokud dosadíme jednotlivé výrazy pro bity carry do vyšších bitů (třeba c_1 do výrazu pro c_2), dostáváme postupně polynomy vyšších řádů s mnoha sčítanci (termy) různých řádů.

Například ze začátku máme

$$c_1 = a_0 * b_0, \text{ 1 term řádu 2}$$

$$c_2 = a_1 * b_1 \text{ xor } a_1 * a_0 * b_0 \text{ xor } b_1 * a_0 * b_0, \text{ 1 term řádu 2 a 2 termy řádu 3}$$

$$c_3 = a_2 * b_2 \text{ xor } a_2 * c_2 \text{ xor } b_2 * c_2, \text{ 1 term řádu 2 a } 2 * (1 \text{ term řádu } 2+1 \text{ a } 2 \text{ termy řádu } 3+1) = 1 \text{ term řádu 2 a 2 termy řádu 3 a 4 termy řádu 4}$$

$$c_4 = a_3 * b_3 \text{ xor } a_3 * c_3 \text{ xor } b_3 * c_3$$

.....

$$c_{30} = a_{29} * b_{29} \text{ xor } a_{29} * c_{29} \text{ xor } b_{29} * c_{29}$$

$$c_{31} = a_{30} * b_{30} \text{ xor } a_{30} * c_{30} \text{ xor } b_{30} * c_{30}$$

počet termů znázorňuje tabulka.

term řádu	0	1	2	3	4	5	6	7	8	9	10	11	1231	Součet
bit carry															
c1			1												1
c2			1	2											3
c3			1	2	4	0									7
c4			1	2	4	8	0	0	0	0	0	0	0	0	15
c5			1	2	4	8	16	0	0	0	0	0	0	0	31
c6			1	2	4	8	16	32	0	0	0	0	0	0	63
c7			1	2	4	8	16	32	64	0	0	0	0	0	127
c8			1	2	4	8	16	32	64	128	0	0	0	0	255
c9			1	2	4	8	16	32	64	128	256	0	0	0	511
c10			1	2	4	8	16	32	64	128	256	512	0	0	1023
c11			1	2	4	8	16	32	64	128	256	512	1024	0	2047
c12			1	2	4	8	16	32	64	128	256	512	1024	0	4095
c13			1	2	4	8	16	32	64	128	256	512	1024	0	8191
c14			1	2	4	8	16	32	64	128	256	512	1024	0	16383
c15			1	2	4	8	16	32	64	128	256	512	1024	0	32767
c16			1	2	4	8	16	32	64	128	256	512	1024	0	65535
c17			1	2	4	8	16	32	64	128	256	512	1024	0	131071
c18			1	2	4	8	16	32	64	128	256	512	1024	0	262143
c19			1	2	4	8	16	32	64	128	256	512	1024	0	524287
c20			1	2	4	8	16	32	64	128	256	512	1024	0	1048575
c21			1	2	4	8	16	32	64	128	256	512	1024	0	2097151
c22			1	2	4	8	16	32	64	128	256	512	1024	0	4194303
c23			1	2	4	8	16	32	64	128	256	512	1024	0	8388607
c24			1	2	4	8	16	32	64	128	256	512	1024	0	16777215
c25			1	2	4	8	16	32	64	128	256	512	1024	0	33554431
c26			1	2	4	8	16	32	64	128	256	512	1024	0	67108863
c27			1	2	4	8	16	32	64	128	256	512	1024	0	134217727
c28			1	2	4	8	16	32	64	128	256	512	1024	0	268435455
c29			1	2	4	8	16	32	64	128	256	512	1024	0	536870911
c30			1	2	4	8	16	32	64	128	256	512	1024	536870912	1073741823
														součet	2147483616

Tabulka: termy v součtu dvou 32 bitových celých čísel

Jedinou operací typu $a + b$ tak v moderních procesorech vznikne 32 polynomů, které dohromady obsahují 64 lineárních členů (termů řádu 1) a přes dvě miliardy (2.147.483.616) termů řádu 2 až 31. Technologickým zázrakem je tedy operace sčítání...

BMW

Algoritmus BMW používá pouze operace XOR a ADD. V popisu nalezneme i operace bitových posunů (shift right, shift left, \gg , \ll) nebo cyklických bitových posunů (cyclic right shift, cyclic left shift, \ggg , \lll), což zdržuje pouze v SW. Tam to stojí navíc takt procesoru, zatímco v hardware je tato operace zadarmo, neboť se jedná jen o vedení spojů na určitá místa.

Podobně je na tomto principu založen i algoritmus EDON-R a další. To, že mnozí autoři nemohli opisovat od sebe návrhy a dospěli ke stejnému východisku, není náhoda. Okrajové

podmínky na řešení dané úlohy byly dané a stejné pro všechny. Použití operací ADD a XOR bylo logickým vyústěním nejrychlejších kandidátů.

Závěr

Vítězem soutěže a novým hašovacím standardem bude silný algoritmus. Slabý totiž nemůže přestát soustředěné několikaleté zkoumání jeho kvalit mezinárodní komunitou kryptologů, které ho čeká do konce roku 2012 (i v letech následujících).

Poděkování

V textu byly použity některé výňatky z článku ve Sdělovací technice 02/2009.

Literatura

Varování ministra fair-play: jediné nezávislé stránky v následujícím seznamu jsou stránky NIST. Ostatní internetové stránky produkují skupiny a lidé, kteří mají své zájmy v soutěži. Většina těchto stránek to skrývá, nelze je považovat za nezávislé, i když se o to všemožně snaží, a často to tak působí.

- [1] (nezávislá) oficiální domácí stránka NIST k projektu SHA-3: <http://csrc.nist.gov/groups/ST/hash/index.html>
- [2] stránka autora s novinkami k projektu SHA-3 a algoritmům BMW a EDON-R: http://cryptography.hyperlink.cz/BMW/BMW_CZ.html (zde naleznete všechny linky)
- [3] Stránka kandidátů, kteří postoupili do prvního kola (NIST): http://csrc.nist.gov/groups/ST/hash/sha-3/Round1/submissions_rnd1.html
- [4] Stránka SHA-3 na wiki: <http://en.wikipedia.org/wiki/SHA-3>
- [5] Stránka SHA-3 projektu ECRYPT: http://ehash.iaik.tugraz.at/index.php/The_SHA-3_Zoo
- [6] Seznam všech autorů všech kandidátů: http://ehash.iaik.tugraz.at/wiki/SHA-3_submitters
- [7] Stránka NIST, věnovaná první konferenci kandidátů SHA-3: <http://csrc.nist.gov/groups/ST/hash/sha-3/Round1/Feb2009/program.html>
- [8] Stránka o SW výkonnosti algoritmů eBash: <http://bench.cr.yt.to/results-hash.html>
- [9] Stránka o HW výkonnosti algoritmů (ECRYPT): http://ehash.iaik.tugraz.at/wiki/SHA-3_Hardware_Implementations
- [10] Srovnávací stránka (Fleischmann-Forler-Gorski): http://www.uni-weimar.de/cms/fileadmin/medien/medsicherheit/Research/SHA3/Classification_of_the_SHA-3_Candidates.pdf
- [11] Srovnávací stránka (Niels Ferguson): <http://www.skein-hash.info/sha3-engineering>
- [12-14] Domovská stránka DN, HDN a SNMAC, v češtině, http://cryptography.hyperlink.cz/SNMAC/SNMAC_CZ.html ,
- [12] V. Klima: O návrhu speciálních blokových šifer a speciálních hašovacích funkcí, *MKB 2007*, Praha, Hotel Olympik, December, 6. – 7., 2007, http://cryptography.hyperlink.cz/2007/Klima_MKB_2007_sbornik.pdf
- [13] V. Klima: Special block cipher family DN and new generation SNMAC-type hash function family HDN, IACR ePrint archive Report 2007/050, February, 2007, IACR ePrint archive Report 2007/050, February, 2007, <http://eprint.iacr.org/2007/050.pdf>, v češtině na http://cryptography.hyperlink.cz/SNMAC/DN_HDN_CZ.pdf
- [14] V. Klima: A New Concept of Hash Functions SNMAC Using a Special Block Cipher and NMAC/HMAC Constructions, IACR ePrint archive Report 2006/376, <http://eprint.iacr.org/2006/376.pdf>, October, 2006, v češtině na http://cryptography.hyperlink.cz/SNMAC/SNMAC_CZ.pdf
- [15] Vlastimil Klíma: O kolizích hašovacích funkcí Turbo SHA-2, IACR ePrint archive Report 2008/003, January, 2008, <http://eprint.iacr.org/2008/003.pdf> v češtině na http://cryptography.hyperlink.cz/2008/Klima_TurboSHA_CZ.pdf