Crypto-World

Informační sešit GCUCMP

ISSN 1801-2140

Ročník 12, číslo 2/2010 **14. únor 2010**

2/2010

Připravil: Mgr. Pavel Vondruška

Sešit je přednostně distribuován registrovaným čtenářům.
Starší sešity jsou dostupné na adrese
http://crypto-world.info
(1355 registrovaných odběratelů)

Obsah :

A. Analýza Blue Midnight Wish – útok na vzor

Vlastimil Klíma, kryptolog konzultant, Praha
(http://cryptography.hyperlink.cz, v.klima@volny.cz)
Prof. Danilo Gligoroski, Norwegian University of Science and Technogy, Norway (danilog@item.ntnu.no , http://www.item.ntnu.no/people/personalpages/fac/danilog/start)

Článek navazuje na příspěvek v čísle 1 Crypto-Worldu 2010, s nímž má společnou skoro celou úvodní stranu a několik obrázků. Volně také navazuje na články o BMW v 3/2009, 7-8/2009 a 12/2009. V čísle 1 jsme se zabývali hledáním vzoru (úloha první), nyní se budeme zabývat hledáním kolize (úloha druhá). Chceme stimulovat analýzy a útoky na BMW a prezentovat otevřené problémy. Ty by se mohly stát předmětem studentských prací. Proč? Velkou výhodou oproti jiným tématům je, že tyto rozbory jsou nyní velmi žádané, ať s negativním nebo pozitivním výsledkem. Když bude problém vyřešen nebo naopak bude ukázáno, že je složitý, je to v obou případech velmi dobře publikovatelný výsledek.

Označení

Článek bude využívat označení zavedené v Crypto-Worldu 12/2009. Připomeňme jen šířku slova w = 32 nebo 64 bitů, délku bloku zprávy a průběžné haše n = 16*w (16 slov) a výpočet haše:

1. Předzpracování

(a) Doplň zprávu M jednoznačným definovaným způsobem o délku zprávy v bitech a doplněk
(b) Rozděl zprávu na celistvý násobek (N) m-bitových bloků $M^{(1)}$, ..., $M^{(N)}$.
(c) Nastav počáteční hodnotu průběžné haše $H^{(0)}$ na konstantu ($CONST^0$).

2. Výpočet haše

For i = 1 to N: $H^{(i)} = f(M^{(i)}, H^{(i-1)})$.

3. Finalizace

$H^{final} = f(H^{(N)}, CONST^{final})$, kde $CONST^{final}$ je konstanta.

4. Závěr

H(M) = dolních n bitů z hodnoty H^{final}.

BMW vždy projde minimálně dvě iterace

Jak ukazuje schéma, BMW vždy projde minimálně dvě iterace kompresní funkce f (obr. 1 a 2) - a to první a poslední. Kromě toho projde volitelně podle délky zpracovávané zprávy ještě určité množství tzv. vnitřních bloků mezi prvním a posledním (obr. 2 a 3). První a poslední blok mají pevně nastavenu hodnotu H. U prvního bloku má hodnotu $CONST^0$, u posledního bloku $CONST^{final}$. V první iteraci kompresní funkce $f(M^{(1)}, CONST^0)$ zpracovává první blok zprávy $M^{(1)}$ a konstantu $CONST^0$. Pokud tento blok zprávy není zároveň blokem posledním, následují ještě vnitřní iterace. Výsledkem je poslední průběžná haš H, která vstupuje v roli bloku zprávy (první argument f) do finalizace $f(H, CONST^{final})$.

Úloha druhá - hledání kolize

Dobrá zpráva pro útočníka je, že úloha hledání kolize je obecně a téměř jistě i u BMW mnohem snazší, než hledání vzoru. A kromě toho, i nalezení pseudokolizí by pravděpodobně BMW vyřadilo z finále o standard SHA-3 (kam v srpnu 2010 postoupí 5 kandidátů). Přitom pseudokolize jsou ještě mnohem snazší než kolize! Navíc, protože kandidátů je ještě mnoho, i ukázání nějakých blízkých pseudokolizí by ohrozilo BMW. Pak tu jsou ještě blízké pseudokolize, které jsou opět mnohem jednodušší než pseudokolize. Vezmeme v úvahu všechny typy kolizí (tj. **kolize, pseudokolize, blízké kolize a blízké pseudokolize**), ale budeme většinou hovořit krátce jen o kolizích. Pojem pseudokolize znamená, že útočník si může volit obě dvě hodnoty H a M, vstupující do kompresní funkce a dociluje kolize na výsledku kompresní funkce, tj. na průběžné hašovací hodnotě a nikoli na hodnotě hašovací funkce jako celku. Jedná se tedy o vlastně o kolizi kompresní funkce, která má dva vstupy (s dvojnásobným počtem stupňů volnosti - H i M), zatímco pro kolizi hašovací funkce musíme najít dvě různé zprávy. Pojem blízké kolize znamená, že rovnost (kolize) proměnných (haší, průběžných haší) nemusí platit v celé šíři, ale jen na části proměnné. Pokud se jedná o blízkou kolizi na kompresní funkci, jedná se o blízkou pseudokolizi, pokud na hašovací funkci, je to blízká kolize.

Odolnost proti blízkým kolizím

Je zřejmé, že význam blízkých kolizí a pseudokolizí je u hašovacích funkcí s dvojitou rourou (double pipe) a přídavným závěrečným zpracováním posledního výsledku kompresní funkce (viz "finalizace" u BMW) velmi malý ve srovnání s kompresními funkcemi s jednoduchou rourou bez závěrečného zpracování (například SHA-1, SHA-2). U jednoduché roury je totiž blízká kolize (pseudokolize) na výsledku poslední kompresní iterace zároveň blízkou kolizí (pseudokolizí) celé hašovací funkce, neboť ta u jednoduché roury přebírá celý výsledek. Naproti tomu u hašovacích funkcí s dvojitou rourou přídavné závěrečné zpracování posledního výsledku kompresní funkce případné blízké hodnoty s velkou pravděpodobností rozptýlí do náhodně vzdálených hodnot.

NIST stanovil odolnost kandidátů proti kolizím, nikoli proti pseudokolizím. Pokud hašovací funkce zabraňuje i pseudokolizím, šlechtí jí to, ale není to požadováno. Pseudokolize neznamenají ohrožení žádné potřebné a využívané vlastnosti hašovací funkce. Odolnost proti pseudokolizím zvyšuje důvěru v hašovací funkci, ale také něco stojí. BMW není a priori stavěná proti pseudokolizním útokům, takže zde má útočník velké pole působnosti.

Obecně má studium všech typů kolizí význam pro poznání vlastností dané hašovací funkce. U kvalitní hašovací funkce může být útočník velice spokojen i s takovým výsledkem jako je pseudokolize, i když není přímo použitelný.

Kolize počáteční, vnitřní a závěrečné iterace

Protože BMW má tři hlavní kroky - počáteční iteraci, (žádné nebo nějaké) vnitřní iterace a závěrečnou iteraci, odpovídají tomu i typy kolizí pro tyto tři typy iterací. U počáteční a závěrečné iterace je (blízká, pseudo) kolize složitější, protože útočník má k dispozici o jednu proměnnou méně. Hodnota průběžné haše je v těchto případech konstantní ($CONST^0$, $CONST^{final}$). Mezi počáteční a závěrečnou iterací je zase podstatný rozdíl v tom, že u závěrečné iterace máme docílit kolize na n bitech výstupu, zatímco u počáteční na 2n bitech. U vnitřní iterace musí útočník sice docílit kolize na 2n bitech, ale má k dispozici jak volbu průběžné haše o 2n bitech, tak volbu bloku zprávy o 2n bitech. Porovnáme-li počty rovnic, které vznikají a počty proměnných, dostáváme:

- počáteční iterace: proměnná M_1 (2n bitů) a M_2 (2n bitů), 2n rovnic (shoda na H: $f(M_1, H_1) = f(M_2, H_2)$), tj. **2n stupňů volnosti**
- vnitřní iterace: proměnná M_1 (2n bitů) a M_2 (2n bitů), proměnná H_1 (2n bitů) a H_2 (2n bitů), 2n rovnic (shoda na H: $f(M_1, H_1) = f(M_2, H_2)$), tj. **6n stupňů volnosti**
- závěrečná iterace: proměnná M_1 (2n bitů) a M_2 (2n bitů), n rovnic (shoda na polovině H, H^{final}: 8_lswords_of $f(M_1, H_1)$ = 8_lswords_of $f(M_2, H_2)$), tj. **3n stupňů volnosti**

Nejjednodušší se jeví vnitřní a závěrečná iterace. Vnitřní kolize je pro útočníka výhodná, protože ji může prodlužovat libovolným shodným pokračovacím blokem u obou (pseudo) kolidujících zpráv. To neplatí pro blízkou pseudokolizi ani blízkou kolizi, protože přídavný blok téměř jistě získané blízké hodnoty průběžné haše opět náhodně rozmíchá.

Postačující složitost
Ukážeme nejprve, že pokud útočník nalezne kolizi hašovací funkce BMW (ať na ní přijde jak chce), bude znát kolizi závěrečné iterace nebo pseudokolizi vnitřní iterace.

Věta
Znalost kolize hašovací funkce BMW implikuje buď znalost kolize závěrečné iterace nebo znalost pseudokolize vnitřní iterace.

Důkaz.
Důkaz vyplývá z faktu, že pokud nastane kolize BMW, můžeme u každé z kolidujících zpráv jít od posledního bloku směrem k prvnímu a zjišťovat, zda jsou odpovídající bloky obou kolidujících zpráv stejné. Pokud je různý poslední blok, útočník získal **kolizi závěrečné iterace**: 8_lswords_of $f(M_1, CONST^{final})$ = 8_lswords_of $f(M_2, CONST^{final})$. Pokud je poslední blok stejný ($M_1 = M_2$), pro předposlední hodnoty průběžných haší (H_1, H_2) a bloků zpráv (m_1, m_2) platí $f(m_1, H_1) = M_1 = M_2 = f(m_2, H_2)$, tj. $f(m_1, H_1) = f(m_2, H_2)$. Pokud je nyní $(m_1, H_1) = (m_2, H_2)$, jdeme ještě o krok zpět, dokud nenarazíme na $(m_1, H_1) \neq (m_2, H_2)$. Na takové bloky narazit musíme, protože útočník nalezl dvě různé zprávy. Při zpětném postupu od posledního bloku k prvnímu musíme tedy najít $(m_1, H_1) \neq (m_2, H_2)$ takové, že $f(m_1, H_1) = f(m_2, H_2)$, což je právě **pseudokolize vnitřní iterace.**

K důkazu složitosti nalezení kolize hašovací funkce postačí ukázat, že je příliš složité jak nalezení kolize pro závěrečnou iteraci, tak nalezení pseudokolize pro vnitřní iteraci. Nyní budeme analyzovat každý z těchto problémů zvlášť.

Kolize závěrečné iterace
Zabývejme se nyní kolizí závěrečné iterace. Útočník při ní docílí kolize na výstupní haši o 8 slovech a hledá dva různé vstupní bloky M_1 a M_2 o 16 slovech, přičemž hodnota průběžné haše H je konstanta $CONST^{final}$. Zároveň s touto úlohou bychom mohli na pozadí uvažovat, že hodnoty výstupní haše nemusí být stejné, ale blízké. Útočníkovi by například postačilo, aby se složitostí menší než při použití narozeninového paradoxu našel "blízkou" závěrečnou kolizi takovou, že obě haše se rovnají pouze na jednom slově a na zbylých jsou náhodně různé, přičemž složitost nalezení takové blízké kolize by musela být menší než $2^{w/2}$. Podobně pro dvě shodná slova by k úspěšnosti potřeboval složitost menší než 2^w, apod.
Hledání kolize (blízké kolize) závěrečné iterace je ekvivalentní hledání dvou různých bloků zpráv M_1 (2n bitů) a M_2 (2n bitů) tak, aby byla splněna ("blízce" splněna) rovnost hodnot $hash_1$ a $hash_2$ z následující soustavy rovnic (S1), (S2).

$Q_{1,a} = A_2(A_1(M_1 \oplus CONST^{final})) + ROTL^1(CONST^{final})$, (S1,1)
$Q_{1,b} = T^L(T^U(Q_{1,a}) + ((B(rotM_1) + K) \oplus ROTL^7(CONST^{final})))$, (S1,2)
$G_1 = (M_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b}))$, (S1,3)
$hash_1 = 8\_lswords\_of(\ f_6(G_1)\ )$. (S1,4)

$Q_{2,a} = A_2(A_1(M_2 \oplus CONST^{final})) + ROTL^1(CONST^{final})$, (S2,1)
$Q_{2,b} = T^L(T^U(Q_{2,a}) + ((B(rotM_2) + K) \oplus ROTL^7(CONST^{final})))$, (S2,2)
$G_2 = (M_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b}))$, (S2,3)
$hash_2 = 8\_lswords\_of(\ f_6(G_2)\ )$. (S2,4)

Tedy máme

$Q_{1,a} = A_2(A_1(M_1 \oplus CONST^{final})) + ROTL^1(CONST^{final})$, **(S3,1)**
$Q_{2,a} = A_2(A_1(M_2 \oplus CONST^{final})) + ROTL^1(CONST^{final})$, **(S3,2)**
$Q_{1,b} = T^L(T^U(Q_{1,a}) + ((B(rotM_1) + K) \oplus ROTL^7(CONST^{final})))$, **(S3,3)**
$Q_{2,b} = T^L(T^U(Q_{2,a}) + ((B(rotM_2) + K) \oplus ROTL^7(CONST^{final})))$, **(S3,4)**
$G_1 = (M_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b}))$, **(S3,5)**
$G_2 = (M_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b}))$, **(S3,6)**
$8\_lswords\_of(\ f_6(G_1)\ ) = 8\_lswords\_of(\ f_6(G_2)\ )$. **(S3,7)**

Budeme zkoumat tyto rovnice. Protože bloky **M_1 a M_2 se liší**, liší se také jejich bijektivní obrazy **$Q_{1,a}$** a **$Q_{2,a}$**. Obě tyto hodnoty vstupují do rovnice pro **G**. Tam se mohou jejich diference vzájemně vyrušit nebo je může vyrušit změna změna **$Q_{1,b}$** a **$Q_{2,b}$** nebo obojí nebo se změna může dále propagovat do **G_1 a G_2**. Pokud by útočník chtěl, aby se změny vyrušily v hodnotě G, docílil by vlastně kolize kompresní funkce v plné šíři 16 slov. Tím by zkoumání jednodušší úlohy kolize na 8 slovech přeměnil na zkoumání složitější úlohy kolize na 16 slovech. Proto v případě závěrečné iterace budeme zejména zkoumat možnost, že změna se propaguje do hodnot G a teprve po transformaci f_6 dojde na dolních 8 slovech $f_6(G)$ ke shodě. Aby útočník mohl změny v hodnotě G ovlivňovat, pravděpodobně bude muset minimálně dobře prostudovat diferenční chování funkcí

$Q_a : M \dashrightarrow Q_a(M) = A_2(A_1(M \oplus CONST^{final})) + ROTL^1(CONST^{final})$, (S3a)
$Q_b : M \dashrightarrow Q_b(M) = T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(CONST^{final})))$, (S3b)
tj. chování funkcí

$Q_a : M \dashrightarrow Q_a(M) = A_2(A_1(M \oplus c_1)) + c_2$, **(S3c)**
$Q_b : M \dashrightarrow Q_b(M) = T^L(T^U(A_2(A_1(M \oplus c_1)) + c_2) + ((B(rotM) + c_3) \oplus c_4))$, **(S3d)**

kde c_i jsou nějaké (obecně různé) konstanty.

Pokud se ukáže, že tyto funkce nemají predikovatelné diferenční chování, útočník se bude muset zaměřit na funkci G jako celek, což je pravděpodobně ještě složitější úloha.

<u>Výzkum vlastností funkcí $Q_a(M)$ a $Q_b(M)$ je klíčový.</u>

Pseudokolize vnitřní iterace
Blízká pseudokolize u vnitřní iterace útočníkovi nestačí, protože závěrečná iterace by blízkou pseudokolizi znáhodnila. Hledáme tedy čtyři proměnné $(M_1, H_1) \neq (M_2, H_2)$ takové, že průběžná haš z nich spočítaná, je stejná: $f(M_1, H_1) = f(M_2, H_2)$. Je-li průběžná haš stejná, jsou stejné i hodnoty G. Tuto úlohu můžeme napsat následovně:

$Q_{1,a} = A_2(A_1(M_1 \oplus H_1)) + ROTL^1(H_1)$,
$Q_{1,b} = T^L(T^U(Q_{1,a}) + ((B(rotM_1) + K) \oplus ROTL^7(H_1)))$,
$G = (M_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b}))$,
$f(M_1, H_1) = f_6(G)$.

$Q_{2,a} = A_2(A_1(M_2 \oplus H_2)) + ROTL^1(H_2)$,
$Q_{2,b} = T^L(T^U(Q_{2,a}) + ((B(rotM_2) + K) \oplus ROTL^7(H_2)))$,
$G = (M_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b}))$,
$f(M_2, H_2) = f_6(G)$,

neboli

$$\mathbf{Q_{1,a} = A_2(A_1(M_1 \oplus H_1)) + ROTL^1(H_1),} \quad \mathbf{(S4,1)}$$
$$\mathbf{Q_{2,a} = A_2(A_1(M_2 \oplus H_2)) + ROTL^1(H_2),} \quad \mathbf{(S4,2)}$$
$$\mathbf{Q_{1,b} = T^L(T^U(Q_{1,a}) + ((B(rotM_1) + K) \oplus ROTL^7(H_1))),} \quad \mathbf{(S4,3)}$$
$$\mathbf{Q_{2,b} = T^L(T^U(Q_{2,a}) + ((B(rotM_2) + K) \oplus ROTL^7(H_2))),} \quad \mathbf{(S4,4)}$$
$$\mathbf{(M_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b})) = (M_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b})).} \quad \mathbf{(S4,5)}$$

V této soustavě máme 4 volné proměnné o 16 slovech (M_1, H_1, M_2, H_2) a jednu rovnici (S4,5) o šířce 16 slov, tedy 48 volných slov. Můžeme **(S4) řešit v této obecnosti**, tj. (S4,1) - (S4,5) nebo se pokusit najít nějaká **speciální řešení**. Nyní vybereme několik přímočarých postupů, které soustavu (S4) zjednodušují. Jedná se jen o ilustraci možností řešení, nic jiného.

Varianta 1: Volíme (M_1, H_1) a hledáme (M_2, H_2).
Ve skutečnosti se jedná o úlohu hledání pseudovzoru, neboť hodnotu $c_0 = f(M_1, H_1)$ známe a hledáme (M_2, H_2) tak, aby $f(M_2, H_2) = c_0$, tedy pseudovzor hodnoty c_0. Je to řešení této soustavy pro neznámé H, M, Q_a, Q_b:

$$\mathbf{Q_a = A_2(A_1(M \oplus H)) + ROTL^1(H),} \quad \mathbf{(S5,1)}$$
$$\mathbf{Q_b = T^L(T^U(Q_a) + ((B(rotM) + K) \oplus ROTL^7(H))),} \quad \mathbf{(S5,2)}$$
$$\mathbf{(M \oplus L_a(Q_b)) + (Q_a \oplus L_b(Q_b)) = c_0.} \quad \mathbf{(S5,3)}$$

Varianta 2: Volíme M_1, M_2 a hledáme H_1, H_2.
Je to problém:

$$\mathbf{Q_{1,a} = A_2(A_1(c_1 \oplus H_1)) + ROTL^1(H_1),} \quad \mathbf{(S6,1)}$$
$$\mathbf{Q_{2,a} = A_2(A_1(c_2 \oplus H_2)) + ROTL^1(H_2),} \quad \mathbf{(S6,2)}$$
$$\mathbf{Q_{1,b} = T^L(T^U(Q_{1,a}) + (c_3 \oplus ROTL^7(H_1))),} \quad \mathbf{(S6,3)}$$
$$\mathbf{Q_{2,b} = T^L(T^U(Q_{2,a}) + (c_4 \oplus ROTL^7(H_2))),} \quad \mathbf{(S6,4)}$$
$$\mathbf{(c_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b})) = (c_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b})).} \quad \mathbf{(S6,5)}$$

Pokud $Q_{i,a}$ jako funkce H_i je pro útočníka jednosměrná, nemůže z hodnoty $Q_{i,a}$ určovat H_i, ale musí naopak z hodnoty H_i určovat $Q_{i,a}$. Jenže H_i jsou jediné volné proměnné v soustavě. Pokud jedno z nich volíme, dostáváme opět úlohu nalezení pseudovzoru. Pokud žádné z nich nevolíme celé, máme soustavu (S6), kde funkce $Q_{i,a}$ jsou jednosměrné. Podobnou úvahu můžeme učinit pro hodnoty $Q_{i,b}$. Pokud $Q_{i,a}$ a $Q_{i,b}$ jsou pro útočníka jednosměrné funkce, pak i kdyby z rovnice (S6,5) získal nějaké informace o $Q_{i,a}$, $Q_{i,b}$ (i celé jejich hodnoty), díky

jednosměrnosti z nich bude obtížně zjišťovat hodnoty H_i. V této úloze je tedy důležité zjistit co nejvíce informací o funkcích $Q_{i,a}$, $Q_{i,b}$, zejména **zda a do jaké míry platí**

Hypotéza QAH:
$$\mathbf{Q_a : H \dashrightarrow Q_a(H) = A_2(A_1(c_1 \oplus H)) + ROTL^1(H)} \qquad \mathbf{(S7)}$$
je jednosměrná náhodná funkce proměnné H,

Hypotéza QBH:
$$\mathbf{Q_b : H \dashrightarrow Q_b(H) = T^L(T^U(A_2(A_1(c_1 \oplus H)) + ROTL^1(H)) + (c_3 \oplus ROTL^7(H)))}, \qquad \mathbf{(S8)}$$
je jednosměrná náhodná funkce proměnné H.

Zcela základní úlohou je (S7), neboť (S8) využívá výsledku zkoumání (S7).

Předpokládejme, že útočník vyzkoumá vlastnosti $Q_a(H)$ velmi dobře a bude tak schopen

najít (velkou) množinu hodnot H, pro něž $Q_a(H)$ je konstantní. **(S9)**

V tom případě může řešit soustavu (S6) pro získanou množinu hodnot H. Dostává tak

$$\mathbf{Q_{1,a} = c_5,} \qquad \mathbf{(S6a,1)}$$
$$\mathbf{Q_{2,a} = c_6,} \qquad \mathbf{(S6a,2)}$$
$$\mathbf{Q_{1,b} = T^L(c_7 + (c_3 \oplus ROTL^7(H_1))),} \qquad \mathbf{(S6a,3)}$$
$$\mathbf{Q_{2,b} = T^L(c_8 + (c_4 \oplus ROTL^7(H_2))),} \qquad \mathbf{(S6a,4)}$$
$$\mathbf{(c_1 \oplus L_a(Q_{1,b})) + (c_5 \oplus L_b(Q_{1,b})) = (c_2 \oplus L_a(Q_{2,b})) + (c_6 \oplus L_b(Q_{2,b})).} \qquad \mathbf{(S6a,5)}$$

Kdybychom soustavu (S6a) dost zjednodušili a místo (S6a,5) řešili zcela jednoduchou rovnici $Q_{1,b} = Q_{2,b}$, pak bychom hledali H_1 a H_2 tak, že

$$T^L(c_7 + (c_3 \oplus ROTL^7(H_1))) = T^L(c_8 + (c_4 \oplus ROTL^7(H_2))). \qquad (S10)$$

Protože obě strany rovnice jsou bijektivní a snadno invertovatelné obrazy, můžeme volit H_1 libovolně a H_2 jen z (S10) dopočítat. Tím bychom úlohu ve variantě 2 mohli vyřešit a získat dokonce velkou množinu řešení. Zbývá pouze umět řešit úlohu (S9).

Ukazuje se také užitečnost problému nalezení obecného řešení rovnice (S6a,5) neboli

vyzkoumat chování funkce
$$\mathbf{G_{qb} : Q \dashrightarrow G_{qb}(Q) = (c_0 \oplus L_a(Q)) + (c_1 \oplus L_b(Q))} \qquad \mathbf{(S11)}$$

Připomeňme, že $L = L_a \oplus L_b$ je bijekce a tudíž je velmi zajímavé vyzkoumat možnost aproximace funkce G_{qb} funkcí typu $L(Q) \oplus c$ nebo $L(Q) + c$. Dále je zajímavé naopak vyzkoumat množinu hodnot Q, na nichž je G_{qb} konstantní. Ty totiž dávají velmi mnoho dvojic řešení rovnice (S6a,5), která nás velice zajímá.

Útočník má možností mnohem více, dalším klíčovým místem jsou vlastnosti funkce (S8).

Varianta 3: Volíme H_1, H_2 a hledáme M_1, M_2.
Je to problém ekvivalentní soustavě:

$$\mathbf{Q_{1,a} = A_2(A_1(M_1 \oplus c_1)) + c_2,} \quad \mathbf{(S12,1)}$$
$$\mathbf{Q_{2,a} = A_2(A_1(M_2 \oplus c_3)) + c_4,} \quad \mathbf{(S12,2)}$$
$$\mathbf{Q_{1,b} = T^L(T^U(Q_{1,a}) + ((B(rotM_1) + K) \oplus c_5)),} \quad \mathbf{(S12,3)}$$
$$\mathbf{Q_{2,b} = T^L(T^U(Q_{2,a}) + ((B(rotM_2) + K) \oplus c_6)),} \quad \mathbf{(S12,4)}$$
$$\mathbf{(M_1 \oplus L_a(Q_{1,b})) + (Q_{1,a} \oplus L_b(Q_{1,b})) = (M_2 \oplus L_a(Q_{2,b})) + (Q_{2,a} \oplus L_b(Q_{2,b})).} \quad \mathbf{(S12,5)}$$

Tento problém je analogický předchozímu problému, je však o něco složitější díky přítomnosti funkce B(rotM).

Obecná poznámka: Pro jednoduchost lze všechny uvedené rovnice a soustavy uvažovat nejprve pro nulové konstanty.

Příklad č.1.
Zajímavý příklad pro nulové konstanty poskytuje hledání pseudokolize ve Variantě 2, a to na úrovni proměnné Q_a: $Q_{1,a} = Q_{2,a}$, kde

$$Q_{1,a} = A_2(A_1(c_1 \oplus H_1)) + ROTL^1(H_1), \quad (S6,1)$$
$$Q_{2,a} = A_2(A_1(c_2 \oplus H_2)) + ROTL^1(H_2), \quad (S6,2)$$

Máme tedy nulové M_1 a M_2 a hledáme H_1 a H_2 takové, že
$A_2(A_1(H_1)) + ROTL^1(H_1) = A_2(A_1(H_2)) + ROTL^1(H_2)$.

Je to soustava rovnic:

$$\begin{array}{lcl}
H^1_1 + s_0(H^1_5 - H^1_7 + H^1_{10} + H^1_{13} + H^1_{14}) & = & H^2_1 + s_0(H^2_5 - H^2_7 + H^2_{10} + H^2_{13} + H^2_{14}) \\
H^1_2 + s_1(H^1_6 - H^1_8 + H^1_{11} + H^1_{14} - H^1_{15}) & = & H^2_2 + s_1(H^2_6 - H^2_8 + H^2_{11} + H^2_{14} - H^2_{15}) \\
H^1_3 + s_2(H^1_0 + H^1_7 + H^1_9 - H^1_{12} + H^1_{15}) & = & H^2_3 + s_2(H^2_0 + H^2_7 + H^2_9 - H^2_{12} + H^2_{15}) \\
H^1_4 + s_3(H^1_0 - H^1_1 + H^1_8 - H^1_{10} + H^1_{13}) & = & H^2_4 + s_3(H^2_0 - H^2_1 + H^2_8 - H^2_{10} + H^2_{13}) \\
H^1_5 + s_4(H^1_1 + H^1_2 + H^1_9 - H^1_{11} - H^1_{14}) & = & H^2_5 + s_4(H^2_1 + H^2_2 + H^2_9 - H^2_{11} - H^2_{14}) \\
H^1_6 + s_0(H^1_3 - H^1_2 + H^1_{10} - H^1_{12} + H^1_{15}) & = & H^2_6 + s_0(H^2_3 - H^2_2 + H^2_{10} - H^2_{12} + H^2_{15}) \\
H^1_7 + s_1(H^1_4 - H^1_0 - H^1_3 - H^1_{11} + H^1_{13}) & = & H^2_7 + s_1(H^2_4 - H^2_0 - H^2_3 - H^2_{11} + H^2_{13}) \\
H^1_8 + s_2(H^1_1 - H^1_4 - H^1_5 - H^1_{12} - H^1_{14}) & = & H^2_8 + s_2(H^2_1 - H^2_4 - H^2_5 - H^2_{12} - H^2_{14}) \\
H^1_9 + s_3(H^1_2 - H^1_5 - H^1_6 + H^1_{13} - H^1_{15}) & = & H^2_9 + s_3(H^2_2 - H^2_5 - H^2_6 + H^2_{13} - H^2_{15}) \\
H^1_{10} + s_4(H^1_0 - H^1_3 + H^1_6 - H^1_7 + H^1_{14}) & = & H^2_{10} + s_4(H^2_0 - H^2_3 + H^2_6 - H^2_7 + H^2_{14}) \\
H^1_{11} + s_0(H^1_8 - H^1_1 - H^1_4 - H^1_7 + H^1_{15}) & = & H^2_{11} + s_0(H^2_8 - H^2_1 - H^2_4 - H^2_7 + H^2_{15}) \\
H^1_{12} + s_1(H^1_8 - H^1_0 - H^1_2 - H^1_5 + H^1_9) & = & H^2_{12} + s_1(H^2_8 - H^2_0 - H^2_2 - H^2_5 + H^2_9) \\
H^1_{13} + s_2(H^1_1 + H^1_3 - H^1_6 - H^1_9 + H^1_{10}) & = & H^2_{13} + s_2(H^2_1 + H^2_3 - H^2_6 - H^2_9 + H^2_{10}) \\
H^1_{14} + s_3(H^1_2 + H^1_4 + H^1_7 + H^1_{10} + H^1_{11}) & = & H^2_{14} + s_3(H^2_2 + H^2_4 + H^2_7 + H^2_{10} + H^2_{11}) \\
H^1_{15} + s_4(H^1_3 - H^1_5 + H^1_8 - H^1_{11} - H^1_{12}) & = & H^2_{15} + s_4(H^2_3 - H^2_5 + H^2_8 - H^2_{11} - H^2_{12}) \\
H^1_0 + s_0(H^1_{12} - H^1_4 - H^1_6 - H^1_9 + H^1_{13}) & = & H^2_0 + s_0(H^2_{12} - H^2_4 - H^2_6 - H^2_9 + H^2_{13})
\end{array}$$

Příklad č.2
Pro pevné H, nalézt blízké M_1 a M_2 tak, že
$Q_{1,a} = A_2(A_1(M_1 \oplus H)) + ROTL^1(H)$, (S6,1)
$Q_{2,a} = A_2(A_1(M_2 \oplus H)) + ROTL^1(H)$, (S6,2)
jsou speciálně blízké. Definujeme speciálně blízké hodnoty M_1 a M_2 tak, že jsou si rovny na všech slovech, kromě posledního patnáctého, kde jsou shodné na co možná nejvíce bitech. Například $Q_{1,a}$ a $Q_{2,a}$ jsou si speciálně blízké, když

$Q_{1,a}[0] = Q_{2,a}[0]$,
$Q_{1,a}[1] = Q_{2,a}[1]$,
...

$Q_{1,a}[14] = Q_{2,a}[14]$,
$Q_{1,a}[15] = Q_{2,a}[15] \oplus 1$.

Místo jedničky může být ovšem libovolná konstanta. Můžeme zkoušet nejprve konstanty (w-bitová slova) obsahující jeden jedničkový bit, pak dva bity atd.

Závěr

V tomto článku jsme uvedli několik dílčích úloh a problémů k řešení, které se objevují při hledání kolize hašovací funkce BMW. Řešení všech těchto úloh je otevřené. Některé vypadají velmi jednoduše a budeme rádi, pokud nás přesvědčíte o tom, že nejen vypadají. Velice doporučujeme začít s analýzou těch nejjednodušších, což je zkoumání vlastností funkcí $Q_a(M)$, $Q_b(M)$ a $G_{qb}(Q)$ nebo se podívat na hypotézy QAH, QBH a problémek (S9). Naopak Soustavu (S3) a (S4) si můžeme nechat nakonec a dříve řešit mnohem jednodušší speciální případy (S5), (S6) a (S12).

Na samotný závěr dáváme provokativní otázku a výzvu. Zdá se Vám funkce
$Q_a : M \dashrightarrow Q_a(M) = A_2(A_1(M \oplus c_1)) + c_2$, (S3c)
příliš složitá? Asi ne, ale ještě ji zjednodušíme. Zkoumejme jenom

$Q_a : M \dashrightarrow Q_a(M) = A_2(A_1(M))$. **(S3c,0)**

Jednoduchoučké, že. Dokážete vyzkoumat diferenční vlastnosti této funkce? Postačí říci cokoli použitelného o chování $A_2(A_1(M \oplus dif))$ nebo o $A_2(A_1(M + dif))$ ve vztahu k $A_2(A_1(M))$ pro diference dif.

Doufejme, že s příspěvky čtenářů na toto téma vás budeme moci seznámit (anonymně nebo se souhlasem) v některém z dalších čísel Crypto-worldu.

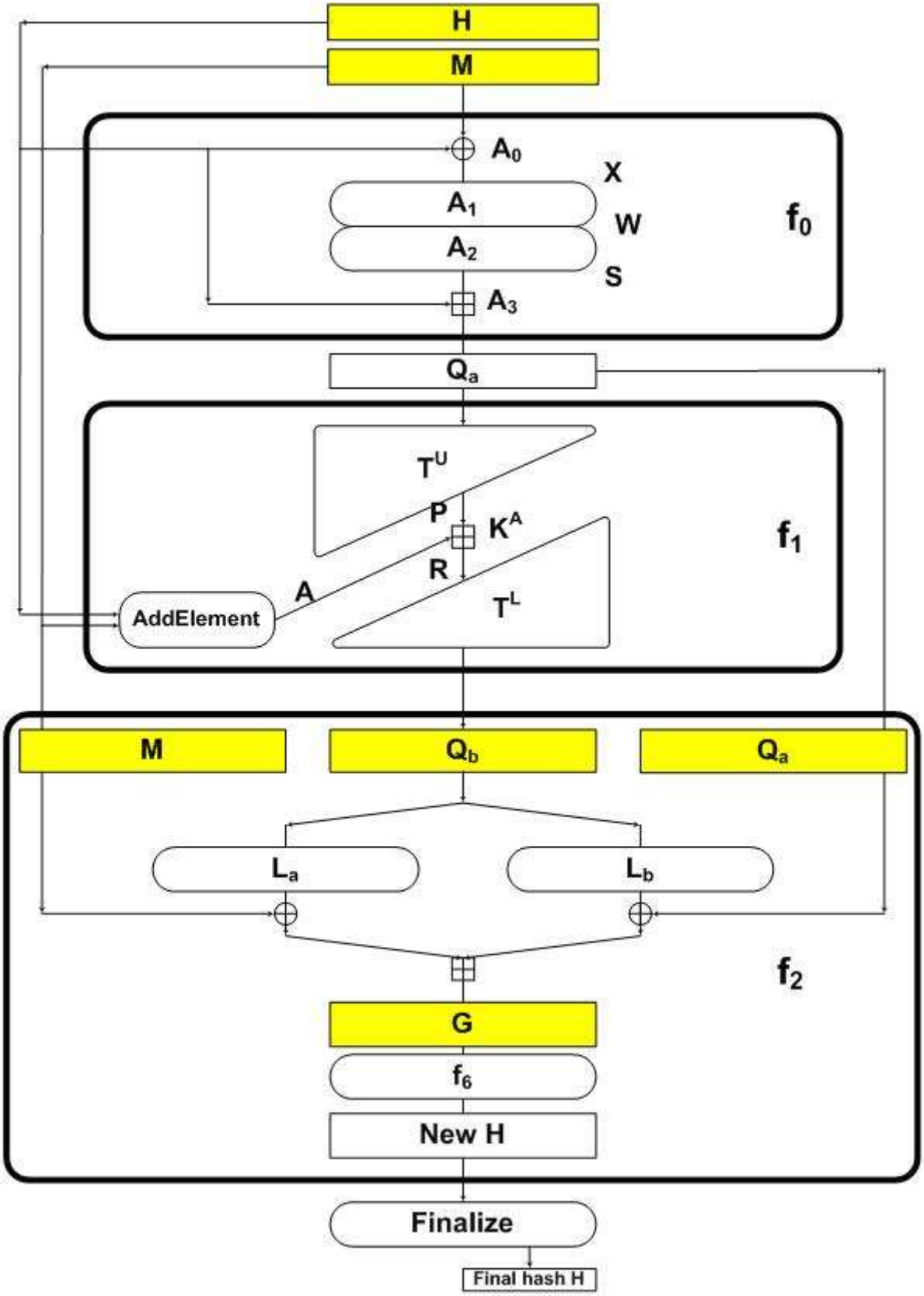
Errata

V minulém dílu došlo k menší drobné chybě, když jsme na několika místech zaměnili H^{final} za $CONST^{final}$. Čtenář to jistě postřehne, nicméně nás to mrzí a tímto se omlouváme.

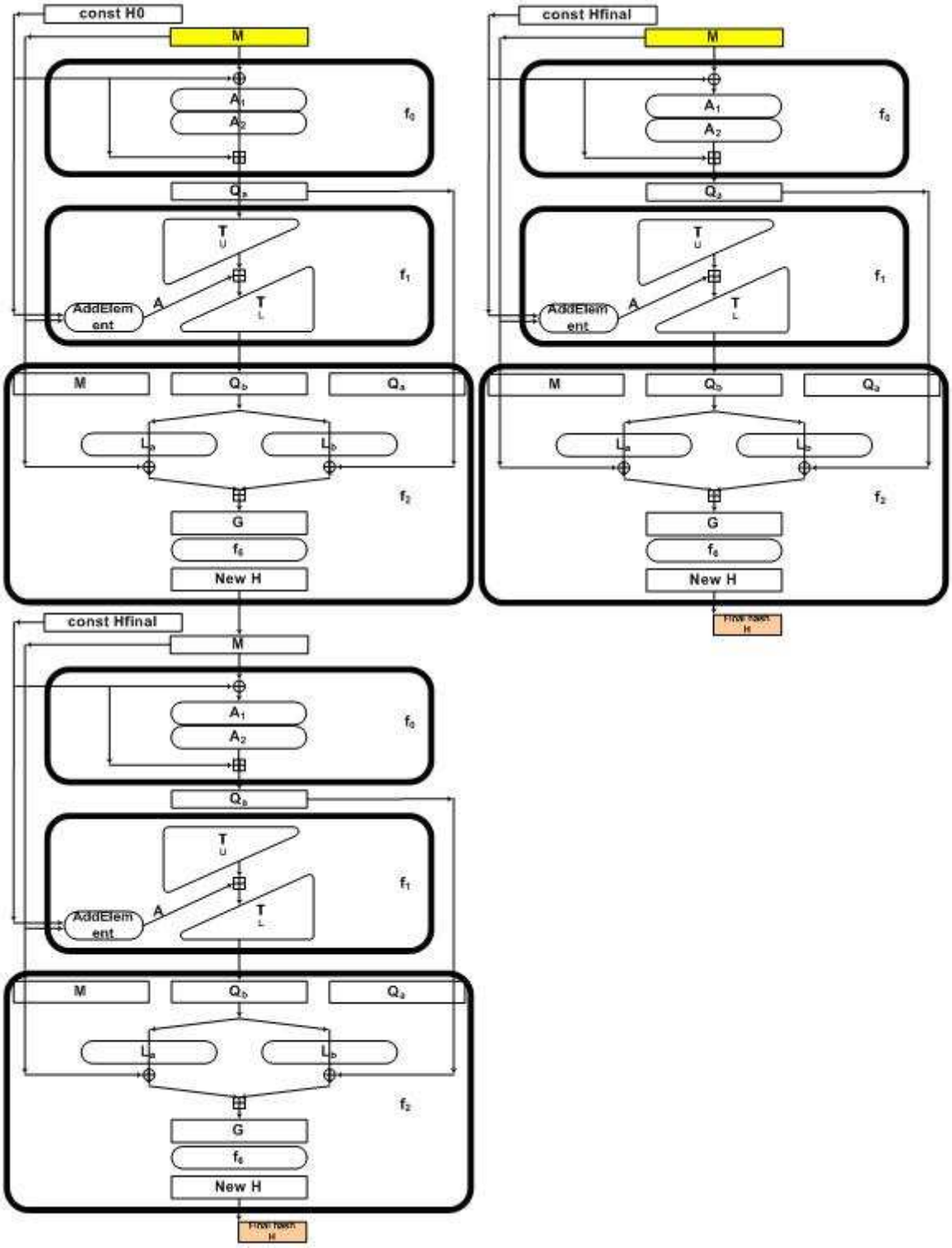
Literatura

[1] domácí stránka týmu BMW: http://www.q2s.ntnu.no/sha3_nist_competition/start
[2] stránka NIST k projektu SHA-3: http://csrc.nist.gov/groups/ST/hash/index.html
[3] dokumenty a analýzy BMW a průběžné novinky k projektu SHA-3: http://cryptography.hyperlink.cz/BMW/BMW_CZ.html
[4] Danilo Gligoroski, Vlastimil Klima, On Blue Midnight Wish Decomposition, SantaCrypt 2009, Dec. 3-4, 2009, Prague, Czech Republic, Proceedings of SantaCrypt 2009, ISBN 978-80-904257-0-5, pp. 41-51

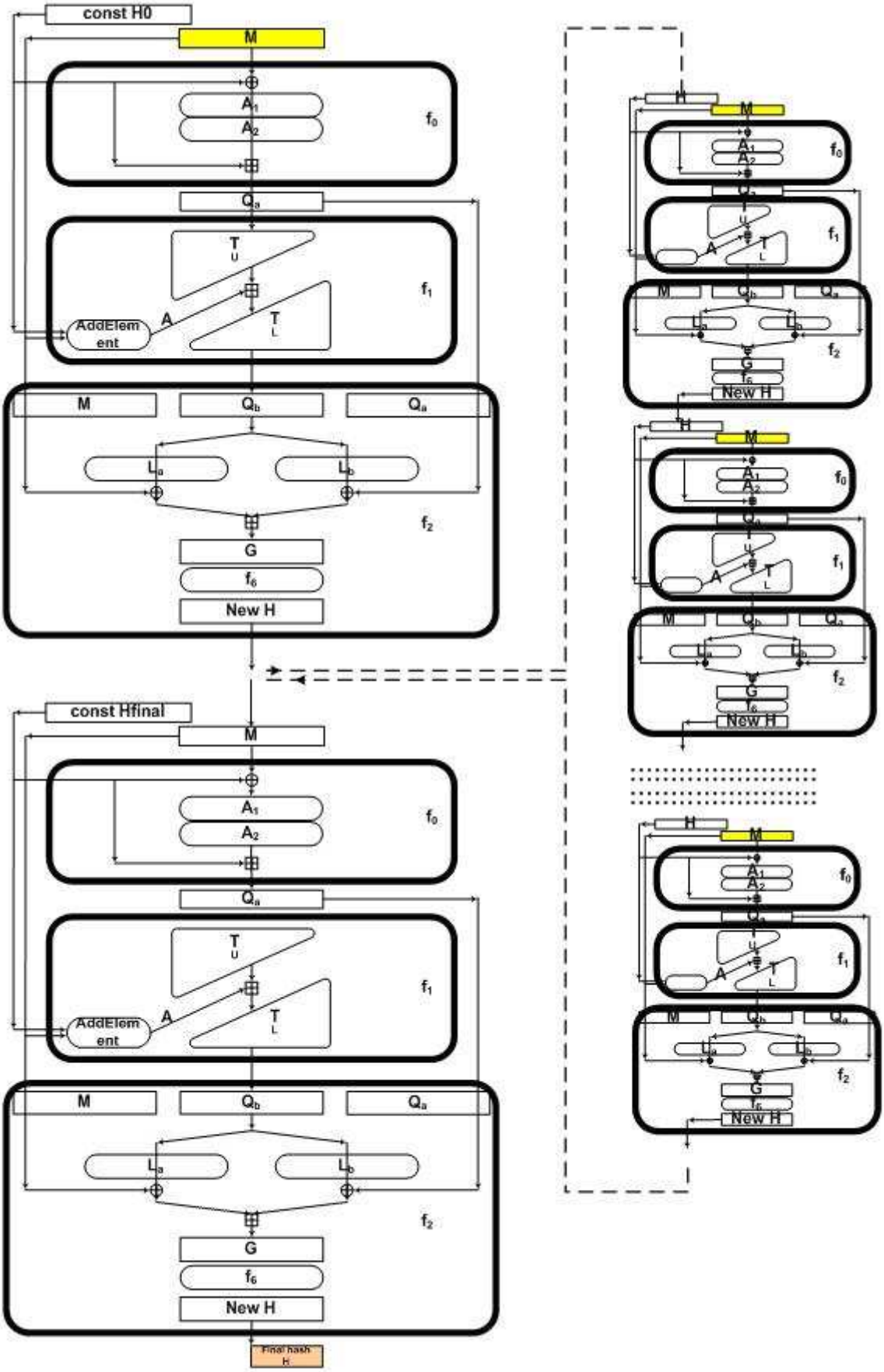
Příloha – Obrázky



Obr.1: Kompresní funkce BMW



Obr.2: Rozdíl složitosti BMW bez a s přidanou finalizací (jediný vstup je jeden blok na počátku, výstup je na konci, vše ostatní je funkce zpracování zprávy o délce jednoho bloku)



Obr.3: BMW s vnitřními bloky

B. Kryptologie, šifrování a tajná písma - ukázka z knihy

Pavel Vondruška (pavel.vondruska@crypto-world.info)

Kniha ***Kryptologie, šifrování a tajná písma*** vyšla v roce 2006 v nakladatelství Albatros v edici OKO. V současné době je již téměř rozebrána. Objednat se dá v některých internetových obchodech nebo přímo v nakladatesltví Albatros.
Více informací o knize a další ukázky najdete na stránce autora
http://crypto-world.info/oko/index.php

Ukázka z kapitoly 5. Důležitá data a mezníky v dějinách kryptologie

1883

Holandský kryptolog Auguste Kerckhoffs (1835-1903) vydal knihu *La Cryptographie Militaire* (Vojenská kryptografie).

Kniha by byla významnou prací již pouze tím, že v ní Kerckhoffs publikuje metodu, jak rozluštit obecnou polyalfabetickou šifru s neperiodickým klíčem za předpokladu, že klíč byl použit vícekrát. Z hlediska vývoje kryptografie Kerckhoffsovu knihu proslavila především skutečnost, že hledal odpovědi na praktické problémy, které vyvstaly před kryptologií v nových podmínkách (masové nasazení polní šifry, potřeba šifrovat telegrafní zprávy, jednoduchost provozu). Autor v knize uvádí řadu požadavků, které by měl vojenský šifrovací systém splňovat. Tyto požadavky jsou známy jako tzv. Kerckhoffsovy principy. Snad nejdůležitější zásadou, kterou lze z těchto principů snadno odvodit, je, že systém musí odolat i za předpokladu, kdy protivník zná šifrovací systém a nezná pouze šifrovací klíče. Ještě nyní se totiž objevují přístupy, že se šifrovací algoritmy tají. Takovému přístupu se říká "security through obscurity", český ekvivalent se zatím nevžil, možná bychom řekli "bezpečnost založená na neznalosti". Utajení vlastního šifrovacího systému je možné využít jako doplňkového bezpečnostního opatření. Aplikuje se například u šifrovacích systémů použitých špionážními službami a ozbrojenými silami, ale v žádném případě nesmí sloužit jako opatření nahrazující nebo garantující kvalitu šifrování nebo ochrany. V současné době se vžilo pravidlo, že u systémů, které jsou určeny pro veřejnost, by měl být popis šifrovacího systému veřejně dostupný.

Kerckohoffsovy hlavní zásady:

První zásada: Je nutné zásadně rozlišovat mezi šifrovacím systémem, který má sloužit k okamžité výměně dopisů mezi několika samostatnými, izolovanými osobami, a mezi

kryptografickou metodou, podle které by se řídila korespondence mezi jednotlivými armádami pro neomezenou dobu.

Druhá zásada : Pouze kryptoanalytici jsou schopni ohodnotit, do jaké míry je systém spolehlivý a bezpečný.

Šest Kerckhoffsových speciálních požadavků na polní šifry:

1) Systém musí být teoreticky nerozluštitelný nebo alespoň nerozluštitelný v praxi.
2) Vyzrazení systému nesmí mít nepříjemné následky pro dopisovatele.
3) Klíč musí být takový, aby se dal zapamatovat bez písemných poznámek a musí být snadno měnitelný.
4) Kryptogramy se musí dát posílat telegrafickou cestou.
5) Přístroj nebo dokumenty musí být přenosné a musí dovolovat manipulaci pouze jedné osobě.
6) Systém musí být jednoduchý, nesmí klást nároky na úroveň vědomostí ani nesmí být zvládnutelný pomocí dlouhého seznamu pravidel ani nesmí vyvolávat příliš velké duševní namáhání.

Kniha značně zpopularizovala a zvýšila zájem o tuto problematiku u luštitelů – amatérů i profesionálů sloužících ve francouzské armádě, a zajistila tak Francii vedoucí postavení ve vojenské kryptologii na konci 19. století (od roku 1873 byl Kerckhoffs naturalizovaný Francouz).

Po vydání knihy zaměřil A. Kerckhoffs všechny své síly na jinou profesní oblast a věnoval se budování tehdy nesmírně oblíbeného umělého „světového jazyka“ (volapük). Z Francie se tento jazyk šířil do celého světa a právě A. Kerckoffs byl jedním z jeho nejaktivnějších propagátorů. V roce 1887 byl v Mnichově na sjezdu stoupenců tohoto jazyka zvolen ředitelem mezinárodní akademie volapüku. V roce 1889 mělo hnutí již 210 000 stoupenců. Na přelomu století se však hnutí zhroutilo a upadlo v zapomenutí. Pro představu, jak jazyk vypadal, alespoň jedna věta: *El Paris binon cifazif Fransäna* (Paříž je hlavní město Francie).

1888

Francouz de Viaris (1847-1901, vlastním jménem Marquis Gaetan Henri Leon Viarizio di Lesegno) publikoval v odborném časopise *Le Génie Civil* dvoudílný příspěvek, který později vyšel jako kniha pod názvem *Cryptographie*. Příspěvek je cenný tím, že pomocí vzorců osvětlil konstrukci polyalfabetických systémů. K tomuto účelu zavedl nejprve převod abecedy

na čísla (A = 0, B = 1, …, y = 24, z = 25). Potom zavedl označení pro libovolný znak šifrového textu jako řecké písmeno χ (chí), libovolný znak klíče Γ (gama) a libovolný znak otevřeného textu písmeno *c*. Následně dokázal, že algebraický vzorec $c + \Gamma = \chi$ popisuje Vigenérovo šifrování, a to samozřejmě bez ohledu na to, jak je technicky realizováno (tabulkou, kryptografickým proužkem, šifrovacím kotoučem). Jeho značení mělo vliv na používanou terminologii. Dokonce ještě sto let po tomto článku se používalo ve slangové hantýrce mezi kryptology označení pro klíč nebo heslo slovo *gama*.

Použijeme-li dnes běžnější značení (*K* znak klíče, *O* znak otevřeného textu a *Š* znak šifrového textu), pak vzorce jednoznačně popisující nejpoužívanější polyalfabetické systémy jsou:

systém	**šifrování**	**dešifrování**
Vigenére	$O + K = Š$	$Š – K = O$
Beaufort	$K – O = Š$	$K – Š = O$
Varianta Beaufort	$O – K = Š$	$Š + K = O$

1890

Historici si byli vědomi toho, že dokumenty zašifrované Velkou šifrou, kterou navrhl Rossignol, mohou poskytnout unikátní pohled na události, jež se ve Francii odehrály v 17. století. Roku 1890 vojenský historik Victor Gendron našel svazek do té doby neznámých dopisů Ludvíka XIV. zašifrovaných Velkou šifrou. Předal je zkušenému luštiteli z šifrovacího oddělení ministerstva zahraničí Étiennu Bazeriesovi. Ten nad nimi strávil následující tři roky života a podařilo se mu systém prolomit. Mezi dopisy, které byly po dvou stech letech přečteny, byl jeden velmi zajímavý, který pravděpodobně vysvětluje jednu z největších záhad 17. století – identitu Muže se železnou maskou. Plyne z něj, že nešlo o dvojče Ludvíka XIV., uvězněné proto, aby se předešlo jakémukoli sporu o trůn, ale vojenského velitele Viviena de Bulonde. Muže, který ohrozil tažení francouzské armády do Itálie. Při útoku na město Cuneo ležící na francouzsko-italských hranicích zbaběle z místa utekl a ponechal zde spoustu zraněných vojáků a munice. Za svoji zradu byl potrestán vězením, které bylo zpřísněno o nošení masky.

1894

Literát, politik a duchovní vůdce kubánského lidu José Martí (1853-1895) padl během povstání, které vypuklo v roce 1895. V přípravě revolučního boje a jeho řízení využíval numerickou variantu Vigenérovy šifry. Přitom v té době již byla známo, že šifra je luštitelná. Martí navíc použil krátké a lehce uhádnutelné klíčové heslo HABANA.

1894

Alfred Dreyfus (1859-1935), člen francouzského generálního štábu, byl zatčen pro podezření z toho, že napsal jakýsi dokument, kterým byly nabídnuty vojenské informace Německu. Zpráva se dostala do novin, kde byl Dreyfus obviněn ze špionáže ve prospěch Německa a Itálie. Italský vojenský přidělenec Alessandro Panizzardi však o tomto muži nic nevěděl a neznal jej ani jeho německý kolega. Z tohoto důvodu reagoval na zprávu o zatčení tím, že napsal na ministerstvo zahraničních věcí v Římě. „Jestliže ***kapitán*** Dreyfus neměl žádné styky s Vámi, bylo by účelné pověřit velvyslance, aby učinil oficiální dementi, aby se **zabránilo** komentářům v tisku."

K zašifrování této zprávy použil volně dostupný obchodní telegrafní kód Baravelli, z důvodu bezpečnosti byly jednotlivé kódy přešifrovány. Telegram, který sehrál v tomto příběhu rozhodující roli, zněl:

913 44 ***7836*** 527 3 88 706 6458 71 18 0288 5715 3716 7567 7943 2107 0018 **7606** 4891 6165

Telegram se luštitelům ministerstva zahraničí podařilo dešifrovat, a přestože jasně potvrzoval Dreyfusovu nevinu, nebyl předložen jako doličný předmět a Dreyfus byl shledán vinným a byl za vlastizradu internován na Ďábelské ostrovy.

Následovalo pětileté období zákulisních tahanic a pokusů text telegramu použít jako důkaz jeho neviny a nebo naopak interpretovat v jeho neprospěch. Dreyfuse v roce 1899 nezbavilo viny dokonce ani opakované nezávislé prokázání správného dešifrování textu. Trvalo ještě dalších sedm let, než se mu dostalo v roce 1906 spravedlnosti, byl plně rehabilitován a dostal řád Čestné legie.

Mezitím se zjistilo, že skutečným pisatelem memoranda byl major Ferdinand Walsin Esterhazy (1847–1923). Mezi dokumenty, které u něj byly objeveny při zatčení, se našlo několik otočných mřížek, které pravděpodobně používal pro komunikaci s německým vojenským přidělencem.

Poznámka:

Přešifrování kódů prováděli Italové následovně: první číslici otevřeného kódu zaměnili za její doplněk do devíti (0 = 9, 1 = 8, … , 8 = 1, 9 = 0) a zapsali na druhé místo šifrového kódu. Druhou číslici otevřeného kódu převedli podle tabulky (0 = 1, 1 = 3, 2 = 5, 3 = 7, 4 = 9, 5 = 0, 6 = 2, 7 = 4, 8 = 6, 9 = 8) a zapsali na první místo zašifrovaného kódu. Podle tohoto systému se např. kód z obchodní kódové telegrafní knihy Baravelli pro slovo capitano (kapitán) = 1336 přešifruje na ***7836***, slovo evitare (vyhnout se/zabránit) =3306 na **7606** atd.

C. Chcete si zaluštit? Díl 3.

Martin Kolařík (marram.mail@gmail.com)

Únorová dávka luštění. Tentokrát jsem vybral „keše“, které jsou jejich autory hodnoceny vyšší obtížností řešení, zda je tomu opravdu tak ponechám na vás.
Jedny z parametrů keší jsou jejich **Obtížnost** a **Terén**. Oba parametry jsou hodnoceny od jedné do pěti hvězdiček, s krokem půl hvězdičky. Hodnocení je čistě na autorovi a to co jednomu přijde lehké je pro jiného obtížné, takže jde většinou o hodnocení subjektivní. Snad jen u 5* jde téměř vždy o náročnější provedení a u terénu to pak znamená, že by mělo být k odlovu zapotřebí nějaké vybavení, např. horolezecké.

A teď již slíbené **únorové GeoŠifry**:

Corrida	(http://coord.info/GC1JA1K)
U-864	(http://coord.info/GC1DGWA)
Strom zivota	(http://coord.info/GC19PDQ), zde je obtížnost řešení dána hlavně dalšími částmi, které se člověk dozví až na místě, ale i první část je zajímavou ukázkou co nabízí geocaching.

Na závěr mám na čtenáře prosbu. Mám na svém kontě nejednu obtížně řešitelnou keš, ale před časem jsme s kolegou narazili na jednu, se kterou si opravdu nevím rady, ale mezi čtenáři se najdou jistě schopnější kryptoanalytikové a snad mi pomůžou.

RNFCTEPUOAFTTSI TDICTQRGDFASTTBI
QUFSIFUUHGFITBT IKTSIHPHQFSATIRI
BVLOYLDVRUEIYHT YATJIKFMBLCHYIAT

K šifře není žádná nápověda, jen příběh čerpá z bájí o Kerberovi a u této části se hovoří o řece Styx, zda je to možný klíč, ale nevím.

Výsledkem jsou dvě tříciferná čísla ??? a ???. Vaši pomoc uvítám (nápady zašlete na e-mail v záhlaví).

Přeji úspěšné luštění a šťastný lov.

Martin

D. Matrix - tak trochu jiná šifrovačka

Michal Kesely & Michal Švagerka
Občanské sdružení Velký vůz, matrix@velkyvuz.cz

*Šifrovací hry jsou již delší dobu v České republice fenoménem. Mají tu stálou základnu stovek hráčů, a přesto se pokusy zorganizovat podobnou hru v zahraničí vyskytují jen sporadicky (bratislavská Haluz, Dortmunder Nachtschicht). A když se to podaří, očekávání organizátorů i hráčů jsou velké, protože české hry nastavují laťku velmi vysoko. Konkurence her je veliká, a proto se každá hra snaží něčím odlišit. TMOU si zakládá na tradici, Bedna na náročnosti šifer a internetový Sendvič na přístupnosti odkudkoliv. Co by mělo hráče přesvědčit, aby se zúčastnili právě pátého pokračování šifrovací hry Matrix, které se odehraje **17. -18. 4. 2010**?*

Informace o hře

Matrix je šifrovací závod. To znamená, že hlavní část závodu tvoří luštění šifer. Výsledek šifry většinou budete potřebovat, abyste věděli, kam postupovat dál. Neočekávejte, že z pohledu na šifru budete vědět, co s ní dělat. Řešení šifry je často posloupností několika logických kroků. V šifrách můžete čekat a hledat cokoliv - morseovku, převod abecedy na čísla, posuny v abecedě. Anebo i všechno najednou.

Matrix je noční závod. Trvá 20-22 hodin a převážná část se odehrává v noci. Slábnoucí světlo baterky a lezavá zima ve 3 ráno dává luštění šifer i orientaci v terénu úplně jinou dimenzi.

Matrix je týmový závod. Závodíte v týmech o 3-5 lidech. Dobře sestavený tým je velkou výhodou. Mějte v týmu inteligenta, který exceluje v logickém uvažování, sportovce, který hravě splní jakoukoliv fyzickou aktivitu, chodící knihovnu, která všechno ví, všechno zná a všude byla, znalce umění nebo ještě lépe šikovného hudebníka či kreslíře, někoho zdatného v orientaci a čtení map, schopného organizátora a taktika, který vždy bude vědět, koho je nejlepší kam poslat, někoho, kdo po celou dobu hry bude udržovat vaši morálku a zvedat vám náladu v krizových situacích písněmi proti trudomyslnosti.

Matrix neobsahuje jen šifry. Matrix se může pochlubit i s nešifrovými aktivitami, které otestují vaši sílu, odvahu, schopnost reagovat v nečekaných situacích.

Matrix je nelineární závod. Připravte se na to, že vaše cesta nebude lineární řetězec stanovišť, ale bude se větvit, proplétat a zase spojovat. V mnoha případech bude užitečné se jako tým rozdělit. Na druhou stranu, pokud nějakou šifru nevyluštíte, není vše ztraceno, můžete ji většinou obejít alternativní cestou. V případě největší nouze vám může pomoci i nápověda, budete ovšem muset obětovat váš drahocenný čas a síly na její vyzvednutí.

Matrix je motivován příběhem. Po třech ročnících inspirovaných Matrixem a jednom inspirovaném filmem Kostka přichází ročník inspirovaný československým seriálem Návštěvníci. CML – centrální mozek lidstva – nezná řešení blížící se hrozby, a proto se účastníci hry vrátí z roku 2410 o 400 let nazpět, aby upravili jeho původní nastavení.

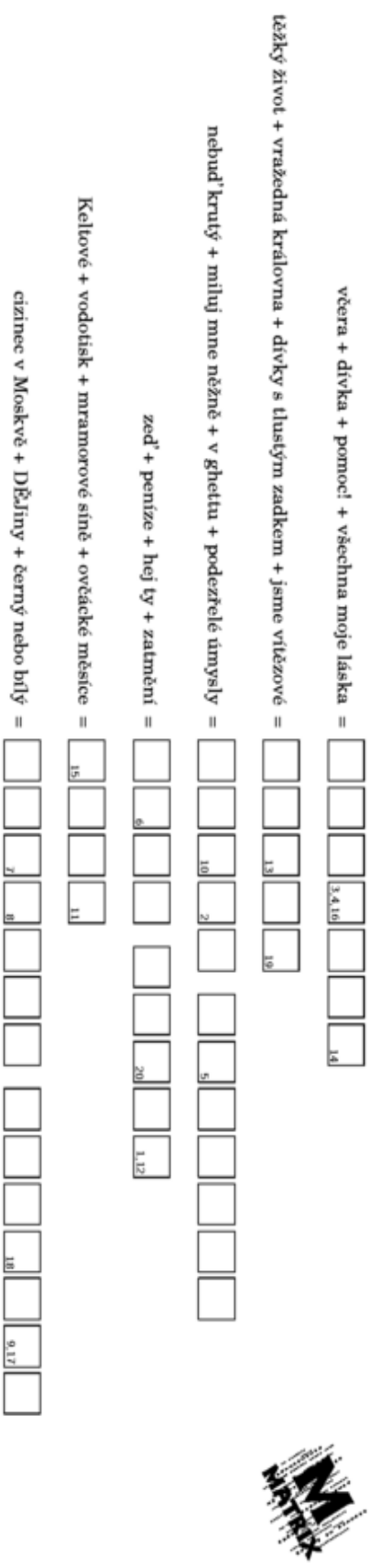
Registrace

Registrace týmů bude zahájena 17. 2. 2010. Pokud jsme ve vás vzbudili nějaký zájem, navštivte naši stránku http://www.velkyvuz.cz/matrix , kde se dozvíte více. Jestliže máte jakékoliv otázky nebo návrhy, zkuste nám napsat e-mail na adresu matrix@velkyvuz.cz .

Těší se na vás

organizátoři šifrovací hry Matrix.

PS: Jestli si chcete zkusit vyluštit šifry z již proběhlých Matrixů, tady jsou dvě. Jejich řešení a další šifry najdete na již zmíněných stránkách.
http://velkyvuz.adam.cz/matrix/2006/sifry/1b2r.php
http://velkyvuz.adam.cz/matrix/2006/sifry/1c2r.php



Byl jednou jeden had, který pořád spal a spal a skrýval tajemství Matrixu. Když jste ho probudili, lekl se a začal utíkat pryč. Tím odhalil svoje tajemství.

N	A	L	O	N
T	K	T	M	A
K	E	5	A	I
O	S	E	R	N
S	S	M	2	O

E. O čem jsme psali v únoru 2000 – 2009

Crypto-World 2/2000

Crypto-World 2/2001

Crypto-World 2/2002

Příloha: Program pro naše čtenáře : "Hašák ver. 0.9" (viz. letem šifrovým světem) hasak.

Crypto-World 2/2003

Příloha : Crypto_p2.pdf
Přehled dokumentů ETSI, které se zabývají elektronickým podpisem
(ETSI - European Telecommunication Standards Institute) 10 stran

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@zoner.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/