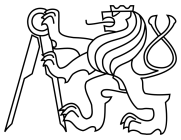
ČESKÉ VYSOKÉ UČENÍ TECHNICKÉ V PRAZE
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ



ZADÁNÍ DIPLOMOVÉ PRÁCE

Název:	Lineární kryptoanalýza šifry Anubis
Student:	Bc. Šárka Hatašová
Vedoucí:	Ing. Josef Kokeš
Studijní program:	Informatika
Studijní obor:	Počítačová bezpečnost
Katedra:	Katedra počítačových systémů
Platnost zadání:	Do konce letního semestru 2016/17

Pokyny pro vypracování

Seznamte se s technikou lineární kryptoanalýzy.
Seznamte se s šifrou Anubis, popište její součásti a zhodnoťte její vhodnost pro kryptoanalýzu.
Zpracujte rešerši aktuálního stavu lineární kryptoanalýzy této šifry.
Navrhněte model šifry Anubis vhodný pro lineární kryptoanalýzu.
Implementujte program, který provede útok na zvolený model pomocí technik lineární kryptoanalýzy.
Zhodnoťte dosažené výsledky.

Seznam odborné literatury

Dodá vedoucí práce.

L.S.

prof. Ing. Róbert Lórencz, CSc.
vedoucí katedry

prof. Ing. Pavel Tvrdík, CSc.
děkan

V Praze dne 9. ledna 2016

České vysoké učení technické v Praze
Fakulta informačních technologií
Katedra počítačových systémů

Diplomová práce

Lineární kryptoanalýza šifry Anubis

Bc. Šárka Hatašová

Vedoucí práce: Ing. Josef Kokeš

6. května 2016

Poděkování

Děkuji vedoucímu diplomové práce Ing. Jiřímu Kokešovi za trpělivost, podnětné nápady a čas, který vedení této práce věnoval. Mé poděkování dále patří rodině a příteli za podporu ve studiu.

Prohlášení

Prohlašuji, že jsem předloženou práci vypracovala samostatně a že jsem uvedla veškeré použité informační zdroje v souladu s Metodickým pokynem o etické přípravě vysokoškolských závěrečných prací.

Beru na vědomí, že se na moji práci vztahují práva a povinnosti vyplývající ze zákona č. 121/2000 Sb., autorského zákona, ve znění pozdějších předpisů, zejména skutečnost, že České vysoké učení technické v Praze má právo na uzavření licenční smlouvy o užití této práce jako školního díla podle § 60 odst. 1 autorského zákona.

V Praze dne 6. května 2016 .....

Odkaz na tuto práci

Hatašová, Šárka. *Lineární kryptoanalýza šifry Anubis.* Diplomová práce. Praha: České vysoké učení technické v Praze, Fakulta informačních technologií, 2016.

Abstrakt

Tato práce se zabývá konstrukcí a verifikací zmenšeného modelu šifry Anubis. Na tento redukovaný model je v několika různých variantách veden útok technikou lineární kryptoanalýzy. Výsledky útoků jsou podrobně rozebrány a dány do souvislosti s možnými dopady na samotný Anubis.

Klíčová slova Bezpečnost, lineární kryptoanalýza, Anubis, šifra, šifrování, nalezení klíče.

Abstract

This master thesis deals with design and verification of a reduced model of the Anubis cipher. The reduced model is subjected to various types of attacks by using linear cryptanalysis. Results of attacks are discussed in detail and associated with possible impact on the Anubis.

Keywords Security, linear cryptanalysis, Anubis, cipher, encryption, key recovery.

Obsah

Seznam obrázků

Seznam tabulek

Úvod

Utajování zpráv je záležitost stará jako lidstvo samo. Stejně jako se vyvíjí lidská společnost, činí pokroky i technika šifrování. Moderní šifry dneška stojí na pokročilých matematických konceptech, které podléhají neustálé inovaci. Kritickým pohledem na jejich strukturu můžeme předejít řadě zranitelností.

Motivace k šifrování je zjevná: zajistit ochranu dat a zabránit úniku informací. Jak ale poznat bezpečnou šifru, nebo jinak, jak detekovat potenciální vady v chování šifry? Tato diplomová práce má ambici odpovědět na otázku bezpečnosti šifrovacích algoritmů z kryptoanalytického úhlu pohledu, kdy předmětem analýzy je šifra Anubis.

První kapitola seznamuje s pojmem lineární kryptoanalýza a buduje důležitý matematický aparát, o který se celá teorie opírá. V závěrečné části této kapitoly je na názorném příkladu převedena teorie do praxe. Druhá kapitola se věnuje průzkumu současného stavu na poli kryptoanalýzy a vyhodnocuje, které šifry by bylo podnětné zkoumat. Anubis, šifra označená v rešeršní části jako zajímavá a v tomto směru dostatečně neprobádaná, je detailně představena ve třetí kapitole. Čtvrtá kapitola se zabývá konstrukcí zmenšeného modelu šifry. Krok po kroku jsou předkládány jednotlivé úpravy jejích dílčích částí – a to za účelem připravit vhodné podmínky pro aplikaci lineárně kryptoanalytických postupů. A konečně pátá kapitola, jádro celé práce, dává do souvislosti znalosti z první kapitoly se samotnou nově vytvořenou verzí šifry, předkládá výsledky a interpretuje je.

Kapitola 1

Lineární kryptoanalýza

Kryptologii, vědu o šifrách, můžeme rozdělit na dva hlavní proudy: *kryptografii*, která se zabývá tvorbou šifer, a *kryptoanalýzu*, která šifry zkoumá a hledá způsob, jak je prolomit. Jednu z často používaných metod analýzy blokových šifer představuje právě lineární kryptoanalýza.

Základy lineární kryptoanalýzy položil roku 1993 Mitsuru Matsui, když publikoval lineární kryptoanalýzu jako nový teoretický koncept a následně ji aplikoval na šifru DES [16] s tím, že podobného postupu lze využít u celé řady dalších blokových šifer.

Následující text ve stručnosti shrnuje základní principy lineární kryptoanalýzy. Pro podrobnější informace odkazuji na materiály [9, 14], ze kterých jsem čerpala.

1.1 Hlavní myšlenka

Lineární kryptoanalýza je nástroj k linearizaci takových částí šifry[1], které lineární nejsou. Hledá tedy lineární závislosti v rámci daných nelineárních vztahů. Lineární kryptoanalýza je útok typu *known ciphertext* – k otevřenému textu (OT) známe příslušný šifrový text (ŠT), ale nemůžeme si je libovolně zvolit. Cílem útoku je zjistit šifrovací klíč.

Pokud se zaměříme na strukturu blokových šifer, tak nelineární část zpravidla představují S-boxy[2]. Na příkladu jednoduché substituční a permutační sítě (SPN), jejíž jedinou nelinearitu přestavují právě S-boxy, ukážeme konstrukci aproximativního lineárního výrazu pro celou šifru a jeho význam při odhadování bitů klíče.

[1] Můžeme analyzovat celou šifru (dovolí-li to její velikost) nebo jen její určitou část – pouze S-boxy či libovolný počet rund.

[2] S-box neboli substituční box je zobrazení, které transformuje posloupnost vstupních bitů na výstupní. Od dobrého S-boxu očekáváme, že malá změna vstupních hodnot vede k velkým změnám na výstupu. Špatně navržený S-box výrazně zlepšuje úspěšnost lineární kryptoanalýzy.

1.2 Konstrukce lineárního výrazu

Jak už bylo naznačeno, lineární kryptoanalýza nahrazuje nelineární části šifry lineárním výrazem, který je aproximativní, tedy platí s nějakou pravděpodobností. Čím vyšší pravděpodobnost přísluší danému výrazu, tím lépe pro úspěšnost analýzy. Takový výraz obecně vyjadřujeme pomocí vstupních a výstupních bitů a zapisujeme jej v následujícím tvaru:

$$\begin{aligned} X &= [X_1, X_2, \ldots, X_n] \quad \text{(vstupní bity)}, \\ Y &= [Y_1, Y_2, \ldots, Y_m] \quad \text{(výstupní bity)}, \end{aligned}$$

$$\begin{aligned} &X_1 \oplus X_2 \oplus \ldots \oplus X_n \oplus Y_1 \oplus Y_2 \oplus \ldots \oplus Y_m = 0, \\ &\text{kde } \oplus \text{ je operace XOR, tj. součet (mod 2).} \end{aligned} \tag{1.1}$$

Sledujeme, jaká je pravděpodobnost výskytu rovnic ve tvaru (1.1) pro různé kombinace vstupních a výstupních bitů.

1.2.1 Odchylka lineární pravděpodobnosti

Odchylka lineární pravděpodobnosti (LPB, Linear Probability Bias) je elementárním pojmem celé lineární kryptoanalýzy. Představme si černou skříňku, *black box*[3], o které nevíme nic jiného, než že každý jednobitový vstup X transformuje právě na jednobitový výstup Y. Pokud by platilo, že chování této tajemné skříňky je absolutně náhodné, nemohli bychom vůbec předpovídat, jaký bit se v budoucnu objeví na výstupu. To, že nevíme, jaké Y bude odpovědí[4] na vstupní X, lze pomocí pravděpodobnosti formulovat tímto způsobem:

$$P(Y = X) = P(Y \neq X) = P(Y = 0) = P(Y = 1) = \frac{1}{2}. \tag{1.2}$$

Pokud by byla vnitřně tato tajemná krabička realizována jako šifra a neznali bychom nějakou další podstatnou informaci (např. část klíče nebo aproximaci, která se blíží skutečnému klíči), zdálo by se, že se šifra chová jako náhodné orákulum[5]. Máme-li ovšem takovou informaci k dispozici, můžeme ji využít ve svůj prospěch. Pravděpodobnost, že se na výstupu objeví Y stejné, respektive odlišné od X, nebude přesně $\frac{1}{2}$, ale bude se od případu, kdy nic dalšího nevíme, *odchylovat*.

$$\begin{aligned} &P(Y = X) = \frac{1}{2} + \varepsilon, \\ &P(Y \neq X) = 1 - P(Y = X) = \frac{1}{2} - \varepsilon, \quad \varepsilon \in \langle -1/2, 1/2 \rangle. \end{aligned} \tag{1.3}$$

[3] Zařízení, o kterém víme, jak se projevuje navenek; neznáme ovšem mechanismy, které probíhají uvnitř.

[4] Nutno podotknout, že možnosti jsou jenom dvě: buď jsou X a Y stejné, nebo se liší.

[5] Náhodné orákulum odpovídá na vstup náhodným výběrem z množiny výstupů vždy tak, aby platilo, že pro dva stejné vstupy jsou shodné i jejich výstupy.

Pokud známe pravděpodobnost p platnosti určitého výrazu typu (1.1), stanovit jeho odchylku ε znamená odečíst od pravděpodobnosti p právě $\frac{1}{2}$. Velikost odchylky ε je pak absolutní hodnota výrazu:

$$| \varepsilon | = \left| p - \frac{1}{2} \right|. \tag{1.4}$$

Znalost odchylek různých výrazů sestavených kombinací vstupních a výstupních bitů je pro lineární kryptoanalýzu velice cenná. Nenulové odchylky, zejména pak výskyt vysokých nebo naopak nízkých (záporných) hodnot LPB, upozorňují na možné slabiny šifry.

1.3 Piling-Up věta

Piling-Up věta je podle Matsuie [16] naprosto nezbytný matematický princip pro sestavení výhodné lineární aproximace. Můžeme ho chápat jako návod, který říká, jak spočítat celkovou pravděpodobnost složenou z více dílčích odchylek. V této části se zaměřím na jeho odvození a plné znění, aplikaci Piling-up principu najdete v navazující sekci 1.4 s modelovým příkladem.

1.3.1 Odvození

Mějme dvě nezávislé náhodné binární proměnné X_1 a X_2. Nechť $X_1 \oplus X_2 = 0$ je lineární výraz a $X_1 \oplus X_2 = 1$ afinní výraz. Předpokládejme následující pravděpodobnostní rozdělení pro X_1 a X_2:

$$\begin{aligned} P(X_1 = i) &= \begin{cases} p_1 & , \ i = 0 \\ 1 - p_1 & , \ i = 1 \end{cases} \\ P(X_2 = i) &= \begin{cases} p_2 & , \ i = 0 \\ 1 - p_2 & , \ i = 1 \end{cases} \end{aligned} \tag{1.5}$$

Využijme vzájemné nezávislosti X_1 a X_2. Pro sdruženou pravděpodobnost pak platí:

$$P(X_1, X_2) = \begin{cases} P(X_1 = 0, X_2 = 0) & = p_1 p_2, \\ P(X_1 = 0, X_2 = 1) & = p_1(1 - p_2), \\ P(X_1 = 1, X_2 = 0) & = (1 - p_1)p_2, \\ P(X_1 = 1, X_2 = 1) & = (1 - p_1)(1 - p_2). \end{cases} \tag{1.6}$$

Díky (1.6) můžeme odvodit pravděpodobnost výrazu $X_1 \oplus X_2 = 0$:

$$\begin{aligned} P(X_1 \oplus X_2 = 0) &= P(X_1 = X_2) \\ &= P((X_1 = 0 \wedge Y_1 = 0) \vee (X_1 = 1 \wedge Y_1 = 1)) \\ &= P(X_1 = 0, Y_1 = 0) + P(X_1 = 1, Y_1 = 1) \\ &= p_1 p_2 + (1 - p_1)(1 - p_2) \end{aligned} \tag{1.7}$$

Nakonec můžeme vyjádřit celkovou odchylku $X_1 \oplus X_2 = 0$. Nechť $p_1 = \frac{1}{2} + \varepsilon_1$ a $p_2 = \frac{1}{2} + \varepsilon_2$. Pak

$$P(X_1 \oplus X_2 = 0) = \frac{1}{2} + \underbrace{2\varepsilon_1\varepsilon_2}_{\substack{\text{celková} \\ \text{LPB}}}. \tag{1.8}$$

1.3.2 Piling-Up lemma

Víme-li, jak vypadá LPB pro dvě nezávislé binární náhodné proměnné (1.8), můžeme dosažený výsledek zobecnit pro n náhodných proměnných.

Lemma 1 (Piling-Up [16])**.** *Pro n nezávislých náhodných binárních proměnných $X_1, X_2, \ldots, X_n$ platí, že*

$$P(X_1 \oplus \cdots \oplus X_n = 0) = \frac{1}{2} + 2^{n-1} \prod_{i=1}^{n} \varepsilon_i$$

nebo ekvivalentně

$$\varepsilon_{1,2,\ldots,n} = 2^{n-1} \prod_{i=1}^{n} \varepsilon_i,$$

kde ε_i přísluší výrazu $X_i = 0$ a $\varepsilon_{1,2,\ldots,n}$ je LPB výrazu $X_1 \oplus \cdots \oplus X_n = 0$.

1.4 Modelový příklad

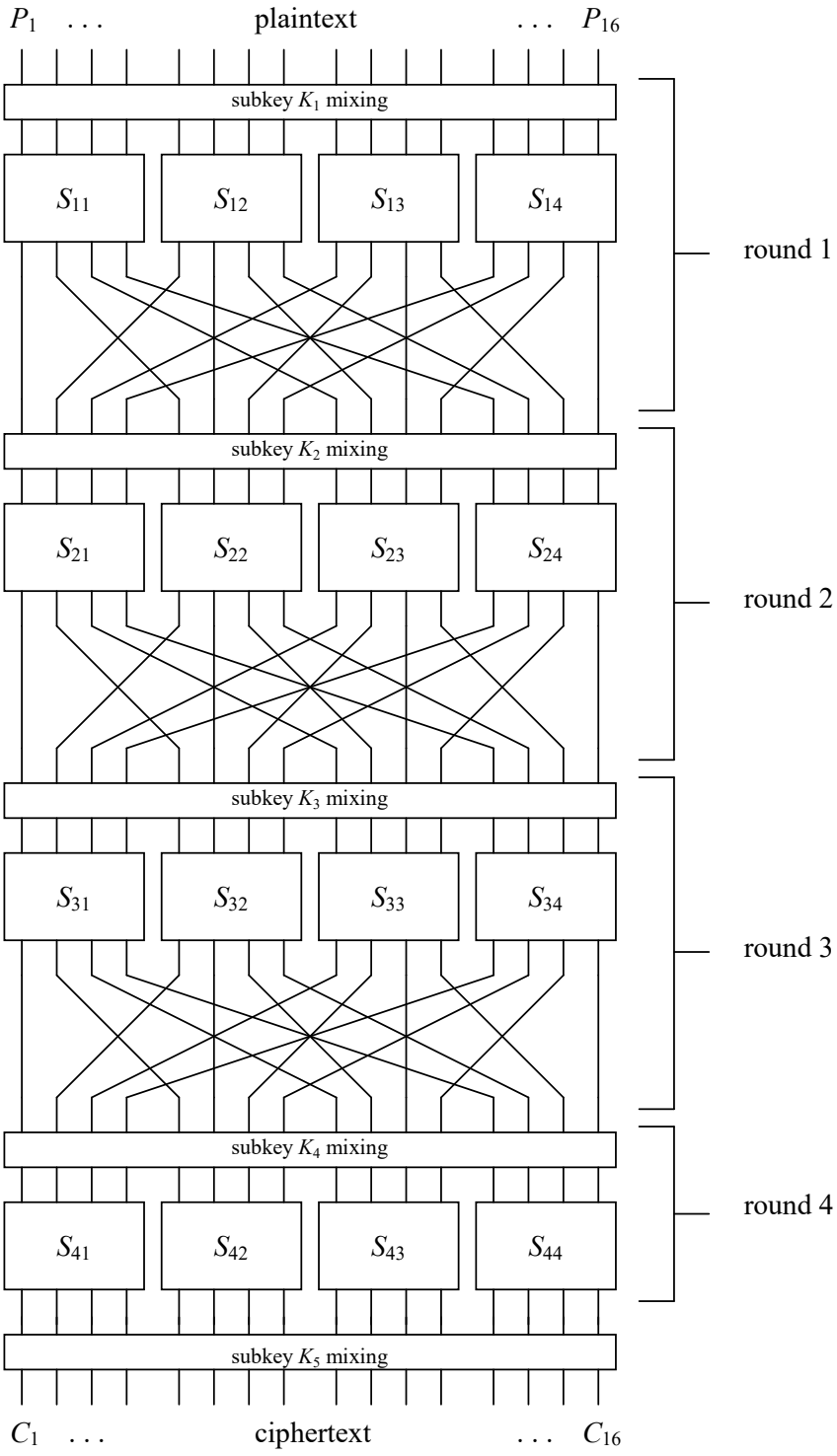
Prvním krokem pro provedení úspěšné kryptoanalýzy je najít výrazné odchylky v chování S-boxu. Na jejich základě je možné sestavit lineární aproximaci pro celou šifru, která bude končit v místě posledního přičtení rundovního klíče. Na tento klíč je veden útok; ukážu, jak jej se znalostí příslušné lineární aproximace zjistit.

Kryptoanalytické postupy demonstruji na příkladu jednoduché SPN převzaté z [9]. Tato SPN (viz obrázek 1.1) pracuje s blokem o velikosti 16 bitů, kdy se každá runda skládá z přičtení 16-bitového rundovního klíče, čtyř 4-bitových identických S-boxů provádějících substituci a následné permutaci výstupů S-boxu.

1.4.1 Analýza S-boxu

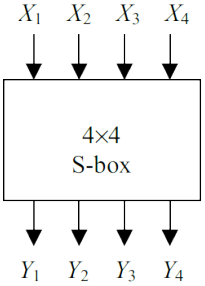
S-box je v našem případě realizován pevně danou permutací, která 4-bitový vstup $X = [X_1, X_2, X_3, X_4]$ transformuje na 4-bitový výstup $Y = [Y_1, Y_2, Y_3, Y_4]$ přesně podle obrázku 1.2. Výsledkem analýzy S-boxu je lineární aproximační tabulka (obrázek 1.4) o $2^4 \times 2^4$ buňkách, kde každá buňka reprezentuje odchylku od ideálního počtu shod mezi rovnicí sestavenou ze vstupních a výstupních bitů vstupujících a vystupujících z S-boxu; obě části rovnice jsou hexadecimálně zakódované předpisy ve tvaru (1.1).

Obrázek 1.1: Substituční a permutační síť. Zdroj: [9].



Obrázek 1.2: S-box substituční a permutační sítě. Zdroj: [14].

Vstup	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Výstup	E	4	D	1	2	F	B	8	3	A	6	C	5	9	0	7



Cesta k získání lineární aproximační tabulky vede přes kontrolu počtu shod kombinací všech možných vstupů a výstupů S-boxu pro každé možné ohodnocení jejich jednotlivých bitů. Náznak, odkud se data pro lineární aproximační tabulku generují, je na obrázku 1.3.

Rozeberme si nyní rovnici $X_2 \oplus X_3 = Y_1 \oplus Y_3 \oplus Y_4$, ekvivalentně též

$$X_2 \oplus X_3 \oplus Y_1 \oplus Y_3 \oplus Y_4 = 0, \tag{1.9}$$

jež je na obrázku 1.3 podbarvena zeleně a v lineární aproximační tabulce 1.4 označena souřadnicemi $[6, \mathrm{B}]$.

Na obrázku 1.3 vidíme výsledek vstupní a výstupní části rovnice (1.9) pro konkrétní hodnoty X a Y. Pokud by přesně polovina zeleně vyznačených řádků byla shodná, museli bychom konstatovat, že (1.9) je perfektně nelineární, čemuž odpovídá nulová LPB. Nyní se ale nacházíme v situaci, kdy se shoda vyskytla na 12 řádcích z 16. Odečtením poloviny od 12/16 získáme LPB příslušící rovnici (1.9). Výsledná odchylka je $+\frac{4}{16}$; do 1.4 zapíšeme pro přehlednost jen hodnotu čitatele. Souřadnice `Input Sum` a `Output Sum` v lineární aproximační tabulce jsou jednoduše odvozeny z předpisu rovnice (1.9):

$$\begin{aligned} \texttt{Input Sum} &= (\underbrace{0}_{X_1}\underbrace{1}_{X_2}\underbrace{1}_{X_3}\underbrace{0}_{X_4})_2 = (6)_{16}, \\ \texttt{Output Sum} &= (\underbrace{1}_{Y_1}\underbrace{0}_{Y_2}\underbrace{1}_{Y_3}\underbrace{1}_{Y_4})_2 = (\mathrm{B})_{16}. \end{aligned} \tag{1.10}$$

Stejným způsobem stanovíme LPB všech rovnic typu (1.1) – 2^4 kombinací vstupních bitů $\times$ 2^4 kombinací výstupních bitů dává dohromady 256prvkovou lineární aproximační tabulku .

V obecném případě, je-li vstup S-boxu n-bitový, tj. $X = [X_1, X_2, \ldots, X_n]$, a výstup m-bitový, tj. $Y = [Y_1, Y_2, \ldots, Y_m]$, pak počet rovnic k prověření je

Obrázek 1.3: Ukázka výpočtu lineární aproximace S-boxu. Zdroj: [9].

X_1	X_2	X_3	X_4	Y_1	Y_2	Y_3	Y_4	$X_2 \oplus X_3$	$Y_1 \oplus Y_3 \oplus Y_4$	$X_1 \oplus X_4$	Y_2	$X_3 \oplus X_4$	$Y_1 \oplus Y_4$
0	0	0	0	1	1	1	0	0	0	0	1	0	1
0	0	0	1	0	1	0	0	0	0	1	1	1	0
0	0	1	0	1	1	0	1	1	0	0	1	1	0
0	0	1	1	0	0	0	1	1	1	1	0	0	1
0	1	0	0	0	0	1	0	1	1	0	0	0	0
0	1	0	1	1	1	1	1	1	1	1	1	1	0
0	1	1	0	1	0	1	1	0	1	0	0	1	0
0	1	1	1	1	0	0	0	0	1	1	0	0	1
1	0	0	0	0	0	1	1	0	0	1	0	0	1
1	0	0	1	1	0	1	0	0	0	0	0	1	1
1	0	1	0	0	1	1	0	1	1	1	1	1	0
1	0	1	1	1	1	0	0	1	1	0	1	0	1
1	1	0	0	0	1	0	1	1	1	1	1	0	1
1	1	0	1	1	0	0	1	1	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1	0	1	0
1	1	1	1	0	1	1	1	0	0	0	1	0	1

$2^n \times 2^m = 2^{m+n}$. Počet řádků lineární aproximační tabulky je roven 2^n a počet sloupců 2^m. Označíme-li si $s \in \{0, 1, 2, \ldots, 2^n\}$ jako počet shodných řádků pro libovolnou rovnici typu (1.1), potom přípustné pravděpodobnostní odchylky nabývají hodnot $\varepsilon = \frac{s}{2^n} - \frac{1}{2}$.

1.4.2 Sestavení lineární aproximace

Kritéria pro vytvoření vhodné lineární aproximace kompletní šifry by se dala shrnout takto:

1. Vybírej takové průchody S-boxem, které mají v absolutní hodnotě co největší odchylku.
2. Je výhodné, pokud je po průchodu S-boxem počet aktivních bitů menší, než byl na jeho vstupu.
3. Udržuj počet aktivních S-boxů co nejnižší.

Průchod šifrou, jež je v souladu s těmito třemi zásadami, je vyznačen na obrázku 1.5. Držme se tohoto obrázku a označme bity otevřeného textu P_i, bity rundovního klíče přičtené v rundě r jako $K_{r,i}$, vstupy S-boxů jako $U_{r,i}$ a výstupy S-boxů jako $V_{r,i}$, kde kde $r \in \{1, 2, 3, 4\}$ zastupuje číslo rundy a $i \in \{1, 2, 3, \ldots, 16\}$ pořadové číslo příslušného bitu (číslováno zleva doprava).

Obrázek 1.4: Lineární aproximační tabulka. Zdroj: [9].

		Output Sum															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Input Sum	0	+8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
	1	0	0	−2	−2	0	0	−2	+6	+2	+2	0	0	+2	+2	0	0
	2	0	0	−2	−2	0	0	−2	−2	0	0	+2	+2	0	0	−6	+2
	3	0	0	0	0	0	0	0	0	+2	−6	−2	−2	+2	+2	−2	−2
	4	0	+2	0	−2	−2	−4	−2	0	0	−2	0	+2	+2	−4	+2	0
	5	0	−2	−2	0	−2	0	+4	+2	−2	0	−4	+2	0	−2	−2	0
	6	0	+2	−2	+4	+2	0	0	+2	0	−2	+2	+4	−2	0	0	−2
	7	0	−2	0	+2	+2	−4	+2	0	−2	0	+2	0	+4	+2	0	+2
	8	0	0	0	0	0	0	0	0	−2	+2	+2	−2	+2	−2	−2	−6
	9	0	0	−2	−2	0	0	−2	−2	−4	0	−2	+2	0	+4	+2	−2
	A	0	+4	−2	+2	−4	0	+2	−2	+2	+2	0	0	+2	+2	0	0
	B	0	+4	0	−4	+4	0	+4	0	0	0	0	0	0	0	0	0
	C	0	−2	+4	−2	−2	0	+2	0	+2	0	+2	+4	0	+2	0	−2
	D	0	+2	+2	0	−2	+4	0	+2	−4	−2	+2	0	+2	0	0	+2
	E	0	+2	+2	0	−2	−4	0	+2	−2	0	0	−2	−4	+2	−2	0
	F	0	−2	−4	−2	−2	0	+2	0	0	−2	+4	−2	−2	0	+2	0

Využijme již spočtených lineárních aproximací pro aktivní S-boxy:

- $S_{1,2} : X_1 \oplus X_3 \oplus X_4 = Y_2$ má pravděpodobnostní odchylku $+\frac{1}{4}$,
- $S_{2,2} : X_2 = Y_2 \oplus Y_4$ má pravděpodobnostní odchylku $-\frac{1}{4}$,
- $S_{3,2} : X_2 = Y_2 \oplus Y_4$ má pravděpodobnostní odchylku $-\frac{1}{4}$,
- $S_{3,4} : X_2 = Y_2 \oplus Y_4$ má pravděpodobnostní odchylku $-\frac{1}{4}$.

Víme, že vstup do S-boxů v první rundě je získán přičtením klíče k otevřenému textu, tedy $U_{1,i} = P_i \oplus K_{1,i}$. Pro 1. rundu pak platí:

$$\begin{aligned} V_{1,6} &= U_{1,5} \oplus U_{1,7} \oplus U_{1,8} \\ &= (P_5 \oplus K_{1,5}) \oplus (P_7 \oplus K_{1,7}) \oplus (P_8 \oplus K_{1,8}) \end{aligned} \tag{1.11}$$

Odchylka rovnice (1.11) je $+\frac{1}{4}$, celková odchylka lineární aproximace po první rundě je proto též $+\frac{1}{4}$.

Pro 2. rundu platí:

$$\begin{aligned} V_{2,6} \oplus V_{2,8} &= U_{2,6} \\ &= V_{1,6} \oplus K_{2,6} \\ &= P_5 \oplus P_7 \oplus P_8 \oplus K_{1,5} \oplus K_{1,7} \oplus K_{1,8} \oplus K_{2,6} \end{aligned} \tag{1.12}$$

Odchylka rovnice (1.12) je sice $-\frac{1}{4}$, ale celkovou odchylku pro první a druhou rundu získáme aplikací piling-up lemmatu[6] (1): $\varepsilon_{1,2} = 2 \cdot \frac{1}{4} \cdot (-\frac{1}{4}) = -\frac{1}{8}$.

Ve 3. rundě jsou aktivní dva S-boxy, máme tedy dvě rovnice, každou s odchylkou $-\frac{1}{4}$:

$$\begin{aligned} V_{3,6} \oplus V_{3,8} &= U_{3,6} \\ &= V_{2,6} \oplus K_{3,6} \end{aligned} \tag{1.13}$$

$$\begin{aligned} V_{3,14} \oplus V_{3,16} &= U_{3,14} \\ &= V_{2,8} \oplus K_{3,14} \end{aligned} \tag{1.14}$$

Sloučením (1.13) a (1.14) po dosazení dostáváme:

$$\begin{aligned} &V_{3,6} \oplus V_{3,8} \oplus V_{3,14} \oplus V_{3,16} \\ &= P_5 \oplus P_7 \oplus P_8 \oplus K_{1,5} \oplus K_{1,7} \oplus K_{1,8} \oplus K_{2,6} \oplus K_{3,6} \oplus K_{3,14} \end{aligned} \tag{1.15}$$

A konečně při vstupu do 4. rundy:

$$\begin{aligned} U_{4,6} &= V_{3,6} \oplus K_{4,6} \\ U_{4,8} &= V_{3,8} \oplus K_{4,8} \\ U_{4,14} &= V_{3,14} \oplus K_{4,14} \\ U_{4,16} &= V_{3,16} \oplus K_{4,16} \end{aligned} \tag{1.16}$$

Spojením všech vstupů (1.16) obou do S-boxů ve 4. rundě můžeme psát:

$$U_{4,6} \oplus U_{4,8} \oplus U_{4,14} \oplus U_{4,16} = P_5 \oplus P_7 \oplus P_8 \oplus \sum K, \tag{1.17}$$

kde $\sum K$ vyjadřuje součet přes všechny zúčastněné klíče $K_{r,i}$. Hodnota sumy je pevně daná, nabývá 0 nebo 1 v závislosti na konkrétním šifrovacím klíči.

$$\begin{aligned} \sum K = &K_{1,5} \oplus K_{1,7} \oplus K_{1,8} \oplus K_{2,6} \oplus K_{3,6} \\ &\oplus K_{3,14} \oplus K_{4,6} \oplus K_{4,8} \oplus K_{4,14} \oplus K_{4,16}. \end{aligned} \tag{1.18}$$

Označme L_0 rovnici $U_{4,6} \oplus U_{4,8} \oplus U_{4,14} \oplus U_{4,16} \oplus P_5 \oplus P_7 \oplus P_8 \oplus \sum K = 0$ a L_1 rovnici $U_{4,6} \oplus U_{4,8} \oplus U_{4,14} \oplus U_{4,16} \oplus P_5 \oplus P_7 \oplus P_8 \oplus \sum K = 1$. Hodnota $\sum K$ rovna 0, respektive 1, ovlivní pravděpodobnost aproximačního výrazu (1.17) tímto způsobem:

$$\sum K = \begin{cases} 0: & P(L_0) = \frac{1}{2} + \varepsilon_{1,2,3,4} = \frac{15}{32} \\ 1: & P(L_1) = 1 - P(L_0) = 1 - \frac{15}{32} = \frac{17}{32} \end{cases} \tag{1.19}$$

[6] Nemůžeme nijak zaručit, že předpoklad lemmatu o vzájemné nezávislosti proměnných je splněný. I přes tuto skutečnost funguje piling-up lemma jako dobrý odhad dané lineární aproximace.

Vidíme, že absolutní hodnota dříve vypočtené odchylky (1.20) se nezměnila. Dokonce můžeme prohlásit, že na absolutní hodnotu odchylky nemá konstantní $\sum K$, ať už je jakákoli, žádný vliv.

Celkovou odchylku lineární aproximace šifry končící před poslední úrovní S-boxů odvodíme opět za pomoci piling-up lemmatu (1):

$$\varepsilon_{1,2,3,4} = 2^3 \cdot \frac{1}{4} \cdot \left(-\frac{1}{4}\right) \cdot \left(-\frac{1}{4}\right) \cdot \left(-\frac{1}{4}\right) = -\frac{1}{32}. \tag{1.20}$$

1.4.3 Hledání klíče

Nyní známe lineární aproximaci šifry až do předposlední rundy a k tomu příslušnou odchylku (1.20). K útoku, jehož cílem je zjistit bity posledního rundovního klíče (konkrétně úseku $K_{5,5}$ až $K_{5,8}$ a $K_{5,13}$ až $K_{5,16}$ vyznačených na obrázku 1.5), je třeba vygenerovat velké množství dvojic otevřeného a šifrového textu. Pro všechny takové otevřené texty platí, že jsou libovolné, ale vzájemně odlišné. K jejich zašifrování je použit neznámý, avšak pevně daný šifrovací klíč.

Neměnnost šifrovacího klíče zaručuje, že XOR jeho konkrétních expandovaných částí je sice též neznámý, ale konstantní, jak ukazuje rovnice (1.18). Navíc je díky (1.19) zřejmé, že v určitém smyslu na $\sum K$ nezáleží; můžeme začít s útokem.

Osm bitů hledaného podklíče znamená 2^8 různých možností, kterých může reálně nabývat. Pro každou z 256 variant zopakujeme tento algoritmus:

1. Vezmi jeden šifrovací podklíč $K \in \{0, \ldots, 255\}$ a doplň ho na místech, která pro tento výpočet nejsou důležitá, nulami.

$$\underbrace{0}_{K_1}\underbrace{0}_{K_2}\underbrace{0}_{K_3}\underbrace{0}_{K_4}\underbrace{?}_{K_5}\underbrace{?}_{K_6}\underbrace{?}_{K_7}\underbrace{?}_{K_8}\underbrace{0}_{K_9}\underbrace{0}_{K_{10}}\underbrace{0}_{K_{11}}\underbrace{0}_{K_{12}}\underbrace{?}_{K_{13}}\underbrace{?}_{K_{14}}\underbrace{?}_{K_{15}}\underbrace{?}_{K_{16}} \tag{1.21}$$

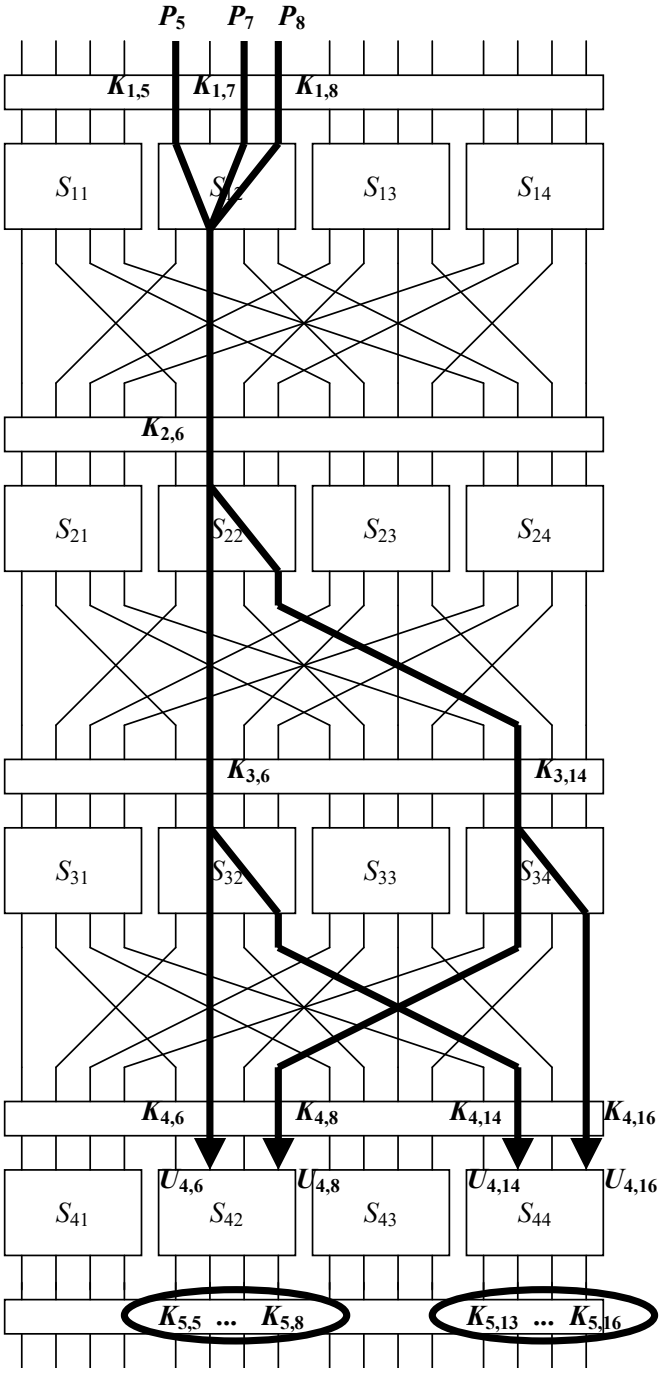
2. Vytvoř k němu co nejvíce[7] dvojic otevřeného a šifrového textu (PT, CT). Nastav čítač na hodnotu 0.

3. Pro každou dvojici (PT, CT) proveď tyto kroky:

   (a) Získej $U_{4,5}, \ldots, U_{4,8}$ a $U_{4,13}, \ldots, U_{4,16}$ provedením zpětné substituce součtu (mod 2) šifrového textu s šifrovacím klíčem K poslední

[7] I u takto skromné šifry je výpočetně náročné projít všechny možné kombinace, omezíme se na vzorek dat vhodné (tedy statisticky významné) velikosti, např. 10 000 párů (PT, CT).

Obrázek 1.5: Hledání lineární aproximace. Zdroj: [9].



rundy:

$$\begin{aligned} U_{4,5} &= S_{4,2}[K_{5,5} \oplus CT_5]^{-1}, \\ U_{4,6} &= S_{4,2}[K_{5,6} \oplus CT_6]^{-1}, \\ U_{4,7} &= S_{4,2}[K_{5,7} \oplus CT_7]^{-1}, \\ U_{4,8} &= S_{4,2}[K_{5,8} \oplus CT_8]^{-1}, \\ U_{4,13} &= S_{4,4}[K_{5,13} \oplus CT_{13}]^{-1}, \\ U_{4,14} &= S_{4,4}[K_{5,14} \oplus CT_{14}]^{-1}, \\ U_{4,15} &= S_{4,4}[K_{5,15} \oplus CT_{15}]^{-1}, \\ U_{4,16} &= S_{4,4}[K_{5,16} \oplus CT_{16}]^{-1}. \end{aligned} \tag{1.22}$$

(b) Dosaď do rovnice L_0 hodnoty získané v předchozím kroku spolu s bity otevřeného textu P_5, P_7 a $P_8 \in PT$. Pokud je splněna, tj. rovná nule, inkrementuj čítač.

4. Sleduj, kolikrát proběhla shoda pro všech 2^8 klíčů, jež přicházejí v úvahu. Počet shod označ q, počet vzorků $n = 10\,000$.

5. Stanov odchylku ε pro každou sadu vzorků.

$$| \varepsilon | = \frac{\left| q - \frac{n}{2} \right|}{n} = \frac{| q - 5\,000 |}{10\,000} \tag{1.23}$$

6. Nejvyšší absolutní hodnota odchylky, kterou takto získáme, patří vzorku, na který byl použit klíč K, jež jsme tímto nalezli.

Jak postupovat dále: K získání celého rundovního klíče můžeme buď sestavit novou lineární aproximaci a postupovat obdobně, nebo za pomoci hrubé síly dopočítat zbylou polovinu neznámých hodnot z celkových 16 bitů. Pro získání hlavního, tedy neexpandovaného šifrovacího klíče, jsou také dvě možnosti: určit poslední rundovní klíč, dále předposlední atd. až k prvnímu rundovnímu klíči. Pokud je způsob expanze klíče známý a lze jej projít zpětným chodem, tj. jedná-li se o invertovatelnou funkci, je možné hlavní šifrovací klíč dopočítat přímo z jakéhokoli již známého rundovního klíče.

Ukázala jsem, jak využít lineární kryptoanalýzu ke zjištění šifrovacího klíče. Pro konkrétní výsledky útoku na SPN doporučuji prohlédnout zdroje [9, 14].

Kapitola 2

Výběr šifry

Šifru, kterou by bylo vhodné podrobit lineární kryptoanalýze, jsem vybírala z finalistů soutěže o federální šifrovací algoritmus (AES) amerického úřadu pro standardizaci (NIST), starších kryptografických standardů, účastníků projektu NESSIE[8] a několika méně známých šifer. Klíčem k posouzení, zda je šifra vhodná pro aplikaci kryptoanalytických metod, byla kombinace několika následujících faktorů:

- složitost šifry,
- význam šifry,
- skutečnost, zda šifra již byla v minulosti prolomena touto metodou.

Pro všechny diskutované šifry platí, že jsou symetrické a blokové. Symetrické šifry používají k šifrování a dešifrování stejný klíč. Jako blokové označujeme šifry, které pracují s bloky pevně dané délky; vstupní data jsou nejprve rozdělena do stejně velikých úseků (bloků) a poslední blok je dle potřeby doplněn výplní (tzv. *padding*). Všechny bloky jsou šifrovány stejnou transformací. To samé platí pro průběh dešifrování, jen je obráceno pořadí transformací.

2.1 DES a Triple DES

DES je vítězem veřejné soutěže o šifrovací standard z roku 1977 pro ochranu citlivých neutajovaných dat ve státní správě USA [15]. Jedná se o šifru Feistelova typu s velikostí bloku 64 bitů a celkovou délkou klíče 64 bitů, kde v 56 bitech je nesena informace o klíči a každý osmý bit je paritní. DES je iterovaná šifra s 16 rundami.

[8] Projekt v rámci programu IST Evropské komise. Hlavním cílem projektu bylo předložení silných kryptografických primitiv [18].

Nevyhovujícím parametrem této šifry je především délka klíče. Z tohoto důvodu je šifra prolomitelná během relativně krátké doby hrubou silou. Teoreticky je útok lineární kryptoanalýzou na plný počet rund možný, ale nemá žádný praktický důsledek – generování dostatečného počtu párů otevřeného a šifrového textu by trvalo řádově desítky dnů, stejně tak hledání samotného klíče (nemluvě o paměťových nárocích takového útoku) [16].

Nástupce šifry DES, Triple DES, je třikrát za sebou zopakovaný DES se dvěma nebo třemi různými klíči. Problém s příliš krátkým klíčem je tímto odstraněn. Také počet rund je 3x vyšší než u DES, z čehož vyplývá pravděpodobný neúspěch při použití metod lineární kryptoanalýzy.

2.2 AES

Platný, dnes patrně nejpoužívanější šifrovací standard. Původní název šifry – Rijndael – odkazuje na své dva tvůrce: Vincenta Rijmena a Joanu Daemen. AES je šifra typu SPN s bloky délky 128 bitů a volitelnou délkou klíče: 128, 192 nebo 256 bitů. V závislosti na délce klíče je počet rund 10, 12 nebo 14. V současné době není znám způsob, jak tuto šifru efektivně pomocí kryptoanalytických metod prolomit; nicméně existují studie její zmenšené verze, které linearitu některých nelineárních vztahů naznačují [6, 12].

2.3 Blowfish a Twofish

Blowfish je šifra Feistelova typu s 16 rundami, délkou bloku 64 bitů a variabilní velikostí klíče: 32 až 448 bitů. Nelinearitu šifry posilují klíčově závislé S-boxy. Díky nim je šifra dobře odolná vůči lineární kryptoanalýze a je dodnes neprolomená [19].

Následník Blowfish – Twofish – má velice podobnou strukturu jako její předchůdce. Opět se jedná o šifru Feistelova typu s 16 rundami. Velikost bloku je vždy 128 bitů, přípustné délky klíče jsou 128, 192 a 256 bitů. Nelinearitu představují již zmíněné klíčově závislé S-boxy; u Twofish jsou vylepšené tak, aby skutečně každý bit klíče měl na výslednou podobu S-boxu vliv.

Twofish, finalista AES soutěže, je jednou ze šifer nabízených ve standardu OpenPGP [23]. Pomocí softwaru OpenPGP lze podepisovat zprávy, šifrovat a dešifrovat e-mailovou komunikaci, soubory či disk.

2.4 Serpent

Serpent, další účastník AES soutěže, je šifra postavená na principu SPN s 32 rundami, velikostí bloku 128 bitů a třemi možnými délkami klíče: 128, 192 nebo 256 bitů. Jednotlivé S-boxy je možné při šifrování nebo dešifrování paralelizovat, ovšem ani s tímto zrychlením nedosahuje Serpent rychlosti šifry

AES. Vysoký počet rund znemožňuje řadu kryptoanalytických útoků, nicméně redukovaná verze Serpentu se útoku lineární kryptoanalýzou neubránila [3].

2.5 Mars

Mars, finalista AES soutěže vytvořený odborníky z IBM, je šifra Feistelova typu s 32 rundami, velikostí bloku 128 bitů a přípustnými délkami klíče 128, 192 a 256 bitů. Nelinearity představují tyto operace: pronásobování dat, datově závislé rotace a substituce [17]. Díky těmto silným nelinearitám se Mars pro lineární kryptoanalýzu příliš nehodí.

2.6 RC5 a RC6

RC5 je šifra Feistelova typu s velikostní bloku 32/64/128 bitů, variabilní délkou klíče, která dosahuje až 2040 bitů, a 1-255 rundami. Doporučené parametry jsou: 64 bitů blok, 128 bitů klíč a 12 rund. Jediná nelineární transformace RC5 jsou datově závislé rotace.

V roce 2001 byl publikován článek o aplikaci metod lineární a diferenciální kryptoanalýzy na tuto šifru; autoři v něm uvádí, že 12 rund poskytuje dostatečně dobré zabezpečení proti kryptoanalytickým útokům [11]. V roce 2006 byl tento závěr částečně vyvrácen zveřejněním úspěšného útoku diferenciální kryptoanalýzou – autoři této studie doporučují pro posílení šifry používat alespoň 16 rund [5]. V dalším článku z roku 2001, kde jsou lineární kryptoanalýze podrobeny dvě verze šifry RC5 (64 bitů blok, 32 bitů klíč, 10 rund a 128 bitů blok, 64 bitů klíč, 15 rund), autoři usuzují, že jimi popisovaný útok by teoreticky měl tyto dvě verze šifry prolomit. Dále diskutují možné dopady na RC6 (šifru odvozenou z RC5); důsledek prolomitelnosti RC5 neshledávají pro RC6 jako závažný [7].

Šifra RC6 byla do AES soutěže navržena s následujícími parametry: 128 bitů blok, 128/192/256 bitů klíč a 20 rund. Strukturou je velmi podobná RC5; k nelinearitě v podobě datově závislých rotací jsou přidány multiplikativní operace. Jejich cílem je zvýšit závislost rotace na každém bitu a tím zvýšit i potenciální odolnost vůči kryptoanalytickým útokům. Je očekáváno, že kombinaci 18 rund a slabého klíče lze aplikací lineární kryptoanalýzy prolomit [25].

2.7 Skipjack

Skipjack byl vytvořen a používán NSA, k odtajnění došlo v roce 1998. Jedná se o šifru Feistelova typu, kdy otevřený text je na počátku rozdělen na dvě nestejně dlouhé poloviny (tzv. *unbalanced Feistel*). Velikost bloku je 64 bitů, klíč délky 80 bitů a počet rund 32; nevyhovujícím parametrem je příliš krátký

klíč. Na prvních 16 rund šifry byl matematicky popsán útok diferenciální kryptoanalýzou – ovšem neověřený v praxi [8]. Další útok (již na plných 32 rund) odhalil fatální slabinu v šifře: perioda generování rundovních klíčů je konstantní, konkrétně se jedná o číslo 5 [10]. Díky této zranitelnosti je šifra plně prolomena a nedoporučuje se její používání.

2.8 Anubis

Struktura šifry Anubis, účastníka projektu NESSIE, vychází z šifry AES; nejedná se o náhodu, tvůrci Anubise jsou Vincent Rijmen (autor AESu) a Paulo S. L. M. Barreto. Jedná se o SPN s velikostí bloku vždy 128 bitů, délkou klíče čtyř až desetinásobku 32 bitů a počtu rund 12 až 18, přičemž 12 rund přísluší délce klíče 128 bitů, 13 rund klíči o délce 160 bitů, atd. Nelineárním prvkem šifry jsou S-boxy, které byly vytvořeny pseudonáhodně a jejich statistické vlastnosti vyhovují řadě předem stanovených kritérií [1].

Ačkoliv Anubis úspěšně prošel všemi statistickými testy [21] první fáze projektu NESSIE, kde bylo zkoumáno, jestli je jeho chování dostatečně náhodné, nepostoupil do užšího výběru. Jako důvod je v oficiálním dokumentu uvedena značná podobnost se šifrou AES – výhody, které by mohl Anubis před AES nabídnout, nejsou dostačující, aby byl vybrán jako alternativa k AESu [20]. Navíc předběžná zpráva NESSIE projektu upozorňuje na možné nedostatky šifry v oblasti lineární kryptoanalýzy [2].

Zatím nebyla publikována žádná studie, která by aplikovala kryptoanalytické metody na tuto šifru. V literatuře ovšem existuje zmínka, že díky podobnosti s šifrou AES by bylo možné na Anubis útočit podobným způsobem jako na AES [4].

2.9 Shrnutí

Základní poznatky o šifrách popsaných v této kapitole jsou uvedeny v tabulce 2.1. Podrobit lineární kryptoanalýze šifru (ať už v plné či zmenšené verzi), která již byla v minulosti prověřena touto metodou, by víceméně znamenalo zopakovat nějaký již existující postup a ověřit, zda se závěry práce shodují se závěry příslušné studie. Přínosnější by naproti tomu bylo aplikovat lineární kryptoanalýzu na zatím kryptoanalyticky neprověřenou šifru; zde se nabízí čtyři možnosti: Anubis, Blowfish, Mars a Twofish.

Anubis se jeví jako vhodný objekt pro tento účel, proto se tato práce bude dále zabývat aplikací lineární kryptoanalýzy na šifru Anubis. Díky jisté podobnosti se šifrou AES by lineární kryptoanalýza šifry Anubis mohla přinést nový vhled na způsob, jak přistupovat ke kryptoanalýze samotného AESu.

Tabulka 2.1: Srovnání šifer

Šifra	**Typ**	**Nelinearita**	**Aplikována LK**
AES$^{\star\bullet}$	SPN	S-box	ano
Anubis$^{\bullet}$	SPN	S-box	ne
Blowfish	Feistel	klíčově závislý S-box	ne
DES	Feistel	S-box	ano
Mars*	Feistel	násobení, S-box, datově závislé rotace	ne
RC5	Feistel	datově závislé rotace	ano
RC6$^{\star\bullet}$	Feistel	násobení, datově závislé rotace	ano
Serpent*	SPN	S-box	ano
Skipjack	unbalanced Feistel	S-box	ano
Triple DES	Feistel	S-box	ano
Twofish*	Feistel	klíčově závislý S-box	ne

$\star$ soutěž AES NIST, $\bullet$ projekt NESSIE.

Kapitola 3

Anubis

Základní rysy šifry jsou představeny v kapitole zabývající se výběrem vhodné šifry k lineární kryptoanalýze, konkrétně v sekci 2.8. Tato kapitola se zaměřuje na přesný popis šifry a jejích jednotlivých součástí. Použitá terminologie a matematické značení jsou převzaty z oficiálního dokumentu od autorů šifry Anubis [1].

Anubis byl navržen tak, aby všechny použité rundovní transformace měly involuční charakter, tedy byly samy sobě inverzí. To neznamená nic jiného, než že šifrování a dešifrování zajišťuje jedna a tatáž funkce; šifrování se od dešifrování liší pouze ve způsobu, jakým je naloženo s rundovními klíči. Šifrovací a dešifrovací schéma šifry je zobrazeno na obrázku 3.4 a 3.5.

V následujícím textu jsou rozebrány jednotlivé části šifry. V závěru této kapitoly je definován kompletní předpis šifry a předložen důkaz o korektnosti dešifrovacího procesu.

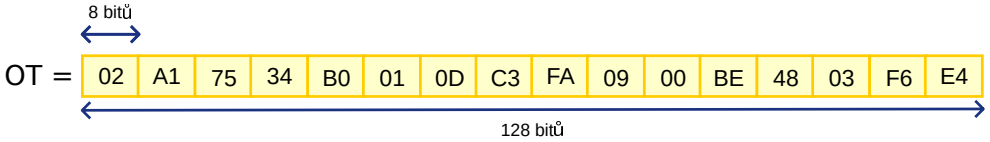
3.1 Vstup a výstup

Vstupem šifry je 128 bitový blok dat, který chceme zašifrovat nebo dešifrovat, a klíč o velikosti $32 \times N$ bitů, kde $4 \leq N \leq 10$. Na vstupní blok můžeme nahlížet jako na 16 za sebou jdoucích 8-bitových hodnot a na klíč jako na posloupnost $4N$ 8-bitových hodnot. Vstupní blok mapujeme do matice 4×4 a šifrovací klíč do matice $N \times 4$ dle následujícího předpisu:

$$\begin{aligned} &\mu : \mathrm{GF}(2^8)^{4N} \to \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)], \\ &\mu(a) = b \iff b_{ij} = a_{4i+j}, \quad 0 \leq i \leq N-1, \quad 0 \leq j \leq 3. \end{aligned} \tag{3.1}$$

Zobrazení μ pro transformaci bitů otevřeného textu do vnitřní maticové formy je demonstrováno obrázkem 3.1. Při daném předpisu je jednoduché si představit, jak vypadá inverzní zobrazení, tj. μ^{-1}, které se používá pro zobrazení z matice zpět do proudu bytů, jež je výstupem šifry.

Obrázek 3.1: Zobrazení μ



μ(OT) =

02	A1	75	34
B0	01	0D	C3
FA	09	00	BE
48	03	F6	E4

3.2 Generování rundovních klíčů

Vstupní šifrovací klíč $K \in \mathrm{GF}(2^8)^{4N}$ je potřeba expandovat na posloupnost rundovních klíčů, které budou přičteny před první rundou a na závěr každé další rundy. Vygenerování rundovních klíčů $K^0, \ldots, K^R \in \mathcal{M}_{4\times 4}[\mathrm{GF}(2^8)]$ zajišťují funkce označené jako **rozvoj klíče** *(key evolution function)* a **selekce klíče** *(key selection function)*.

3.2.1 Rozvoj klíče $\psi[c^r]$

Vstupní šifrovací klíč $K \in \mathrm{GF}(2^8)^{4N}$ je převeden do maticové formy pomocí zobrazení μ, tedy $\kappa^0 = \mu(K)$. Z předklíče κ^0 je následně vytvořena celá sada předklíčů $\kappa^0, \ldots, \kappa^R$. Každý další předklíč je získán z předchozího předklíče aplikací funkce *rozvoje klíče* $\psi[c^r] \equiv \sigma[c^r] \circ \theta \circ \pi \circ \gamma$, tj.

$$\kappa^r = (\sigma[c^r] \circ \theta \circ \pi \circ \gamma)(\kappa^{r-1}), \quad r > 0. \tag{3.2}$$

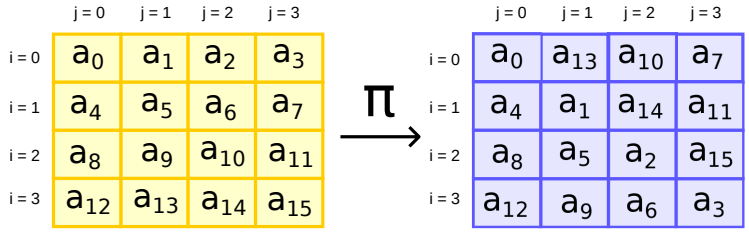
3.2.1.1 Cyklická permutace π

Cyklická permutace π je spolu s operací *přičtení rundovní konstanty* c^r jediná transformace, která figuruje výhradně v *rozvoji klíče* $\psi[c^r]$ a v dalších částech šifry se již nevyskytuje.

Cyklická permutace π (viz obrázek 3.2) je definována jako zobrazení $\mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)]$, jež provádí cyklický posun každého sloupce matice v závislosti na jeho *j-tém* indexu:

$$\pi(a) = b \iff b_{ij} = a_{(i-j) \bmod N, j}, \quad 0 \le i \le N-1, \quad 0 \le j \le 3. \tag{3.3}$$

Obrázek 3.2: Cyklická permutace π



3.2.1.2 Rundovní konstanta c^r

Rundovní konstanta c^r je matice vhodné velikosti s prvním řádkem vygenerovaným pomocí S-boxu; ostatní elementy matice jsou nuly. Tedy *r-tá* rundovní konstanta pro $r > 0$ je matice $c^r \in \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)]$ definovaná následujícím způsobem:

$$\begin{aligned} c^r_{0j} &= S[4(r-1)+j], \quad 0 \le j \le 3, \\ c^r_{ij} &= 0, \quad 1 \le i < N, \quad 0 \le j \le 3. \end{aligned} \tag{3.4}$$

3.2.2 Selekce klíče ϕ

Rundovní klíče jsou získány z předklíčů $\kappa^0, \ldots, \kappa^R$ uplatněním mapování $\phi \equiv \tau \circ \omega \circ \gamma$ na každý předklíč zvlášť. Díky tomu je *selekci klíče* (na rozdíl od *rozvoje klíče*) možné paralelizovat.

$$K^r = (\tau \circ \omega \circ \gamma)(\kappa^r), \quad 0 \le r \le R. \tag{3.5}$$

3.2.2.1 Extrakce klíče ω

Extrakce klíče ω je součástí přípravy rundovních klíčů; matice selekce klíče je zleva vynásobena maticí V se speciálními vlastnostmi.

$$\begin{aligned} V &= \begin{pmatrix} \text{'01'} & \text{'01'} & \text{'01'} & \cdots & \text{'01'} \\ \text{'01'} & \text{'02'} & \text{'02'}^2 & \cdots & \text{'02'}^{N-1} \\ \text{'01'} & \text{'06'} & \text{'06'}^2 & \cdots & \text{'06'}^{N-1} \\ \text{'01'} & \text{'08'} & \text{'08'}^2 & \cdots & \text{'08'}^{N-1} \end{pmatrix}, \\ \omega &: \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{4\times 4}[\mathrm{GF}(2^8)], \\ \omega(a) &= b \iff b = V \cdot a. \end{aligned} \tag{3.6}$$

V je Vandermondova matice *(Vandermonde matrix)* obsahující vybrané prvky $(\text{'01'}, \text{'02'}, \text{'06'}, \text{'08'}) \in \mathrm{GF}(2^8)$. Na každý řádek V se můžeme dívat jako na geometrickou posloupnost.

3.3 Nelineární vrstva γ

Úkolem *nelineární vrstvy* γ je vzít každý prvek matice a provést jeho substituci pomocí S-boxu. Substituce mohou být prováděny paralelně; hodnota prvku po substituci nezávisí na jeho poloze v matici.

$$\begin{aligned} &\gamma : \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)], \\ &\gamma(a) = b \iff b_{ij} = S[a_{ij}], \quad 0 \le i \le N-1, \quad 0 \le j \le 3. \end{aligned} \tag{3.7}$$

S-box je zobrazení $S : \mathrm{GF}(2^8) \to \mathrm{GF}(2^8)$ a pro $\forall x \in \mathrm{GF}(2^8)$ platí:

$$S[S[x]] = x. \tag{3.8}$$

Je zřejmé, že S je involuční zobrazení. Snadno tedy prohlédneme, že požadavek na involučnost splňuje i *nelieární vrstva* γ.

3.3.1 Odvození S-boxu

Substituční box je pseudonáhodně vygenerovaná tabulka o 2^8 prvcích, kde hodnoty, které se v S-boxu objevují, byly vybrány tak, aby splňovaly předem stanovené podmínky a přísná statistická kritéria.

Aby byl význam omezujících podmínek konstrukce S-boxu zcela jasný, je nejdříve potřeba uvést několik definic:

1. **Nelineární řád** v *(nonlinear order)* Booleovské funkce $f : \mathrm{GF}(2)^n \to \mathrm{GF}(2)$ je maximální řád termů, které se vyskytují v její algebraické normální formě[9].

2. **Nelineární řád** v_S S-boxu S je nejmenší nelineární řád, který můžeme libovolnou lineární kombinací jednotlivých složek S vyrobit:

$$\begin{aligned} &v_S = \min_{\alpha\in\mathrm{GF}(2)^n} \{v(l_\alpha \circ S)\}, \\ &l_\alpha : \mathrm{GF}(2)^n \to \mathrm{GF}(2) : \quad l_\alpha(x) = \bigoplus_{i=0}^{n-1} \alpha_i \cdot x_i. \end{aligned}$$

3. **Tabulka diferencí** e *(difference table)* S-boxu S:

$$e_S(a,b) = \#\{c \in \mathrm{GF}(2^n) \mid S[c \oplus a] \oplus S[c] = b\}.$$

4. **Korelace** $c(f,g)$ *(correlation)* dvou Booleovských funkcí f a g:

$$c(f,g) = 2^{1-n} \cdot \#\{x \mid f(x) = f(g)\} - 1.$$

[9] Mějme $m \in \mathbb{N}$ od sebe vzájemně různých Booleovských proměnných. Jejich součin se nazývá součin m-tého řádu. Booleovskou funkci můžeme zapsat právě pomocí sumy součinů m-tého řádu jejich argumentů nad GF(2), $0 \le m \le n$.

5. **Parametr** λ S-boxu S představuje maximální hodnotu korelace mezi vstupním a výstupním mapováním S-boxu S:

$$\lambda_S = \max_{(i,j)\neq(0,0)} c(l_i, l_j \circ S).$$

   Pro současnou podobu S-boxu je parametr $\lambda = 13 \times 2^{-6}$.

6. **Parametr** δ S-boxu S:

$$\delta_S = \frac{1}{e_S(0,0)} \cdot \max_{a\neq 0,b} e_S(a,b).$$

Omezující podmínky konstrukce S-boxu S:

(a) S má involuční strukturu,

(b) v S neexistuje žádný fixní bod, tj. $S[x] \neq x$, $\forall x \in \mathrm{GF}(2^8)$,

(c) pro každou hodnotu diference[10] $x \oplus S[x]$ platí, že nastává přesně dvakrát,

(d) parametr δ nesmí překročit hodnotu 8×2^{-8},

(e) parametr λ nesmí překročit hodnotu 16×2^{-6},

(f) nelineární řád v je maximálně 7.

Omezující podmínky mají zabránit řadě potenciálních zranitelností šifry, které mají souvislost s S-boxy – zejména ztížit lineární, diferenciální či algebraickou kryptoanalýzu.

Generování S-boxu se sestává ze dvou hlavních částí: generování pseudonáhodné sekvence a na jejím základě založené výrobě samotného S-boxu. Jako PRNG[11] je použit generátor velmi podobný generátoru parametrů pro DSA (standard pro digitální podpis).

3.3.1.1 Generování pseudonáhodné sekvence

1. Klíč PRNG je 20-bytové pole k, které je otiskem[12] řetězce m zakódovaného v ASCII.

$$m = \text{'An offering which Anubis gives'}$$
$$k = \text{SHA-1}(m)$$

[10] Rozdíl vstupních a výstupních bitů; s diferencemi mimo jiné operuje *diferenciální kryptoanalýza.*

[11] Generátor takové posloupnosti čísel, která sice není ve skutečnosti náhodná, ale bez znalosti dalších parametrů generátoru se takto produkované informace mohou jevit jako náhodné.

[12] Otisk nebo-li haš *(hash)*, výstup hašovací funkce. V tomto konkrétním případě se jedná o hašovací funkci SHA-1.

2. Mějme dvě 20-bytové pole: $s^{(i)}$ pro uložení stavu a $b^{(i)}$ jako buffer. Inicializujme obě pole na hodnotu 0. Hodnota pole $b^{(i)}$ je určena z klíče k a stavu $x^{(i)}$. Pseudonáhodné byty jsou převzaty z pole $b^{(i)}$, postupuje se od nejpravějšího indexu pole ($i = 19$) k nejlevějšímu ($i = 0$). Počet zbývajících nenaplněných bytů $b^{(i)}$, tedy těch, co byly inicializovány na 0, je sledován.

3. Jakmile počet nenaplněných bytů $b^{(i)}$ klesne na 0, hodnota $b^{(i)}$ a $x^{(i)}$ je aktualizována:

   - $b^{(i+1)} = \text{SHA-1}((k + s^{(i)}) \mod 2^{160})$,
   - $s^{(i+1)} = (s^{(i)} + b^{(i)} + 1) \mod 2^{160}$.

   Takto definovaná pole reprezentují 160-bitové číslo ($20 \times 8 \times 2$) čtené jako little-endian[13], kdy hodnota $x^{(i)}$ je $\sum_{t=0}^{19} 256^t \cdot x^{(i)}[t]$.

3.3.1.2 Generování S-boxu

Na základě pseudonáhodné sekvence získané předchozím algoritmem je vygenerován S-box S:

1. Vygeneruj seznam diferencí p, který obsahuje náhodné permutace všech 255 nenulových 8-bitových hodnot.

2. Všechny vstupy S-boxu S označ jako nedefinované.

3. Každý vzor x S-boxu S je vypočten následujícím způsobem:

   a) Sekvenčně projdi p a vyber první nepoužitou diferenci d takovou, že $S[x \oplus d]$ je označeno jako nedefinované. Pokud tuto podmínku žádná diference nesplňuje, vygeneruj nové p a začni znovu.

   b) Nechť $y = x \oplus d$. Nastav $S[x] = y$ a $S[y] = x$. Diference d je tedy použitá a vstupy $S[x]$ a $S[y]$ přešly z nedefinovaného stavu na definovaný.

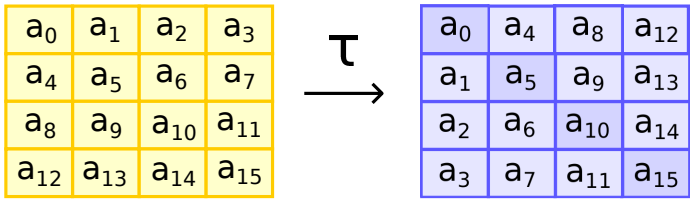
   c) Poslední prvek q z p je použit k zaplnění mezery, tj. $S[q] = 0$ a $S[0] = q$.

Na konci algoritmu obsahuje S pseudonáhodně vygenerované elementy, kdy každý z nich $\in \text{GF}(2^8)$ a S splňuje první tři omezující podmínky (a), (b), (c), definované v sekci 3.3.1. Takto vygenerovaný S-box je třeba otestovat, zda vyhovuje zbylým omezujícím podmínkám. Pokud testy selžou, je potřeba zopakovat celý proces a vygenerovat nový S-box.

Současná podoba S-boxu vzešla z více než 600 miliónů opakovaných běhů tohoto algoritmu, kdy jako nejvhodnější byla vybrána taková podoba S-boxu, která v testech vykazovala nejnižší hodnoty parametrů δ a λ.

[13] Nejméně významný byte (LSB) je přečten z nejnižší adresy, za ním následují další byty až po nejvíce významný byte (MSB).

Obrázek 3.3: Transpozice τ



3.4 Transpozice τ

Transpozice τ je zobrazení, které jednoduše transponuje jednotlivé argumenty matice. Není těžké prohlédnout, že toto zobrazení opět splňuje požadavek involučnosti. Obrázek 3.3 ukazuje, jak *transpozice* τ zrcadlově převrací argumenty matice podle hlavní diagonály.

$$\begin{aligned} &\tau : \mathcal{M}_{4\times4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{4\times4}[\mathrm{GF}(2^8)], \\ &\tau(a) = b \iff b = a^t \iff b_{ij} = a_{ji}, \quad 0 \leq i, j \leq 3. \end{aligned} \tag{3.9}$$

3.5 Lineární difúzní vrstva θ

Lineární difúzní vrstva θ provádí maticové násobení vnitřního stavu šifry s Hadamardovou maticí H *(Hadamard matrix)*. Ze všech vhodných matic H byla vybrána taková matice, jejíž koeficienty měly nejmenší Hammingovu váhu[14]. Pro matici H platí, že je symetrická ($H^T = H$) a unitární ($H^T = H^{-1}$), z čehož plyne splnění požadavku na involučnost pro *difúzní vrstvu* θ.

$$H^T = H^{-1} \iff H \cdot H^T = I \iff H \cdot H = I \tag{3.10}$$

$$H = \begin{pmatrix} \text{'01'} & \text{'02'} & \text{'04'} & \text{'06'} \\ \text{'02'} & \text{'01'} & \text{'06'} & \text{'04'} \\ \text{'04'} & \text{'06'} & \text{'01'} & \text{'02'} \\ \text{'06'} & \text{'04'} & \text{'02'} & \text{'01'} \end{pmatrix} \tag{3.11}$$

Definice *lineární difúzní vrstvy* θ:

$$\begin{aligned} &\theta : \mathcal{M}_{N\times4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{N\times4}[\mathrm{GF}(2^8)], \\ &\theta(a) = b \iff b = a \cdot H. \end{aligned} \tag{3.12}$$

[14]Hammingova váha čísla x je počet jedniček v binárním zápisu čísla x.

3.6 Přičtení klíče $\sigma[k]$

Rundovní operace *přičtění klíče* $\sigma[k]$ očekává na vstupu matici, která reprezentuje vnitřní stav šifry, a aktuální rundovní klíč, reprezentovaný též maticově. Nad prvky se stejným indexem obou matic je provedena operace XOR. *Přičtení klíče* je involuční zobrazení, neboť $(x \oplus x) \oplus x = x$ pro $x \in \mathrm{GF}(2)$.

Definice *přičtení klíče* $\sigma[k]$:

$$\begin{aligned} &\sigma[k] : \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)] \to \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)], \\ &\sigma[k](a) = b \iff b_{ij} = a_{ij} \oplus k_{ij}, \quad 0 \le i \le N-1, \quad 0 \le j \le 3. \end{aligned} \tag{3.13}$$

3.7 Šifrování

Anubis je definován pro šifrovací klíč $K \in \mathrm{GF}(2^8)^{4N}$ jako zobrazení $\mathrm{GF}(2^8)^{16} \to \mathrm{GF}(2^8)^{16}$, které je složeno z těchto transformací:

$$\mathrm{ANUBIS}[K] \equiv \mu^{-1} \circ \alpha_R[K^0, \ldots, K^R] \circ \mu, \tag{3.14}$$

kde

$$\alpha_R[K^0, \ldots, K^R] = \sigma[K^R] \circ \tau \circ \gamma \circ (\circ_1^{r=R-1} \sigma[K^r] \circ \theta \circ \tau \circ \gamma) \circ \sigma[K^0]. \tag{3.15}$$

Jako R je označen počet rund, který je $R = 8+N$ pro $4 \le N \le 10$, přičemž šifrovací klíč je délky $32N$ bitů.

Rundovní funkce je definovaná jako mapování $\rho[K^r] \equiv \sigma[K^r] \circ \theta \circ \tau \circ \gamma$. Poslední runda je oproti plné rundě zkrácena o *lineární difúzní vrstvu* θ; jedná se o mapování $\rho'[K^R] \equiv \sigma[K^r] \circ \tau \circ \gamma$.

3.8 Dešifrování

Dešifrování se liší od šifrování pouze ve způsobu, jak je naloženo s rundovními klíči. Pro důkaz této skutečnosti jsou nejprve potřeba dvě pomocná lemmata.

Lemma 2. $\tau \circ \gamma = \gamma \circ \tau$

Důkaz. Transpozice τ transponuje matici a γ provádí náhradu každého elementu nezávisle na jeho pozici v matici. Z toho plyne nezávislost pořadí těchto operací. □

Lemma 3. $\theta \circ \sigma[K^r] = \sigma[\theta(K^r)] \circ \theta$

Důkaz. Pro $\forall a \in \mathcal{M}_{N\times 4}[\mathrm{GF}(2^8)]$ platí:

$$(\theta \circ \sigma[K^r])(a) = \sigma(K^r \oplus a) = \theta(K^r) \oplus \theta(a) = (\sigma[\theta(K^r)] \circ \theta)(a)$$

□

Tvrzení 1. *Nechť* $\bar{K^0} \equiv K^R$, $\bar{K^R} \equiv K^0$ *a* $\bar{K^r} \equiv \theta(K^{R-r})$ *pro* $0 < r < R$. *Pak* $\alpha_R^{-1}[K^0, \ldots, K^R] = \alpha_R[\bar{K^0}, \ldots, \bar{K^R}]$.

Důkaz. Transformace $\alpha_R[K^0, \ldots, K^R]$ je dle definice:

$$\alpha_R[K^0, \ldots, K^R] = \sigma[K^R] \circ \tau \circ \gamma \circ (\circ_1^{r=R-1} \sigma[K^r] \circ \theta \circ \tau \circ \gamma) \circ \sigma[K^0].$$

Využijme faktu, že $\sigma[k]$, θ, τ a γ jsou involuční. Pokud je poskládáme v opačném pořadí, vytvoříme inverzní transformaci:

$$\alpha_R^{-1}[K^0, \ldots, K^R] = \sigma[K^0] \circ (\circ_1^{r=R-1} \gamma \circ \tau \circ \theta \circ \sigma[K^r]) \circ \gamma \circ \tau \circ \sigma[K^R].$$

Díky Lemma 2 dostáváme:

$$\alpha_R^{-1}[K^0, \ldots, K^R] = \sigma[K^0] \circ (\circ_1^{r=R-1} \tau \circ \gamma \circ \theta \circ \sigma[K^r]) \circ \tau \circ \gamma \circ \sigma[K^R].$$

A aplikace Lemma 3 vede k:

$$\alpha_R^{-1}[K^0, \ldots, K^R] = \sigma[K^0] \circ (\circ_1^{r=R-1} \tau \circ \gamma \circ \sigma[\theta(K^r)] \circ \theta) \circ \tau \circ \gamma \circ \sigma[K^R].$$

Asociativní zákon nám dovoluje přeskládat pořadí operací:

$$\alpha_R^{-1}[K^0, \ldots, K^R] = \sigma[K^0] \circ \tau \circ \gamma \circ (\circ_1^{r=R-1} \sigma[\theta(K^r)] \circ \theta \circ \tau \circ \gamma) \circ \sigma[K^R].$$

Posledním krokem je substituce $\bar{K^r}$:

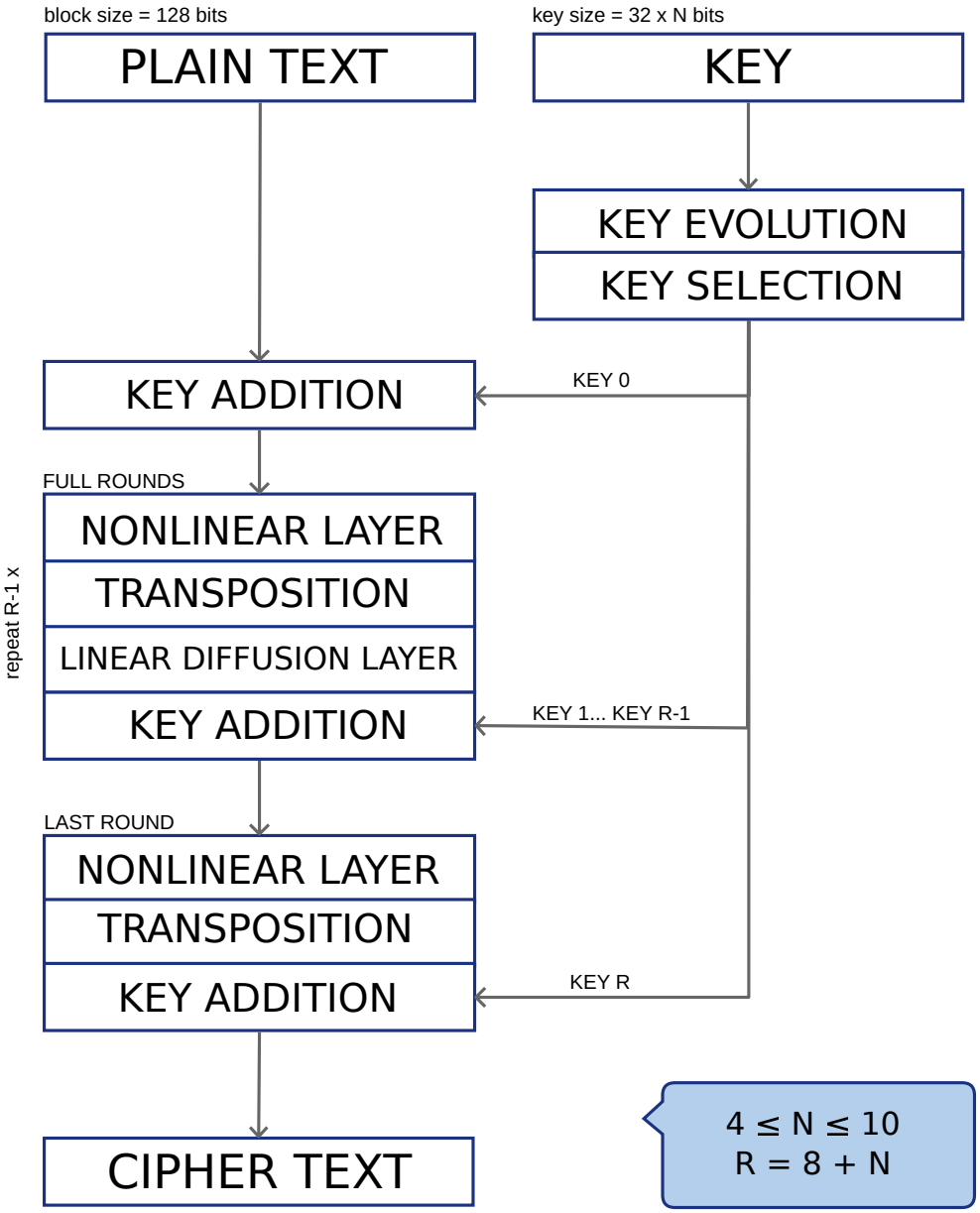
$$\alpha_R^{-1}[K^0, \ldots, K^R] = \sigma[\bar{K^R}] \circ \tau \circ \gamma \circ (\circ_1^{r=R-1} \sigma[\bar{K^r}] \circ \theta \circ \tau \circ \gamma) \circ \sigma[\bar{K^0}].$$

Tedy platí:

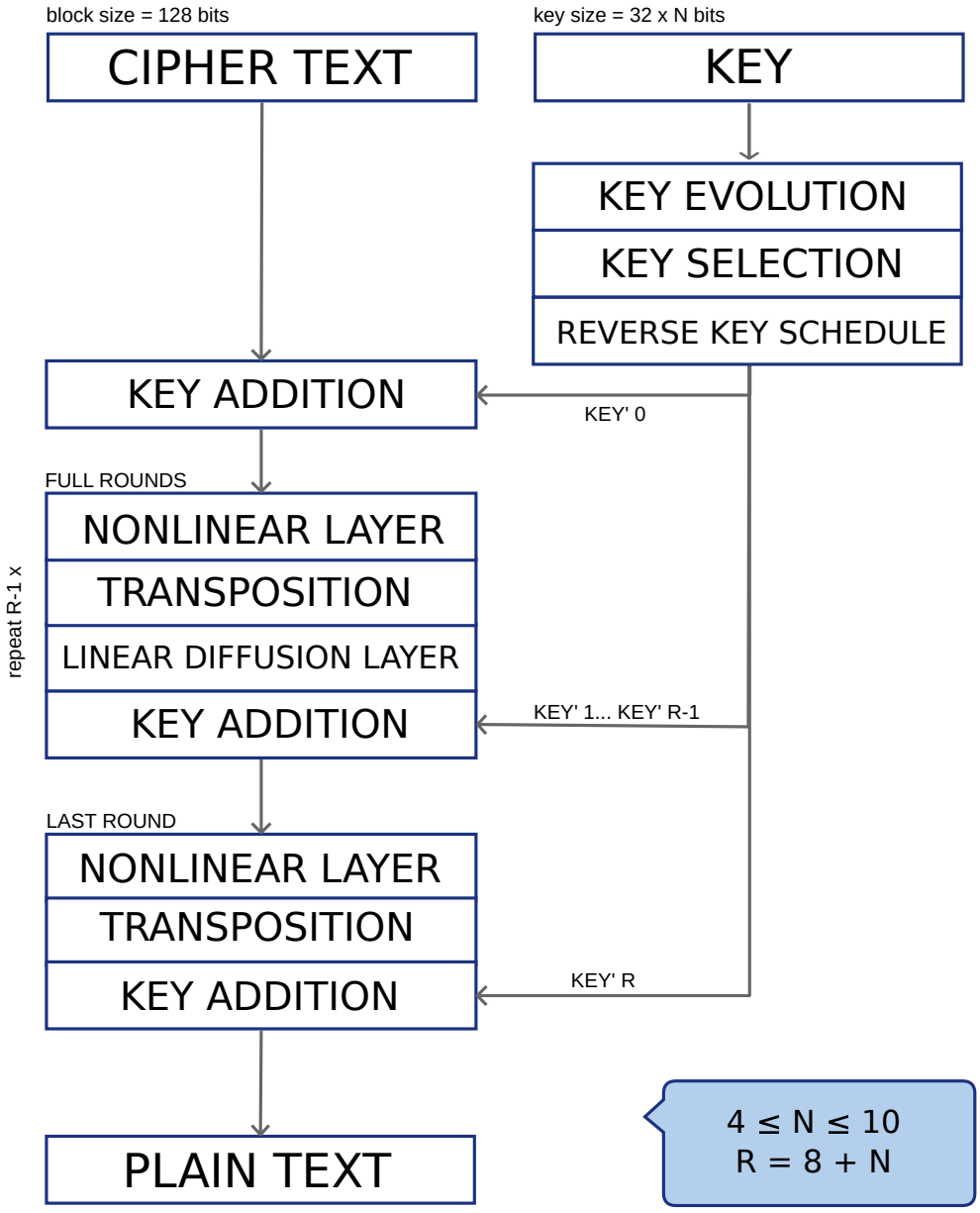
$$\alpha_R^{-1}[K^0, \ldots, K^R] = \alpha_R[\bar{K^0}, \ldots, \bar{K^R}].$$

□

Obrázek 3.4: Anubis - schéma šifrování



Obrázek 3.5: Anubis - schéma dešifrování



Kapitola 4

Baby Anubis

Plná verze šifry Anubis ve své nejstřízlivější variantě se 128-bitovým klíčem přestavuje příliš velké sousto pro lineární kryptoanalýzu. Už jen vygenerování všech 2^{128} otevřených a k nim příslušných šifrových textů by kromě přílišné výpočetní náročnosti otevřelo nový problém – kde takto obrovské množství informací skladovat. Proto se přirozeně nabízí přechod z plné verze Anubise k takovému modelu, který by se neodchýlil od jeho vlastností a z jehož konstrukce by přitom nevyplýval nějaký závažný výpočetní problém.

Existují v zásadě dva základní přístupy, jak by se dalo docílit zmenšení plné verze šifry Anubis:

- **Redukovat počet rund** – tuto variantu lze použít vždy.
- **Proporcionálně zmenšit celou šifru**, tedy redukovat počet bitů vstupu, výstupu a vnitřního stavu šifry. Tato možnost není zaručena automaticky a je ji třeba prověřit.

Díky specifickým vlastnostem šifry přichází v úvahu ještě jeden přístup:

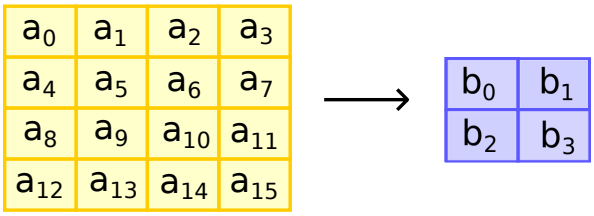
- **Využít pseudonáhodně generovaného S-boxu a přegenerovat si jej** dle svých potřeb vhodně tak, aby stále splňoval omezující podmínky popsané v sekci 3.3.1.

Tvorbě nového S-boxu podle algoritmu ze sekce 3.3.1.1 se nebudu v této práci věnovat, i když se jedná o možný validní krok při vytváření zmenšeného modelu šifry. Šifra s takto přegenerovaným S-boxem (nabízí se varianta 4-bitového vstupu/výstupu) se až příliš odlišuje od původní plné verze.

4.1 Konstrukce zmenšeného modelu

Základní parametry zmenšeného modelu šifry Anubis – Baby Anubis – jsou zobrazeny v tabulce 4.1. Vstup a výstup Baby Anubise je 32-bitový, model

Obrázek 4.1: Baby Anubis - redukce velikosti matice.



Tabulka 4.1: Porovnání plné verze šifry se zmenšeným modelem.

parametry	Anubis	Baby Anubis
velikost bloků [bity]	128	32
velikost klíče [bity]	128	32
rozměr matice vnitřního stavu	4×4	2×2
těleso	$\mathrm{GF}(2^8)$	$\mathrm{GF}(2^8)$
ireducibilní polynom	$x^8 + x^4 + x^3 + x^2 + 1$	$x^8 + x^4 + x^3 + x^2 + 1$

tedy nutně musí operovat s redukovanou velikostí matice vnitřního stavu. Obrázek 4.1 schématicky naznačuje redukci plné verze šifry s maticí 4×4 na zmenšenou verzi pracující s maticí 2×2.

Další text této kapitoly je věnován definici vlastností Baby Anubise[15] a ověření, zda tento zmenšený model odpovídá principům, dle kterých byla vytvořena původní plná verze šifry.

4.2 Vstup a výstup

Mapování $\overline{\mu}$ je pro Baby Anubis upraveno tak, aby stále zobrazovalo vstupní blok dat do matice vnitřního stavu a jeho inverze $\overline{\mu}^{-1}$ převáděla vnitřní stav zpět na výstup:

$$\begin{aligned} &\overline{\mu} : \mathrm{GF}(2^8)^4 \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\mu}(a) = b \iff b_{ij} = a_{2i+j}, \quad i,j \in \{0,1\}. \end{aligned} \tag{4.1}$$

Při porovnání s předpisem μ (3.1) pro plnou verzi šifry je na první pohled zřejmé, že úprava $\overline{\mu}$ je naprosto korektní.

[15] Pro odlišení zmenšeného modelu Baby Anubis od původní šifry Anubis z 3. kapitoly je veškerá použitá matematická notace ve shodě s plnou verzí s tím rozdílem, že význačná mapování a rundovní operace jsou označeny navíc nadtržítkem.

4.3 Generování rundovních klíčů

Definice *rozvoje klíče* $\overline{\psi}[\overline{c^r}]$ a *selekce klíče* $\overline{\phi}$ zůstává pro Baby Anubis prakticky beze změny:

$$\begin{aligned} \overline{\psi}[\overline{c^r}] &\equiv \overline{\sigma}[\overline{c^r}] \circ \overline{\theta} \circ \overline{\pi} \circ \overline{\gamma}, \\ \overline{\phi} &\equiv \overline{\tau} \circ \overline{\omega} \circ \overline{\gamma}. \end{aligned} \tag{4.2}$$

Úprava rundovních operací je rozebírána dále v textu, následuje proto popis a zdůvodnění operací typických pro získání klíčů: *cyklická permutace* $\overline{\pi}$, přičtení *rundovní konstanty* $\overline{c^r}$ a *extrakce klíče* $\overline{\omega}$.

4.3.1 Cyklická permutace $\overline{\pi}$

Adaptace *cyklické permutace* $\overline{\pi}$ na práci se čtvercovou maticí menších rozměrů je jen technická záležitost, korekce definice proběhla pouze na bázi úpravy hodnot indexů i a j:

$$\begin{aligned} &\overline{\pi} : \mathcal{M}_{2\times2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times2}[\mathrm{GF}(2^8)], \\ &\overline{\pi}(a) = b \iff b_{ij} = a_{(i-j) \bmod 2, j}, \quad i, j \in \{0, 1\}. \end{aligned} \tag{4.3}$$

4.3.2 Rundovní konstanta $\overline{c^r}$

Rundovní konstanta $\overline{c^r}$ je matice 2×2 naplněná koeficienty $\in \mathrm{GF}(2^8)$, kde r představuje pořadové číslo rundy:

$$\overline{c^r} = \begin{pmatrix} S[4(r-1)] & S[4(r-1)+1] \\ 0 & 0 \end{pmatrix}, \quad r > 0. \tag{4.4}$$

Při porovnání s původní definicí c^r (3.4) je na první pohled zřejmé, že zmenšená *rundovní konstanta* $\overline{c^r}$ je v pořádku.

4.3.3 Extrakce klíče $\overline{\omega}$

Redukce matice V je netriviální záležitost; nejprve je nutné pečlivě prošetřit, zda šifra používající Vandermondovu matici o rozměrech 2×2 není až příliš zjednodušená. K ověření vhodnosti nové matice $\overline{V}$ proběhly dva na sobě nezávislé běhy statistických testů:

(i) Nová matice $\overline{V}$ byla zakomponována do šifry. Za pomoci pseudonáhodného generátoru[16] [22] bylo vygenerováno 25 000 dvojic otevřeného textu

[16] Třída `Random` není kryptograficky bezpečná, což ale nevylučuje její použití právě na tomto místě – slouží k vytvoření vstupních hodnot šifry. Naopak nevhodné by bylo použít generátor zprostředkovaný třídou `Random` pro získání dat, která jsou citlivá a mají zůstat utajena.

a klíčů. Zašifrováním těchto dat vzniklo 100 000 bytů, které byly jako jeden celistvý datový tok testovány na náhodnost. Tento postup je v souladu s průběhem testování Anubise v rámci projektu NESSIE [21].

K testování jsem použila statistickou testovací sadu NIST [13]. Testy neodhalily žádnou odchylku od náhodného chování.

Při vypuštění matice $\overline{V}$ z procesu generování klíčů a dalším otestování výstupního zašifrovaného proudu bytů dopadly testy náhodnosti opět úspěšně. Zdá se, že *extrakce klíče* $\overline{\omega}$ nemá v případě zmenšené verze šifry zásadní vliv na náhodnost výstupu celé šifry, v případě potřeby by se tedy dala tato část přípravy klíčů u Baby Anubise vynechat.

(ii) Další kolo statistických testů se týkalo *selekce klíčů* $\overline{\phi}$. Stejný pseudonáhodný generátor vygeneroval posloupnost 120 šifrovacích klíčů K. Z každého takového šifrovacího klíče K byly odvozeny rundovní klíče pro sto rund, tedy 400 bytů informací ze sto běhů *rozvoje klíče* $\overline{\psi}$ a *selekce klíče* $\overline{\phi}$, kdy na vstupu byla 4-bytová hodnota K.

Stejná statistická analýza jako v předchozím případě potvrdila, že výstupní informace jsou dostatečně náhodné.

Pro statistická testování byla matice $\overline{V}$ postupně nastavena na tyto hodnoty:

$$\overline{V} \in \left\{ \begin{pmatrix} \text{`01'} & \text{`01'} \\ \text{`01'} & \text{`02'} \end{pmatrix}, \begin{pmatrix} \text{`01'} & \text{`02'} \\ \text{`01'} & \text{`03'} \end{pmatrix}, \begin{pmatrix} \text{`01'} & \text{`02'} \\ \text{`01'} & \text{`04'} \end{pmatrix} \right\}. \tag{4.5}$$

Mezi jednotlivými variantami $\overline{V}$ nebyl objeven žádný zásadní rozdíl, proto jsem jako finální verzi $\overline{V}$ zvolila první matici z výčtu výše, která je zároveň nejpodobnější původní matici V. Nyní je možné zadefinovat zmenšenou *extrakci klíče* $\overline{\omega}$:

$$\begin{aligned} &\overline{\omega} : \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\omega}(a) = b \iff b = \overline{V} \cdot a, \\ &\overline{V} = \begin{pmatrix} \text{`01'} & \text{`01'} \\ \text{`01'} & \text{`02'} \end{pmatrix}. \end{aligned} \tag{4.6}$$

Takto definovaná *extrakce klíče* $\overline{\omega}$ je plně v souladu s principy, pomocí nichž byla zkonstruována ω (3.6).

4.4 Nelineární vrstva $\overline{\gamma}$

S-boxy Baby Anubise zůstavají totožné s S-boxy plné verze šifry. Jediné, co je potřeba mírně upravit, je předpis *nelineární vrstvy* $\overline{\gamma}$:

$$\begin{aligned} &\overline{\gamma} : \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\gamma}(a) = b \iff b_{ij} = S[a_{ij}], \quad i, j \in \{0, 1\}. \end{aligned} \tag{4.7}$$

Jak je vidět, zmenšení *nelineární vrstvy* $\overline{\gamma}$ je naprosto korektní.

4.5 Transpozice $\overline{\tau}$

Úprava *transpozice* $\overline{\tau}$ je opět bezproblémová záležitost, korekce předpisu je triviální:

$$\begin{aligned} &\overline{\tau} : \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\tau}(a) = b \iff b = a^t \iff b_{ij} = a_{ji}, \quad i,j \in \{0,1\}. \end{aligned} \tag{4.8}$$

4.6 Lineární difúzní vrstva $\overline{\theta}$

Pro *lineární difúzní vrstvu* θ je jasně definovaná Hadamardova matice H (3.11), která násobí vnitřní stav šifry zprava. Pro situaci, kdy je matice vnitřního stavu velikosti 2×2, je potřeba najít novou násobící matici $\overline{H}$, jež by vyhovovala svými vlastnostmi.

Počet symetrických a unitárních matic o rozměru 2×2 s prvky $\in \mathrm{GF}(2^8)$ je konečný[17], existuje jich přesně 256:

$$\begin{aligned} &\begin{pmatrix} \text{`00'} & \text{`01'} \\ \text{`01'} & \text{`00'} \end{pmatrix}, \begin{pmatrix} \text{`01'} & \text{`00'} \\ \text{`00'} & \text{`01'} \end{pmatrix}, \begin{pmatrix} \text{`02'} & \text{`03'} \\ \text{`03'} & \text{`02'} \end{pmatrix}, \begin{pmatrix} \text{`03'} & \text{`02'} \\ \text{`02'} & \text{`03'} \end{pmatrix}, \\ &\qquad \ldots, \begin{pmatrix} \text{`FE'} & \text{`FF'} \\ \text{`FF'} & \text{`FE'} \end{pmatrix}, \begin{pmatrix} \text{`FF'} & \text{`FE'} \\ \text{`FE'} & \text{`FF'} \end{pmatrix}. \end{aligned} \tag{4.9}$$

Kompaktnější zápis předchozího výčtu matic:

Pro $\forall \mathcal{M}_{2\times 2}$ platí, že jsou symetrické a unitární, pokud mají tvar

$$\begin{pmatrix} x & x-1 \\ x-1 & x \end{pmatrix} \quad \text{nebo} \quad \begin{pmatrix} x-1 & x \\ x & x-1 \end{pmatrix}, \tag{4.10}$$

kde $x \in \mathrm{GF}(2^8)$ s ireducibilním polynomem $x^8 + x^4 + x^3 + x^2 + 1$ a $x \in \{2k+1\},\ k \in \{\texttt{0}, \texttt{1}, \texttt{2}, \ldots, \texttt{7F}\}$.

Pro Baby Anubis by byla teoreticky vhodná každá z nalezených matic kromě prvních dvou – nulové prvky by dostatečně nerozptýlily informace obsažené v matici vnitřního stavu. Proto jsem jako matici $\overline{H}$ zvolila první vhodnou matici z výčtu výše – s nejnižšími možnými koeficienty a Hammingovou váhou:

$$\overline{H} = \begin{pmatrix} \text{`02'} & \text{`03'} \\ \text{`03'} & \text{`02'} \end{pmatrix}. \tag{4.11}$$

[17] Experimentálně zjištěno vlastním programem, který hrubou silou prohledává celý stavový prostor matic daného rozměru. Tento program je součástí přiloženého média, výstup programu viz `Výsledky/Zmenšený model/H-matrix.txt`.

Kompletní definice *lineární difúzní vrstvy* $\overline{\theta}$ vypadá následovně:

$$\begin{aligned} &\overline{\theta} : \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\theta}(a) = b \iff b = a \cdot \overline{H}. \end{aligned} \tag{4.12}$$

4.7 Přičtení klíče $\overline{\sigma}[k]$

Přičtení klíče $\overline{\sigma}[k]$ zůstalo prakticky beze změny:

$$\begin{aligned} &\overline{\sigma}[k] : \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)] \to \mathcal{M}_{2\times 2}[\mathrm{GF}(2^8)], \\ &\overline{\sigma}[k](a) = b \iff b_{ij} = a_{ij} \oplus k_{ij}, \quad i,j \in \{0,1\}. \end{aligned} \tag{4.13}$$

4.8 Šifrování a dešifrování

Šifrovací a dešifrovací proces Baby Anubise je analogií k témuž procesu plné verze šifry (3.14). Bez újmy na obecnosti lze transformovat důkaz Tvrzení 1 a ukázat tak korektnost dešifrovacího procesu i pro případ Baby Anubise. Pro úplnost zde uvedu definici šifrování:

$$\begin{aligned} &K \in \mathrm{GF}(2^8)^4, \\ &R \in \mathbb{N} \text{ (volitelný počet rund, standardně 12)}, \\ &\mathrm{ANUBIS} : \mathrm{GF}(2^8)^4 \to \mathrm{GF}(2^8)^4, \\ &\mathrm{ANUBIS}[K] \equiv \overline{\mu}^{-1} \circ \overline{\alpha}_R[K^0, \ldots, K^R] \circ \overline{\mu}, \\ &\overline{\alpha}_R[K^0, \ldots, K^R] = \overline{\sigma}[K^R] \circ \overline{\tau} \circ \overline{\gamma} \circ (\circ_1^{r=R-1} \overline{\sigma}[K^r] \circ \overline{\theta} \circ \overline{\tau} \circ \overline{\gamma}) \circ \overline{\sigma}[K^0]. \end{aligned} \tag{4.14}$$

Praktická použitelnost šifry plynoucí z jejího teoretického základu byla ověřena implementačně. Dešifrování již zašifrovaného textu s použitím shodného klíče vedlo na původní otevřený text, tedy:

$$\begin{aligned} &x \ \ldots \text{otevřený text}, \ x \in \mathrm{GF}(2^8)^4, \\ &k \ \ldots \text{(de)šifrovací klíč}, \ k \in \mathrm{GF}(2^8)^4, \\ &\text{pro libovolné } x \text{ a } k \text{ platí, že: } \boxed{D(E(x,k),k) = x.} \end{aligned}$$

Kapitola 5

Lineární kryptoanalýza Baby Anubis

S ověřeným zmenšeným modelem šifry a znalostmi z 1. kapitoly je konečně možné přikročit k samotné lineární kryptoanalýze. Za tímto účelem jsem vytvořila program[18], který automatizuje dílčí kryptoanalytické postupy. Všechny prezentované výsledky v této kapitole jsou získány pomocí tohoto programu.

Na modelovém příkladu ze sekce 1.4 jsem demonstrovala kryptoanalytické metody; abych mohla stejného postupu využít i v případě šifry Baby Anubis, je výhodné upravit její tvar tak, aby svou strukturou odpovídala substituční a permutační síti.

5.1 Baby Anubis jako SPN

Na obrázku 5.1 vidíme Baby Anubis převedený na tvar SPN. Ke čtyřem bytům otevřeného textu ($P_1 - P_8, P_9 - P_{16}, P_{17} - P_{24}, P_{25} - P_{32}$) jsou přičteny čtyři byty klíče ($K_{0,1}, K_{0,2}, K_{0,3}, K_{0,4}$). *Nelineární vrstva* $\overline{\gamma}$ a *transpozice* $\overline{\tau}$ si oproti specifikaci (4.14) vyměnily pořadí – nejdříve se druhý byte prohodí s třetím a až poté oba vstupují do S-boxu ($S_{x,1} - S_{x,4},\ x \in \{1, 2, 3, \ldots, N\}$). Naštěstí díky lemmatu 2 víme, že výměnu $\overline{\gamma}$ s $\overline{\tau}$ můžeme uskutečnit.

Na lineární *difúzní vrstvu* $\overline{\theta}$ (na obrázku označenou písmenem L) lze také v jistém slova smyslu nahlížet jako na speciální případ S-boxu. Jak sám název napovídá, obsahuje tato vrstva pouze takové transformace, které se dají vyjádřit lineárně, což ale samo o sobě není vůbec na překážku. Pro připomenutí, *lineární difúzní vrstva* $\overline{\theta}$ násobí matici na vstupu s maticí $\overline{H}$ (4.11), proto první řádek výsledné matice (v SPN tedy první dva byty) je lineární kombinací složenou z prvních dvou bytů vstupní matice a celé matice $\overline{H}$. Třetí a čtvrtý výstupní byte L jsou lineární kombinací třetího a čtvrtého vstupního bytu s maticí $\overline{H}$.

[18] Program je součástí USB disku, jež je přiložen k této práci.

Výhoda, která plyne z výměny $\overline{\gamma}$ a $\overline{\tau}$, je nasnadě: jsou-li oba druhy S-boxů ($S_{x,1} - S_{x,4}$ a $L_{x,1}, L_{x,2}$) umístěny za sebou, nic nám nebrání v tom je spojit v jeden jediný nelineární prvek s 16-bitovým vstupem a výstupem (spojení je dále označeno jako $S + L$, v matematické notaci pak jako $\overline{\gamma} + \overline{\theta}$)[19].

Na závěr každé rundy se opět po bytech přičte rundovní klíč. Operace prováděné ve všech rundách jsou totožné s výjimkou poslední rundy, kdy dochází k vynechaní lineární difúzní vrstvy $\overline{\theta}$ přesně podle předpisu (4.14).

5.2 Analýza nelineární vrstvy

Zaměřme se nyní na 8-bitové S-boxy, které jsou shodné pro plnou i zmenšenou verzi šifry. Lineární aproximační tabulka získaná analýzou vstupů a výstupů S-boxu má 2^8 řádků a 2^8 sloupců. Kompletní tabulka o 2^{16} prvcích je dostupná na přiloženém médiu. Všechny nalezené odchylky jsou seřazeny v tabulce 5.1, souřadnice nejvyšších odchylek pak v tabulce 5.2. Výsledky analýzy S-boxu jsou v souladu s průzkumem možných slabin šifry v rámci projektu NESSIE [2].

Obrázek 5.2 zachycuje výskyt sedmi nejvyšších odchylek a jejich rozložení v lineární aproximační tabulce. Na ose y jsou naneseny vstupní sumy, na ose x pak výstupní sumy; nicméně díky involučnosti S-boxu je obrázek zrcadlově souměrný podle přímky $y = x$ a význam os je tudíž zaměnitelný.

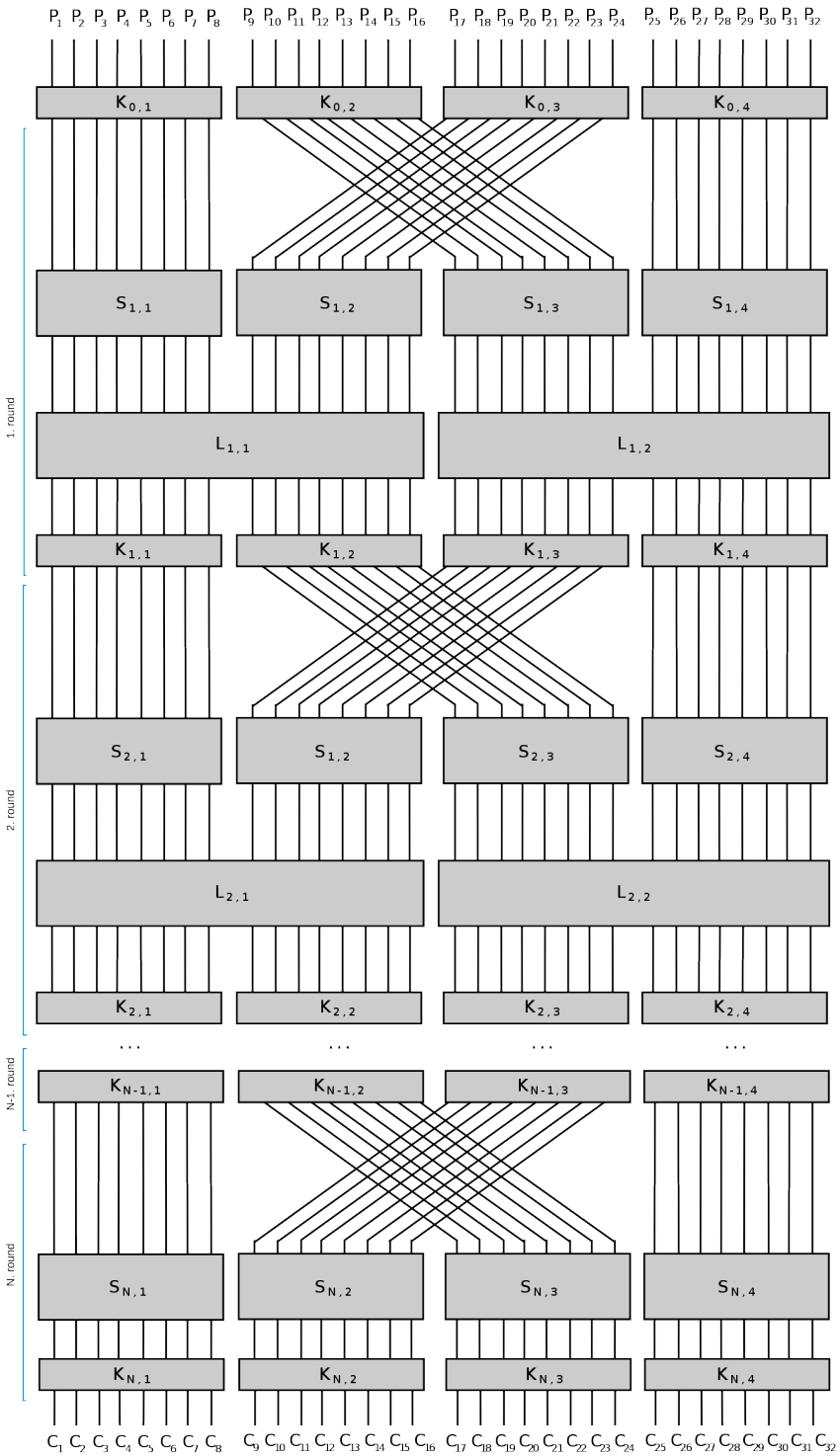
Několik postřehů z analýzy nelineární vrstvy $\overline{\gamma}$

- Lineární aproximační tabulka obsahuje celkově 32 různých hodnot LPB. Kdyby nás nezajímalo znaménko odchylky, rozdílných hodnot je 19.
- Nejvyšší vyskytující se odchylka[20] je $\varepsilon = -\frac{34}{256} = -\frac{17}{128} \approx -0,1328$.
- Odchylky rovné 0 tvoří 10,5% z celkového počtu odchylek.
- Z obrázku 5.2 nelze usoudit, že by některé z nejlepších hodnot LPB tvořily shluky; jejich rozložení se zdá v pořádku.
- Pravidlo *čím lepší odchylka, tím menší je její zastoupení* signalizuje, že S-boxy byly navrženy velmi pečlivě.

[19] Na tento krok přímo navazuje podkapitola 5.3.

[20] Striktně vzato, nejvyšší hodnota odchylky je $+\frac{128}{2^8}$ pro vstupní a výstupní sumu [0,0]. Ta ale není v tuto chvíli zajímavá, protože říká, že příslušné aproximace se neúčastní žádné bity vstupu a výstupu.

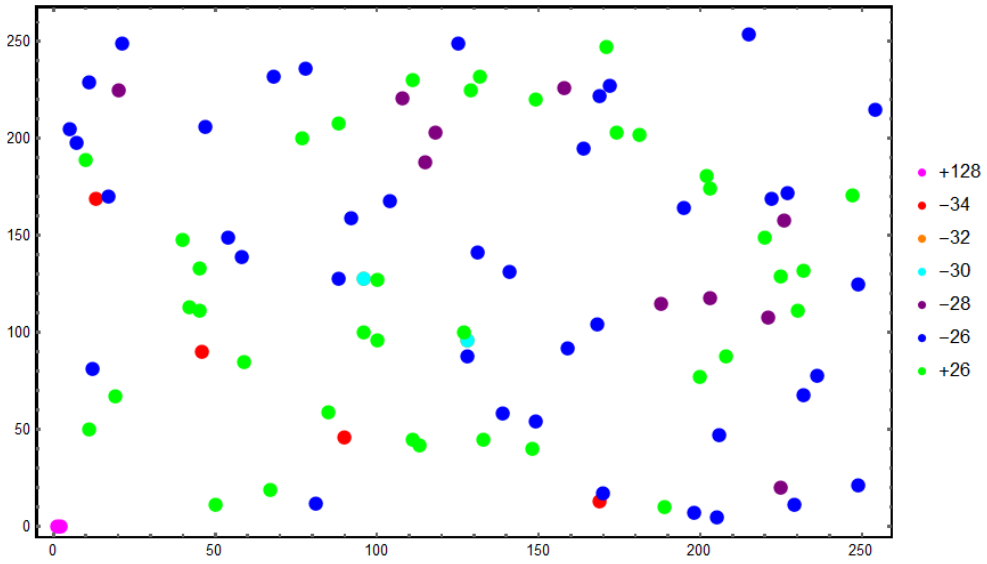
Obrázek 5.1: Baby Anubis ve tvaru SPN



Tabulka 5.1: Anubis – analýza 8-bitového S-Boxu

LPB	počet výskytů
128	1
-34	4
-32	2
-30	2
-28	10
-26	40
26	38
-24	78
24	60
-22	130
22	152
20	282
-20	290
18	510
-18	536
-16	876
16	985
-14	1426
14	1334
-12	2158
12	2177
10	2988
-10	2982
-8	3903
8	3847
6	5074
-6	4904
-4	5691
4	5640
2	6242
-2	6278
0	6896
součet	65536

Obrázek 5.2: Vybrané hodnoty z lineární aproximační tabulky S-Boxu



Tabulka 5.2: Anubis – souřadnice výskytu nejvyšších odchylek

LPB	[InputSum, OutputSum]
-34	[0D, A9], [2E, 5A], [5A, 2E], [A9, D0]
-32	[60, 80], [80, 60]
-30	[1A, D2], [D2, 1A]
-28	[14, E1], [6C, DD], [73, BC], [76, CB], [9E, E2], [BC, 73], [CB, 76], [DD, 6C], [E1, 14], [E2, 9E]

5.3 Analýza sloučení nelineární a lineární difúzní vrstvy

K analýze 16-bitového nelineárního prvku vytvořeného spojením *nelineární* a *lineární difúzní vrstvy* nebylo možné přistupovat tak přímočaře, jako tomu bylo u originálního 8-bitového S-boxu. Limitujícím faktorem se stal výpočetní čas, respektive značné množství výpočtů, které je pro sestavení kompletní lineární aproximační tabulky nezbytné.

5.3.1 Odvození časové a paměťové složitosti

- Pro 2^{16} vstupů X a 2^{16} výstupů Y potřebujeme lineární aproximační tabulku o $2^{32} \approx 4,3$ miliardách buňkách, aby byly postihnuté všechny možné vstupně-výstupní kombinace.

- Stanovení jedné buňky tabulky znamená projít všechny hodnoty X a Y, kterých mohou nabývat.
- Při načtení každé dvojice $x \in X$ a $y \in Y$ je potřeba vyhodnotit příslušné sumy daných bitů pro x a y zvlášť, porovnat výsledky a popřípadě inkrementovat čítač.
- Asymptoticky je pak výpočetní čas pro získání lineární aproximační tabulky ζ přibližně roven

$$T(\zeta) \approx 2 \times 2^{16} \times 2^{32} \times O(k) \approx O(k \cdot 2^{49}),$$
kde k zastupuje operace prováděné s každou dvojicí x a y. (5.1)

- Paměť potřebná k uložení celé aproximační tabulky, tj. 4,3 miliard záznamů uchovaných v 32-bitovém datovém typu `Integer`[21], je

$$32 \times 2^{32}\ \texttt{b} = \frac{2^5}{2^3} \times 2^{32}\ \texttt{B} = 2^2 \times 2^{32}\ \texttt{B} = 2^{34}\ \texttt{B} = 16\ \texttt{GiB}. \tag{5.2}$$

Nejenže je proces výpočtu v domácích podmínkách příliš časově náročný[22], paměťové nároky pro uložení celé lineární aproximační tabulky jsou v řádech gigabytů. Naštěstí existuje způsob, jak časovou a paměťovou náročnost obejít.

Jedinou nelinearitu ukrytou ve sjednocení $\overline{\gamma}+\overline{\theta}$ je již analyzovaný původní 8-bitový S-box. Za dvěma 8-bitovými S-boxy následuje lineární promíchání jejich výstupů realizované maticovým násobením, které původní odchylku zachová nebo zhorší, nikdy však nezlepší. Jednoduchou úvahou dojdeme k závěru, že nejlepší odchylky vrstvy $\overline{\gamma} + \overline{\theta}$ budou jistě výsledkem takových kombinací vstupů, které generují výrazné odchylky v 8-bitovém S-boxu.

Využila jsem proto této myšlenky a vytvořila heuristiku, jež počítá pouze vybrané části lineární aproximační tabulky. Vstupy, jež dávají nejlepší odchylky, jsou známé (viz např. tabulka 5.2). Ty jsou po bytech vzájemně nakombinované a odeslané k provedení dílčích výpočtů.

Pro pět nejvyšších odchylek $(+\frac{128}{256}, -\frac{34}{256}, -\frac{32}{256}, -\frac{30}{256}, -\frac{28}{256})$[23] se analyzuje $n = 19$ různých vstupů, které se ještě navíc dále kombinují mezi sebou (5.3), čímž vznikne $19^2 = 361$ vzájemně různých vstupních sum. Pro každou vstupní sumu program prochází všech 2^{16} výstupních sum a sbírá získané odchylky.

$$\begin{aligned} \texttt{InputSum} &= \alpha\beta,\ \alpha\beta \neq \texttt{0}, \\ \alpha, \beta &\in \{\texttt{00}, \texttt{0D}, \texttt{14}, \texttt{19}, \texttt{1A}, \texttt{2E}, \texttt{5A}, \texttt{60}, \texttt{6C}, \texttt{73}, \\ &\qquad \texttt{76}, \texttt{80}, \texttt{9E}, \texttt{A9}, \texttt{BC}, \texttt{CB}, \texttt{D2}, \texttt{DD}, \texttt{E1}, \texttt{E2}\}. \end{aligned} \tag{5.3}$$

[21] Reálně sice počítáme s 16-bitovými hodnotami, ale v jazyku Java, v němž probíhala implementace, neexistuje žádný 16-bitový typ; nejblíže je právě `Integer`.

[22] Při konfiguraci procesor Intel® Core™ i5-2467M CPU 1,60 GHz, 4GB RAM, Windows 10, NetBeans IDE 8.0.2, Java SE 8 se jeden řádek tabulky (tedy 65 536 položek) počítá ≈ 3 minuty. Vygenerovat celou tabulku pak znamená zaměstnat takový počítač na 136 dní.

[23] Vstupní souřadnice odchylky $+\frac{128}{256}$ je v tomto případě užitečná, protože ji kombinujeme s dalšími již nenulovými 8-bitovými hodnotami.

Počet získaných elementů aproximační tabulky je n^2, je-li na vstupu výpočtu n nejlepších vstupních sum do 8-bitového S-boxu. Časová složitost výpočtu malého úseku ζ' lineární aproximační tabulky je pak:

$$T(\zeta') \approx 2 \times 2^{16} \times n^2 \times 2^{16} \times O(k) \approx O(n^2 \cdot k \cdot 2^{33}). \tag{5.4}$$

Nebude-li n^2 příliš velké, tj. bude se pohybovat v řádu desítek až stovek, snížíme dobu výpočtu na přijatelnou mez. Porovnáním $T(\zeta)$ z rovnice (5.1) s $T(\zeta')$ vidíme, že omezením vstupních hodnot se sníží doba výpočtu[24] právě tolikrát, kolikrát méně generujeme dat. Paměť se při dodržení stejných předpokladů jako v (5.2) zmenší na jednotky až desítky `MiB`.

5.3.2 Zajímavosti z částečné analýzy $\overline{\gamma} + \overline{\theta}$

- Nejlepší nalezená odchylka $\varepsilon = -\frac{8\,704}{2^{16}} = -\frac{8\,704}{65\,536} \approx -0,1328$ se vyskytuje hned 8×, a to na souřadnicích [`000D`, `FD54`], [`002E`, `772D`], [`005A`, `3917`], [`00A9`, `8B86`], [`0D00`, `FD54`], [`2E00`, `772D`], [`5A00`, `3917`] a [`A900`, `8B86`].
- Aby se během výpočtu zbytečně nekumulovaly neužitečné LPB, program eviduje hodnoty odchylek, jež jsou v absolutní hodnotě vyšší nebo rovny 500. Takových bylo nalezeno 149. Pokud by nás nezajímalo jejich znaménko, rozdílných odchylek je pak 81.
- Odchylek v absolutní hodnotě nad 5 000 je 182 a tvoří méně než 1‰ ze všech evidovaných hodnot.

5.4 Útok na poslední rundovní klíč

Na základě dat získaných při částečné analýze $\overline{\gamma} + \overline{\theta}$ je možné realizovat útok na poslední rundovní klíč. Schéma útoku vychází z modelového příkladu z 1. kapitoly.

5.4.1 Baby Anubis ve dvourundové verzi

Nejlepší odchylku, která byla nalezena při analýze $\overline{\gamma} + \overline{\theta}$, lze přímo využít při útoku na Baby Anubis se dvěma rundami. Odchylka $\varepsilon = -\frac{8\,704}{65\,536}$ byla objevena na více místech, zvolila jsem tedy souřadnice [`002E`, `772D`]. Detailně útok popisuje obrázek 5.3. Vybraná lineární aproximace dává do souvislosti bity třetího bytu otevřeného textu, v první rundě prochází 16-bitovým S-boxem $(S+L)_{1,1}$ a končí před vstupem do dvou 8-bitových S-boxů $S_{2,1}$ a $S_{2,3}$. Cílem útoku jsou podklíče $K_{2,1}$ a $K_{2,3}$.

Pro statistickou analýzu úspěšnosti útoku jsem testovala 10 000 náhodných[25] šifrovacích klíčů. Pro každý takový klíč jsem vygenerovala 500 náhodných dvojic (OT, ŠT) a sledovala, jestli se hledaná část rundovního klíče

[24] Z původních 136 dnů jsme nyní na 18 hodinách.

[25] Náhodnost zajistila kryptograficky bezpečná třída `SecureRandom` [24].

Tabulka 5.3: Útok na dvourundový Baby Anubis

Průměrná pozice klíče	1,00030
Nejlepší pozice klíče	1
Nejhorší pozice klíče	2
Směrodatná odchylka	0,01732
Medián	1

vyskytne mezi výsledky, jež algoritmus zmiňovaný v sekci 1.4.3 označí jako pravděpodobný podklíč.

Výsledky útoku[26] shrnuje tabulka 5.3. Dvě rundy šifry Baby Anubis není problém prolomit, dokonce můžeme tvrdit, že správný klíč nalezneme téměř vždy a s poměrně velkou jistotou, že se nemýlíme. Jen ve třech případech označil algoritmus správný klíč až na druhé pozici, což i tak není vůbec špatný výsledek.

5.4.2 Baby Anubis ve třírundové verzi

Útok na tři rundy můžeme rozdělit na dvě varianty:

(a) V první a ve druhé rundě je shodně aktivní jen jeden S-box $\overline{\gamma}+\overline{\theta}$, celkově útok počítá se dvěma aktivními S-boxy $\overline{\gamma}+\overline{\theta}$.

(b) V první rundě jsou aktivní dva S-boxy $\overline{\gamma}+\overline{\theta}$ a jejich výstup je vhodně zvolen tak, aby se vstupoval pouze do jednoho S-boxu $\overline{\gamma}+\overline{\theta}$ ve druhé rundě.

5.4.2.1 Varianta se dvěma aktivními S-boxy

Pro sestavení cesty skrz dvě rundy šifry jsem napsala automatizovaný vyhledávač, jež vybírá průchody s nejlepší celkovou hodnotou odchylky. Cestu vyhledává na základě dat získaných z předchozí analýzy $\overline{\gamma}+\overline{\theta}$.

Algoritmus vyhledávání

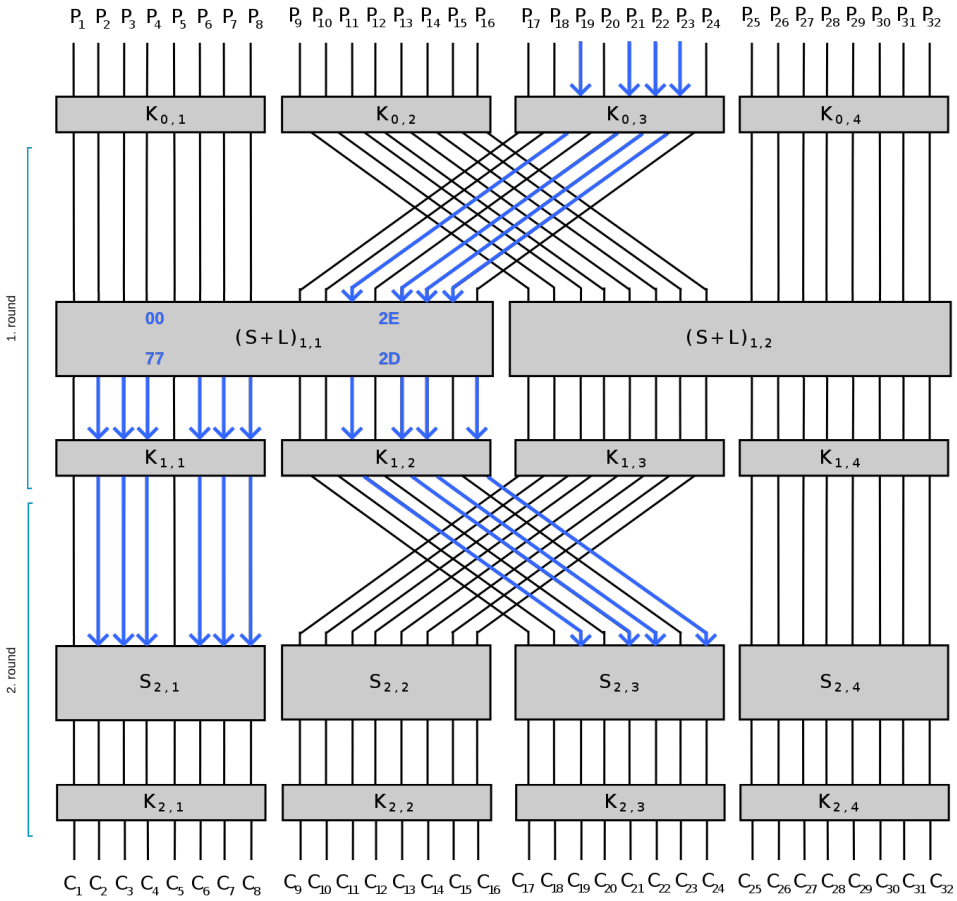
- Vytvoř seznam `InputSums` vstupních sum nejlepších odchylek, které budou prověřovány.
- Pro všechny položky `iSum` $\in$ `InputSums` proveď následující posloupnost kroků:
  - Je-li jeden z bytů `iSum` nulový, projdi všechny výstupní sumy `oSum` $\in$ `OutputSums` každého záznamu. Ty jsou předem seřazeny podle velikosti LPB.

[26] Kompletní data viz `Výsledky/Útok/2-rundy.txt`.

– Nalezneš-li takovou shodu, že `iSum` == `oSum`, ulož cestu, spočítej celkovou odchylku aproximace a pokračuj s hledáním pro další `iSum`.

- Seřaď cesty v sestupném pořadí podle celkové odchylky a zobraz je.

Obrázek 5.3: Lineární aproximace pro dvě rundy



Tabulka 5.4: Útok na třírundový Baby Anubis, 2 aktivní S-boxy

Průměrná pozice klíče	4 910,01
Nejlepší pozice klíče	2
Nejhorší pozice klíče	46 430
Směrodatná odchylka	8 633,45
Medián	1 180

Pro 182 nejlepších jednotlivých aproximací S-boxu $\overline{\gamma} + \overline{\theta}$, které mají LPB v absolutní hodnotě nad $\frac{5\,000}{65\,536}$, objevil algoritmus 620 průchodů. První dva výsledky měly shodnou hodnotu celkové odchylky, vybrala jsem k analýze první z nich, viz též obrázek 5.4:

$$\begin{aligned} &\text{1. runda: } [\texttt{2E80}, \texttt{006C}], \quad \varepsilon_1 = -\frac{1\,496}{65\,536}, \\ &\text{2. runda: } [\texttt{006C}, \texttt{B36E}], \quad \varepsilon_2 = -\frac{7\,168}{65\,536}, \\ &\varepsilon_{1,2} = 2 \cdot \varepsilon_1 \cdot \varepsilon_2 = 4,99343 \times 10^{-3}. \end{aligned} \tag{5.5}$$

Nejdříve jsem k vyhodnocení použila stejný objem vzorků (OT, ŠT) jako v případě dvourundového útoku, tj. 500, což se ale ukázalo jako naprosto nedostatečný počet. Nakonec byl útok vyhodnocen s parametry 2^{16} dvojic (OT, ŠT) pro 100 náhodně vybraných šifrovacích klíčů. Statistická data průběhu útoku[27] s použitím aproximace (5.5) shrnuje tabulka 5.4.

Výsledky lineární kryptoanalýzy nejsou již tak přesvědčivé, jak tomu bylo v případě nižšího počtu rund. V průměrném případě se správný klíč vyskytoval až téměř kolem pozice 5 000, tedy v prvních 7,5% seznamu kandidátních klíčů. Poměrně velká propast mezi hodnotou průměru a mediánu může ukazovat na to, že některé hledané klíče vyhovují dané aproximaci lépe než jiné. Podklíče, jež dosáhly nejlepších pozic, jsem prověřila, ale nenašla jsem mezi nimi žádnou spojitost. Stejně tak jsem učinila i u klíčů, které skončily až v druhé polovině seznamu. Ty byly pouze čtyři (`5794`, `C34F`, `658F`, `97B7`) a výrazně se podílely na zhoršení průměrné pozice klíče. Ovšem i zde se mi nic prokázat nepodařilo.

5.4.2.2 Varianta se třemi aktivními S-boxy

Pro průchod přes tři aktivní S-boxy $\overline{\gamma} + \overline{\theta}$ jsem mírně modifikovala algoritmus na prohledávání cest tak, aby do druhé rundy dosadil aproximaci s vysokým LPB a prohledával všechny ostatní jednobytové výstupní sumy, aby dohromady odpovídaly vstupní sumě vybrané vysoké odchylky.

Pro stejný objem nejlepších odchylek jako v případě aproximace skrz dva S-boxy $\overline{\gamma} + \overline{\theta}$ nebyla žádná cesta označena jako vhodná. Seznam souřadnic postupně dosazovaných do třetí rundy bylo nutné rozšířit až na 3 804 položek.

[27] Viz též `Výsledky/Útok/3-rundy-2-aktivni-Sboxy.txt`.

Tabulka 5.5: Útok na třírundový Baby Anubis, 3 aktivní S-boxy

Průměrná pozice klíče	31 029,20
Nejlepší pozice klíče	114
Nejhorší pozice klíče	63 786
Směrodatná odchylka	18 580,82
Medián	32 706

Z 6 013 nalezených cest popisuje obrázek 5.5 tu nejlepší. Podívejme se blíže na její vlastnosti:

$$\begin{aligned} &\text{1. runda, oba S-boxy shodně: } [\texttt{2E80}, \texttt{006C}], \quad \varepsilon_1 = -\frac{1\,496}{65\,536}, \\ &\text{2. runda: } [\texttt{6C6C}, \texttt{DDDD}], \quad \varepsilon_2 = +\frac{1\,568}{65\,536}, \\ &\varepsilon_{1,2,3} = 2^2 \cdot \varepsilon_1 \cdot \varepsilon_1 \cdot \varepsilon_2 = 4,99868 \times 10^{-5}. \end{aligned} \tag{5.6}$$

Celková odchylka aproximace je dva o řády horší než s jakou počítala varianta se dvěma aktivními S-boxy $\overline{\gamma} + \overline{\theta}$; nedá se proto očekávat, že by útok[28] s využitím slabší aproximace dosáhl lepších výsledků. Při konfiguraci 2^{16} náhodných párů (OT, ŠT) a 100 náhodných šifrovacích klíčů se správný podklíč nacházel v průměru těsně pod hranicí poloviny seznamu kandidátních klíčů, viz tabulka 5.5. Nejlepší zaznamenaná pozice klíče je 114., nejhorší pozice se nachází až na 1 750. místě od konce seznamu.

V tomto případě by se dala lineární kryptoanalýza zhodnotit jako neúspěšná – vybraná aproximace je příliš slabá. Kromě toho tři rundy šifry již poměrně dobře rozptylují informaci o klíči.

5.5 Zhodnocení výsledků

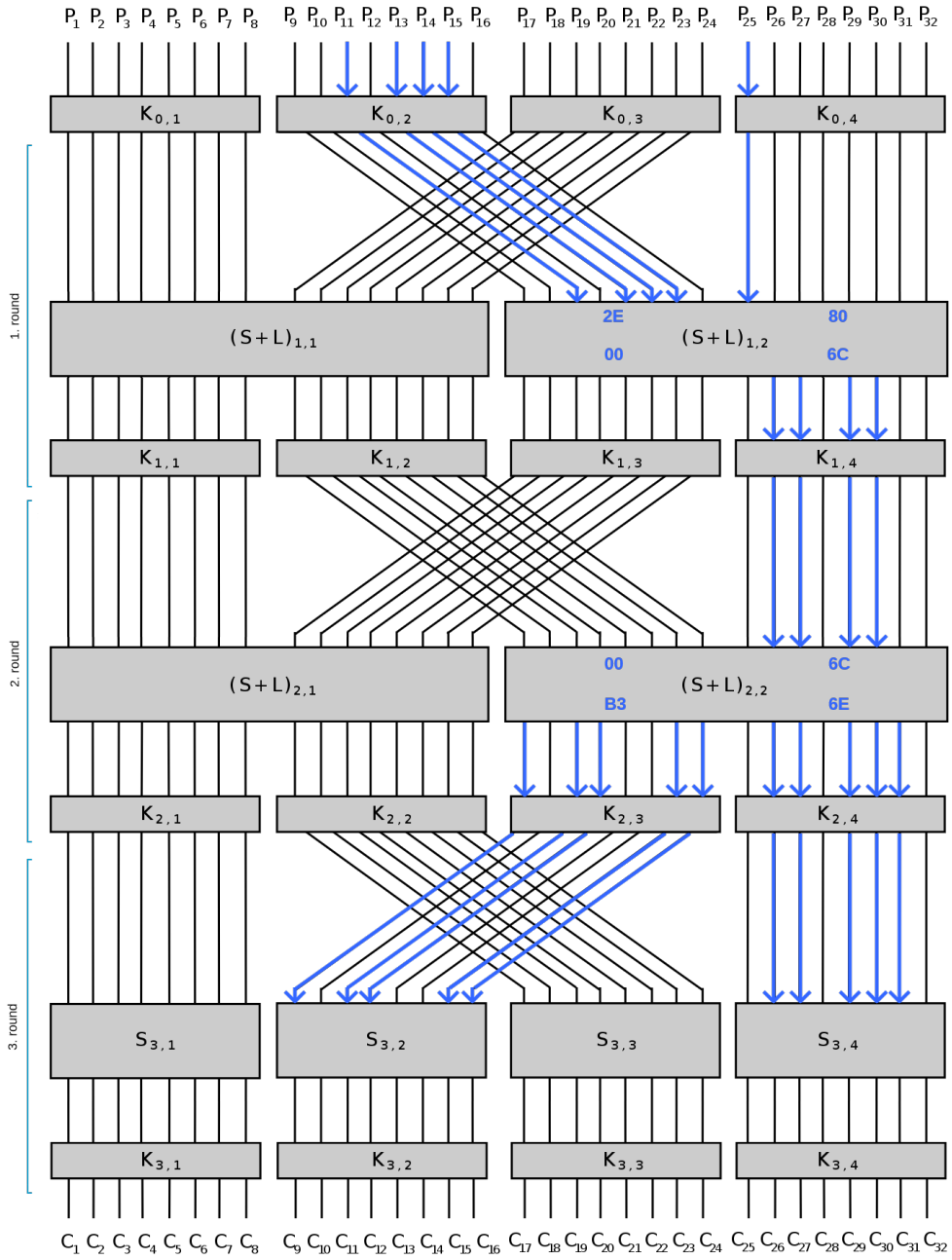
Není pochyb o tom, že na Baby Anubis lze aplikovat techniku lineární kryptoanalýzy. V sekci 5.1 jsem ukázala, jak šifru vhodně upravit tak, aby se snadno prováděly jednotlivé kryptoanalytické postupy. Od analýzy nelineární vrstvy $\overline{\gamma}$ jsem plynule přešla k části 5.3, kde jsem zkoumala možnosti spojení nelineární a lineární difúzní vrstvy. Tento krok se ukázal jako velmi užitečný; na jeho základě jsem sestavila lineární aproximace pro dvou a třírundový útok a realizovala je.

Dvě rundy se mi podařilo úspěšně prolomit, zatímco výsledky pro tři rundy již tak přesvědčivé nejsou. V případě tří rund ale situace není až tak ztracená – vygenerování další vhodné části lineární aproximační tabulky $\overline{\gamma} + \overline{\theta}$, a tím pádem zvětšení prostoru pro hledání cesty skrz šifru, by mohlo vést k lepším výsledkům[29].

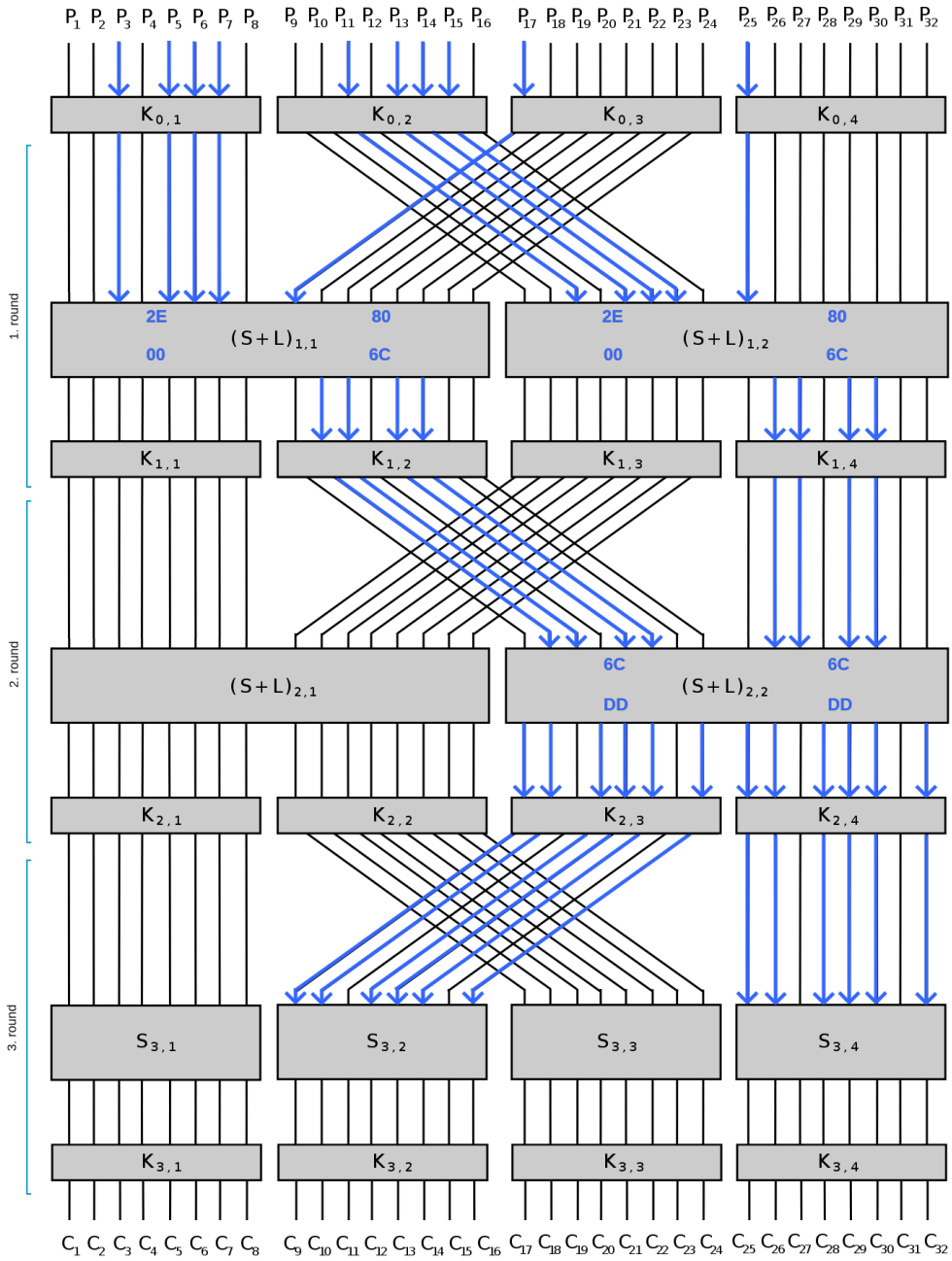
[28] Data útoku viz `Výsledky/Útok/3-rundy-3-aktivni-Sboxy.txt`.

[29] Tato skutečnost by mohla posloužit jako námět k dalšímu pokračování této práce.

Obrázek 5.4: Lineární aproximace pro tři rundy, varianta (a)



Obrázek 5.5: Lineární aproximace pro tři rundy, varianta (b)



5.5.1 Jak aplikovat výsledky na šifru Anubis

Jednotlivé komponenty zmenšeného modelu byly pečlivě voleny tak, aby se co nejvíce blížily šifře Anubis. I když není možné z výpočetních důvodů ověřit dosažené výsledky vyplývající z lineární kryptoanalýzy Baby Anubise, přece jen lze několik závěrů pro plnou verzi šifry učinit:

- Převést Anubis do tvaru SPN lze obdobným způsobem, jako jsem učinila u Baby Anubise. Předpokládejme nejmenší možnou variantu Anubise: 128 bitů blok a 128 bitů klíč. Vstup do lineární difúzní vrstvy θ bude 4-bytový, v jedné rundě se ocitnou na stejné úrovni čtyři lineárně difúzní prvky. Pokud budeme chtít využít triku se spojením $\gamma + \theta$ a analyzovat tento rozšířený S-box, musíme se připravit na exponenciální zhoršení výpočetního času. Lineární aproximační tabulka spojení $\gamma + \theta$ bude v 2^{32} řádcích a 2^{32} sloupcích obsahovat neuvěřitelných $2^{64} \approx 1,84 \times 10^{19}$ hodnot.

- Pokud by se nějakým způsobem podařilo získat data sloučení $\gamma + \theta$ a sestavit lineární aproximaci pro více než dvě rundy, očekávala bych, že výsledky kryptoanalýzy by byly ještě o něco horší, než vyplývá z prezentovaných výsledků pro Baby Anubis. Plná šifra by měla nutně s každou další rundou více aktivních S-boxů $\gamma + \theta$; a jak plyne z piling-up lemma (1), čím více odchylek různých od $\pm\frac{1}{2}$ násobíme, tím se velikost celkové odchylky snižuje. Nižší odchylka aproximace nutně implikuje menší šanci na prolomení šifry.

- Analýza 8-bitového S-boxu z podkapitoly 5.2 je okamžitě aplikovatelná na Anubis, neboť jednou z vlastností Baby Anubise je, že tyto 8-bitové S-boxy nelineární vrstvy jsou shodné s původní šifrou.

Závěr

Cílem této diplomové práce bylo za pomoci metod lineární kryptoanalýzy potvrdit či vyvrátit, zda nevykazuje šifra Anubis jisté anomálie ve svém chování. Šifru jsem důkladně prostudovala, navrhla její zmenšený model a ověřila jej. Na verifikovaný model šifry jsem aplikovala lineárně kryptoanalytické postupy a z výsledků jsem vyvodila dopady pro Anubis.

Na základě provedené analýzy se Anubis jeví jako robustní šifra, ze které její velikost a dobře zvolené jednotlivé komponenty činí poměrně nesnadné sousto pro lineární kryptoanalýzu. Neshledávám žádné důvody, kvůli nimž by bylo možné její používání označit jako riskantní.

Toto tvrzení vychází především z výsledků útoků lineární kryptoanalýzou na redukovaný model šifry Baby Anubis. Opakované útoky, jejichž cílem bylo odhadnout určitou část posledního rundovního klíče, dopadly v případě dvou rund úspěšně; dvourundovou verzi Baby Anubise se podařilo prolomit. U třírundové šifry jsem uvažovala dva případy: verze (a) počítala s lineární aproximací, jež prostupovala přes dva aktivní rozšířené S-boxy, verze (b) využívala aproximace skrz tři aktivní rozšířené S-boxy. Varianta (a) se prolomení šifry přiblížila, správný klíč se průměrně umísťoval v první desetině seznamu všech kandidátních klíčů. Varianta (b) dopadla o poznání hůře, průměrně se správný klíč nacházel až kolem poloviny seznamu. To je nejhorší možný výsledek, jakého lze vůbec dosáhnout, protože podobných hodnot by dosahovaly i náhodné tipy klíče.

Z těchto výsledků je patrné, že více aktivních nelineárních prvků významně snižuje šanci na úspěch útoku. Anubis se svými rozšířenými S-boxy dvojnásobné velikosti oproti zmenšenému modelu, celkově větším výskytem nelinearit a vysokým počtem rund (nejméně 12), je z tohoto důvodu dobře odolný vůči takovým formám útoku.

Literatura

[1] Barreto, P. a Rijmen, V.: *The Anubis Block Cipher.* [online]. [cit. 2015-12-19]. Dostupné z: `http://www.larc.usp.br/~pbarreto/AnubisPage.html`.

[2] Biham, E., Dunkelman, O., Furman, V. a Mor, T.: *Preliminary Report on the NESSIE Submissions: Anubis, Camellia, Khazad, IDEA, Misty1, NIMBUS, and Q.* [online]. [cit. 2015-12-23]. Dostupné z: `https://www.cosic.esat.kuleuven.be/nessie/reports/phase1/tecwp3-011b.pdf`.

[3] Biham, E., Dunkelman, O. a Keller, N.: Linear Cryptanalysis of Reduced Round Serpent. *Lecture Notes in Computer Science*, 2001, č. 2355: s. 16-27.

[4] Biryukov, A.: Analysis of Involutional Ciphers: Khazad and Anubis. *Lecture Notes in Computer Science*, 2003, č. 2887: s. 45-53.

[5] Biryukov, A. a Kushilevitz, E.: Improved cryptanalysis of RC5. *Lecture Notes in Computer Science*, 2006, č. 1403: s. 85-99.

[6] Bizaki, H. K., Mansoori, S. D. a Falahati, A.: Linear Cryptanalysis on Second Round Mini-AES. *Information and Communication Technologies, ICTTA*, ročník 2006, č. 1: s. 1958-1962.

[7] Borst, J., Preneel, B. a Vandewalle, J.: Linear Cryptanalysis of RC5 and RC6. *Lecture Notes in Computer Science*, 2001, č. 1636: s. 16-30.

[8] Granboulan, L.: Flaws in Differential Cryptanalysis of Skipjack. *Lecture Notes in Computer Science*, 2001, č. 2355: s. 328-335.

[9] Heys, H. M.: A Tutorial on Linear and Differential Cryptanalysis. *Technical Report CORR*, 2001, č. 17. [online]. [cit. 2016-3-16]. Dostupné z: `http://www.engr.mun.ca/~howard/PAPERS/ldc_tutorial.pdf`.

[10] Chung-Wei Phan, R.: Cryptanalysis of full Skipjack block cipher. *Electronics Letters*, 2002, č. 38: s. 69-71.

[11] Kaliski, B. S. Jr. a Yin, Y. L.: On Differential and Linear Cryptanalysis of the RC5 Encryption Algorithm. *Lecture Notes in Computer Science*, 2001, č. 963: s. 171-184.

[12] Kokeš, J.: *Kryptoanalýza šifry Baby Rijndael.* ČVUT, 2013, [online]. [cit. 2015-12-14]. Dostupné z: `https://dip.felk.cvut.cz/browse/pdfcache/kokesjo1_2013dipl.pdf`.

[13] Lawrence, E. B. III, Rukhin, A., Soto, J., Nechvatal, J., Smid, M., Barker, E., Leigh, S., Levenson, M., Vangel, M., Banks, D., Heckert, A., Dray, J. a Vo, S.: *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications.* [online]. [cit. 2016-3-11]. Dostupné z: `http://csrc.nist.gov/groups/ST/toolkit/rng/documents/SP800-22rev1a.pdf`.

[14] Lórencz, R.: *Pokročilá kryptologie: Lineární kryptoanalýza.* ČVUT, 2013, [online]. [cit. 2016-3-16]. Dostupné z: `https://edux.fit.cvut.cz/courses/MI-KRY/`.

[15] Lórencz, R.: *Pokročilá kryptologie: Symetrická kryptografie.* ČVUT, 2013, [online]. [cit. 2015-12-13]. Dostupné z: `https://edux.fit.cvut.cz/courses/MI-KRY/`.

[16] Matsui, M.: Linear Cryptanalysis Method for DES Cipher. *Lecture Notes in Computer Science*, ročník 1994, č. 765: s. 386–397.

[17] Matyas, S. M., O'Connor, L., Peyravian, M., Safford, D. a Zunic, N.: *MARS - A Candidate Cipher for AES.* NIST AES Proposal, 1998, [online]. [cit. 2015-12-17]. Dostupné z: `http://www.nada.kth.se/kurser/kth/2D1449/99-00/mars.pdf`.

[18] *NESSIE: Project main goals.* [online]. [cit. 2015-12-13]. Dostupné z: `https://www.cosic.esat.kuleuven.be/nessie/`.

[19] Nie, T. a Zhang, T.: A study of DES and Blowfish encryption algorithm. *TENCON 2009 - 2009 IEEE Region 10 Conference*, ročník 2009, s. 1-4.

[20] Preneel, B., Van Rompay, B., Granboulan, L., Martinet, G., Murphy, S., Shipsey, R., White, J., Dichtl, M., Serf, P., Schafheutle, M., Biham, M., Dunkelman, O., Ciet, M., Quisquater, J-J., Sica, F., Knudsen, L. a Raddum, H.: *NESSIE Phase I: Selection of Primitives.* [online]. [cit. 2015-12-23]. Dostupné z: `https://www.cosic.esat.kuleuven.be/nessie/deliverables/Decision.pdf`.

[21] Raddum, H.: *The Statistical Evaluation of the NESSIE Submission Anubis.* [online]. [cit. 2015-12-23]. Dostupné z: `https://www.cosic.esat.kuleuven.be/nessie/reports/phase1/anubis_report.pdf`.

[22] *Random (Java Platform SE 7).* [online]. [cit. 2016-3-11]. Dostupné z: `https://docs.oracle.com/javase/7/docs/api/java/util/Random.html`.

[23] *RFC 4880 - OpenPGP Message Format.* [online]. [cit. 2015-12-15]. Dostupné z: `https://tools.ietf.org/pdf/rfc4880.pdf`.

[24] *SecureRandom (Java Platform SE 7).* [online]. [cit. 2016-4-21]. Dostupné z: `https://docs.oracle.com/javase/7/docs/api/java/security/SecureRandom.html`.

[25] Shimoyama, T., Takenaka, M. a Koshiba, T.: Multiple Linear Cryptanalysis of a Reduced Round RC6. *Lecture Notes in Computer Science*, 2002, č. 2365: s. 76-88.

Příloha **A**

Seznam použitých zkratek

AES Advanced Encryption Standard

DES Data Encryption Standard

DSA Digital Signature Algorithm

IST Information Society Technologies

LSB Least Significant Bit

MSB Most Significant Bit

NESSIE New European Schemes for Signatures, Integrity and Encryption

NIST National Institute of Standards and Technology

NSA National Security Agency

PGP Pretty Good Privacy

PRNG Pseudo Random Number Generator

RC5 Rivest Cipher 5

SHA Secure Hash Algorithm

SPN Substitution-Permutation Network

Příloha B

Obsah přiloženého USB disku

- `/`
  - `readme.txt` ..... stručný popis obsahu USB
  - `Text`
    - `dp.pdf` ..... text práce ve formátu PDF
    - `src` ..... zdrojové soubory práce ve formátu LaTeX
  - `Literatura` ..... veřejně dostupné materiály, které jsou uvedeny v seznamu Literatura
  - `Implementace` ..... zdrojové kódy v jazyce Java; pro procházení doporučuji použít vývojové prostředí NetBeans
    - `BabyAnubis` ..... implementace šifry Baby Anubis a dílčích kroků lineární kryptoanalýzy
    - `HadamardMatrix` ..... program prohledávající prostor matic 2×2
  - `Výsledky` ..... důležité výstupy z implementační části
    - `Analýza S-boxu`
      - `LAT-8-bit.txt` ..... lineární aproximační tabulka 8-bitového S-boxu, zdroj pro tabulku 5.2 a obrázek 5.2
      - `LAT-16-bit.txt` ..... část lineární aproximační tabulky 16-bitového S-boxu, zdroj dat pro podsekci 5.3.2
      - `LAT-8-bit-list.txt` ..... seznam odchylek nalezených v lineární aproximační tabulce 8-bitového S-boxu, zdroj pro tabulku 5.1
      - `LAT-16-bit-list.txt` ..... seznam odchylek nalezených v lineární aproximační tabulce 16-bitového S-boxu, zdroj dat pro podsekci 5.3.2
    - `Zmenšený model` ..... výpočty nutné k ověření zmenšeného modelu
      - `H-matrix.txt` .... seznam všech nalezených matic vhodných pro použití v lineární difúzní vrstvě
    - `Aproximace` ..... nejlepší nalezené aproximace pro průchod šifrou
      - `3-rundy-aprox-2-Sboxy` ..... zdroj pro obrázek 5.4
      - `3-rundy-aprox-3-Sboxy` ..... zdroj pro obrázek 5.5
    - `Útok` ..... data z útoku LK na daný počet rund šifry ve formátu: seznam testovaných klíčů, hledaných podklíčů a pozice, na níž byla správná část klíče nalezena; na konci souboru je celková statistika útoku
      - `2-rundy` ..... zdroj pro tabulku 5.3
      - `3-rundy-2-aktivni-Sboxy` ..... zdroj pro tabulku 5.4
      - `3-rundy-3-aktivni-Sboxy` ..... zdroj pro tabulku 5.5