Crypto-World

Informační sešit GCUCMP

ISSN 1801-2140

Ročník 13, číslo 3/2011 **16. březen**

3/2011

Připravil: Mgr. Pavel Vondruška

Sešit je přednostně distribuován registrovaným čtenářům.
Starší sešity jsou dostupné na adrese
http://crypto-world.info
(1360 registrovaných odběratelů)

Obsah :

A. Československé šifry z obdobia 2. svetovej vojny Diel 3., Šifra „Rímska osem“

Jozef Kollár, jmkollar@math.sk
KMaDG, SvF STU v Bratislave

Mnohé informácie o československých šifrách z obdobia 2. svetovej vojny mi stále chýbajú. Preto ak niekto viete doplniť, prípadne opraviť mnou uvádzané popisy šifier (TTS, Rímska 2, 8, 9, 10, 13, Eva, Marta, Růžena, Utility a Palacký), alebo máte akékoľvek informácie o ďalších československých šifrách z obdobia 2. svetovej vojny, poteší ma, ak mi o tom pošlete správu.

3 Šifra „Rímska osem“

Šifra „Rímska osem“ bola typu SP, čiže pozostávala zo substitúcie znakov dvojcifernými číslami a následného pričítania periodického hesla. Použitá substitúcia bola homofónna. Túto šifru používali radiostanice Libuše, Božena a Lída výsadkových skupín Silver A, Silver B a Zinc. Popis šifry „Rímska osem“ je uvedený v knihe [2] (str. 117–118) a ešte lepšie v knihe [3] (str. 77–81). V druhej uvedenej knihe je naznačený aj postup lúštenia tejto šifry.

3.1 Všeobecný popis a príklad šifrovania depeší

Postup šifrovania je veľmi jednoduhý. Najskôr sa substituovali znaky za čísla a následne sa k takto získanej číselnej postupnosti pričítalo periodické heslo. Substitučná tabuľka mala rozmer 10x10 a abeceda sa do nej zapisovala dosť komplikovaným spôsobom. Zrejme sa pre rôzne varianty tejto šifry používali rôzne abecedy, pretože informácie z [2] a [3] sa v zostavovaní substitučnej tabuľky rozchádzajú. Kým v príklade z [3] sa používajú písmená a číslice ako pri 45 znakovej českej abecede, používanej pri šifre TTS, a štyri špeciálne znaky (. , / -), v uvedenom poradí, v príklade z [2] sa používa až osem špeciálnych znakov (. : , " - / ? !), v uvedenom poradí. Keďže sa táto šifra používala minimálne v troch variantách (v troch rôznych operáciach s troma rôznymi stanicami), je dosť pravdepodobné, že obe verzie substitučnej abecedy sú správne. V našom príklade použijeme substitučnú tabuľku od pána Hanáka, z [2], obsahujúcu osem špeciálnych znakov.

Vypĺňanie substitučnej tabuľky predstavovalo zrejme najnáročnejšiu časť procesu šifrovania a práve v tejto fáze je pravdepodobnosť chyby najvyššia. Najskôr bolo potrebné 10-znakové heslo. Toto heslo bolo zrejme používateľom

pevne pridelené vopred a menilo sa len občas. V našom príklade ako heslo použijeme slovo CHOBOTNICA. Teraz si ho vyčíslime obvyklým spôsobom, pričom cifra 0 bude posledná v poradí:

C	H	O	B	O	T	N	I	C	A
3	5	8	2	9	0	7	6	4	1

Po vyčíslení hesla ním označíme vodorovne (stĺpce) a zvislo (riadky) tabuľky 10x10. Následne budeme pomerne komplikovaným spôsobom vypĺňať túto tabuľku znakmi. Použijeme cifry 1 až 0, písmená rovnaké ako v 45 znakovej abecede pri šifre TTS a špeciálne znaky (. : , " - / ? !), v uvedenom poradí.

1. Začneme s ciframi. Číslicu 1 zapíšeme do prvého riadku tabuľky pod číslo 1 vodorovného hesla. Číslicu 2 do druhého riadku tabuľky pod číslo 2 vodorovného hesla atď. Nakoniec číslicu 0 do posledného riadku tabuľky pod číslo 0 vodorovného hesla.

2. Pokračujeme s písmenami. Tabuľku budeme vypĺňať po diagonálach zľava doprava a zhora nadol. Pritom políčka, ktoré sú už vyplnené, preskakujeme. Začíname písmenom A na diagonále, ktorá začína pri čísle 1 zvislého hesla. Pokračujeme na diagonále začínajúcej číslom 1 vodorovného hesla. Postupne prejdeme všetky diagonály podľa poradia čísel hesla (2,3,...,9,0), pričom najskôr ideme po diagonále začínajúcej pri zvislom hesle a potom po diagonále začínajúcej pri vodorovnom hesle. Zapíšeme teda celú použitú abecedu od A po Ž a osem špeciálnych znakov . : , " - / ? !. Toto nazývame prvé diagonály a na obrázku sú vyznačené zelenou farbou. Spolu máme vyplnených 49 políčok tabuľky.

3. Pokračovať budeme po diagonálach, ale už len písmenami abecedy. Výnimku tvoria písmená Q a X, namiesto ktorých píšeme znak -. Druhá sada písmen sa nám do tabuľky vojde celá. Toto budeme nazývať druhé diagonály a na obrázku sú vyznačené žltou farbou.

4. Do zvyšných diagonál zapíšeme ešte raz písmená A až R, pričom Q opäť nahradíme znakom -. Toto budeme nazývať tretie diagonály a na obrázku sú vyznačené modrou farbou.

Vyplnenú tabuľku z nášho príkladu máme na strane 4. Pri tejto substitučnej tabuľke sa vlastne jedná o druh homofónnej substitúcie. Žiaľ, neprináša nám to to, kvôli čomu sa vlastne homofónne substitúcie začali používať, čiže zníženie frekvencie častých znakov v texte. V našom prípade má každé písmeno, bez ohľadu na jeho frekvenciu v texte, 2 prípadne 3 rôzne vyjadrenia.

	3	**5**	**8**	**2**	**9**	**0**	**7**	**6**	**4**	**1**
3	M	/	-	F	Ě	O	O	H	V	1
5	X	N	?	2	G	F	P	P	I	W
8	3	Y	O	!	Y	H	G	-	-	J
2	B	Ř	Z	P	A	Z	I	H	4	R
9	C	5	S	Ž	Q	B	Ž	J	I	R
0	K	Č	C	Š	.	R	C	6	K	J
7	K	L	D	Č	T	:	7	Č	A	L
6	Ě	L	8	E	D	U	,	Ř	D	B
4	T	F	M	M	9	E	V	”	S	E
1	A	U	G	N	N	0	Ě	W	-	Š

Tabuľka 1: Zostavovanie substitučnej tabuľky pre „rímsku osem“

Výnimky tvoria písmena `Q` a `X`, ktoré sa v českých a slovenských textoch vyskytujú veľmi zriedkavo a majú len jedno vyjadrenie a ešte znak `-`, ktorý sa používa namiesto medzery a má v tabuľke štyri rôzne vyjadrenia.

Pred samotným šifrovaním sa malo posunúť vodorovné aj zvislé heslo substitučnej tabuľky. Tým dostaneme tzv. „šifrovacie heslo“. Podľa [2] (str. 117) to mal šifrant urobiť pre každú jednu depešu. Tento posun hesiel bolo treba oznámiť príjemcovi v indikačných skupinách, čo si my ukážeme až na konci nášho príkladu. Toto opatrenie bezpečnosť nijako nezvýšilo, ale aspoň to komplikovalo život šifrantom a zvyšovalo pravdepodobnosť chýb.[1] V našom príklade teda zrotujeme heslá tak, že vodorovné heslo bude začínať číslom 8 a zvislé heslo bude začínať číslom 9. Dostaneme substitučnú tabuľku, ktorú máme uvedenú na strane 5. Ak si teraz prepíšeme vyjadrenia jednotlivých znakov z nášho príkladu, dostávame tabuľku:

A	B	C	Č	D	E	Ě	F	G	H	I	J	K	L	M	N	O
28	68	19	30	57	86	24	90	07	76	64	41	13	35	98	02	79
67	46	14	31	53	85	58	82	29	91	03	75	38	52	89	20	94
33	55	48	12	39	50	97	06	74	61	43	15	18	32	80	27	96

P	Q	R	Ř	S	Š	T	U	V	W	X	Y	Z	Ž	-	:	,
60	47	16	51	83	25	88	22	93	05	08	72	69	40	23	36	54
01		65	62	49	10	37	56	84	21		77	66	44	73		
04		45												99		
														71		

[1] Presne podľa hesla: *Skratka bola síce dlhšia, ale o to horšia bola cesta.*

	8	**2**	**9**	**0**	**7**	**6**	**4**	**1**	**3**	**5**
9	M	/	-	F	Ě	O	O	H	V	1
0	X	N	?	2	G	F	P	P	I	W
7	3	Y	O	!	Y	H	G	-	-	J
6	B	Ř	Z	P	A	Z	I	H	4	R
4	C	5	S	Ž	Q	B	Ž	J	I	R
1	K	Č	C	Š	.	R	C	6	K	J
3	K	L	D	Č	T	:	7	Č	A	L
5	Ě	L	8	E	D	U	,	Ř	D	B
8	T	F	M	M	9	E	V	”	S	E
2	A	U	G	N	N	0	Ě	W	-	Š

Tabuľka 2: Substitučná tabuľka pre „rímsku osem“ s posunutými heslami

"	.	/	?	!	1	2	3	4	5	6	7	8	9	0		
81	17	92	09	70	95	00	78	63	42	11	34	59	87	26		

Takže substitučnú tabuľku už máme pripravenú a môžme sa pustiť do šifrovania textu. Ako príklad budeme šifrovať Ovidiov výrok:

> *Co je nad kámen tvrdší, co nad vodu měkčí být může?*
> *Tvrdý však kámen voda měkká vyhloubí přec.*
>
> *Ovidius*[2]

Tento text je pomerne krátky, takže ho nebudeme rozdeľovať a zašifrujeme ho ako jednu depešu. V praxi sa dlhšie texty rozdeľovali, čo vyplýva z poznámky v [3]. V dostupných zdrojoch popis rozdeľovania textu nie je uvedený, ale dá sa predpokladať, že pre rozdeľovanie textu platili rovnaké pravidlá ako pri šifrách TTS a STT. My však nič rozdeľovať nebudeme, takže si len celý text prepíšeme pomocou znakov substitučnej tabuľky, medzery nahradíme znakom -, špeciálne znaky nenachádzajúce sa v tabuľke vynecháme a medzery za špeciálnymi znakmi zrušíme. Dostaneme upravený text:

```
CO-JE-NAD-KAMEN-TVRDŠI,CO-NAD-VODU-MĚKČI-BYT-MUŽE?
TVRDY-VŠAK-KAMEN-VODA-MĚKKA-VYHLOUBI-PŘEC.OVIDIUS
```

K textom depeší sa, podľa [3] (str. 81), na začiatok a koniec pridávali adresovacie a podpisové znaky. Boli to skupiny písmen, ktoré sa podľa všetkého

[2]Pôvodná verzia v latinčine: *Quid magis est saxo durum, quid mollius unda? Dura tamen molli saxa cavantur aqua. Ovidius (Ars.I,475)*

odvádzali zo šifrovacích hesiel. Na začiatku prvej depeše seriálu boli uvedené adresovacie znaky príjemcu a v poslednej depeši seriálu boli na konci uvedené podpisové znaky odosielateľa. Naše heslo je `CHOBOTNICA`, takže ako podpisové znaky si zvolíme `ICA`. Ako adresovacie znaky príjemcu si zvolíme `TER`, čo bol, podľa [3], jeden z adresovacích znakov ústredia v Londýne. Takže naša depeša v textovej podobe bude:

```
TER.CO-JE-NAD-KAMEN-TVRDŠI,CO-NAD-VODU-MĚKČI-BYT-MUŽE?
TVRDY-VŠAK-KAMEN-VODA-MĚKKA-VYHLOUBI-PŘEC.OVIDIUS.ICA
```

Podľa substitučnej tabuľky si teraz prepíšeme našu depešu do číselnej podoby a dostávame číselnú podobu nášho textu:

```
88861 61719 79234 18573 02285 79913 67985 02071 37936 55325
64541 49423 27333 97384 96572 29989 24383 00371 68728 82380
56408 60937 93455 37773 84102 81899 13679 88502 71937 93933
23895 83818 28738 47276 35942 24643 99606 25048 17969 36457
03568 31743 1967
```

Všimnime si, že posledná skupina cifier je neúplná (celkový počet cifier nie je násobok 5). Chýbajúce cifry môžme doplniť náhodne. Pri dešifrovaní nanajvýš dostaneme jeden, maximálne dva náhodné znaky. Tieto ale budú figurovať až za podpisom, prípadne znakmi určujúcimi nadväznosť dielov, takže bude zrejmé, že ide o náhodné znaky a môžme ich vynechať. V našom príklade doplníme na koniec cifru 9.

K textu v číselnej podobe ešte musíme pripočítať periodické heslo. Za toto heslo sa bralo (posunuté) 10 ciferné vodorovné heslo zo substitučnej tabuľky 2. V našom príklade bude toto heslo: `82907 64135`. Pripočítavanie sa vykonávalo jednotkovo, bez prenosu desiatok. Inak povedané, text s heslom sa sčítava modulo 10. V našom príklade bude pričítanie periodického hesla k textu depeše vyzerať nasledovne:

```
Text:   88861 61719 79234 18573 02285 79913 67985 02071 37936
Heslo:  82907 64135 82907 64135 82907 64135 82907 64135 82907
-----------------------------------------------------------
Depeša: 60768 25844 51131 72608 84182 33048 49882 66106 19833


Text:   55325 64541 49423 27333 97384 96572 29989 24383 00371
Heslo:  64135 82907 64135 82907 64135 82907 64135 82907 64135
-----------------------------------------------------------
Depeša: 19450 46448 03558 09230 51419 78479 83014 06280 64406
```

```
Text:   68728 82380 56408 60937 93455 37773 84102 81899 13679
Heslo:  82907 64135 82907 64135 82907 64135 82907 64135 82907
-----------------------------------------------------------
Depeša: 40625 46415 38305 24062 75352 91808 66009 45924 95476


Text:   88502 71937 93933 23895 83818 28738 47276 35942 24643
Heslo:  64135 82907 64135 82907 64135 82907 64135 82907 64135
-----------------------------------------------------------
Depeša: 42637 53834 57068 05792 47943 00635 01301 17849 88778


Text:   99606 25048 17969 36457 03568 31743 19679
Heslo:  82907 64135 82907 64135 82907 64135 82907
-----------------------------------------------
Depeša: 71503 89173 99866 90582 85465 95878 91576
```

Ako sme už skôr spomenuli, šifrant musel k správe pridať indikačné skupiny, v ktorých príjemcovi oznámil, ako sú posunuté zvislé a vodorovné heslá. Na začiatok správy sa pridávala jedna päťciferná indikačná skupina a na koniec správy jedna päťciferná kontrolná skupina. Tvorba kontrolnej skupiny závisela aj na dni šifrovania. Predpokladajme preto, že našu správu šifrujeme 18. deň v mesiaci. Prvá cifra indikačnej skupiny je prvá cifra zvislého hesla, druhá cifra indikačnej skupiny je prvá cifra vodorovného hesla a ďalšie tri cifry sú ľubovoľné. V našom príklade bude mať indikačná skupina tvar `98xxx` a môže ňou byť napríklad pätica `98634`. Ako prvé dve cifry kontrolnej skupiny sa berú prvé dve cifry indikačnej skupiny a k obom sa pripočíta jednotková cifra dňa šifrovania modulo 10. Ďalšie tri cifry kontrolnej skupiny sa doplnili ľubovoľne. V našom príklade šifrujeme 18. dňa v mesiaci, čiže k cifrám 9 a 8 pripočítame 8 modulo 10. Naša kontrolná skupina bude mať preto tvar `76xxx` (pretože $(9+8)mod10 = 7$ a $(8+8)mod10 = 6$) a môže ňou byť napríklad pätica `76102`. Napokon už len na začiatok pridáme návestie depeše v tvare `xxx-yyy-zz`, kde `xxx` je poradové číslo depeše, `yyy` je počet cifier depeše a `zz` je deň šifrovania depeše. V našom príklade dostaneme, na odoslanie pripravenú, depešu v tvare:

```
029-225-18
98634 60768 25844 51131 72608 84182 33048 49882 66106 19833
19450 46448 03558 09230 51419 78479 83014 06280 64406 40625
46415 38305 24062 75352 91808 66009 45924 95476 42637 53834
57068 05792 47943 00635 01301 17849 88778 71503 89173 99866
90582 85465 95878 91576 76102
```

3.2 Postup pri šifrovaní

V tejto časti budeme vychádzať z nasledovných predpokladov:

a. Máme k dispozícii text na šifrovanie.

b. Je dané 10 znakové šifrovacie heslo.

c. Daný je deň šifrovania.

d. Sú dané adresovacie a podpisové znaky príjemcu a odosieľateľa depeše.

Potom šifrovanie depeše bude prebiehať podľa nasledovných krokov:

1. Šifrovacie heslo vyčíslime bežným spôsobom a vyčísleným heslom označíme riadky a stĺpce tabuľky 10x10.

2. Zostavíme substitučnú tabuľku tak, ako to bolo popísané v predošlom texte:

   (a) Do tabuľky najskôr zapíšeme cifry 1 až 0 (nula bude posledná, najvyššia, cifra). Cifru i zapíšeme do i-teho riadku a stĺpca označeného číslom i vyčísleného hesla.

   (b) Ďalšie znaky budeme do tabuľky zapisovať po diagonálach zhora nadol a zľava doprava. Budeme postupovať podľa cifier vyčísleného hesla v poradí 1, 2, 3, 4, 5, 6, 7, 8, 9, 0 a vždy najskôr vypĺňame diagonálu začínajúcu pri cifre i zvislého hesla a potom diagonálu začínajúcu pri cifre i vodorovného hesla. Políčka, ktoré už sú vyplnené preskakujeme.

   (c) Najprv do tabuľky zapíšeme 31 písmen, `A, B, C, Č, D, E, Ě, F, G, H, I, J, K, L, M, N, O, P, Q, R, Ř, S, Š, T, U, V, W, X, Y, Z, Ž`, v uvedenom poradí.

   (d) Potom do tabuľky zapíšeme znaky `. : , " - / ? !`, v uvedenom poradí.

   (e) Opäť do tabuľky zapíšeme písmena z bodu 2c, avšak namiesto písmen `Q` a `X` píšeme znak `-`.

   (f) Nakoniec do tabuľky zapíšeme písmená `A` až `R`, zo zoznamu v bode 2c, ale namiesto písmena `Q` píšeme znak `-`.

   (g) Zvolíme si dve cifry zo zoznamu 1 až 0 a posunieme zvislé a vodorovné heslá substitučnej tabuľky tak, aby sa začínali týmito ciframi. Tento posun číselných hesiel by sa mal, podľa [2], robiť pre každú depešu, resp. pre každú časť seriálu.

3. Text, ktorý ideme šifrovať, prepíšeme len pomocou znakov obsiahnutých v substitučnej tabuľke, čiže nahradíme písmená a vynecháme špeciálne znaky, ktoré sa v substitučnej tabuľke nevyskytujú.

4. Medzery medzi slovami nahradíme pomlčkou. Pokiaľ sa medzi niektorými slovami textu nachádza niektorý zo špeciálnych znakov obsiahnutých v substitučnej tabuľke, tak sa za týmto znakom medzera vynecháva.

5. Text rozdelíme na časti približne 100 znakov dlhé tak, aby každá časť vždy končila kompletným slovom.

6. Na koniec prvej časti pridáme, kvôli nadväznosti dielov `/A`. Na začiatok druhej časti pridáme `A/`, na koniec druhej časti pridáme `/B` atď. Každá časť textu (okrem prvej a poslednej) bude mať na začiatku písmeno identické s koncovým písmenom predošlej časti a znak `/` a na konci textu znak `/` a písmeno identické s písmenom označujúcim nasledovnú časť textu. Písmena na označovanie častí berieme podľa abecedy. Prvá časť má označenie len na konci a posledná časť len na začiatku.

7. Na začiatok prvej časti seriálu pridáme adresovacie znaky príjemcu a od textu ich oddelíme bodkou. Podobne na koniec poslednej časti seriálu pridáme bodku a podpisové znaky odosieľateľa.

8. Každú časť textu šifrujeme zvlášť a každá časť textu tvorí samostatnú depešu s vlastným návestím.

9. Všetky znaky nahradíme číslami podľa substitučnej tabuľky. Podľa informácii z [2] na strane 117 by sa malo vodorovné aj zvislé heslo substitučnej tabuľky posúvať pre každú depešu a dokonca aj pre každú časť seriálu. Pri ručnom šifrovaní by to bolo extrémne náročné a spôsobovalo by to veľké množstvo chýb.

10. Potom skontrolujeme počet cifier jednotlivých častí. Počet cifier každej časti musí byť násobok 5. Pokiaľ nie je, tak náhodne doplníme potrebný počet cifier.

Nasledujúce body popisujú spracovávanie jednej časti depeše:

11. Postupnosť čísel, ktorú sme dostali rozdelíme na skupiny po 5 cifier.

12. Pod cifry textu si opakovane zapíšeme vodorovné heslo zo substitučnej tabuľky použitej pre príslušnú časť textu a sčítame modulo 10, t.j. jednotkové sčítanie bez prenosu desiatok. Takto dostaneme prešifrovaný text.

13. Vytvoríme indikačnú skupinu tak, že zoberieme prvú cifru zvislého a vodorovného hesla v uvedenom poradí a zvyšné tri cifry doplníme náhodne. Túto indikačnú skupinu pridáme na začiatok prešifrovaného textu.

14. Vytvoríme kontrolnú skupinu tak, že vezmeme prvé dve cifry indikačnej skupiny a jednotkovo, bez prenosu desiatok, k nim pripočítame jednotkovú cifru dátumu dňa šifrovania. Ďalšie tri cifry zvolíme náhodne. Túto kontrolnú skupinu pridáme na koniec prešifrovaného textu.

15. Na začiatok depeše pridáme ešte návestie v tvare `xxx-yyy-zz`, kde `xxx` je poradové číslo depeše, `yyy` je počet cifier depeše (aj s náhodne pridanými ciframi, t.j. toto číslo musí byť násobkom 5) a `zz` je deň šifrovania depeše. Týmto je šifrovanie depeše ukončené a depeša je pripravená na odoslanie.

3.3 Postup pri dešifrovaní

V tejto časti budeme vychádzať z nasledovných predpokladov:

a. Máme k dispozícii kompletný text zašifrovanej depeše aj s návestím.

b. Je dané 10 znakové šifrovacie heslo.

Potom dešifrovanie depeše bude prebiehať v nasledovných krokoch:

1. Na základe návestia si overíme kompletnosť depeše (počet cifier).

2. Vynecháme návestie depeše, ktoré už nebudeme potrebovať.

3. Podľa prvej (indikačnej) a poslednej (kontrolnej) päťčlennej skupiny depeše určíme prvé cifry posunutého zvislého a vodorovného hesla substitučnej tabuľky. Potom prvú a poslednú skupinu vynecháme.

4. Obvyklým spôsobom vyčíslime heslo a pomocou neho zostavíme substitučnú tabuľku tak, ako je to popísané v bode 2 postupu šifrovania.

5. Zvislé a vodorovné heslá substitučnej tabuľky posunieme tak, aby sa začínali ciframi, ktoré sme určili v bode 3.

6. Pod cifry depeše opakovane zapíšeme vodorovné heslo zo substitučnej tabuľky a spravíme rozdiel modulo 10, t.j. jednotkové odčítanie bez prenosu desiatok. Dostaneme otvorený text depeše v číselnom tvare.

7. Podľa zostavenej substitučnej tabuľky zameníme čísla za znaky. Pritom sa môže stať, že na konci depeše sa objavia 1 alebo 2 náhodné znaky. Tieto prípadné náhodné znaky ale budú až za znakmi označujúcimi nadväznosť, prípadne za podpisovými znakmi, takže ich ľahko spoznáme a môžme vynechať.

8. Pomlčky nahradíme medzerami a rovnako doplníme medzery za špeciálne znaky v texte. Týmto sme dostali pôvodný text depeše.

9. Pokiaľ sa jedná o seriál, tak text zostavíme v správnom poradí podľa označenia na začiatku a konci jednotlivých častí seriálu.

3.4 Lúštenie

Šifra „Rímska osem“ bola šifrou typu SP. Použitá substitúcia síce bola homofónna, ale stále to bola len substitúcia a ako taká odhaliteľná a lúštiteľná. Naviac spôsob, akým bola robená substitučná tabuľka, bol veľmi zle zvolený. Preto výsledná substitúcia nemala ani len tie výhody, ktoré by správne urobená homofónna substitúcia mať mohla. Okrem toho, ako už bolo v texte spomenuté, substitučná tabuľka sa pripravovala veľmi komplikovaným spôsobom. To nijako nezvyšovalo bezpečnosť šifry, ale zvyšovalo to pravdepodobnosť chýb a následne nutnosť opakovaného posielania tých istých depeší. Lúštenie podobných substitúcií zvládali kryptoanalytici bez problémov už v časoch Rossignolovcov, t.j. pár storočí pred 2. svetovou vojnou. Na analýzu bolo potrebné len dostatočné množstvo zašifrovaného textu, a to nemeckí lúštitelia mali.

Druhá časť šifry bola robená pomocou pričítania periodického hesla. Pri dostatočnom množstve zašifrovaného textu sa dĺžka použitého periodického hesla dá odhaliť pomocou Kasiského metódy, ktorá bola v literatúre popísaná už v druhej polovici 19. storočia. V čase 2. svetovej vojny už boli známe aj sofistikovanejšie metódy na zistenie periodického hesla, ale aj uvedená Kasiského metóda bola postačujúca. V praxi na lúštenie stačila jediná dlhšia depeša. Po odhade dĺžky hesla Kasiského metódou sa použije útok pomocou predpokladaného textu. Metóda lúštenia takýchto šifier s periodickým heslom bola zverejnená už v roku 1925 v štúdii Marcela Giviergeho: *Cours de Cryptographie*[3] a nemeckým lúštiteľom bola známa najneskôr od 30. rokov XX. storočia. Navyše sa ako periodické heslo používali vodorovné súradnice substitučnej tabuľky. Preto ak lúštiteľom už bol známy princíp použitej šifry, napr. z už skôr rozlúštených depeší, tak po zistení periodického hesla už mali, takmer zadarmo, aj substitučnú tabuľku ako bonus.

[3]Zdroj: [3], strana 80.

Postup lúštenia naznačil v knihe [3] a v knihe [4] (str. 119–122) pán Janeček. Dá sa predpokladať, že v praxi bolo lúštenie šifry „Rímska osem“ zrejme oveľa menej náročné ako napr. lúštenie šifry STT.

Literatúra

[1] Grošek Otokar, Vojvoda Milan, Zajac Pavol: Klasické šifry
STU v Bratislave, 2007

[2] Hanák Vítězslav: Muži a radiostanice tajné války
Elli Print, 2002

[3] Janeček Jiří: Gentlemani (ne)čtou cizí dopisy
Books Bonus A, 1998

[4] Janeček Jiří: Odhalená tajemství šifrovacích klíčů minulosti
Naše vojsko, 1994

[5] Janeček Jiří: Válka šifer – výhry a prohry československé vojenské rozvědky (1939–1945)
Votobia, 2001

Obr. 1: plk. František Moravec. Počas vojny pôsobil v londýnskej exilovej vláde na ministerstve národnej obrany ako náčelník II. (spravodajského) odboru. Bol to človek zodpovedný za spôsob akým sa používali jednotlivé šifry. Zdroj: `http://cs.wikipedia.org/wiki/František_Moravec_(generál)`

B. Blinkry blikají aneb komentář ke zprávě NIST o výběru finalistů SHA-3

Vlastimil Klíma, kryptolog, KNZ, s.r.o., Praha
(http://cryptography.hyperlink.cz, vlastimil.klima@knzsro.cz)

Když soutěž začínala, NIST ve velkém stylu nejprve podrobil veřejné kritice návrhová kritéria [5]. Z toho vznikla upřesněná návrhová kritéria [6], která byla vyhlášena a závazná pro soutěžící. Základním požadavkem bylo [6]:

"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. NIST also desires that the SHA–3 hash functions will be designed so that a possibly successful attack on the SHA–2 hash functions is unlikely to be applicable to SHA–3."

Na základě těchto kritérií ([6]) soutěž začala. Mohl někdo v té době vědět, že může tato kritéria porušit? A bylo by správné, aby takový algoritmus postoupil? A přesto, pokud se podíváme na pět dnešních finalistů, pouze jeden splňuje počáteční požadavky. Jak je to možné? Je to jednoduché, týmy, které pracovaly na návrzích prostě musely vyřešit na první pohled nesmyslný požadavek - vyšší bezpečnost a vyšší rychlost - po svém. Neměly žádné informace o soupeřích, a tak hledaly to nejlepší, co mohly nabídnout, v první řadě bezpečnost. A v první řadě rychlost. Mnohé z týmů takové řešení prostě nenašly, a tak nabídly alespoň to, co našly. Neuspěl ani tým Bruce Schneiera se Skeinem. Neuspěly mnohé jiné týmy. Některé týmy braly požadavek NISTu tak, jak to bylo logické a v soutěžích obvyklé, že požadavky soutěže jsou něčím, co se MUSÍ dodržet, jinak nesplní základní podmínku soutěže a budou z ní vyřazeny. A tak šly "na hranu" a musely přijít s něčím novým, neprověřeným. Když přeskočíme období dvou roků analýz, NIST právě tyto algoritmy vyřadil kvůli svému špatnému pocitu. Naproti tomu se do finále dostaly tři algoritmy, které výše uvedené kritérium nesplňují velmi markantně.

Jak známo, NIST 9. prosince 2010 oznámil pět finalistů soutěže SHA-3 [1]. Se značným zpožděním poté publikoval zprávu o tom, proč a jak vybíral těchto pět finalistů ze 14 kandidátů [2]. V tomto článku budeme toto rozhodnutí NISTu komentovat. Není možné skrýt, že autor článku je spoluautorem BMW, nejrychlejšího z kandidátů, který ale do finále nepostoupil. Není také možné, aby se autor oprostil od tohoto faktu, i kdyby nakrásně chtěl. Jsou proto dvě možnosti, snažit se to potlačit a být objektivní nebo se o to nesnažit. Pokusím se jít první cestou a ponechávám na čtenářích Crypto-Worldu, aby případné neobjektivnosti sami odfiltrovali.

NIST ve zprávě [2] probírá jeden algoritmus za druhým. U každého uvádí jeho základní stavební prvky, bezpečnost a výkon. Ve výkonu je zahrnuta i rychlost i náročnost realizace v čipech a omezených prostředích. Na závěr je vždy uveden důvod, proč algoritmus byl nebo nebyl vybrán.

NIST neměl žádnou formální povinnost cokoli vysvětlovat, ale přesto důvody výběru nebo odmítnutí uvedl. Člověk by se rád z těchto vysvětlení něco dozvěděl, třeba proto, aby se poučil, co si NIST myslí o stavebních prvcích těch nebo oněch. Ve zdůvodněních ovšem to, co hledáme, nenalezneme. Právě tam, kde by zpráva NIST mohla být přínosem, je zdůvodnění skryto v zaklínací formulce, že něco nevzbuzovalo důvěru nebo že NIST měl pocit nebo kryptografové NIST měli pocit. Bohužel, z těchto pocitů NISTu se světová kryptografie těžko může poučit.

V tabulce, která pochází z domácí stránky Skeinu [4], jsou seřazeni kandidáti podle svého výkonu v softwarové realizaci. Není to všezahrnující, ale dostatečné pro orientaci ve schopnostech algoritmů. Zároveň uvádíme v originále hlavní důvod, proč NIST algoritmus vyřadil. Pod pomyslnou čarou jsou algoritmy, které nesplňují základní kritérium NISTu - vyšší rychlost a vyšší bezpečnost než SHA-2. To lze snadno prokázat právě uvedenými čísly a NIST to ani neskrývá, jak uvidíme dále.

Algoritmus	64bit. proc. hash 256 bitů / hash 512 bitů	32bit. proc. hash 256 bitů / hash 512 bitů	**Důvod vyřazení**
SHA-1	10/10	10/10	
BMW	7/3	7/12	the attacks on the algorithm, even after an extensive tweak, **did not provide confidence** in the security of the algorithm
Shabal	8/8	10/10	... raised concerns among NIST's cryptographers about **the possibility of more powerful attacks in the future**
BLAKE	8/9	9/12	
SIMD	11/12	12/13	SIMD was not selected as a finalist, due to its large area requirements and **the existence of the symmetric states**
Skein	7/6	21/20	
CubeHash	13/13	13/13	NIST felt that **an additional year of study would not be enough to determine whether or not the symmetric properties pose a threat**.
SHA-2	20/13	20/40	
JH	16/16	21/21	
Luffa	13/23	13/25	the security margin of the compression function is quite small, and full distinguishers on the sub-permutations have also been discovered
Hamsi	25/25	36/36	second-preimage attacks

Grøstl	22/30	23/36	
SHAvite-3	26/38	35/55	... the lack of security in the key schedule of the underlying block cipher, leading to a relatively low security margin for the 512-bit version of the hash function. In addition, SHAvite-3 has a relatively low throughput-to-area ratio.
Keccak	10/20	31/62	
Echo	28/53	32/61	Although ECHO appears to be a simple secure design, it was not selected as a finalist, due to its all-around poor performance.
Fugue	28/56	36/72	Fugue is an innovative design and has decent, all-around performance. ...NIST felt that it would not be possible to establish confidence in the hash algorithm after another year of cryptanalysis; therefore, it was not selected as a finalist.

Vidíme, že z prvních šesti míst byly čtyři algoritmy vyřazeny z důvodu stísněných pocitů NIST ohledně jejich bezpečnosti a místo nich byli do finále zařazeni tři outsideři, u nichž NIST špatné pocity ohledně bezpečnosti neměl. Jedná se o algoritmy JH, Keccak a Grostl.

Přitom algoritmus JH je teoreticky prolomený (nemá plnou odolnost proti útoku nalezením druhého vzoru) jak podle vlastních kritérií NIST, tak podle stránky ecrypt [3]. NISTu to ovšem nevadí, neboť přímo uvádí, že žádná z analýz JH nevzbuzuje žádné bezpečnostní obavy. To je pravda, ve mě také nevzbuzuje bezpečnostní obavy útok se složitostí 2**507 místo požadovaných 2**512. Jenže NIST nestanovil v požadavcích soutěže, že by něco mělo nebo nemělo vzbuzovat jeho obavy, ale stanovil návrhová kritéria a pravidla, a ta to tedy porušuje.

Algoritmy Keccak a Grostl jsou zase vůči dvěma zbývajícím kandidátům BLAKE a Skein natolik pomalé, že s nimi nemohou soupeřit.

Zbývá otázka, proč Keccak, Grostl a JH byli do výběru finalistů vůbec zařazeni. Stačí se podívat na důvody NISTu. Když vyřadíme ty algoritmy, u nichž má NIST stísněné pocity, nikdo jiný už na výběr není a zbývá pouze uvedených pět, žádné jiné už nejsou. Je tedy skutečně možné potvrdit, že bezpečnost byla při rozhodování NIST na prvním místě a zbylo jen pět algoritmů, o nichž se NIST vůbec chce dále bavit.

Závěrem uvedu (vytržený, ale přesto) citát ze zprávy NIST [2] (str. 5): "... However, during the analysis of the second round candidates, it became apparent that significant improvement in efficiency while fulfilling the security requirements was not easily attainable."

Teď bych poprosil čtenáře, aby mi věřili, že v tom není nic osobního. Pokud by NIST vybral BMW do finále, soutěžil by BLAKE, Skein a BMW o vítěze. BMW mohlo být podrobeno ještě kritice, a kdyby se nic nenašlo, a zvítězil by, mohlo být dosaženo cíle NIST - nový standard by byl podstatně rychlejší než SHA-2. Teď je to dosažitelné o něco hůře pouze pomocí BLAKE. Skein bohužel nenabídne takovou rychlost, ovšem je konzervativní. V tom je také jádro této maličko zmařené soutěže - NIST chtěl novou technologii, ale cokoli se nového objevilo, to označil za dosud nepříliš probádané nebo měl z toho stísněné pocity bezpečnosti (SIMD, Shabal, BMW). Pochopitelně, osobně bych měl také stísněné pocity bezpečnosti u

nových věcí. Jenže bez toho se nové rychlosti dosáhnout nedá. Současná "hašovací technologie" je prostě na hraně možností a Skein dosahuje mírného zvýšení rychlosti pouze a jen u 64-bitových procesorů. Jenže NIST na počátku deklaroval, že chce SHA-3 rychlejší než SHA-2 i pro 32-bitové i pro 64-bitové procesory. Od toho teď de facto upustil a vybraní finalisté přešvihávají rychlostní (nebo spíše pomalostní) charakteristiky jako na běžícím pásu (kromě BLAKE). To, že to není fair, "... se hlásit nemusí, to přece vidíme...", notabene, když NIST sám ve výše uvedeném citátu přiznal, že "...blinkry blikají...".

Rádi bychom také diskutovali o důvodech vyřazení BMW, SIMD a Shabal, ale bylo by to zbytečné.

Pokud se však podíváme jen na detailní návrh algoritmu Shabal a jeho rychlost a kdo za ním stojí (francouzský tým 14 lidí - DCSSI, EADS, Fr. Telecom, Gemalto, INRIA, Cryptolog, Sagem), je poněkud smutné, když je takový algoritmus vyřazen z důvodu, že "...by se v budoucnu mohly objevit nějaké útoky...".

Zpráva NISTu není vědecká. Nepřináší žádná poučení a nad jejími závěry zůstávají velké otazníky. NIST však vykonal v posledních 40 letech tolik práce a má tolik zásluh o rozvoj aplikované i teoretické kryptologie, že je to velmi ohromující a velmi záslužné. Poslední rozhodnutí NIST můžeme proto brát jako výjimku, která potvrzuje pravidlo.

Nakonec je tu ještě možný relativně velmi dobrý konec, pokud bude vybrán BLAKE. Ovšem, to záleží na NISTu. Ale že bych věřil na objektivnost poté, co NIST předvedl, to bohužel už nemohu říci.

[1] http://csrc.nist.gov/groups/ST/hash/sha-3/Round3/submissions_rnd3.html

[2]http://csrc.nist.gov/groups/ST/hash/sha-3/Round2/documents/Round2_Report_NISTIR_7764.pdf

[3] http://ehash.iaik.tugraz.at/wiki/The_SHA-3_Zoo

[4] http://skein-hash.info/sha3-engineering

[5] Announcing the Development of New Hash Algorithm(s) for the Revision of Federal Information Processing Standard (FIPS) 180–2, Secure Hash Standard, Federal Register / Vol. 72, No. 14 / Tuesday, January 23, 2007 / Notices 2861, http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Jan07.pdf

[6] Announcing Request for Candidate Algorithm Nominations for a New Cryptographic Hash Algorithm (SHA–3) Family, NIST, Federal Register / Vol. 72, No. 212 / Friday, November 2, 2007 / Notices, http://csrc.nist.gov/groups/ST/hash/documents/FR_Notice_Nov07.pdf

C. Charakteristiky Booleovských funkcí 8 proměnných

RNDr. Petr Tesař, Ph.D. , Katedra informatiky, VŠFS v Praze, 22901@mail.vsfs.cz

Každý moderní kryptografický algoritmus je složen z elementárních stavebních prvků, které jsou vhodně propojeny.
Jedním z hlavních stavebních prvků jsou Booleovské funkce. Booleovské funkce jsou obecně libovolná zobrazení vstupních *N*-rozměrným binárních vektorů do prostoru *B*, kde *B = {0,1}* je Booleovská 1-dimenzionální množina.

V první části uvedeme několik standardních definic z oblasti popisu kryptografické kvality Booleovských funkcí. Ve druhé části jsou experimentálně získané hodnoty statistik těchto kritérií pro Booleovské funkce s $N = 8$.

Definice 1 – Hammingova váha

Hammingova váha $w_t(f)$ Booleovské funkce *f* s jedním výstupem, je počet jedniček v množině výstupních hodnot. Hammingova vzdálenost dvou Booleovských funkcí *f* a *g* je definována jako Hammingova váha $w_t(f \oplus g)$, kde $\oplus$ je logická funkce XOR.

Definice 2 – Vybalancovaná Booleovská funkce

Booleovskou funkci *f* o *N* proměnných nazveme *vybalancovanou*, pokud splňuje podmínku

(1) $$w_t(f) = 2^{N-1}$$

Definice 3 – Lineární Booleovská funkce

Lineární Booleovskou funkci příslušnou vektoru $\alpha \in B^N$ označíme

(2) $$L_\alpha(x) = \alpha_1 x_1 \oplus \alpha_2 x_2 \oplus \ldots \oplus \alpha_N x_N$$

Kde $\alpha_i\, x_i$ značí logickou funkci AND i-tého bitu vektorů α a *x*.

Definice 4 – Afinní Booleovská funkce

Množina afinních Booleovských funkcí je složena z množiny lineárních Booleovských funkcí a jejich komplementů

(3) $$A_{\alpha,c}(x) = L_\alpha(x) \oplus c$$

kde $c \in B$.

Definice 5 – Walsh-Hadamard transformace

Pro Booleovskou funkci *f*, definujeme Walsh-Hadamard transformaci $\hat{F}_f$ výrazem

(4) $$\hat{F}_f(\alpha) = \sum_{x \in B^N} \hat{f}(x)\hat{L}_\alpha(x)$$

kde $\hat{f}(x) = (-1)^{f(x)}$ je tzv. polaritní reprezentace funkce *f*.

Označíme $WHT_{max}(f)$ maximum absolutní hodnoty Walsh-Hadamard transformace dané výrazem

(5) $$WHT_{\max}(f) = \max_{\alpha \in B^N} \left| \hat{F}_f(\alpha) \right|$$

Definice 6 – Nelinearita Booleovské funkce

Míru nelinearity Booleovské funkce f (na bitové úrovni) vyjadřujeme hodnotou N_f, která je definována jako minimum Hammingovy vzdálenosti w_t mezi danou funkcí f a prostorem afinních funkcí.

(6) $$N_f = \min_{d \in A_{\alpha,c}} (w_t(f \oplus d))$$

Toto kritérium, v této verzi, bylo poprvé popsáno v [1]. Pro výpočet nelinearity funkce f využíváme vztah

(7) $$N_f = \frac{1}{2}(2^N - WHT_{\max}(f))$$

Platí vzorec známý jako Parsevalova věta:

(8) $$\sum_{\alpha \in B^N} (\hat{F}_f(\alpha))^2 = 2^{2N}$$

Z Parsevalovy věty přímo plyne

(9) $$WHT_{\max}(f) \geq 2^{\frac{N}{2}}$$

Dosazením do (7) dostáváme pro sudá N horní mez pro nelinearitu Booleovské funkce f:

(10) $$N_f \leq 2^{N-1} - 2^{N/2-1}$$

Booleovské funkce s maximální nelinearitou (např. pro $N=8$ je to 120) se nazývají *bent funkcemi*. Bent funkce není nikdy vybalancovaná. Dosažitelná maximální nelinearita u vybalancovaných Booleovských funkcí není obecně známa.

Definice 7 – Autokorelační transformace

Autokorelační transformaci Booleovské funkce f definujeme výrazem:

(11) $$\hat{r}_f(\alpha) = \sum_{x \in B^N} \hat{f}(x) \hat{f}(x \oplus \alpha)$$

Autokorelací funkce f budeme označovat hodnotu AC_f , kde

(12) $$AC_f = \max_{\alpha \in B^N} \left| \sum_{x \in B^N} \hat{f}(x) \hat{f}(x \oplus \alpha) \right|$$

Kritérium je popsáno např. v [2].

Definice 8 – Booleovská diference

Nechť je dána Booleovská funkce *f*. Potom výraz

(13) $$\frac{df}{dx_i} = f(x_1,...,x_i = 0,...,x_N) \oplus f(x_1,...,x_i = 1,...,x_N)$$

nazveme Booleovskou diferencí funkce *f* podle proměnné x_i .
Definice je převzata z [3].

Je zřejmé, že Booleovskou diferencí funkce *f* podle proměnné x_i je Booleovskou funkcí *N-1* proměnných, a to $x_1,...,x_{i-1},x_{i+1},...,x_N$. Jestliže Booleovská funkce *f* nezávisí na proměnné x_i platí, že $\frac{df}{dx_i} = 0$ pro všechny (*N-1*)-bitové vektory hodnot proměnných $x_1,...,x_{i-1},x_{i+1},...,x_N$.
Na druhé straně, jestliže Booleovská funkce *f* závisí na proměnné x_i, existuje *(N-1)* bitový vektor hodnot proměnných $x_1,...,x_{i-1},x_{i+1},...,x_N$ pro který $\frac{df}{dx_i} = 1$

Definice 9 – Přísná propagace změny

Booleovská funkce *f* vyhovuje přísné propagaci změny podle proměnné x_i *i=1,...,N,* jestliže na množině (*N-1*)-bitových vektorů platí:

(14) $$w_t(\frac{df}{dx_i}) = 2^{N-2}$$

Definice 10 – Kriterium propagace změny Booleovské funkce

Hodnotou propagace změny Booleovské funkce *f* pro proměnnou x_i nazýváme veličinu

(15) $$w_t(\frac{df}{dx_i})$$

Kritériem propagace změny Booleovské funkce *f* (angl. Avalanche criterion) s *N* proměnnými nazveme veličinu

(16) $$\vartheta(f) = \sum_{i=1}^{N} \left| w_t(\frac{df}{dx_i}) - 2^{N-2} \right|$$

Čím je menší hodnota definovaná vztahem (16), tím kryptograficky kvalitnější je Booleovská funkce. Kritérium je popsáno např. v [3] nebo [4].

Definice 11 – Kriterium složitosti algebraické normální formy Booleovské funkce

Složitost algebraické normální formy Booleovské funkce *f* je počet nenulových členů v algebraicky normálním tvaru této funkce. Označíme jej výrazem *Z(f)*. V experimentu je toto kritérium počítáno dle vzorce:

(17) $$ANF(f) = 2^{N-1} - |Z(f) - 2^{N-1}|$$

Algebraická normální forma Booleovská funkce je součet (= XOR operace) výrazů, které jsou všemi možnými konjunkcemi vstupních proměnných. Úplný výpis Booleovské funkce s N vstupními proměnnými má až 2^N nenulových sčítanců. Kryptograficky není dobrý ani velký ani malý počet těchto sčítanců. Ideální je počet blízký hodnotě 2^{N-1}.

Jako poslední kritérium popsané v této práci, je speciální kritérium zaměřené proti využití diferenčních luštících metod. Toto kritérium popsané v [5] je spíše používáno u S-boxů (tj. Booleovských funkcí o N vstupních proměnných a M výstupních hodnotách), ale lze jej použít i na Booleovskou funkci s jedním výstupem. Kritérium je hodnota maximálního prvku v tzv. Input-Output matici diferencí (mimo první řádek).

Nechť T je NxM S-box. Potom matice Input-Output diferencí MIO má rozměr $2^N x 2^M$, kde každý prvek této matice $MIO[i,j]$ obsahuje počet případů, kdy pro vstupní vektory x_k a x_p platí: $x_k \oplus x_p = i$ a současně platí $T(x_k) \oplus T(x_p) = j$

Všechny prvky matice MIO jsou sudá čísla.
Pokud matice MIO má následující hodnoty, je příslušný S-box imunní proti diferenční luštící metodě.

(18) $MIO[0,0] = 2^N$
$MIO[0,j] = 0$ pro $j=1, \ldots, 2^M-1$
$MIO[i,j] = A$ pro $i=1, \ldots, 2^N-1$ a pro $j=0, \ldots, 2^M-1$
kde $A = \frac{2^N}{2^M}$

V případě invertibilního S-boxu ($N=M$) je A=1, což ovšem nemůže nastat (matice MIO obsahuje pouze sudá čísla). Optimální NxN S-Box by měl mít matici MIO obsahující (mimo první řádek) směs 0 a 2. V našem případě, kdy je N = 8, M = 1, je optimální hodnota A = 128, která je zároveň dolní hranicí kritéria MIO-Max.

Definice 12 – Kriterium MIO-Max Booleovské funkce

Nechť f je Booleovská funkce o N proměnných. Spočteme její matici Input-Output diferencí MIO. Kritérium MIO-Max je vyjádřeno vzorcem

(19) $$\psi(f) = \max_{i=1,\ldots,2^N-1,\, j=0,1}(MIO[i, j])$$

Nižší hodnota $\psi(f)$ je kryptograficky výhodnější.

Metodou náhodného výběru bylo vygenerováno 10 000 vybalancovaných Booleovských funkcí 8 proměnných, pro které byly vypočteny hodnoty výše popsaných kritérií.

- Nelinearita podle vzorce (6) (zkratka NELI) - vyšší je lepší
- Autokorelace podle vzorce (12) (zkratka AKOR) - nižší je lepší
- Avalanche kritérium podle vzorce (16) (zkratka AVAL) - nižší je lepší
- Složitost ANF podle vzorce (17) (zkratka ANFA) - vyšší je lepší
- MIO-Max podle vzorce (19) (zkratka MIOX) - nižší je lepší

Rozdělení hodnot těchto kritérií pro 10 000 náhodně vybraných vybalancovaných Booleovských funkcí 8 proměnných je v následující tabulce.

Kritérium	Průměr	Odchylka	Šikmost	Špičatost	Minimum	Medián	Maximum
NELI	103,53	2.876	-0.850	4.285	86	104	110
AKOR	67.62	8.963	0.598	3.690	40	64	112
AVAL	35.69	9.956	0.306	3.087	8	36	88
ANFA	121.65	4.827	-1.017	4.046	95	123	128
MIOX	161.81	4.482	0.598	3.690	148	160	184

Tab. 1

Ve veřejné literatuře jsou popsány metody generující vybalancované Booleovské funkce 8 proměnných s maximálně dosaženou nelinearitou 116. Netypický přístup je zvolen v [6], kde na rozdíl od ostatních metod, je na počátku místo vybalancované funkce s nízkou nelinearitou (např. získanou náhodným výběrem) vzata bent funkce, která je vhodně „vybalancovávána“, aby se získala maximálně nelineární vybalancovaná funkce. I v tomto případě byla pro $N = 8$ získána funkce s nelinearitou pouze 116. Z veřejné literatury není známo, zda existuje vybalancovaná Booleovská funkce 8 proměnných s nelinearitou 118 nebo zda neexistuje. Rovněž nebyl nalezen kvantitativní popis zde uvedených kritérií pro dostatečně reprezentativní počet vybalancovaných Booleovských funkcí 8 proměnných s maximální známou nelinearitou 116.

Autorovou metodou GaT popsanou v [7] bylo vygenerováno 10 000 vybalancovaných Booleovských funkcí 8 proměnných s nelinearitou 116.

V Tab. 2 jsou uvedeny statistiky kritérií pro tento soubor maximálně nelineárních Booleovských funkcí.

Kritérium	Průměr	Odchylka	Šikmost	Špičatost	Minimum	Medián	Maximum
NELI	116	0	0	0	116	116	116
AKOR	41.98	5.623	0.730	4.599	32	40	80
AVAL	21.48	6.155	0.306	3.062	4	22	46
ANFA	120.68	5.431	-0.972	3.798	95	122	128
MIOX	148.99	2.812	0.730	4.599	144	148	168

Tab. 2

Na všechny kritéria byl použit test normality uvedený v [8] na straně 95. Ve všech případech byla normalita rozdělení zamítnuta na hladině významnosti 0.01.

Byly testovány hypotézy o shodě výběrových průměrů kritérií mezi množinou náhodně vybraných funkcí a množinou funkcí s nelinearitou 116. Protože test shodnosti rozptylů uvedený v [9] na straně 94, zamítl ve všech případech hypotézu o rovnosti rozptylů na hladině významnosti 0.01, byl pro testování hypotéz o shodnosti výběrových průměrů použit Cochran-Coxův test (dále též C-C test) uvedený v [9] na straně 93.

Výsledky testů jsou v Tab. 3. Ve všech testech je kritická hodnota na hladině významnosti 0.01 rovna 2.576323.

Kritérium	C-C test	Hypotéza o rovnosti průměrů
AKOR	242.313422	Zamítá se
AVAL	121.395512	Zamítá se
ANFA	13.349063	Zamítá se
MIOX	242.281856	Zamítá se

Tab. 3

Závěr

Provedený experiment prokázal, ve shodě s veřejnou literaturou, že vysoká nelinearita u vybalancovaných Booleovských funkcí s 8 proměnnými pozitivně (ve smyslu kryptograficky výhodně) ovlivňuje ostatní uvedená kritéria, vyjma kritérium složitosti algebraické normální formy (ANFA).

Výsledky obdobných experimentů, které má autor k dispozici, pro regulární 8x8 S-Boxy naznačují, že kritérium ANFA je nekorelované s nelinearitou.

LITERATURA

[1] Nyberg K.: *On the Construction of Highly Nonlinear Permutations*, EUROCRYPT '92, pp. 92-98.

[2] Clark J. A., Jacob J. L., Stepney S.: *The design of S-Boxes by simulated annealing*, New Generation Computing archive, September 2005, Vol. 23, issue 3.

[3] McCluskey E. J.: *Logic design principles*, Prentice-Hall, New Jersey, 1986.

[4] Webster A. F.: *Plaintext/Ciphertext Bit Dependencies in Cryptographic Systems*, Master's thesis, Department of Electrical Engineering, Queen's University, 1985.

[5] Dawson M. H., Tavares S. E.: An Expanded Set of S-box Design Criteria Based on Information Theory and its Relation to Differential-Like Attacks, EUROCRYPT '91, p. 359.

[6] Izbenko, Y., Kovtun, V., Kuznetsov, A. The design of Boolean functions by modified hill climbing method (datasheet). 17 pages. [Online] Cited 2008. Available at: http://eprint.iacr.org/2008/111.pdf

[7] Tesař, P.: *A New Method for Generating High Non-linearity S-Boxes*, ISSN 1210-2512, Radioengineering, Volume 19, Number 1, April 2010, pp. 23 – 26.

[8] Meloun M., Militký J.: *Statistické zpracování experimentálních dat,* Praha 1998, East Publishing Praha.

[9] Anděl J.: *Matematická statistika,* Praha 1978, SNTL.

D. Odborná skupina kryptologie při JČMF
RNDr. Jaroslav Hrubý (hruby.jar@centrum.cz)

Jednota českých matematiků a fyziků, JČMF (http://jcmf.cz/)
Jednota českých matematiků a fyziků je profesní společnost, sdružující vědce, pedagogy i laické příznivce matematiky a fyziky. Její počátky sahají do roku 1862.
Jednota si klade za cíl podporovat rozvoj matematiky a fyziky nad rámec akademických a průmyslových institucí a to zejména popularizací, péčí o talenty a vydáváním odborných stanovisek.
Jednota českých matematiků a fyziků v rámci svých oborů především

- pořádá konference, semináře a setkání,
- vyhlašuje a organizuje soutěže na všech úrovních škol
- vydává časopisy, učebnice a monografie.
- popularizuje nové i klasické poznatky před laickou veřejností
- věnuje se historii
- vydává stanoviska k vědecké práci.

Odborná skupina kryptologie České matematické sekce (ČMS)
Používaný anglický název a zkratka:
Group of Cryptology Union of Czech Mathematicians and Physicists (GCUCMP**)**

Odborná skupina kryptologie je součástí České matematické sekce při Jednotě českých matematiků a fyziků. Jedná se o profesní skupinu, která sdružuje kryptology, vědce, pedagogy i laické příznivce o tuto oblast. Její počátky sahají do roku 1995, kdy byla založena z podnětu dnešního předsedy RNDr.J.Hrubého.

Skupina si klade za cíl podporovat rozvoj kryptologie nad rámec akademických a průmyslových a silových institucí a to zejména její popularizací.

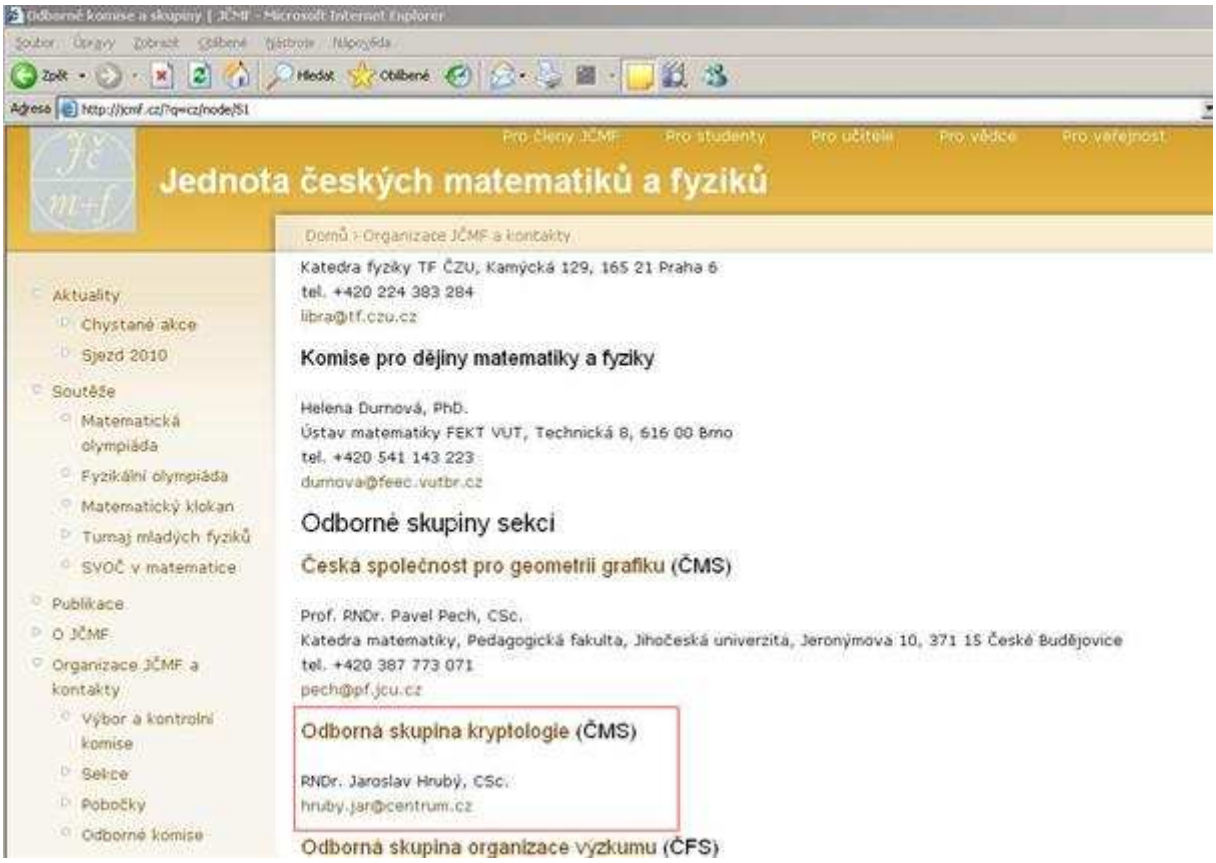
Skupina kryptologie

- od roku 1999 vydává vlastní e-zin Crypto-World
- pořádá setkání členů odborné skupiny
- od roku 2000 vyhlašuje a organizuje soutěže v luštění
- popularizuje nové i klasické poznatky před laickou veřejností
- věnuje se historii v oboru kryptografie
- podporuje konference a setkání a školení z oblasti kryptologie

Jak se stát členem?
a) nejprve je nutné se stát členem JČMF, v přihlášce uveďte, že se chcete stát členem sekce ČMS (Česká matematická společnost)
Přihláška: http://www.jcmf.cz/?q=node/33
Členské příspěvky: http://www.jcmf.cz/?q=node/46
Přihlášku i členské příspěvky lze zařídit elektronicky.

b) následně oznamte svůj zájem o práci v *Odborné skupině kryptologie* předsedovi této skupiny

Tím je registrace ukončena a stáváte se členy **Odborné skupiny kryptologie**.



Struktura odborné skupiny kryptologie

Předseda:	RNDr. Jaroslav Hrubý, CSc.
Místopředseda:	Ing. Jaroslav Pinkava, CSc.
Kontaktní adresa:	hruby.jar@centrum.cz
www:	http://jcmf.cz/?q=cz/node/51

E-zin Crypto-World

Vedoucí redaktor:	Mgr. Pavel Vondruška
Redakce:	RNDr. Vlastimil Klíma
	Ing. Jaroslav Pinkava, CSc.
	Ing. Tomáš Rosa, PhD.
Kontaktní adresa:	ezin@crypto-world.info
www:	http://crypto-world.info/

E. O čem jsme psali v březnu 2000 – 2010

F. 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 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 e-ziny, 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:	Vlastimil Klíma
	Pavel Vondruška
Přehled autorů:	http://crypto-world.info/obsah/autori.pdf
NEWS (výběr příspěvků, komentáře a vkládání na web)	Jaroslav Pinkava Vlastimil Klíma Pavel Vondruška Tomáš Rosa
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@gmail.com ,	http://crypto-world.info/pinkava/
Tomáš Rosa	tomas.rosa@rb.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