Vlastimil Klíma: Současná kryptologie a hashovací funkce v praxi

SmartCard Forum 2009 - informační bezpečnost & čipové technologie, Praha, 21. května 2009, Konferenční sál společnosti OKsystem s.r.o.

Informační společnost

Lidé se dobře adaptovali na výhody, které přináší informační společnost a nové formy komunikace. Samozřejmost, se kterou chtějí mít všechny možné informace a služby všude a hned, jen dále stimuluje rozvoj informačních a komunikačních technologií. Data získala jiný význam než dříve. Oproti minulosti představují často nebývale velmi vysokou hodnotu, jak pro jednotlivce tak pro společnosti. Příkladem budiž finanční transakce, nebo chcete-li data přímo ekvivalentní penězům.

Objev hašovacích funkcí a digitálního otisku

Objev hašovacích funkcí a digitálního otisku umožnil jednoznačně identifikovat jakákoliv data. Jakýkoliv digitální soubor dat může být jednoznačně identifikován poměrně krátkým binárním řetězcem, a to dokonce s mnohem větší jistotou, než otisk prstu identifikuje člověka. Hašovací funkce jsou důležitým mnohostranným vynálezem. Umožňují nejen jednoznačně data identifikovat digitálním otiskem, ale také zajistit jejich neporušenost, autentizaci a nepopiratelnost různých akcí s daty, například autorství, přijetí, odeslání, průkaz jejich znalosti apod. Hašovací funkce jsou natolik složité, že jejich výpočty musí provádět procesory počítačů nebo čipy platebních karet, SIM karet apod. Na druhé straně musí být rychlé, aby neomezovaly vlastní užitné funkce a datové toky mezi zařízeními. To klade vysoké nároky jak na kryptologii, která je vytváří, tak na čipové technologie, které je realizují.

Hašovací funkce

Hašovací funkce jsou funkce, které z libovolného souboru na svém vstupu vytvoří hašovací kód dané délky (například u SHA-256 je to 256 bitů). Tyto funkce mají vlastnost jednosměrnosti a bezkoliznosti. Pojem "bezkoliznosti" je odborný termín, který je myšlenkovou zkratkou, často však nebývá pochopen. Neznamená to, že by kolize neexistovaly - protože hašových kódů je pouze 2^{256} , zatímco datových vstupů je mnohem více, znamená to ale, že složitost nalezení takových kolizí je příliš velká, a to tak velká, že je výpočetně nemožné je nalézt (na daném stupni poznání a stavu matematicko-kryptologických metod). Často také vzniká otázka, jak je to možné a jak se to zajišťuje, aby se ty kolize nenašly? No, a to je právě úloha vědy, kryptologie, aby přinesla dostatečné argumenty pro taková tvrzení. Moderní hašovací funkce mají však více vlastností, kvůli nimž se používají i k jiným účelům než je digitální otisk, tj. jednoznačná identifikace dat. Jsou to vlastnosti pseudonáhodnosti a jednosměrnosti. Hašovací funkce se používají, k autentizaci (prokazování znalosti tajemství), k odvozování klíčů a generování klíčového materiálu libovolné délky a jsou používány téměř v každé kryptografické aplikaci.

Kryptografie

Kryptologie je jednou z mnoha metod informační bezpečnosti[1]. Zabývá se ochranou dat, tj. ochranou soukromí, integrity, neodmítnutelnosti zodpovědnosti, autentičnosti apod. Není třeba ji přeceňovat, avšak její podcenění může být devastující, ostatně, stejně jako podcenění jiných metod ochrany dat. Ztracené soukromí nelze vrátit zpět, podvodná autentizace může mít nedozírné následky. Kryptografie držela dlouhou dobu náskok před informačními technologiemi. Bylo připraveno mnoho kryptografických technik a standardů pro utajení dat, integritu, autentizaci, nepopiratelnost. Avšak současné matematické metody informační bezpečnosti nestačí velmi rychle se zvyšující rychlosti a objemu přenášených dat a tento trend bude nadále progresivně pokračovat. Proto vznikají kompromisy mezi bezpečností a rychlostí, které mohou vést a občas vedou k oslabení nebo prolomení nějaké kryptografické metody. To je i příklad funkcí pro digitální otisky. V roce 2004 byla prolomena velmi populární a rychlá hašovací funkce MD5[2] a v dalších letech byl oslaben její následník SHA-1[3].

[1] V. Klíma: Současná kryptologie v praxi, Information Security Summit 2008, Martinický Palác, 28. – 29. května 2008, Praha,
http://cryptography.hyperlink.cz/2008/Klima_Vlastimil_IS2_2008_presentation_2.pdf

[2] V roce 2009 byl díky slabinám MD5 vytvořen platný certifikát, který ovšem certifikační autorita nikdy neviděla.

Současný platný standard, na něž se musí přecházet od počátku příštího roku (neboť SHA-1 do nových produktů nesmí), je SHA-2. Je mnohem bezpečnější, ale také skoro 3x pomalejší než SHA-1. A to je pro průmysl příliš velká cena za bezpečnost. Navíc v budoucnu se množství informací a rychlost přenosu dále zvýší. Proto je potřeba naopak funkce rychlejší.

SHA-3

Na kvalitě nové hašovací funkce bude ještě více záviset bezpečnost elektronického bankovnictví a ještě více bezpečnost komunikací na celém světě, protože aplikací bude přibývat a budou se ještě více ve světě rozšiřovat. Průmysl tedy klade na kryptology dva protichůdné požadavky - aby zvýšili bezpečnost i rychlost nové hašovací funkce. Ve skutečnosti tím kryptology nutí, aby vyvinuli novou "kryptologickou technologii" hašovací funkce, která bude mít obě vlastnosti, které jsou v přímém rozporu. Z důvodu bezpečnosti je proces tvorby nového návrhu pod širokou mezinárodní kontrolou. Jeho stav je takový, že soutěž na nový hašovací standard SHA-3 byla vypsána v roce 2007 a je organizována americkým úřadem pro standardizaci NIST. V prosinci 2008 bylo z 64 přihlášených kandidátů do prvního kola vybráno 51. V srpnu tohoto roku bude tento počet zúžen na 15 a za další rok na 5. Tyto funkce budou masově používány také v čipových a kartových technologiích, proto se soutěže účastní nejen světoznámí kryptologové, Univerzity a elektronické giganty, ale i největší světoví výrobci z oblasti čipů. Soutěž zaměstnává stovky kryptologů na celém světě, znamená tisíce hodin odborné práce a není pochyb, že bude vybrána vysoce kvalitní funkce s nejlepším světovým know-how. Soutěže se zúčastní 191 kryptologů, pracujících v různých společnostech a zemích, jak ukazuje následující přehled.

Účastníci soutěže na SHA-3

Společnosti z oblasti čipových technologií

- Hitachi
- Hifn, Inc.
- Gemalto
- STMicroelectronics
- Intel Corporation
- NXP Semiconductors
- SRI International
- Orange Labs/Ingenico

Laboratoře

- Massachusetts Institute of Technology, USA
- Sandia National Laboratories, USA
- DCSSI Crypto Lab, France
- IBM T. J. Watson Research Center, USA

Elektronika, komunikace

- Sony Corp.
- Sirrix AG
- Opta Consulting
- QUALCOMM Inc.
- Sagem Sécurité
- BT Group plc
- France Telecom

SW společnosti

- RSA Laboratories
- Certicom
- PGP Corporation
- Microsoft Corporation

[3] V květnu t.r. byla u hašovací funkce SHA-1 snížena složitost nalezení kolize na hodnotu 2^{52} operací. Letos bude v EU i v ČR ukončeno vydávání kvalifikovaných certifikátů s algoritmem SHA-1. Od 1. 1. 2010 budou CA vydávat kvalifikované certifikáty podporující některý z algoritmů SHA-2 (zároveň s tím je stanovena minimální délka kryptografického klíče pro algoritmus RSA na 2048 bitů).

Různé firmy

- Orange Labs
- Mobileye Inc., Israel
- Sirrix AG
- NETS Corporation
- VEST Corporation
- Purple Streak, USA
- Institute for Infocomm Research, Singapore
- Washburn Research
- Tata Consultancy Services
- EADS Secure Networks
- WaveStrong, Inc.
- Cryptolog International
- IDC Herzliya, Israel

Univerzity

- Universidad Nacional de Córdoba, Argentina
- Technische Universität Graz, Austria
- Katholieke Universiteit Leuven, Belgium
- Charles University, Czech Republic
- Technical University of Denmark
- École Normale Supérieure (Ulm), France
- École Nationale Supérieure de Techniques Avancées, France
- Research Centre INRIA Paris-Rocquencourt, France
- Université de Limoges, France
- Université de Versailles, France
- Université de Cergy-Pontoise, France
- Université Bordeaux, France
- Justus Liebig-Universität, Germany
- Bauhaus-University Weimar, Germany
- Tel-Aviv University, Israel
- Technion, Israel
- University of Fukui, Japan
- Nagoya University, Japan
- Kobe University, Japan
- CIST, Korea University, Korea
- University of Luxembourg, Luxembourg
- Macedonian Academy of Sciences and Arts
- Ss. Cyril and Methodius University, Macedonia
- University Goce Delčev, Macedonia
- Norwegian University of Science and Technology, Norway
- University of Alicante, Spain
- Universitat Autònoma de Barcelona, Spain
- Eidgenössische Technische Hochschule Zürich, Switzerland
- Ecole Polytechnique Fédérale de Lausanne, Switzerland
- Fachhochschule Nordwestschweiz, Switzerland
- The National Research Institute of Electronics and Cryptology (UEKAE), Turkey
- Middle East Technical University, Turkey
- Loughborough University, UK
- New York University, USA
- James Madison University, USA
- Boston University, USA
- Oregon State University, USA
- University of Illinois at Chicago, USA
- University of Virginia, USA
- University of Washington, USA
- University of California San Diego, USA
- University of California Santa Barbara, USA

Jednotlivci

- Independent cryptologist - consultant (CZ)
- Private individual (UK)
- Independent cryptography and security consultant (USA)

Požadavky na kandidáty

Nejdůležitějším požadavkem je rychlost, dále spotřeba paměti a poté spousta dalších požadavků. Rychlost se u všech kandidátů měří podle NIST jednotně a na definovaných platformách v počtu cyklů na bajt, tedy jako průměrný počet cyklů procesoru, které je trvá zpracování jednoho bajtu dat. Měření rychlosti je samostatná kapitola, které je věnována internetová stránka eBash (http://bench.cr.yp.to/ebash.html) s desítkami měření na různých platformách a z různých pohledů. Jednoduše řečeno, SHA-1 spotřebovala zhruba 7.5 cyklu na bajt (c/b), SHA-2 20 c/b a SHA-3 musí být výrazně rychlejší než SHA-2. Nároky na paměť by se neměly pohybovat více než v řádu stovek bajtů.

Přehled kandidátů na SHA-3

V následující tabulce uvádíme přehled kandidátů a zároveň velmi subjektivní hodnocení (závislého) pozorovatele (spoluautora BMW a EDON-R). Kandidáti jsou seřazeni podle našeho mínění od nejperspektivnějších, po nejméně perspektivní. Kupodivu výběr prvních 15 kandidátů je (bude) velmi jednoduchý. Spíše než že by přebývaly, kandidátů se nedostává, proto některé, které by při dostatečném počtu byly asi vyřazeny (i když kvůli malichernosti), mohou být vzaty na milost a zařazeny do výběru. Téměř jistě postoupí BMW, Shabal, BLAKE, SIMD a Skein. JH a Luffa mohou postoupit, ale v samotném finále 5 algoritmů mohou být jen tehdy, pokud některý kandidát z první pětice bude prolomen. Do boje o finálních 5 může ještě zasáhnout EDON-R, Cheetah, MD6 a Cubehash, pokud budou připuštěny do druhého kola a bude jim umožněna drobná změna (podmínky změn jsou známy). Ostatní algoritmy nemají šanci z důvodu rychlosti, a pokud budou do výběru zařazeny, bude to pouze "k doplnění počtu". Mohou sem být maximálně ještě přiřazeny Hamsi, Grøstl, Arirang a TIB3. V tabulce uvádíme rychlost kandidátů v cyklech na bajt pro 64/32bitové procesory a pro 256/512bitové varianty hašovacích funkcí (velikosti hašovacího kódu). Pro orientaci je nejdůležitější čtvrté z těchto čísel, protože je pro hašovací funkci nejméně výhodné - 512bitový otisk je náročnější než 256bitový, a to při výpočtu na 32bitovém procesoru, než na 64bitovém. V tabulce je pro orientaci uvedena také stávající SHA-2. Poznamenejme, že nejmenší čísla lze vidět u algoritmu EDON-R, u něhož bylo poukázáno na velmi nepatrné nedostatky. Pokud bude připuštěn do druhého kola s malou opravou zmíněného nedostatku, bude to rázem horký kandidát na vítěze.

Predikce vítězného algoritmu

Autor se domnívá, že vítěz vzejde z těchto šesti algoritmů: EDON-R, BMW, Shabal, BLAKE, SIMD a Skein. Kromě toho, na rozdíl od minulých let, NIST určí ještě "zástupce vítěze", tj. algoritmus, který nastoupí jako standard v případě, že by vítěz utrpěl v průběhu své platnosti nějaké vážnější šrámy. Osobně se domníváme, že vyhraje především rychlost, protože nový standard bude platit v době, kdy objem dat a rychlost jejich přenosu vzrostou nejméně o jeden řád, možná o dva. A takovým rychlostem mohou stačit nanejvýše dva z uvedených kandidátů. Náš odhad je, že to budou dva nejrychlejší, na nichž se nenajde žádná viditelná slabina. Které to budou, nevíme, protože slabina se může objevit u všech. A o tom ta soutěž je.

Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
Algoritmy, které pravděpodobně postoupí do 2.kola, rychlé a bez nedostatků:				
Blue Midnight Wish (BMW)	7/3	7/**12**	**Mezinárodní tým 6 lidí, Danilo Gligoroski** (Norwegian Univ. of Science and Technology), **Vlastimil Klima** (Independent), Svein Johan Knapskog, Mohamed El-Hadedy, Jørn Amundsen, Stig Frode Mjølsnes (all Norwegian Univ. of Science and Technology)	Paměť 264/528 Byte
Shabal	8	**10**	**Francouzský tým 14 lidí** (DCSSI, EADS, France Telecom, Gemalto, INRIA, Cryptolog, Sagem Security)	
BLAKE	8/9	9/**12**	**Mezinárodní tým 4 lidí**, Jean-Philippe Aumasson, Luca Henzen, Willi Meier, Raphael C.-W. Phan, Fachhochschule Nordwestschweiz, Switzerland, Loughborough Univ., UK	
SIMD	11/12	12/**13**	**Francouzský tým 3 lidí**, Gaëtan Leurent, Charles Bouillaguet, Pierre-Alain Fouque, École Normale Supérieure (Ulm), France	
Skein	7/6	21/**20**	**Mezinárodní tým 8 lidí**, **Bruce Schneier** (BT Group), Niels Ferguson (Microsoft), Stefan Lucks (Bauhaus Univ. Weimar), Doug Whiting (Hifn), Mihir Bellare (Univ. of California), Tadayoshi Kohno (Univ. of Washington), Jon Callas (PGP Corp.), Jesse Walker (Intel)	Paměť 100/200B
SHA-2	20/13	20/**40**	NIST, stávající standard	pro srovnání

Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
JH	16	**21**	**Hongjun Wu**, Institute for Infocomm Research, Singapore	pseudokolize triviálně, teor. nevýhoda
Luffa	13/23	13/**25**	**Mezinárodní tým 3 lidí,** Christophe De Canniere (Katholieke Univ. Leuven), Hisayoshi Sato, Dai Watanabe (Hitachi)	
Algoritmy, které mohou být ještě vzaty na milost:				
Edon-R	4/2	6/**10**	**Mezinárodní tým 7 lidí, Danilo Gligoroski**, Svein Johan Knapskog, Rune Steinsmo Ødegård (Norwegian Univ. of Science and Technology), Marija Mihova (Ss. Cyril and Methodius University, Macedonia), Ljupco Kocarev (Univ. of California San Diego, USA and Macedonian Academy of Sciences and Arts), Aleš Drápal (Charles Univ. Czech Rep.), Vlastimil Klima (Independent, Czech Rep.)	Formálně nepatrně prolomený, lze jednoduše opravit. Paměť 256/512B
Cheetah	9/13	15/**30**	**Mezinárodní tým 3 lidí,** Dmitry Khovratovich, Alex Biryukov, Ivica Nikolić (Univ. of Luxembourg)	Formálně prolomený (útok prodloužením zprávy), lze snadno opravit
MD6	28/44	68/**106**	**Mezinárodní tým 15 lidí, Ronald L. Rivest** (MIT), Benjamin Agre (MIT), Daniel V. Bailey (RSA), Christopher Crutchield (MIT), Yevgeniy Dodis (New York Univ.), Kermin Elliott Fleming (MIT), Asif Khan (MIT), Jayant	Paměť > 700B není vhodná pro smart karty

Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
			Krishnamurthy (MIT), Yuncheng Lin (MIT), Leo Reyzin (Boston University, USA), Emily Shen (MIT), Jim Sukha (MIT), Drew Sutherland (MIT), Eran Tromer (MIT), Yiqun Lisa Yin (Independent)	
CubeHash	160	**200**	**Dan Bernstein** (Univ. of Illinois)	možnost široké parametrizace
Algoritmy, které jsou zatím bez nedostatků, ale nenabízí nic zvláštního:				
Hamsi	25	36	Özgül Kücük (Katholieke Univ. Leuven)	
Grøstl	22/30	23/36		
Arirang	15/11	20/55		
Tento algoritmus asi nepostoupí z důvodu teoretických vlastností:				
TIB3	7/6	13/4	Daniel Penazzi, Miguel Montes (Argentina)	Vážnější nedostatky (pseudokolize, nenáhodnost)
Tyto algoritmy asi nepostoupí z důvodu rychlosti:				
SHAvite-3	26/38	35/55		
Keccak	10/20	31/62		
Echo	28/53	32/61	Mnohem rychlejší s Intel AES instrukcemi	Asi ho nezachrání ani Intel instrukce (nejsou všude)
CHI	24/16	49/78		

Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
Fugue	28/56	36/72		
LANE	25/145	40/152		
SANDstorm	37/95	62/297		
Lesamnta	52/51	59/54		
SWIFFTX	57	57		
ESSENSE	64/63	150/176		
FSB		324/507		
Prolomené (prakticky nebo teoreticky, mají větší slabiny nebo všechno dohromady):				
Abacus	37	37		Prolomené
Aurora	15/26	19/35		Prolomené
Blender				Prolomené
Crunch	161/446	298/862		Útok prodloužením zprávy
DCH				Prolomené
Dynamic SHA	27/47	27/47		Útok prodloužením zprávy, kolize
Dynamic SHA2	21/67	21/67		Útok prodloužením zprávy, kolize
ECOH	>1000	7500/10000		Prolomené

Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
EnRUPT				Prolomené
Khichidi-1				Prolomené
MCSSHA-3		60		Prolomené
LUX	10/9	16/28		Prolomené
MeshHash	13/18	42/67		Prolomené
NaSHA	26/26	27/30		Prolomené
Sarmal	9/10	19/23		Prolomené
Sgàil	61			Prolomené
Spectral Hash				Prolomené
StreamHash				Prolomené
Tangle				Prolomené
Twister	15/17	35/39		Prolomené
Vortex	46/56	69/90		Korelace výstupních bitů. Rychlost < 3 c/b použitím budoucích instrukcí Intel CPU
Nejsou již účastníky soutěže:				
Boole	7/7	21/21		Vyňato
HASH 2X				Nepostoupil do

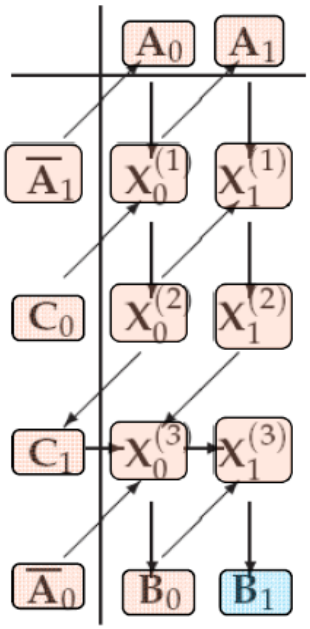
Algoritmus	64bitový procesor, hash 256/ /512 bitů	32bitový procesor, hash 256/ /512 bitů	Autorský tým, poznámka	Poznámka
				1.kola.
Maraca	5			Nepostoupil do 1.kola. Vyžaduje >6kB paměti
NKS2D				Nepostoupil do 1.kola.
Ponic	3000	7000		Nepostoupil do 1.kola.
SHAMATA	8/11	15/22		Vyňato
WaMM	360	360		Vyňato
Waterfall	16	16		Vyňato

Tabulka: Rychlost kandidátů v cyklech na bajt pro 64/32bitové procesory a pro 256/512bitové varianty hašovacích funkcí (poskytujících 256/512bitové velikosti hašovacího kódu)[4]

EDON-R, nejrychlejší kandidát na SHA-3

Pro konkrétní představu uvedeme jádro tohoto algorimu, konkrétně EDON-R256 s 256bitovým výstupním kódem. Pracuje iterativně tak, že na počátku naplní (průběžnou) hašovací hodnotu (P_0) inicializačním vektorem. Poté v každém kroku pomocí tzv. kompresní funkce (f) zpracuje průběžný blok zprávy (M_i) a průběžnou hašovací hodnotu (P_{i-1}) na novou průběžnou hodnotu $P_i = f(P_{i-1}, M_i)$. A tak se postupuje až do konce zprávy, přičemž poslední průběžná hodnota je konečná hodnota haše. Pro představu uvádíme základní stavební blok kompresní funkce. Během výpočtu $P_i = f(P_{i-1}), M_i)$ osmkrát volá následující tzv. kvazigrupovou operaci Q256. Do argumentů Q256 postupně vstupují jak blok zprávy, tak průběžná hash, tak mezivýsledky předchozích operací Q256, blíže viz úplná definice EDON-R na http://cryptography.hyperlink.cz.

[4] Tabulka vznikla přepracováním http://www.skein-hash.info/sha3-engineering



Obr.: Kompresní funkce EDON-R

Z = Q256(X, Y),
Funkce Q256 používá pouze operace sčítání slov (+), xor slov (v programu jako ^) a bitovou rotaci slov, tedy operace velmi rychlé na 32bitovém procesoru. Vstupy X a Y a Z jsou osmice 32bitových slov.

```
Q256(x0,x1,x2,x3,x4,x5,x6,x7,y0,y1,y2,y3,y4,y5,y6,y7,z0,z1,z2,z3,z4,z5,z6,z7)
{
  // funkce prvního argumentu (X)
   t0  = x0 + x1 + x2 + x4 + x7 + 0xaaaaaaaa;
   t1  = x0 + x1 + x3 + x4 + x7;
   t2  = x0 + x1 + x4 + x6 + x7;
   t3  = x2 + x3 + x5 + x6 + x7;
   t4  = x1 + x2 + x3 + x5 + x6;
   t5  = x0 + x2 + x3 + x4 + x5;
   t6  = x0 + x1 + x5 + x6 + x7;
   t7  = x2 + x3 + x4 + x5 + x6;

   t1  = rotl32((t1), 4);
   t2  = rotl32((t2), 8);
   t3  = rotl32((t3),13);
   t4  = rotl32((t4),17);
   t5  = rotl32((t5),22);
   t6  = rotl32((t6),24);
   t7  = rotl32((t7),29);

   t8   = t3 ^ t5 ^ t6;
   t9   = t2 ^ t5 ^ t6;
   t10 = t2 ^ t3 ^ t5;
   t11 = t0 ^ t1 ^ t4;
   t12 = t0 ^ t4 ^ t7;
   t13 = t1 ^ t6 ^ t7;
   t14 = t2 ^ t3 ^ t4;
   t15 = t0 ^ t1 ^ t7;
```

```
  // funkce druhého argumentu (Y)

   t0  = y0 + y1 + y2 + y5 + y7+ 0x55555555;
   t1  = y0 + y1 + y3 + y4 + y6;
   t2  = y0 + y1 + y2 + y3 + y5;
   t3  = y2 + y3 + y4 + y6 + y7;
   t4  = y0 + y1 + y3 + y4 + y5;
   t5  = y2 + y4 + y5 + y6 + y7;
   t6  = y1 + y2 + y5 + y6 + y7;
   t7  = y0 + y3 + y4 + y6 + y7;

   t1  = rotl32((t1), 5);
   t2  = rotl32((t2), 9);
   t3  = rotl32((t3),11);
   t4  = rotl32((t4),15);
   t5  = rotl32((t5),20);
   t6  = rotl32((t6),25);
   t7  = rotl32((t7),27);

  // součet obou dílčích funkcí
   z5   = t8  + (t3 ^ t4 ^ t6);
   z6   = t9  + (t2 ^ t5 ^ t7);
   z7   = t10 + (t4 ^ t6 ^ t7);
   z0   = t11 + (t0 ^ t1 ^ t5);
   z1   = t12 + (t2 ^ t6 ^ t7);
   z2   = t13 + (t0 ^ t1 ^ t3);
   z3   = t14 + (t0 ^ t3 ^ t4);
   z4   = t15 + (t1 ^ t2 ^ t5);
}
```

BMW, druhý nejrychlejší kandidát

BMW používá tytéž operace (+, XOR, avšak jiným způsobem). Základním stavebním prvkem je následující blok operací, které transformují 16 slov Q[0],..., Q[15] na 16 slov Q[16],..., Q[31]. Funkce rj jsou bitové rotace a si jsou funkce, které k danému argumentu přixorují jeho bitový posun (nebo více různých bitových posunů) o několik bitů. Tato transformace se na rozdíl od EDON-R prování jen jednou (nikoli osmkrát), avšak je mnohem složitější, neboť používá 16 aritmetických sčítanců namísto pěti u EDON-R. Tím dosahuje vyšší algebraické složitosti (a je tak o pár procent pomalejší)

```
Q[16] = s1(Q[ 0])+s2(Q[ 1])+s3(Q[ 2])+s0(Q[ 3])+s1(Q[ 4])+s2(Q[ 5])+s3(Q[ 6])+s0(Q[ 7]) + s1(Q[ 8])+s2(Q[
9])+s3(Q[10])+s0(Q[11])+s1(Q[12])+s2(Q[13])+s3(Q[14])+s0(Q[15])+K[0]

Q[17]=s1(Q[ 1])+s2(Q[ 2])+s3(Q[ 3])+s0(Q[ 4])+s1(Q[ 5])+s2(Q[ 6])+s3(Q[ 7]) + s0(Q[ 8]) + s1(Q[
9])+s2(Q[10])+s3(Q[11])+s0(Q[12])+s1(Q[13])+s2(Q[14])+s3(Q[15])+ K[1] + s0(Q[16])

Q[18]=Q[ 2]+r1(Q[ 3])+Q[ 4]+r2(Q[ 5])+Q[ 6]+r3(Q[ 7])+Q[ 8]+r4(Q[ 9])+Q[10]+r5(Q[11])+
Q[12]+r6(Q[13])+Q[14]+r7(Q[15])+ K[2] +s5(Q[16])+s4(Q[17])

Q[19]=Q[ 3]+r1(Q[ 4])+Q[ 5]+r2(Q[ 6])+Q[ 7]+r3(Q[ 8])+Q[ 9]+r4(Q[10])+Q[11]+r5(Q[12])+
Q[13]+r6(Q[14])+Q[15] + K[3] +r7(Q[16])+s5(Q[17])+s4(Q[18])

Q[20]=Q[4]+r1(Q[5])+Q[6]+r2(Q[7])+Q[8]+r3(Q[9])+Q[10]+r4(Q[11])+Q[12]+r5(Q[13])+Q[14]
+r6(Q[15])+K[4]+Q[16]+r7(Q[17])+s5(Q[18])+s4(Q[19])

Q[21]=Q[ 5]+r1(Q[ 6])+Q[ 7]+r2(Q[ 8])+Q[ 9]+r3(Q[10])+Q[11]+r4(Q[12)+Q[13]+r5(Q[14]) +Q[15] + K[5] +
r6(Q[16])+Q[17]+r7(Q[18])+ s5(Q[19])+s4(Q[20])

Q[22]=Q[ 6]+r1(Q[ 7])+Q[ 8]+r2(Q[ 9])+Q[10]+r3(Q[11])+Q[12]+r4(Q[13])+Q[14]+r5(Q[15]) + K[6]
+Q[16]+r6(Q[17])+Q[18]+r7(Q[19])+s5(Q[20])+s4(Q[21])
```

Q[23]=Q[7]+r1(Q[8])+Q[9]+r2(Q[10])+Q[11]+r3(Q[12])+Q[13]+r4(Q[14])+Q[15] + K[7]
+r5(Q[16])+Q[17]+r6(Q[18])+Q[19]+r7(Q[20])+s5(Q[21])+s4(Q[22])

Q[24]=Q[8]+r1(Q[9])+Q[10]+r2(Q[11])+Q[12]+r3(Q[13])+Q[14]+r4(Q[15]) + K[8]+
Q[16]+r5(Q[17])+Q[18]+r6(Q[19])+Q[20]+r7(Q[21])+s5(Q[22])+s4(Q[23])

Q[25]=Q[9]+r1(Q[10])+Q[11]+r2(Q[12])+Q[13]+r3(Q[14])+Q[15] + K[9] +r4(Q[16])+
Q[17]+r5(Q[18])+Q[19]+r6(Q[20])+Q[21]+r7(Q[22])+s5(Q[23])+s4(Q[24])

Q[26]=Q[10]+r1(Q[11])+Q[12]+r2(Q[13])+Q[14]+r3(Q[15]) + K[10]+Q[16]+r4(Q[17])+
Q[18]+r5(Q[19])+Q[20]+r6(Q[21])+Q[22]+r7(Q[23])+s5(Q[24])+s4(Q[25])

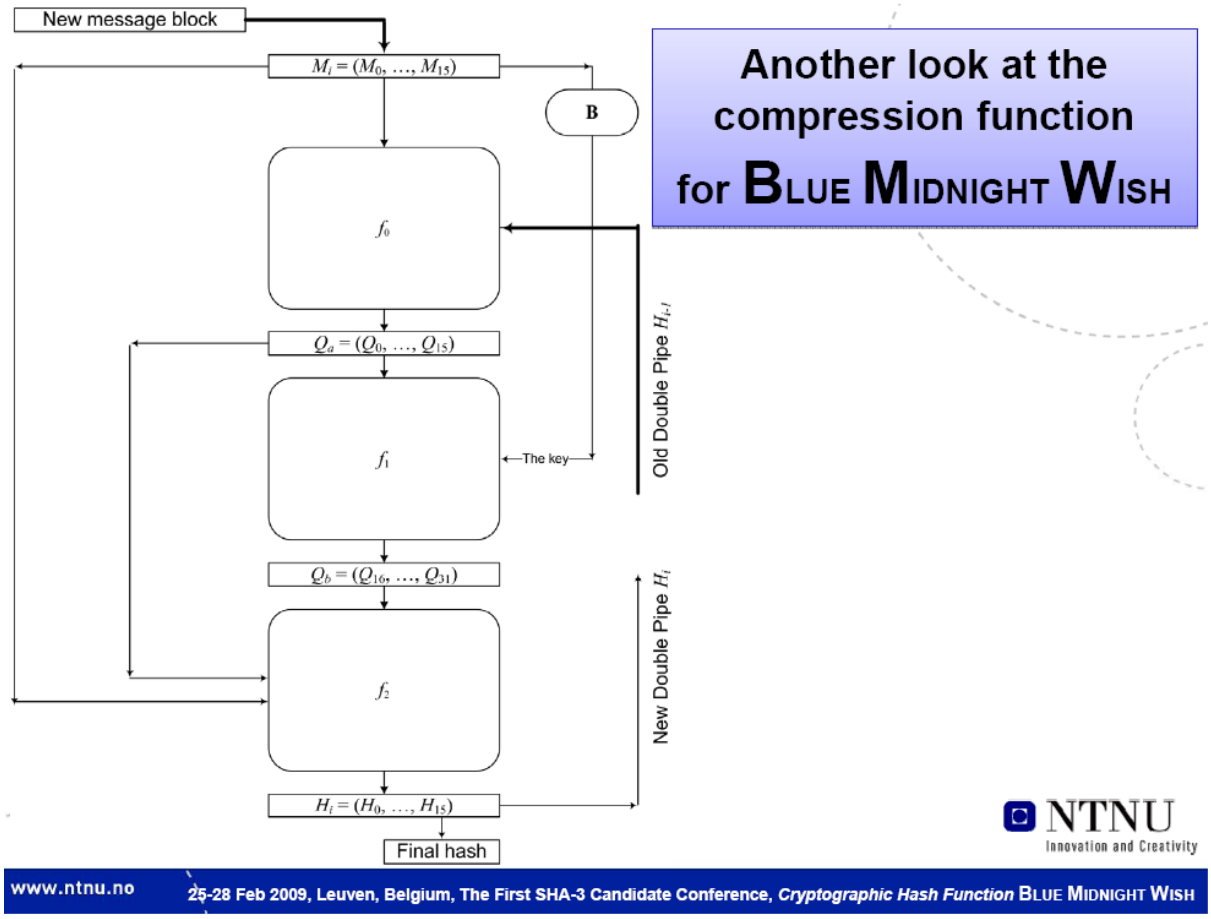
Q[27]=Q[11]+r1(Q[12])+Q[13]+r2(Q[14])+Q[15] + K[11]+r3(Q[16])+Q[17]+r4(Q[18])+
Q[19]+r5(Q[20])+Q[21]+r6(Q[22])+Q[23]+r7(Q[24])+s5(Q[25])+s4(Q[26])

Q[28]=Q[12]+r1(Q[13])+Q[14]+r2(Q[15]) + K[12]+Q[16]+r3(Q[17])+Q[18]+r4(Q[19])+
Q[20]+r5(Q[21])+Q[22]+r6(Q[23])+Q[24]+r7(Q[25])+s5(Q[26])+s4(Q[27])

Q[29]=Q[13]+r1(Q[14])+Q[15] + K[13]+r2(Q[16])+Q[17]+r3(Q[18])+Q[19]+r4(Q[20])+
Q[21]+r5(Q[22])+Q[23]+r6(Q[24])+Q[25]+r7(Q[26])+s5(Q[27])+s4(Q[28])

Q[30]=Q[14]+r1(Q[15]) + K[14]+Q[16]+r2(Q[17])+Q[18]+r3(Q[19])+Q[20]+r4(Q[21])+
Q[22]+r5(Q[23])+Q[24]+r6(Q[25])+Q[26]+r7(Q[27])+s5(Q[28])+s4(Q[29])

Q[31]=Q[15] + K[15]+r1(Q[16])+Q[17]+r2(Q[18])+Q[19]+r3(Q[20])+Q[21]+r4(Q[22])+
Q[23]+r5(Q[24])+Q[25]+r6(Q[26])+Q[27]+r7(Q[28])+s5(Q[29])+s4(Q[30])



Obr.: Celkové schéma BMW, transformace 16 slov Q[0],..., Q[15] na 16 slov Q[16],..., Q[31] je funkce f2 uprostřed

Podstata nové technologie
Algoritmy, které nepřišly se skutečně novou kryptologickou myšlenkou, nemohly "starou technologií" vyřešit ony protichůdné požadavky NISTu - vyšší rychlost a vyšší bezpečnost. Takových návrhů je většina, a tudíž nemají šanci na vítězství. Tou novou kryptologickou technologií je vhodně použitá operace ADD. Je jednak velmi složitá (fakticky obsahuje polynomy se 4 miliardami termů, které nelze přímým způsobem za normálních okolností vůbec vyrobit), a přitom zabírá minimální místo a spotřebovává minimální čas. V dodatku ukazujeme, proč je tomu tak. Poznamenejme, že pouhé aplikování operace ADD kde se zlíbí, by nebylo dobré. Je však vynikajícím stavebním prvkem, který je a musí být využit v harmonii s mnoha dalšími vztahy a zákonitostmi, které je nutné respektovat.

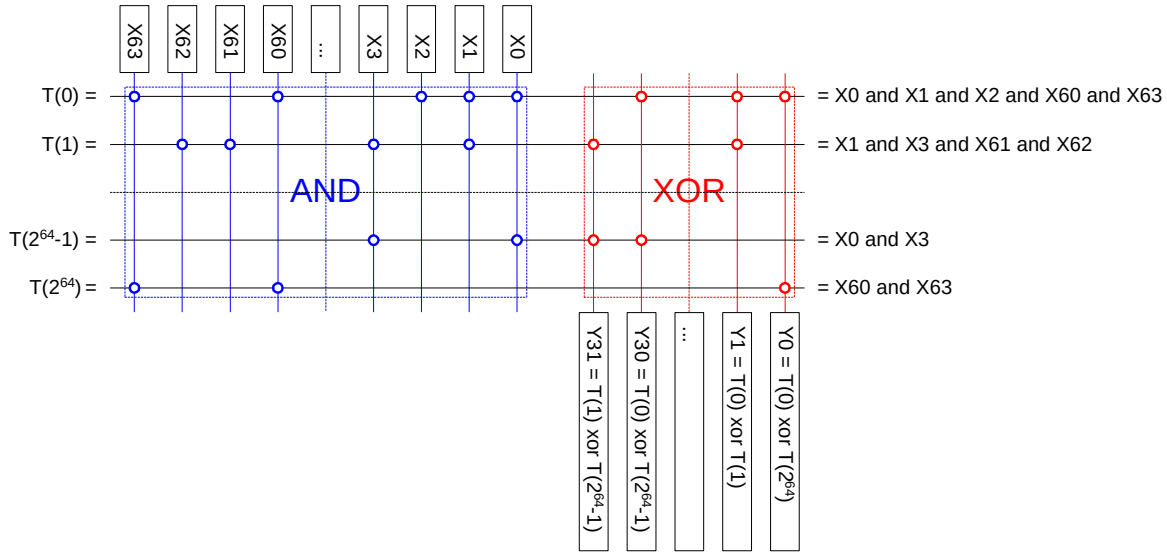
Závěr - Kvalita budoucího standardu
Vybraný algoritmus bude analyzován velmi dlouho velkou množinou nejzkušenějších kryptologů světa. Z dosavadního průběhu soutěže je vidět, že se mu dostane největší pozornosti, jaké se kdy nějakému kryptografickému algoritmu podařilo. Proto bude zcela jistě velmi kvalitním standardem pro další léta. A to je dobrá zpráva pro náruživé uživatele informační společnosti.

-oOo-

Dodatek - Zajímavá matematika v pozadí operace ADD
Tento odstavec zařazujeme pouze jako zajímavost pro čtenáře, kteří si něco pamatují z algebry. Ostatní ho mohou klidně přeskočit.
Uvedené algoritmy mají celou řadu návrhových principů a odlišných metod konstrukce. Posouzení a zhodnocení těchto metod bylo provedeno NISTem na první konferenci SHA-3, který chce mít v užším výběru zastoupení všech možných metod. Proto je také těžké odhadnout, které algoritmy NIST vybere do semifinálové patnáctky a poté do pětice finalistů. My si zde jenom ukážeme jeden z maličkých principů, na nichž stojí dvojice algoritmů BMW a EDON-R.

Ať se již jedná o BMW a EDON-R nebo jiné algoritmy, koneckonců všechny bez výjimky budou "zadrátovány" do hardware a realizovány nějakými logickými prvky XOR, AND, OR, NOT, apod. Každou funkci, nakonec realizujeme obvodem, funkčně ekvivalentním s obvodem na obrázku, a můžeme ji vyjádřit v tzv. algebraické normální formě (ANF) jako součet termů (součinů) typu *XY...Z*, kde součet i součin jsou logické operace XOR a AND (viz obrázek, kde je funkce vyjádřena jako součet termů). Možných termů je pro 64 vstupních bitů ovšem velmi vysoký, 2^64 (!). Nicméně každou funkci můžeme takto vyjádřit tímto obvodem, odpovídajícím matematicky algebraické normální formě, tj. součtu termů typu $X \oplus Z \oplus T \oplus AM \oplus BW \oplus FGTZUI \oplus ... \oplus ABCDEFGHITUVWXYZ$. Každý takový obvod můžeme proto popsat soustavou 32 rovnic (máme 32 výstupních bitů), vyjadřující každý výstupní bit pomocí ANF vstupních bitů obvodu. Kryptografické prolomení příslušné funkce je ekvivalentní naší schopnosti s těmito rovnicemi umět manipulovat. Pokud jsou složité a je jich mnoho, dostáváme se do časově i teoreticky neřešitelných matematických problémů, čili funkce je bezpečnější. Avšak čím více termů je v dané rovnici (tj. funkce bezpečnější), tím je také tento logický obvod větší nebo může trvat déle jeho výpočet nebo je jak velký tak ještě pomalý. Obecně musíme nejprve vytvořit termy (logický součin několika vstupních bitů) a poté je sečíst v binární sčítačce $\oplus$. Bude-li těchto součtů mnoho, například milion, může to znamenat časové zpoždění nebo ohromnou plochu obvodu. Tím máme přesně materiálně vyjádřen základní rozpor mezi bezpečností a rychlostí (plochou a organizací obvodu): čím bezpečnější funkce chceme, tím více termů by měla mít, tím je ovšem obvod větší nebo pomalejší. Funkce BMW a EDON-R využily faktu, že moderní procesory umí provést operaci, která vytvoří zhruba 4 miliardy termů a jejich součtů na velmi malé ploše a během jednoho cyklu. Jedná se operaci ADD. Vskutku, označme *a*, *b*, *s* 32bitové proměnné (čísla), kde 32bitové číslo s vznikne jako aritmetický součet dvou 32bitových čísel $s = a + b$. Trikem je, že bity carry, které vznikají směrem od nejnižšího k nejvyššímu jsou velmi složité termy, které se každým krokem zesložiťují jako polynomy o jeden stupeň. Navíc se počet termů v tomto bitu každým krokem zdvojnásobí, tedy (zkráceně) poslední carry bit bude obsahovat 2^31 termů!!! Skutečně, vezmeme-li si první carry bit, který vzniká, máme c1 = a0*b0. Další carry bit c2 vznikne, když se sčítají bity a1 + b1 + c1. Jednoduše máme c2 = a1*b1 $\oplus$ a1*c1 $\oplus$ b1*c1, což si můžeme napsat jako c2 = a1*b1 $\oplus$ c1*(a1 $\oplus$ b1). Jestliže předchozí bit carry (c1) měl nějaký počet termů, tak následující bit carry (c2) obsahuje kromě prvního termu (a1*b1) ještě c1*(a1 $\oplus$ b1), což jak vidíme je dvojnásobný počet termů než měl c1, a navíc mají každý o jeden vyšší stupeň (je vynásoben a1 nebo b1). Toto pravidlo platí rekurzivně, proto počet a stupeň termů můžeme odvodit exaktně. Vidíme, že počet termů se krok od kroku zhruba

zdvojnásobuje a zvyšuje se jejich stupeň o jednu. Vyjádříme-li bity *si* (*i* = 0, ..., 31) čísla *s* jako ANF bitů *a0, ..., a31* a *b0, ..., b31* dostaneme pro všech 32 bitů dohromady neuvěřitelných 2^32 + 31 = 4.294.967.327 různorodých termů od stupně 1 po stupeň 32 (přesněji *s0* obsahuje 2^0 + 1 termů, *s1* 2^1 + 1 termů, ..., *si* 2^*i* + 1 termů,..., až *s31* obsahuje 2^31 + 1 termů). Uvedli jsme ho pouze jako kuriozitu, která je kryptologům známa, a která je pro laiky většinou velkým překvapením. Pouhé aplikování operace ADD není všespasitelné, je jen stavebním prvkem, který je využit s přihlédnutím k mnoha dalším vztahům a zákonitostem, které je nutné v návrhu respektovat. Využití operace ADD však je onou "technologií", která se snaží vyřešit základní rozpor mezi bezpečnosti funkce a složitostí obvodu - operace ADD je velmi složitá, a přitom zabírá minimální místo a spotřebovává minimální čas.



Obr.: Obvod, který realizuje obecnou ANF (operace ADD by reprezentovala přes čtyři miliardy vyplněných řádků tohoto obvodu, přitom je hardwarově realizována mnohem efektivněji)

O autorovi:

RNDr. Vlastimil Klíma je absolventem Matematicko-fyzikální fakulty Univerzity Karlovy v Praze, nyní nezávislý kryptolog. V ČR je spoluzakladatelem oboru kryptoanalýzy postranními kanály. Je autorem přes 200 příspěvků a přednášek. Ve světě je znám nejrychlejší metodou hledání kolizí MD5, odhalením slabin v OpenPGP a SSL/TLS a návrhem jejich obrany proti hackerům. Navrhl nový koncept hašovacích funkcí SNMAC (HDN) a bezpečných blokových šifer (DN) pro Národní bezpečnostní úřad. V současné době se účastní světové soutěže na nový hashovací standard (SHA-3), a to dvěma nejrychlejšími kandidáty z 51. Osobní stránky: http://cryptography.hyperlink.cz