VŠB – TECHNICKÁ UNIVERZITA OSTRAVA
FAKULTA ELEKTROTECHNIKY A INFORMATIKY
KATEDRA INFORMATIKY

Lineární a diferenciální kryptoanalýza

Linear and Differential Cryptanalysis

2012 **Daria Stepanova**

VŠB - Technická univerzita Ostrava
Fakulta elektrotechniky a informatiky
Katedra informatiky

Zadání diplomové práce

Student: **Bc. Daria Stepanova**

Studijní program: N2647 Informační a komunikační technologie

Studijní obor: 2612T025 Informatika a výpočetní technika

Téma: Lineární a diferenciální kryptoanalýza
Linear and Differential Cryptanalysis

Zásady pro vypracování:

Pro ověření odolnosti kryptografických algoritmů proti bezpečnostním incidentům se používají různé kryptoanalytické metody. Tato práce se věnuje lineární a diferenciální kryptoanalýze blokových symetrických šifrovacích algoritmů.
1. Seznamte se s typy kryptoanalytických útoků, s klasickými a moderními kryptoanalytickými metodami.
2. Seznamte se s principy lineární kryptoanalýzy.
3. Seznamte se s principy diferenciální kryptoanalýzy.
4. Pro demonstraci těchto kryptonalatyických metod vyberte vhodný algoritmus nebo algoritmy.
5. Navrhněte a naimplementujte aplikaci pro názornou kryptoanalýzu vybraného algoritmu (nebo algoritmů).

Seznam doporučené odborné literatury:

1. Matsui M. and Yamagishi A. "A new method for known plaintext attack of FEAL cipher". Advances in Cryptology - EUROCRYPT 1992.
2. Matsui M. "Linear cryptanalysis method for DES cipher". Advances in Cryptology - EUROCRYPT 1993.
3. Biham E. and Shamir A., "Differential Cryptanalysis of the Data Encryption Standard", Springer Verlag, 1993. ISBN 0-387-97930-1, ISBN 3-540-97930-1.
4. Swenson Ch., "Modern Cryptanalysis: Techniques for Advanced Code Breaking", Wiley 2008, ISBN: 978-0-470-13593-8.
4. Dále dle pokynů vedoucího diplomové práce.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

Vedoucí diplomové práce: **RNDr. Eliška Ochodková, Ph.D.**

Datum zadání: 18.11.2011

Datum odevzdání: 04.05.2012

VYSOKÁ ŠKOLA BÁŇSKÁ-TECHNICKÁ UNIVERZITA Fakulta elektrotechniky a informatiky OSTRAVA

doc. Dr. Ing. Eduard Sojka
vedoucí katedry

prof. RNDr. Václav Snášel, CSc.
děkan fakulty

Prohlášení

Prohlašuji, že jsem tuto diplomovou práci vypracovala samostatně. Uvedla jsem všechny literární prameny a publikace, ze kterých jsem čerpala.

Děkuji své vedoucí diplomové práce RNDr. Elišce Ochodkové, Ph.D za cenné rady, připomínky a metodické vedení práce.

V Ostravě 4. května 2012 Daria Stepanova

Abstrakt

Tématem této diplomové práce je lineární a diferenciální kryptoanalýza. Práce však obecně pojednává také o kryptologii a kryptoanalýze jako takové. Jsou zde podrobně probrány blokové šifrovací algoritmy DES, AES (symetrické šifrování) a jejich zjednodušené verze, dále jsou zde popsány různé metody kryptoanalýzy, druhy útoků na jednotlivé typy šifrování, a samozřejmě jsou také rozebrány výhody a nevýhody blokových šifrovacích algoritmů. Cílem diplomové práce pak zejména bylo podrobně prozkoumat obor lineární a diferenciální kryptoanalýzy, a navrhnout aplikaci, která by demonstrovala a názorně ukázala jednotlivé metody kryptoanalýzy v praxi, tedy jak je možné realizovat dané druhy útoků či metody kryptoanalýzy.

Klíčová slova

Kryptoanalýza, Diferenciální, Lineární, Blokové šifry, DES, AES, S-DES, S-AES, útoky, klíče

Abstract

The main topic of this thesis is the Linear and Differential Cryptanalysis. However, work also deals in general with the cryptology and the cryptanalysis as such. In detail are discussed the block encryption algorithms DES, AES (with symmetric encryption) and their simplified versions, then there are described different methods of the cryptanalysis, types of attacks on different types of encryptions, and of course the advantages and disadvantages of the block encryption algorithms are also discussed. The aim of this thesis was to examine particular field of the linear and differential cryptanalysis in detail, and then design and implement an application for demonstration and illustration of various methods of cryptanalysis in practice.

Keywords

Cryptanalysis, Differential, Linear, Block encryption algorithms, DES, AES, S-DES, S-AES, attacks, keys

Seznam použitých zkratek a symbolů

AES - Advanced Encryption Standard

CMEA – Cellular Message Encryption Algorithm

DECT – Digital Enhanced Cordless Telecommunications

DES - Data (Digital) Encryption Standard

DESX – Symetricky šifrovací algoritmus založeny na základě DES

FIPS – Federal Information Processing Standard

S-AES - Simplified Advanced Encryption Standard

S-DES - Simplified Data (Digital) Encryption Standard

TDEA - Triple Data Encryption Algorithm

XAML – Extensible Application Markup Language

XOR - Exkluzivní disjunkce, vylučovací nebo, exkluzivní OR

WPF – Windows Presentation Foundation

Seznam použitých obrázků

Seznam použitých tabulek

Obsah

1. Úvod

Šifrování a odhalování klíčů je již po staletí oblíbeným oborem, kterému velká část lidí věnovala, a čím dál více věnuje značné úsilí. K čemu kdysi stačila obyčejná tužka a papír, a pro rozluštění lidský rozum, v dnešní době velmi sofistikovaných šifrovacích systémů a algoritmů již pouhé oko a chytrá mysl nestačí. Kryptografie se s příchodem digitálního světa rapidně rozvíjí, a to samozřejmě ruku v ruce s kryptoanalýzou. Tendence chránit data co nejsilnějšími šifrovacími systémy bude vždy na prvním místě, a musí jít s dobou. Jinými slovy, musí se stále rozvíjet s tím, jak rychle roste výpočetní výkon současných výpočetních systémů a jak se zlepšují systémy pro prolomení těchto systémů. Kryptoanalýza však není pouze temná stránka, tedy nemusí jít vždy o piráty či hackery, kteří získat cenné informace, ale také může odhalit slabiny a zásadním způsobem zlepšit šifrovací algoritmy.

Je potřeba rozlišovat mezi pojmy kryptologie, kryptografie a kryptoanalýzou. Kryptografie je věda o šifrování, tedy o metodách utajování informací. Kryptoanalýza, jak bylo již řečeno, je naopak věda o odhalování obsahu šifrovaných zpráv, respektive odhalování tajných klíčů. Kryptologie pak zahrnuje oba pojmy kryptografie a kryptoanalýzy, jedná se tedy o vědu o šifrování obecně.

Typy kryptografických útoků pak dělíme do několika základních skupin, a to podle toho jakou informaci má útočník k dispozici. Útočník může mít například k dispozici pouze zašifrovanou zprávu, nebo může mít k dispozici část otevřeného textu spolu s odpovídajícím zašifrovaným textem atd. Dále pak můžeme typy útoků rozdělit na útoky na analytické úrovni a útoky sociální, kde tajný klíč získáme na základě selhání lidského faktoru (například vydírání, podplácení, podvržení zpráv apod.), kdy nám daná osoba sama vydá příslušnou informaci.

Tématem diplomové práce je lineární a diferenciální kryptoanalýza. Tyto typy kryptoanalýzy úzce souvisí s blokovými šifrovacími algoritmy, tedy se symetrickou kryptografií. Mezi blokové šifrovací algoritmy patří například AES (Advanced Encryption Standard) či DES (Data Encryption Standard). V principu blokové šifrovací algoritmy pracují s bloky pevně stanovené délky, tedy větší množství dat se rozdělí na jednotlivé bloky pevné délky (např. 128 bitů) a zbylé místo v posledním bloku je poté vyplněno. Slabou stránkou blokových šifrovacích algoritmů je pak fakt, že pro šifrování i dešifrování jednotlivých bloků je vždy použit stejný klíč, tento fakt je tedy velkým nedostatkem, který však lze částečně upravit.

Cílem práce pak bylo seznámit se s typy kryptoanalytických útoků, dále pak s klasickými a moderními kryptoanalytickými metodami, konkrétně lineární a

diferenciální kryptoanalýzou. Následně pro demonstraci těchto metod vytvořit vhodnou a přehlednou animaci, která by vhodně z praktického pohledu znázornila princip dané kryptoanalýzy.

V první kapitole je detailně popsán princip blokových šifrovacích algoritmů, zejména algoritmů AES a DES, které budou následně použity pro kryptoanalýzu. V další kapitole jsou podrobně uvedeny jednotlivé typy útoků s příslušnými příklady a také důkladně popsány metody kryptoanalýzy, tedy diferenciální a lineární. Poté následuje kapitola, která popisuje vlastní implementaci aplikace a příslušné zvolené technologie na zjednodušené verzi DES a AES.

2. Blokové šifrovací algoritmy

2.1. Základní terminologie

Otevřený text – otevřeným textem se v kryptografii rozumí text nezašifrovaný, čitelný. Jedná se tedy o vstup algoritmu šifrového.

Šifrový text – text zprávy, který je důsledkem transformace otevřeného textu pomoci šifrovacího algoritmu.

Kryptograficky systém – kompletní integrovaný model, který je schopen produkovat oboustranné transformace nad daty.

Šifrovací klíč – je informace, která určuje průběh kryptografického algoritmu. Při šifrování klíč specifikuje transformaci zprávy do šifrového textu, při dešifrování je tomu přesně naopak.

Kryptoanalýza – je věda zabývající se metodami, cílem kterých je získaní s šifrového textu otevřeny.

Kryptoanalýza se zabývá vyhodnocováním silných a slabých stran šifrovacích metod a také vývojem metod prolomení kryptografických systémů.

Kryptoanalytik – člověk zabývající se kryptoanalýzou.

Kryptograficky útok – výsledek kryptoanalýzy konkrétní šifry.

Šifrovaní – způsob transformace otevřeného textu do šifrovaného pomocí algoritmu.

Dešifrovaní – metoda opačná šifrovaní, tedy převod šifrového textu zpět na otevřený text.

Kryptologie – věda zabývající se metodami šifrovaní a dešifrovaní. Kryptologie se skládá ze dvou véd kryptografie a kryptoanalýzy.

Kryptografie – věda o metodách zjištění integrity a autentičnosti informace. Kryptografie vyvíjí metody šifrovaní dat.

Symetrické kryptografické systémy – algoritmus, který používá pro šifrování i dešifrování jediný klič.

Asymetrické kryptografické systémy - je skupina kryptografických metod, ve kterých se pro šifrování a dešifrování používají odlišné klíče.

Substituční šifra je druh šifry, při které dochází k záměně (substituci) nějaké množiny symbolů za jinou množinu symbolů.

Permutační (transpoziční) algoritmy nemění znaky otevřeného textu, ale mění jejich pořadí.

2.2. DES

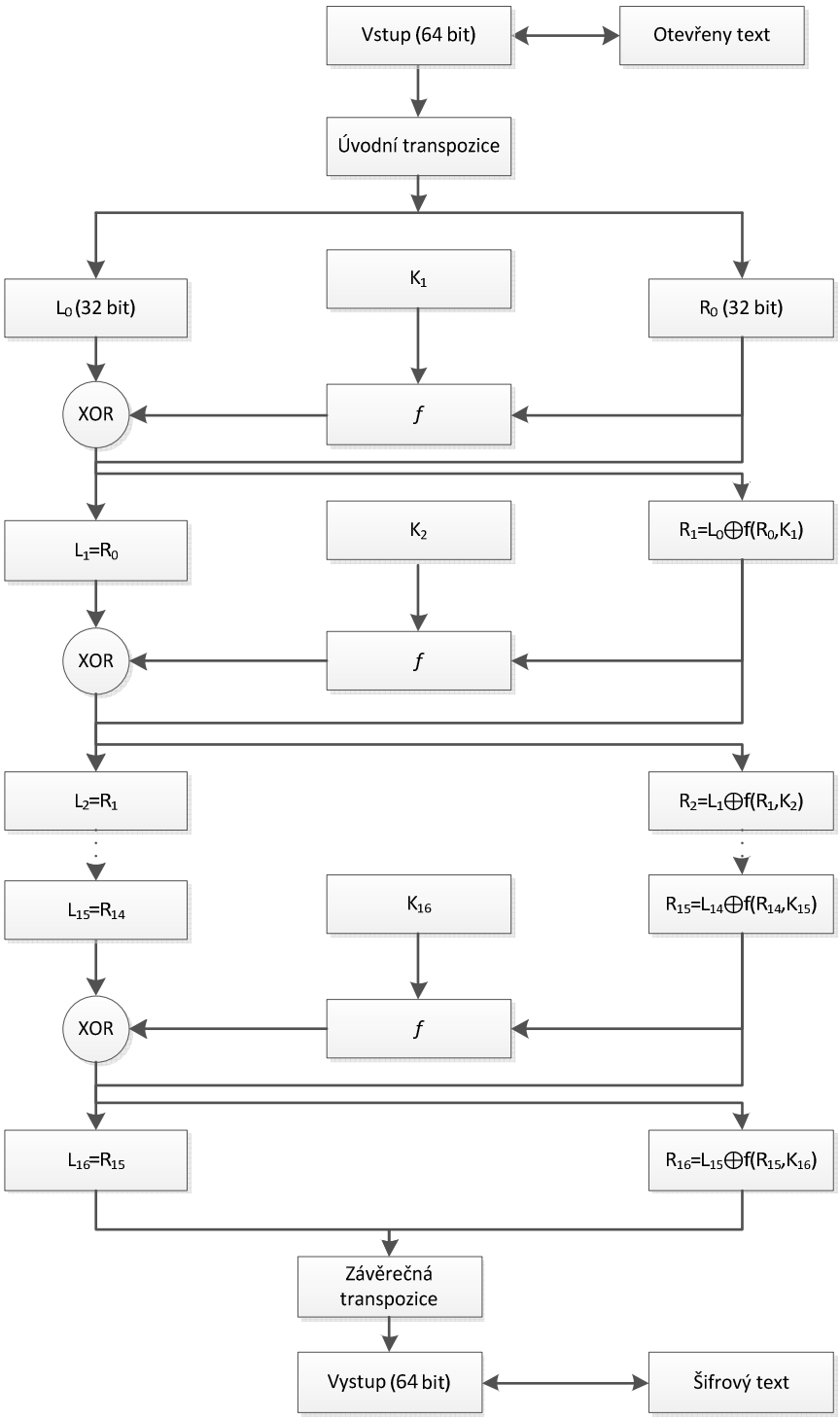
DES (Data Encryption Standard) – symetrický šifrovací algoritmus vyvinutý společností IBM a schválený v roce 1977 americkou vládou jako oficiální standard (FIPS 46-3). DES je blokový algoritmus, který šifruje 64 bitové bloky dat. Do algoritmu vstupuje 64 bitový blok otevřeného textu a vystupuje 64 bitový blok šifrového textu. DES je symetrický algoritmus, tj. pro šifrování a dešifrování používá stejný algoritmus a klič (délka klíčů je 56 bitů). DES se skládá z 16 iterací. Stejná kombinace metod je aplikována na otevřený text 16 krát. Algoritmus používá pouze standardní aritmetiku 64-bitových čísel a logické operace, proto byl snadno realizován v hardwaru již ve druhé polovině 70. let [10].

2.2.1. Schéma algoritmu

Pro popis algoritmu je třeba určit pojem Feistelova síť. Feistelova síť se v kryptografii označuje základní strukturu, která je použitá v mnoha blokových šifrách včetně DES. Kde otevřený text šifrovaného bloku nejprve rozdělí na dvě poloviny a následně se opakuje několik iteraci, při kterých se vždy provede stejná operace[10].

Po první permutace blok se děli na pravou a levou poloviny 32 bitové délky. Pak se provádí 16 iterace stejných operaci, ve kterých se data kombinuji s klíčem. Po 16 iterací se leva a pravá polovina spojuji a algoritmus provádí poslední permutaci.

V každé iteraci se provádí posun bitů klíčů, pak se z 56 bitů vybírá 48 bitů. Pravá polovina dat rozšíří na 48 bitů pomoci expanzní permutace, poté pomocí operace XOR spojí s 48 bity klíče a prochází přes 8 S–boxů. Výstupem bloku je 32 nových bitů, nad kterými se taky provádí permutace. Tyto 4 operace jsou prováděny pomocí funkce f. Poté se výsledek funkce f pomocí operace XOR spojuje s levou polovinou bloku. Tyto kroky se opakuji 16 krát a tvoří 16 iterací DES algoritmu.



Obrázek 1 - Schéma šifrovaní DES

Úvodní permutace (viz Tabulka 1.) se provádí před první iterací. Úvodní a závěrečná permutace nemají vliv na bezpečnost DES.

58	50	42	34	26	18	10	2	60	52	44	36	28	20	12	4
62	54	64	38	30	22	14	6	64	56	48	40	32	24	16	8
57	49	41	33	25	17	9	1	59	51	43	35	27	19	11	3
61	53	45	37	29	21	13	5	63	55	47	39	31	23	15	7

Tabulka 1 - Úvodní permutace

2.2.2. Transformace klíčů

Nejdříve se 64-bitový klíč zmenšuje na 56 bitů pomocí vynechání každého osmého bitu. Tyto bity se používají pro kontrolu parity umožňující kontrolovat správnost klíčů. Po obdržení 56 bitů klíčů pro každou iteraci DES se generuje nový 48-bitový podklíč. Tyto podklíče K_i jsou generovaný následujícím způsobem (viz Tabulka 2.):

57	49	41	33	25	17	9	1	58	50	42	34	26	18
10	2	59	51	43	35	27	19	11	3	60	52	44	36
63	55	47	39	31	23	15	7	62	54	46	38	30	22
14	6	61	53	45	37	29	21	13	5	28	20	12	4

Tabulka 2 - Transformace klíčů

Nejprve se 56-bitový klíč dělí na dvě poloviny o délce 28 bitů. Poté se cyklicky posouvají vlevo o jeden nebo dva bity v závislosti na iteraci (viz Tabulka 3.).

Fáze	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
Číslo	1	1	2	2	2	2	2	2	1	2	2	2	2	2	2	1

Tabulka 3 - Počet bitů pro permutace klíčů

Po permutaci se vybírá 48 bitů z 56. Výsledky této operace jsou představeny v tabulce 4 – komprese.

14	17	11	24	1	5	3	28	15	6	21	10	23	19	12	4
26	8	16	7	27	20	13	2	41	52	31	37	47	55	30	40
51	45	33	48	44	49	39	56	34	53	46	42	50	36	29	32

Tabulka 4 - Komprese

2.2.3. Expanzní permutace

Dana operace rozšiřuje pravou půlku dat R_i z 32 na 48 bitů. Pro každý 4-bitový vstupní blok první a čtvrtý bity reprezentuji dva bity výstupního bloku, druhy a třetí – jeden bit výstupního bloku (viz Tabulka 5.). Přestože výstupní blok je vetší než vstupní, každý vstupní blok generuje unikátní výstupní blok.

32	1	2	3	4	5	6	7	8	9	8	9
8	9	10	11	12	13	12	13	14	15	16	17
16	17	18	19	20	21	20	21	22	23	24	25
24	25	26	27	28	29	28	29	30	31	32	1

Tabulka 5 - Expanzní permutace

2.2.4. Substituce pomoci S-boxů

Po sjednocení bloku po komprese s blokem po operace expanzní permutace pomoci XOR dostaneme 48-bitový výsledek, nad kterým se provádí operace substituce. Substituce se provádí v každém z 8 S-boxů, každý blok obsahuje 6-bitový vstup a 4-bitový vystup.

Každý S-box je tabulka, která se skládá ze 4 řádků a 16 sloupců [6]. Všechny elementy v bloku jsou 4-bitová čísla (viz Tabulka 6. - 13.).

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	14	4	13	1	2	15	11	8	3	10	6	12	5	9	0	7
1	0	15	7	4	14	2	13	1	10	6	12	11	9	5	3	8
2	4	1	14	8	13	6	2	11	15	12	9	7	3	10	5	0
3	15	12	8	2	4	9	1	7	5	11	3	14	10	0	6	13

Tabulka 6 – První S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	15	1	8	14	6	11	3	4	9	7	2	13	12	0	5	10
1	3	13	4	7	15	2	8	14	12	0	1	10	6	9	11	5
2	0	14	7	11	10	4	13	1	5	8	12	6	9	3	2	15
3	13	8	10	1	3	15	4	2	11	6	7	12	0	5	14	9

Tabulka 7 – Druhý S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	10	0	9	14	6	3	15	5	1	13	12	7	11	4	2	8
1	13	7	0	9	3	4	6	10	2	8	5	14	12	11	15	1
2	13	6	4	9	8	15	3	0	11	1	2	12	5	10	14	7
3	1	10	13	0	6	9	8	7	4	15	14	3	11	5	2	12

Tabulka 8 – Třetí S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	7	13	14	3	0	6	9	10	1	2	8	5	11	12	4	15
1	13	8	11	5	6	15	0	3	4	7	2	12	1	10	14	9
2	10	6	9	0	12	11	7	13	15	1	3	14	5	2	8	4
3	3	15	0	6	10	1	13	8	9	4	5	11	12	7	2	14

Tabulka 9 – Čtvrtý S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	2	12	4	1	7	10	11	6	8	5	3	15	13	0	14	9
1	14	11	2	12	4	7	13	1	5	0	15	10	3	9	8	6
2	4	2	1	11	10	13	7	8	15	9	12	5	6	3	0	14
3	11	8	12	7	1	14	2	13	6	15	0	9	10	4	5	3

Tabulka 10 – Pátý S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	12	1	10	15	9	2	6	8	0	13	3	4	15	7	5	11
1	10	15	4	2	7	12	9	5	6	1	13	14	0	11	3	8
2	9	14	15	5	2	8	12	3	7	0	4	10	1	13	11	6
3	4	3	2	12	9	5	15	10	11	14	1	7	6	0	8	13

Tabulka 11 – Šestý S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	4	11	2	14	15	0	8	13	3	12	9	7	5	10	6	1
1	13	0	11	7	5	9	1	10	14	3	5	12	2	15	8	6
2	1	4	11	13	12	3	7	14	10	15	6	8	0	5	9	2
3	6	11	13	8	1	4	10	7	9	5	0	15	14	2	3	12

Tabulka 12 - Sedmý S-box

	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	13	2	8	4	6	15	11	1	10	9	3	14	5	0	12	7
1	1	15	13	8	10	3	7	4	12	5	6	11	0	14	9	2
2	7	11	4	1	9	12	14	2	0	6	10	13	15	3	5	8
3	2	1	14	7	4	10	8	13	15	12	9	0	3	5	6	11

Tabulka 13 – Osmý S-box

Vstupní bity definují prvky S-boxu. Nechť b1, b2, b3, b4, b5 a b6 jsou vstupní bity S-boxu. Bity b1 a b6 se spojuji a tím tvoří dvoubitové číslo od 0 do 3, které odpovídá řádku v tabulce. Prostřední 4 bity b2, b3, b4, b5 tvoří čtyřbitové číslo 0 - 15, které odpovídá sloupcům v tabulce.

Substituce pomocí S–boxů je klíčová fáze DES. Ostatní kroky algoritmu jsou lineární a snadné k analýze. S – boxy jsou nelineární (tj. všichni výstupní bity nejsou lineární kombinacemi vstupních bitů u DES) a proto ve větší míře určuji bezpečnost DES. Výsledkem této iterace je osm 4 bitových bloků, které tvoří jediný 32 bitový blok. Dále tento blok vstupuje do následující permutace pomocí P-boxů.

2.2.5. Permutace pomoci P – boxů

32 bitový výstup substituce pomocí S–boxů se posouvá pomocí P–boxu. Tato permutace přemisťuje každý vstupní bit na jinou pozici (viz Tabulka 14.).

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Tabulka 14 – Permutace P-box

Výsledek dané permutace se pomocí operace XOR spojuje s levou polovinou původního 64-bitového bloku. Poté si levá a pravá část vyměňuje pozice a vstupuji do následující iterace.

2.2.6. Závěrečná permutace

Závěrečná permutace (viz Tabulka 15.) je inverzní k úvodní a provádí se po 16. iteraci.

40	8	48	16	56	24	64	32	39	7	47	15	55	23	63	31
38	6	46	14	54	22	62	30	37	5	45	13	53	21	61	29
36	4	44	12	52	20	60	28	35	3	43	11	51	19	59	27
34	2	42	10	50	18	58	26	33	1	41	9	49	17	47	25

Tabulka 15 – Závěrečná permutace

2.2.7. Dešifrovaní

Dešifrovaní se provádí v opačném pořadí. DES umožnuje používat pro šifrování a dešifrování stejné funkce. Jediný rozdíl spočívá v tom, že klíče musejí být použity v opačném pořadí. Pro dešifrování se klíč posouvá cyklicky doprava a počet bitů posunu se rovná 0, 1, 2, 2, 2, 2, 2, 2, 1, 2, 2, 2, 2, 2, 2, 1.

DES byl do roku 2000 nejpoužívanějším kryptografickým systémem na světě. Proto existovala spousta výzkumů zaměřená na prolomení DES. Byly vyvinuty techniky, které úspěšně napadly DES (např. diferenciální a lineární kryptoanalýza). Dané techniky vyžadují velký výpočetní výkon. Zpočátku byly tyto útoky považovány za nepraktické vzhledem k neexistenci dostatečného výpočetního výkonu. Ale během vývoje technologického pokroku, především s širokým využitím paralelních strojů, DES začal byt méně bezpečným. Tak byl v roce 1999 DES rozšířen na triple-DES, bezpečnější verzi algoritmu [1]. Předpokládá se, že TDEA a AES budou koexistovat jako FIPS (Federal Information Processing Standards). Cílem je poskytnout odolnou kryptografickou bezpečnost pro ochranu citlivých informací v 21. století.

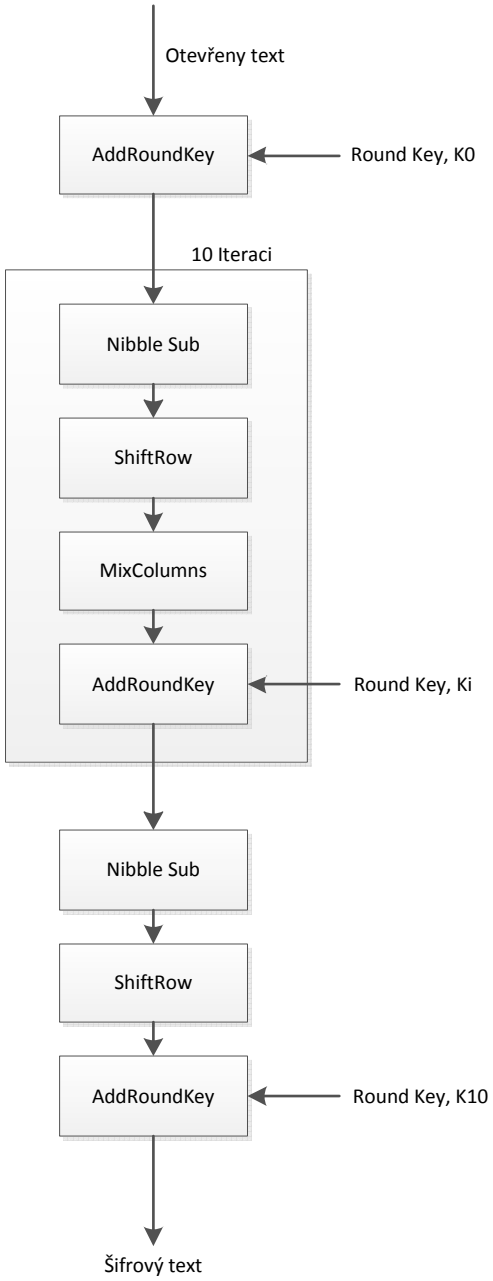
2.3. AES

AES (Advansed Encryption Standard) je symetrický blokový šifrovací algoritmus, přijatý americkou vládou jako šifrovací standard podle výsledku soutěže AES. Tento algoritmus je dobře zkontrolovaný a je nyní široce používán. 26. listopadu 2001 byl AES vyhlášen jako standard symetrického šifrování. Podpora AES byla firmou Intel u procesoru od rodiny x86 (od Core i7 – 980X Extreme Edition, Sandy Bridge). AES na rozdíl od DES nepoužívá Feistelovou síť.

2.3.1. Schéma algoritmu:

AES je standard založený na algoritmu Rijndael. Délka vstupního a výstupního bloku je 128 bitů, délka klíče je 128, 192 nebo 256 bitů. Od délky klíče pak závisí počet

iterací daného algoritmu 10, 12, 14. AES používá pojem stav – přechodný výsledek šifrování. Do stavu se na začátku zapíše všech 128 bitů vstupního bloku.



Obrázek 2 - Schéma šifrovaní AES

Algoritmus používá následující transformace a funkce:

AddRoundKey – transformace používaná během šifrovaní i dešifrování „xoruje“ stav a podklíč odpovídající iterace. Pro každou iteraci je podlíč odvozen z hlavního klíče a má stejnou velikost.

MixColumns – transformace pří šifrovaní, ve které se kombinují 4 bajty v každém sloupci stavu matice a zpracovává každý sloupec jako polynom 4. stupně. Nad těmito polynomy probíhá násobení mod(x^4+1) na polynom c(x) = $3x^3+x^2+x+2$. V kombinaci s ShiftRow poskytuje náhodnost v šifře.

ShiftRows – transformace, která postupně posouvá každý řádek stavu o určitý počet bitů. Nultý řádek zůstává beze změn, první řádek se posouvá o jeden bajt doleva atd.

SubBytes – transformace, která spočívá v substituci bitů S-boxu, pomocí substituční tabulky (viz. Příloha A). Daná transformace zajišťuje nelinearitu šifry.

2.3.2. Dešifrovaní

Dešifrovací algoritmus je potom posloupností odpovídajících inverzních operací uspořádaných v opačném pořadí než při šifrování. Tj. při dešifrování se výhodně využívá vlastnosti linearity některých operací tak, že nakonec lze sestavit ze stejných stavebních prvků, naplněných jinými konstantami (parametry).

Realizované útoky:

V dubnu 2005, Daniel J. Bernstein zveřejnil článek popisující útok (Timing attack), který využívá informace o runtime jednotlivých operací šifrování. Tento útok vyžaduje více než 200 milionů vybraných šifrových textů pro nalezení klíčů [2].

AES dovoluje použití 256-bitových klíčů. Prolomení 256-bitového symetrického klíče brute-force útokem vyžaduje 2^{128} krát vyšší výpočetní výkon než je tomu u 128-bitového klíče. Zařízení, které by mohlo otestovat 10^{18} AES klíčů za sekundu (pokud vůbec je možné, aby takové zařízení existovalo) by teoreticky vyžadovalo okolo $3x10^{51}$ let k vyčerpání 256-bitového klíčového prostoru.

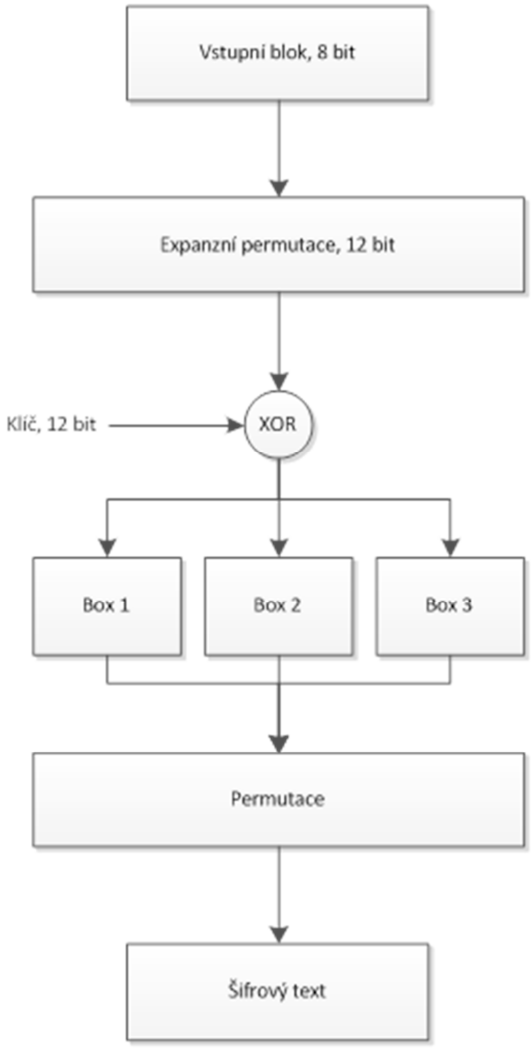
2.4. S-DES

Zjednodušeny DES (S-DES) vyvinut profesorem Univerzity Santa Clara Edwardem Schaeferem a slouží, jako vzdělávací nastroj pro studenty pro šifrování a dešifrovaní s použitím blokových šifru a klíčů s malým počtem bitu [12].

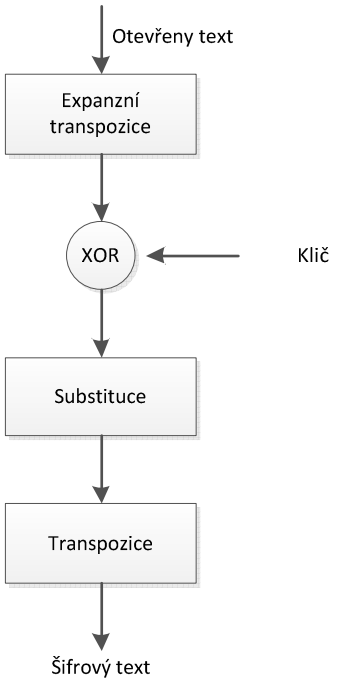
S-DES má podobné vlastnosti jako DES, ale používá mnohem menší bloky a klíče (pracuje z 8 bitovým bloky zprávy a 10 bitovým klíčem). Byl vyvinut jako testovací

blokový šifer pro studium moderních metod kryptoanalýzy jako lineární, diferenciální a lineár – diferenciální kryptoanalýzy.

Struktura S-DES



Obrázek 3 - Podrobnější struktura S-DES



Obrázek 4 - Schéma šifrovaní S-DES

Vstupním blokem je 16-bitová posloupnost, která se rozděluje do dvou částí, na levou a pravou polovinu. Pravá část prochází blokem šifrování a poté se „xoruje“ s levou časti. Šifrová zpráva se skládá ze dvou 2-bitových bloku, pravá část je pravá polovina otevřeného textu, levá část je výsledek XOR operace.

Šifrovací blok dostává na vstup 8-bitový blok, který dále prochází expanzí. Výsledkem daného kroku je 12-bitový text (viz tabulka 16.).

3	4	1	2	6	8	5	7	3	8	2	4

Tabulka 16 – Expanze

Dále probíhá proces „xorování“ 12-bitového textu s 12-bitovým klíčem. Poté se výsledných 12 bitů rozdělí do tří bloků. Vstupem každého bloků je 4 bity, které procházejí odpovídající substituční tabulkou (viz tabulka 17. – 19.). Výstupem prvních dvou bloků 3 bity, výstupem třetího bloků – 2 bity.

Níže představeny substituční tabulky třech bloků:

	000	001	010	011	100	101	110	111
0	4	6	1	3	5	7	2	5
1	5	7	2	4	6	1	3	6

Tabulka 17 – Substituční tabulka prvního bloku

	000	001	010	011	100	101	110	111
0	3	5	7	2	4	6	1	7
1	4	6	1	3	5	7	2	1

Tabulka 18 – Substituční tabulka druhého bloku

	00	01	10	11
00	1	3	2	1
01	2	1	3	2
10	3	2	1	3
11	1	3	2	1

Tabulka 19 – Substituční tabulka třetího bloku

Výstupem dané iterace je 8-bitový blok, který prochází závěrečnou permutace (viz tabulka 20.).

8	7	3	2	5	4	1	6

Tabulka 20 – Substituční tabulka třetího bloku

2.5. S-AES

Zjednodušený AES (S-AES) byl vyvinut profesorem Edwardem Schaeferem z Univerzity Santa Clara a slouží, jako vzdělávací nástroj pro studenty, který pomáhá pochopit strukturu AES s použitím menších bloků a klíče [7].

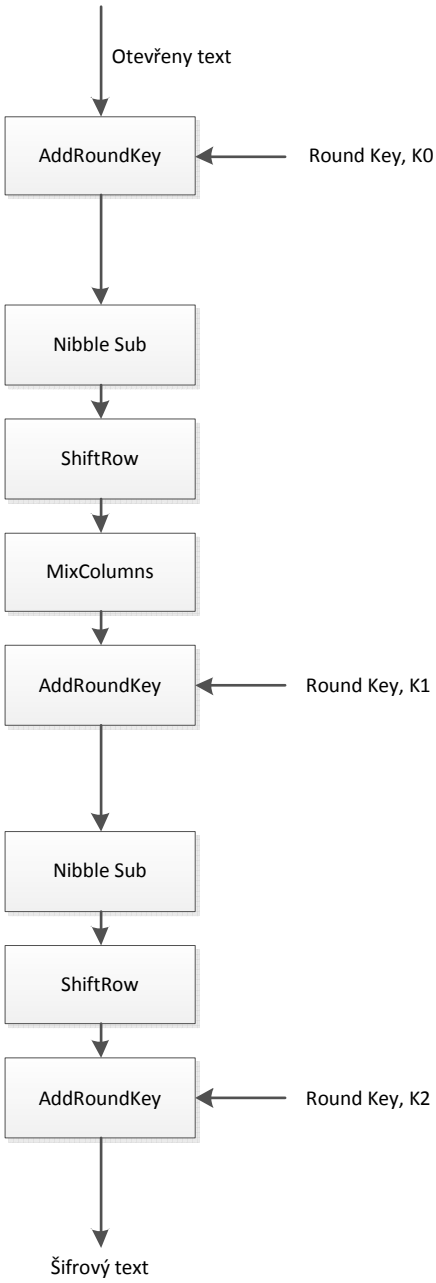
2.5.1. Struktura S-AES

S-AES používá 16-bitový klič k šifrování 16-bitových bloků. S-box je nelineární prvek S-AES, který má na vstupu 4 bity, podle kterých se generuje 4-bitový výstup.

Tabulka číslo 21. zobrazuje výstupní hodnoty pro jednotlivé vstupy.

Vstup	výstup
0000	1110
0001	0100
0010	1101
0011	0001
0100	0010
0101	1111
0110	1011
0111	1000
1000	0011
1001	1010
1010	0110
1011	1100
1100	0101
1101	1001
1110	0000
1111	0111

Tabulka 21 – Substituční tabulka



Obrázek 5 – Schéma šifrovaní S-AES

Substitute nibbles – Bity otevřeného bloku procházejí substituční tabulkou. Výstupem dané transformace je 4-bitový blok.

Shift Rows – Dalším krokem je cyklická rotace prvku. První řádek zůstává beze změn, druhy řádek rotuje o jednu pozici. Výstupem dané transformace je 4 prvková matice.

Mix Columns – V této transformaci se provádí násobení výstupní matice z transformace Shift Rows s konstantní maticí $\begin{bmatrix} 3 & 2 \\ 2 & 3 \end{bmatrix}$.

Add Round Key – Dále na řadu přichází „xorování“ prvku matice z předchozí transformace s klíčem příslušné iterace.

Key Expansion – Daná transformace slouží k expanzi klíče, pomocí které vygenerujeme klíče pro jednotlivé iterace. Celkem zjednodušená verze AES používá 2 iterace k šifrování.

3. Kryptoanalýza

Metoda brute-force (hrubé sily) je jedním z typů útoků. Během daného útoků útočník zkouší všechny možné kombinace klíčů, dokud nebude identifikován správný klíč.

Taky kryptoanalytik muže využit tzv. selhání "lidského faktoru", tj. pokoušet se pomocí vydíraní, podplácení, mučení nebo podobnými způsoby získat informace o systému šifrovaní nebo získat samotný šifrovací klíč.

V tomto případě je metodika odhalení šifry založena na slabosti lidí, jakožto součásti systému bezpečnosti informací. Banditský útok je považován za velmi účinný způsob prolomení systému a mnohokrát za nejlepší cestu odhalení šifry.

Kryptoanalýza je velmi náročná věda, která se zabývá odolností kryptografických systémů. Existuje několik typu útoků, které mohou byt použity k prolomení šifru, v souvislosti s tím, kolik informací má útočník k dispozici. Cílem kryptoanalýzy je získat klič, pomocí kterého rozluštíme skryté informace. Kryptografické útoky založené na metodě hrubé sily jsou nejefektivnější, ale mají velkou časovou náročnost. V tabulce číslo 22. představeny výsledky časové složitosti daného útoku. Předpokládá se, že heslo muže obsahovat 36 symbolu, během sekundy se zkouší 100 000 hesel [13].

Počet symbolu	Počet kombinace	Odolnost	Časová náročnost
1	36	5 bitů	Méně než sekundy
2	1296	10 bitů	Méně než sekundy
3	46 656	15 bitů	Méně než sekundy
4	1 679 616	21 bitů	17 sekund
5	60 466 176	26 bitů	10 minut
6	2 176 782 336	31 bitů	6 hodin
7	78 364 164 096	36 bitů	9 dní
8	$2{,}821\ 109\ 9 \times 10^{12}$	41 bitů	11 měsíců
9	$1{,}015\ 599\ 5 \times 10^{15}$	46 bitů	32 let
10	$3{,}656\ 158\ 4 \times 10^{15}$	52 bitů	1 162 let
11	$1{,}316\ 217\ 0 \times 10^{17}$	58 bitů	41 823 let
12	$4{,}738\ 381\ 3 \times 10^{18}$	62 bitů	1 505 615 let

Tabulka 22 – Časová složitost metody brute-force

3.1. Metody kryptoanalýzy

Klasická kryptoanalýza:

- Frekvenční analýza (Frequency analysis)
- Index shody okolnosti (Index of coincidence)
- Kasiského test (Kasiski examination)
- Útok hrubou silou (Brute-force attack)

Symetrické algoritmy:

- Útok bumerang (Boomerang attack)
- Útok Daviesův (Davies' attack)
- Diferenciální kryptoanalýza (Differential cryptanalysis)
- Neuskutečnitelná diferenciální kryptoanalýza (Impossible differential cryptanalysis)
- Zkrácena diferenciální kryptoanalýza (Truncated differential cryptanalysis)
- Integrální kryptoanalýza (Integral cryptanalysis)
- Lineární kryptoanalýza (Linear cryptanalysis)
- Útok meet in the middle (Meet-in-the-middle attack)
- (mod n) kryptoanalýza ((mod n) cryptanalysis)
- Útok related key (Related-key attack)
- Slide útok (Slide attack)
- XSL útok (XSL attack)

Hash funkce:

- Narozeninový útok (Birthday attack)
- Duhova tabulka (Rainbow table)

Sítové útoky:

- Útok MITM (Man in the middle)
- Útok Replay (Replay attack)

Vnější útoky:

- Kryptoanalýza Rubber-hose (Rubber-hose cryptanalysis)
- Kryptoanalýza Black-bag (Black-bag cryptanalysis)

3.1.1. Klasická kryptoanalýza

Ačkoli pojem kryptoanalýza byl zaveden poměrně nedávno, některé metody odhalení byly vynalezeny již před desítkami století. „Rukopis o dešifrovaní kryptografických zpráv“ jsou první známé písemné poznámky o kryptoanalýze, napsané arabským vědcem Al-Kindi v 9 století. Tato vědecká práce obsahuje popis metod frekvenční kryptoanalýzy [9].

Frekvenční analýza – základní nástroj pro odhalení většiny klasických permutačních a substitučních šifer. Tato metoda je založena na předpokladu existence netriviální statistické rozložení přirozeného jazyku a současny výskyt v otevřeném a šifrovém textu.

Metoda frekvenční analýzy známa z 9. století, přestože její nejznámější aplikace je dešifrovaní egyptských hieroglyfů Jean-François Champollionem v roce 1922. Od poloviny 20. století většina používaných algoritmů šifrování byla posílena oproti frekvenční analýze, a proto se tato kryptoanalýza používá spíše pro vzdělávací účely.

Během 15.-16. století v Evropě vznikaly a vyvíjely se monoalfabetické šifry. Nejznámější z nich je šifra francouzského diplomata Blaise de Vigenère, která je založena na využití sekvence několika Caesarových šifer s různými hodnotami posunu. Během třech století byla Vigenèrova šifra považována za zcela kryptograficky silnou, dokud v roce 1863 Frederick Kasiski nenabídl svůj vlastní způsob prolomení této šifry. Základní myšlenka metody Kasiski byla následující: Jestliže v otevřeném textu mezi stejnými znakovými segmenty je takový blok textu, že jeho délka je násobkem délky klíčového slova, pak tyto stejné segmenty symbolu otevřeného textu během šifrovaní půjdou do stejných segmentů šifrovaného textu. V praxi to znamená, že v případě existence stejných segmentů o délce tři a více znaků v šifrovaném textu, je pravděpodobné, že tyto segmenty odpovídají stejným segmentům otevřeného textu.

Další fáze vývoje kryptoanalýzy je spojena s vynálezem válcových šifrovacích strojů, jako například Enigma (vynalezena Arturem Scherbiusem). Účelem těchto zařízení bylo minimalizovat počet opakovaných segmentů šifrovaného textu. Kryptoanalytikům z Polska se podařilo postavit prototyp dešifrovacího stroje pro verzi Enigmy. Mechanizmus dostal název „Bomba“ proto, že během výkonu vydávala zvuky podobné tikaní hodin [13].

3.1.2. Moderní kryptoanalýza

S vývojem nových metod šifrování se matematika stávala vice významnou. Například, pro frekvenční analýzu musí mít kryptoanalytik znalosti v oblasti lingvistiky a statistiky. Zatímco teoretickou práci kryptoanalýzy Enigmy vykonávali především matematici, jako například Alan Turing. Nicméně díky matematice kryptografie dosáhla takového rozvoje, že množství informací potřebných k prolomení elementárních matematických operaci vzrostlo do astronomických hodnot.

Moderní kryptografie je mnohem více odolná proti kryptoanalýze, než kdysi používané zastaralé metody, kde pro odhalení stačil pouze papír a tužka.

V průběhu vzniku a vylepšování moderní počítačové kryptografie bylo provedeno velké množství vážných útoků na teoretická a praktické kryptografická primitiva, např.[13]:

- V roce 1998 byla nalezena zranitelnost vůči útokům u šifru Madryga, navržené v roce 1984.
- Celá řada útoků od vědecké společnosti doslovně zničila blokovou šifru FEAL, navrženou jako náhradu DES.
- Taky bylo zjištěno, že pomocí běžně dostupných výpočetních zdrojů mohou byt prolomeny proudové šifry A5/1, A5/2, blokové šifry CMEA a standart šifrovaní DECT používané k ochraně mobilních a bezdrátových telefonů během několika hodin či minut, a v nejhorším případě i v reálném čase.
- Útok hrubou silou pomohl prolomit některé aplikované systémy ochrany, například CSS (systém pro ochranu digitálních informaci na DVD).

Ačkoli nejspolehlivější moderní šifry jsou mnohem odolnější proti kryptoanalýze než například Enigma, kryptoanalýza stále hraje důležitou roli v široké oblasti bezpečnosti informací.

3.2. Typy útoku

Kryptoanalytické útoky se liší podle síly a nebezpečí, které představuji pro existující kryptografické systémy.

Bruce Schneier identifikuje 4 základní a 3 další metody kryptoanalýzy za předpokladu znalosti algoritmu šifry [11]:

Základní metody:

1. Útok na základě známého šifrového textu (Ciphertext-only attack)

2. Útok na základě otevřeného textu a odpovídajícího šifrového textu (Known-plaintext attack)

3. Útok na základě vybraného známého otevřeného textu (možnost výběru textu pro šifrování) (Chosen-plaintext attack)

4. Útok na základě adaptivně vybraného zašifrovaného textu (Adaptive chosen-plaintext attack)

Další metody:

1. Útok na základě vybraného šifrového textu (Chosen-ciphertext attack)
2. Útok na základě vybraného klíče (Chosen key attack)
3. Banditský útok (Rubber-hose cryptanalysis)

3.2.1. Útok na základě známého šifrového textu

Jedná se o jeden ze základních typů útoků. Předpokládá se, že kryptoanalytik zná alespoň část šifrového textu a jeho cílem je získat co největší počet odpovídajících otevřených textů nebo klíč použitý pro šifrování. Šifrové texty mohou byt získány jednoduchým odposloucháváním komunikace přes nezabezpečené komunikační kanály. Daný typ útoků je slabý a obtížný.

Příkladem útoku na základě šifrového textu je frekvenční analýza monoalfabetických substitučních šifer, nebo jakýkoliv brute-force útok.

3.2.2. Útok na základě otevřeného textu a odpovídajícího šifrového textu

Typ kryptoanalýzy, při kterém kryptoanalytik má určitou sadu otevřených textů a odpovídajícího šifrového textu.

Pak existují dvě možnosti stanovení cílů:

- Najít klíč použitý pro transformaci otevřeného textu do šifrového.
- Vytvořit algoritmus schopný dešifrovat jakoukoliv zprávu šifrovanou pomoci daného klíče.

Obdržení otevřeného textu hraje klíčovou roli v tomto útoku. Otevřené texty se získávají z různých zdrojů. Například, můžeme odhadnout obsah souboru podle jeho rozšíření.

Tento útok je silnější, než útok na základě šifrovaného textu.

Analytik může být schopen zachytit jednu nebo více zpráv otevřeného textu stejně jako šifrového. Nebo může analytik znát nějaký konkrétní vzor otevřeného textu, který se objeví ve zprávě. Například soubor, který je kódovaný ve formátu postscript začíná vždy stejným vzorem, nebo zde mohou být standardizované hlavičky či bannery elektronického přenosu zpráv apod. U všech se tedy jedná o známý otevřený text.

S touto informací je analytik schopen vydedukovat klíč na základě toho, jakým je otevřený text transformován (lineární kryptoanalýza).

3.2.3. Útok na základě vybraného známého otevřeného textu

Pro uskutečnění daného typu útoku kryptoanalytik musí mít nejen určitý počet otevřených textů a na jejich základě získaných šifrových textů, mimo jiné v takovém případě musí kryptoanalytik mít možnost k vybraným otevřeným textům získat šifrový text.

Získat vstupní data pro tento typ útoku můžeme například takto:

1. Vytvořit a odeslat falešnou nezašifrovanou zprávu údajně od jednoho z uživatelů, který běžné používá šifrování.

2. V některých případech existuje moznost získat odpověď, která bude obsahovat šifrový text citující obsah falešné zprávy.

Při provádění tohoto typu útoku má kryptoanalytik možnost vybírat bloky otevřeného textu, což za určité podmínky muže poskytnout více informací o šifrovacím klíči.

3.2.4. Útok na základě adaptivně vybraného zašifrovaného textu

Útok tohoto typu je výhodnějším konkrétním případem útoku na základě zvoleného šéfového textu.

Snadnost útoku na základě adaptivně zvoleného šifrového textu spočívá v tom, že kromě možnosti vybírat šifrový text, může kryptoanalytik rozhodnout o šifrování textu na základě již získaných výsledků operací šifrování.

Jinými slovy, pro provedení útoku na základě vybraného otevřeného textu kryptoanalytik vybírá jen jeden velký blok otevřeného textu pro následující šifrování,

pak na základě těchto údajů, se začne pokoušet o prolomení systému. V případě organizace adaptivního útoku může kryptoanalytik získávat výsledky šifrování jakýchkoliv bloků otevřeného textu.

Přítomnost zpětné vazby umožňuje útoku, na základě adaptivně zvoleného šifrového textu, mít výhodu před všemi výše uvedenými typy útoků.

Při tomto typu útoku využívá útočník zároveň dotazy na adaptivně zvolený otevřený text a adaptivně zvolený šifrový text. Takový útok je jedním z nejsilnějších z hlediska možností útočníka. Jediné dva příklady takových aktuálně známých útoků jsou bumerangový útok[15] a jojo hra[3].

3.2.5. Útok na základě vybraného šifrového textu

Předpokládejme, že kryptoanalytik má dočasný přístup k dešifrovacímu přístroji. V tomto případě může během omezené doby kryptoanalytik získat ze známých šifrových textů odpovídající otevřené texty a pak se bude muset kryptoanalytik začít zkoušet prolomení systému. Příkladem daného útoku je diferenciální kryptoanalýza.

3.2.6. Útok na základě vybraného klíče

Navzdory svému názvu útok na základě vybraného klíče neoznačuje, že se kryptoanalýza zabývá prohledáváním klíče v naději najít správný. Útok tohoto typu vychází z myšlenky, že kryptoanalytik může sledovat šifrovací algoritmus, ve kterém se používá několik klíčů. Kryptoanalytik neví o přesné hodnotě klíče, ale ví o matematickém vztahu spojující mezi sebou klíče.

Neexistuje kryptoanalytický útok na algoritmus AES, který umožní získat klíč ze známého nebo vybraného šifrového textu za přiměřenou dobu a s přiměřenou paměťovou složitostí. Existující útoky se zaměřují na realizaci a obcházení šifrovacího algoritmu: útoky postranními kanály, útoky proti systémům pro generování klíčů (zneužívající špatné použití generátorů náhodných čísel nebo lajdáckých návyků při tvorbě hesel), útoky, které se zaměřují na koncové body komunikačního systému a útoky na špatné implementace algoritmu atd[14].

3.3. Diferenciální kryptoanalýza

V roce 1990 Eli Biham a Adi Shamir představili koncepce diferenciální kryptoanalýzy [4, 5]. Pomoci této metody Biham a Shamir nalezli způsob prolomení DES s použitím vybraného otevřeného textu, který byl účinnější, než metoda hrubé síly.

Diferenciální kryptoanalýza je metoda, která analyzuje vliv jednotlivých rozdílů ve dvojicích otevřeného textu na rozdíl výsledných dvojic šifrového textu. Tyto rozdíly mohou byt použity pro přiřazení pravděpodobnosti potenciálním klíčům a lokalizace nejvíc pravděpodobného klíče.

Základní myšlenkou dané metody je zašifrovat otevřeny text, pak provést určité změny v otevřeném textu a zašifrovat ho znovu.

Ve svém článku [4] Biham a Shamir provedli analýzu DES a zjistili, že přestože DES jeví jako nelineární ve svých vstupních bitech, při jednotlivých kombinacích vstupních bitu, které modifikovány současné. To znamená, že pro každý vstupní bity S-boxu navrhuje se pravděpodobnostní rozdělení možných výstupních bitu. Biham a Shamir použily tuto vlastnost jako prostředek k identifikaci bitu klíčů. Autory popsali charakteristiky 1., 2., 3. a 5. iterace, které mohou byt použity k útokům na DES do 7 iterace. Na základě charakteristik je možné odvodit informace o výstupech pro další 2 iterace.

Nechť X a X‘ – dvojice vstupu, a ΔX jejich rozdíl. Výstupy Y a Y‘ jsou známy, proto je známy ΔY. Taky známe expanzní permutace a P – blok, proto jsou známy ΔA a ΔC. B a B‘ jsou neznámy, ale známe jejich rozdíl ΔB, který roven ΔA. Kombinace ΔA a ΔC dovoluje předpokládat bitové hodnoty pro *A* XOR K_i a *A‘* XOR K_i. Známe A a A‘ poskytuji informace o K_i.

Kde X – bity otevřeného textu, Y – bity šifrového textu, K_i – bity klíče, A – bity po operace expanzní permutace, C – výstupní bity P-bloku, B – výsledek operace „xorování“ bitu klíče a bitu textu po operace expanzní permutace.

Rozdíly dvojic otevřeného textu vedou k rozdílům získaného šifrového textu s určitou pravděpodobnosti. Tyto pravděpodobnosti lze určit pomoci tabulek (viz Příloha A) pro každý substituční blok. Tabulky vycházejí z následujícího principu: vertikálně uspořádané všechny možné kombinace ΔA, horizontálně – všechny možné kombinace ΔC, hodnoty tabulky – počet shod daného ΔC danému ΔA. Největší počet shod ukazuje na dvojice, pomocí které lze nalézt klič. Dvojice otevřeného textu odpovídající ΔA a ΔC se nazývá správná, dvojice otevřeného textu, která neodpovídá ΔA a ΔC – nesprávná. Správná dvojice napoví správný klíč iterace, nesprávná – náhodný.

Efektivita diferenciální kryptoanalýzy pro různý počet iterací DES je znázorněna v tabulce 23.

Počet iterace	Vybraný otevřeny text	Složitost analýzy
8	2^{14}	2^{9}
9	2^{24}	2^{32}
10	2^{24}	2^{15}
11	2^{31}	2^{32}
12	2^{31}	2^{21}
13	2^{39}	2^{32}
14	2^{39}	2^{29}
15	2^{47}	2^{37}
16	2^{47}	2^{37}

Tabulka 23 – Efektivita diferenciální kryptoanalýzy DES

Je známa řada pokusů rozšířit pojem diferenciální kryptoanalýzy o charakteristiky vyšších řádů. Lars Knudsen používá takzvané částečné diferenciály k prolomení DES z 6 iterace. Ještě jedna metoda kryptografického útoků – lineárně-diferenciální – spojuje techniky lineární a diferenciální kryptoanalýzy. Susan Langford navrhla způsob prolomení DES z 8 iterace, který odhaluje 10-bitový klič s 80% pravděpodobností úspěchu[12].

3.3.1. Přiklad diferenciální kryptoanalýzy S-DES

Pro realizace dané analýzy vezmeme 30 dvojic otevřeného a šifrového textu (viz Příloha A.). Dále se provádí analýza každého bloku, výsledky představeny v Příloha A. Jakmile provedena analýza bloku, lze určit nejlepší ΔA a odpovídající ΔC.

Pro druhy blok optimální dvojice (ΔA, ΔC) – (1111, 1111). Pro třetí blok optimální dvojice – (1111, 01). Z tohoto jednoznačné můžeme určit posledních 8 bit ΔA i 5 bit odpovídajícího ΔC, ΔA = xxxx11111111, ΔC = xxx11101. Pro provedení analýzy prvního bloku najdeme několik dvojic se stejnou pravděpodobností: (00001,010), (1010,100), (1110,111), (1111,101). Z toho vyplývá, že existuje 4 varianty ΔA. Jelikož ΔA je výsledkem „xorování“ bitů po substituce a permutace v daném algoritmu šifrovaní, musí byt stejné následující dvojice bitů: 3 a 9, 8 a 10, 2 a 11, 4 a 12. Dané podmínce odpovídá jednoznačné ΔA = 111111111111 a ΔC = 10111101. Pro nalezení klíče potřebujeme několik dvojic otevřeného textu, pro které ΔA = 11111111 a ΔC = 10111101.

Blok 1.

1. Dvojice (0000000000000001, 0000000011111110)
   0000⊕K1 dává na výstupu 010;
   1111⊕K1 dává na výstupu 111;
   Z tabulky analýzy 1 bloku zjistíme následující varianty:
   0000⊕K1 = 0110; K1 = 0110;
   0000⊕K1 = 1010; K1 = 1010;
   1111⊕K1 = 1001; K1 = 0110;
   1111⊕K1 = 0101; K1 = 1010;
2. Dvojice (0000000000000010, 0000000011111101)
   0000⊕K1 dává na výstupu 010;
   1111⊕K1 dává na výstupu 111;
   0000⊕K1 = 0110; K1 = 0110;
   0000⊕K1 = 1010; K1 = 1010;
   1111⊕K1 = 1001; K1 = 0110;
   1111⊕K1 = 0101; K1 = 1010;
3. Dvojice (0000000000011010, 0000000011100101)
   0100⊕K1 dává na výstupu 011;
   1011⊕K1 dává na výstupu 110;
   0100⊕K1 = 0011; K1 = 0111;
   0100⊕K1 = 1110; K1 = 1010;
   1011⊕K1 = 0001; K1 = 1010;
   1011⊕K1 = 1100; K1 = 0111;

Klič K1 = 1010, se vyskytuje nejčastěji.

Stejným způsobem provedeme analýzu 2 a 3 bloku.

Blok 2.

1. Dvojice (0000000000000001, 0000000011111110)
   0100⊕K2 dává na výstupu 010;
   1011⊕K2 dává na výstupu 101;
   Z tabulky analýzy 2 bloku zjistíme následující varianty:
   0100⊕K2 = 1110; K2 = 1010;
   0100⊕K2 = 0011; K2 = 0111;
   1011⊕K2 = 0001; K2 = 1010;
   1011⊕K2 = 1100; K2 = 0111;

2. Dvojice (0000000000000010, 0000000011111101)
   0001⊕K2 dává na výstupu 011;
   1110⊕K2 dává na výstupu 100;
   0001⊕K2 = 0000; K2 = 0001;
   0001⊕K2 = 1011; K2 = 1010;
   1110⊕K2 = 0100; K2 = 1010;
   1110⊕K2 = 1000; K2 = 0110;
3. Dvojice (0000000000011010, 0000000011100101)
   0011⊕K2 dává na výstupu 110;
   1100⊕K2 dává na výstupu 001;
   0011⊕K2 = 0101; K2 = 0110;
   0011⊕K2 = 1001; K2 = 1010;
   1100⊕K2 = 0110; K2 = 1010;
   1100⊕K2 = 1010; K2 = 0110;

Klič K2 = 1010, se vyskytuje nejčastěji.

Blok 3.

1. Dvojice (0000000000000001, 0000000011111110)
   0100⊕K3 dává na výstupu 11;
   1011⊕K3 dává na výstupu 10;
   0100⊕K3 = 0010; K3 = 0110;
   0100⊕K3 = 0101; K3 = 0001;
   0100⊕K3 = 1000; K3 = 1100;
   0100⊕K3 = 1011; K3 = 1111;
   0100⊕K3 = 1110; K3 = 1010;
   1011⊕K3 = 0000; K3 = 1010;
   1011⊕K3 = 0011; K3 = 1111;
   1011⊕K3 = 0110; K3 = 1100;
   1011⊕K3 = 1001; K3 = 0001;
   1011⊕K3 = 1100; K3 = 0110;

2. Dvojice (0000000000000010, 0000000011111101)
   0000⊕K3 dává na výstupu 10;
   1111⊕K3 dává na výstupu 11;
   0000⊕K3 = 0001; K3 = 0001;
   0000⊕K3 = 0100; K3 = 0100;

0000⊕K3 = 0111; K3 = 0111;
0000⊕K3 = 1010; K3 = 1010;
0000⊕K3 = 1101; K3 = 1101;
1111⊕K3 = 0010; K3 = 1101;
1111⊕K3 = 0101; K3 = 1010;
1111⊕K3 = 1000; K3 = 0111;
1111⊕K3 = 1011; K3 = 0100;
1111⊕K3 = 1110; K3 = 0001;

3. Dvojice (0000000000011010, 0000000011100101)
   0001⊕K3 dává na výstupu 11;
   1110⊕K3 dává na výstupu 10;
   0001⊕K3 = 0010; K3 = 0011;
   0001⊕K3 = 0101; K3 = 0100;
   0001⊕K3 = 1000; K3 = 1001;
   0001⊕K3 = 1011; K3 = 1010;
   0001⊕K3 = 1110; K3 = 1111;
   1110⊕K3 = 0001; K3 = 1111;
   1110⊕K3 = 0100; K3 = 1010;
   1110⊕K3 = 0111; K3 = 1001;
   1110⊕K3 = 1010; K3 = 0100;
   1110⊕K3 = 1101; K3 = 0011;

Klič K3 = 1010, se vyskytuje nejčastěji.

Po provedení diferenciální analýzy získali jsme klíč K = 1010 1010 1010.

3.4. Lineární kryptoanalýza

Lineární kryptoanalýza je jiná metoda kryptoanalytického odhalení vynalezena japonským kryptologem Mitsuruem Matsui. Je to druh útoku na základě otevřeného textu, ve kterém útočník zkoumá lineární aproximace bitových dvojic otevřeného textu, šifrového textu a klíče. Daná analýza používá lineární aproximace k popisu činnosti blokových šifer. Algoritmus navržený v roce 1993 byl původně zaměřen na odhalení DES [8] a FEAL. Následně byla lineární kryptoanalýza rozšířena i na další algoritmy. Využitím pomocných technik lze rozšířit útok k nalezení více bitů klíče.

V blokových šifrách je analýza především zaměřena na S-boxy. Kryptoanalýza se provádí ve dvou krocích. První – sestavení vztahu mezi otevřeným textem a šifrovaným

textem a klíčem, které jsou s vysokou pravděpodobnosti pravdivé. Druhý - použití těchto vztahů společně se známými dvojicemi otevřeného – šifrového textu k získání bitů klíčů.

Poměrně často se lineární kryptoanalýza používá v kombinaci s útokem brute-force. Po nalezení určitých bitu klíčů pomoci lineární analýzy se provádí vyhledávaní všech možných hodnot zbývajících bitů.

Lineární aproximace pro jednotlivé operace v šifře mohou být dále kombinované do aproximace, které jsou platné pro jednu iteraci šifry.

Cílem tohoto algoritmu je získat následující vztahy:

$P[i_1, i_2, \ldots, i_a]$ XOR $C[j_1, j_2, \ldots, j_b] = K[k_1, k_2, \ldots, k_c]$

Kde $i_1, i_2, \ldots, i_a$; $j_1, j_2, \ldots, j_b$; $k_1, k_2, \ldots, k_c$ označuji bitové pozice. Daný vztah může být získán vhodným zřetězením jedné iterace aproximace. Taková aproximace je vhodná, jestliže je její pravděpodobnost odlišná od hodnoty 0,5.

3.4.1. Piling-up lemma

První krok lineární analýzy spočívá v nalezení užitečných aproximaci pro danou šifru. Přestože nejvíce přiblížena lineární aproximace může být jednoduše nalezena vyčerpávajícím způsobem pro jednoduché komponenty, jakými je třeba S-box, vznikne hned řada problémů při pokusu odvodit tuto metodu pro plnohodnotné šifry. První problém spočívá ve výpočtu pravděpodobnosti lineární aproximace.

V podstatě by to znamenalo pro kryptoanalytika projít všechny možné kombinace otevřeného textu a klíče, což je jednoduše v praxi nemožné pro jakoukoliv šifru. Řešením tohoto problému je učinit řadu předpokladů a přiblížit se požadované pravděpodobnosti pomocí Piling-up lemma.

Pro n nezávislých a náhodných binárních proměnných ($X_1, X_2, ..., X_n$) platí:

$$\Pr(X1 \oplus X2 \ldots \oplus Xn) = \frac{1}{2} + 2^{n-1} \prod_{i=1}^{n} \varepsilon_i$$

kde ε_i reprezentuje odchylku lineární pravděpodobnosti $X1 \oplus X2 \oplus \ldots \oplus Xn = 0$.

3.4.2. Známé útoky pomocí lineární kryptoanalýzy

DES obsahující 8 iterací byl prolomen pomoci 2^{21} známých otevřených textů během 40 vteřin. DES obsahující 12 iterací byl prolomen pomoci 2^{33} známých otevřených textů během 50 hodin.

Kromě DES a FEAL existuje další řada algoritmu, které jsou zranitelné lineární analýzou.

Některé varianty DES (DESX, DES s nezávislým podklíčem a Biham-DES), RC5 (v případě že šifrovaný klíč patří do skupiny slabých klíčů), NUSH a Noekeon byly prolomeny lineární analýzou.

V dnešní době se od nových šifrovacích algoritmů vyžaduje odolnost vůči lineární kryptoanalýze.

3.4.3. Přiklad lineární kryptoanalýzy S-DES

Lineární kryptoanalýza se začíná analýzou bloku (viz Příloha A) a nalezením nejefektivnější rovnice.

Blok 1

Pro první blok určeny 3 dvojice: (12,5), (14,1) a (15,2). Na základě principu algoritmu S-DES můžeme sestrojit následující rovnice:

$X11 \oplus X12 \oplus Y7 \oplus Y3 \oplus X7 \oplus X3 = K1 \oplus K2$

P = 1/16, Δ = |1-2p| = 7/8

$X11 \oplus X12 \oplus X9 \oplus Y3 \oplus X3 = K1 \oplus K2 \oplus K3$

P = 13/16, Δ = |1-2p| = 5/8

$X11 \oplus X12 \oplus X9 \oplus X10 \oplus Y4 \oplus X4 = K1 \oplus K2 \oplus K3 \oplus K4$

P = 3/16, Δ = |1-2p| = 5/8

Blok 2

$X13 \oplus Y6 \oplus X6 = K7$

P = 3/16, Δ = |1-2p| = 5/8

$X14 \oplus X16 \oplus Y6 \oplus Y8 \oplus X6 \oplus X8 = K5 \oplus K6$

P = 13/16, Δ = |1-2p| = 5/8

$X14 \oplus X16 \oplus X15 \oplus Y5 \oplus X5 = K5 \oplus K6 \oplus K8$

P = 3/16, Δ = |1-2p| = 5/8

$X14 \oplus X16 \oplus X13 \oplus Y8 \oplus X8 = K5 \oplus K6 \oplus K7$

P = 1/8, Δ = |1-2p| = 3/4

$X14 \oplus X16 \oplus X13 \oplus X15 \oplus Y5 \oplus Y8 \oplus X5 \oplus X8 = K5 \oplus K6 \oplus K7 \oplus K8$

P = 3/16, Δ = |1-2p| = 5/8

Blok 3

$X11 \oplus X16 \oplus X10 \oplus X12 \oplus Y2 \oplus X2 = K9 \oplus K10 \oplus K11 \oplus K12$

P = 7/8, Δ = |1-2p| = 3/4

Dále pomoci algoritmu:

Jestli T > N/2, pak

$K_i = 0$, jestli p > 1/2

$K_i = 1$, jestli p < 1/2

Jestli T < N/2, pak

$K_i = 1$, jestli p > 1/2

$K_i = 0$, jestli p < 1/2

Kde N je počet otevřených textu, T je počet otevřených textu, pro které výsledek „xoroványch“ šifrových bitu rovna 0 a K_i – jednotlivé bity klíčů. Získáme následujíce:

$K1 \oplus K2 = 1$

$K1 \oplus K2 \oplus K3 = 0$

$K1 \oplus K2 \oplus K3 \oplus K4 = 0$

$K5 \oplus K6 \oplus K7 = 0$

$K7 = 1$

$K5 \oplus K6 \oplus K8 = 1$

$K5 \oplus K6 = 1$

$K5 \oplus K6 \oplus K7 \oplus K8 = 0$

Pomoci daných rovnic můžeme najit 4 možných kombinace klíčů:

K1 = 01100110xxxx;

K2 = 10100110xxxx;

K3 = 01101010xxxx;

K4 = 10101010xxxx;

Použitím brute-force metody najdeme chybějící bity klíčů. Jako výsledek dostaneme 64 možné kombinace klíčů. Pro nalezení správného klíče, je třeba zašifrovat libovolný počet dvojic a porovnat s výsledky správných šifrových dvojic. Správný klíč pro dany přiklad je K = 101010101010.

4. Demonstrace útoků

4.1. Zvolená technologie

Pro implementaci animovaných praktických příkladů kryptoanalýzy byla zvolena technologie Microsoft Silverlight.

Silverlight je aplikační platforma vytvořená společností Micorosoft a určená pro vývoj business a multimediálních aplikací. Aplikace vytvořené touto technologií mohou běžet v rámci webového prohlížeče, anebo v režimu „out-of-browser“, tedy ve vlastním okně mimo prohlížeč.

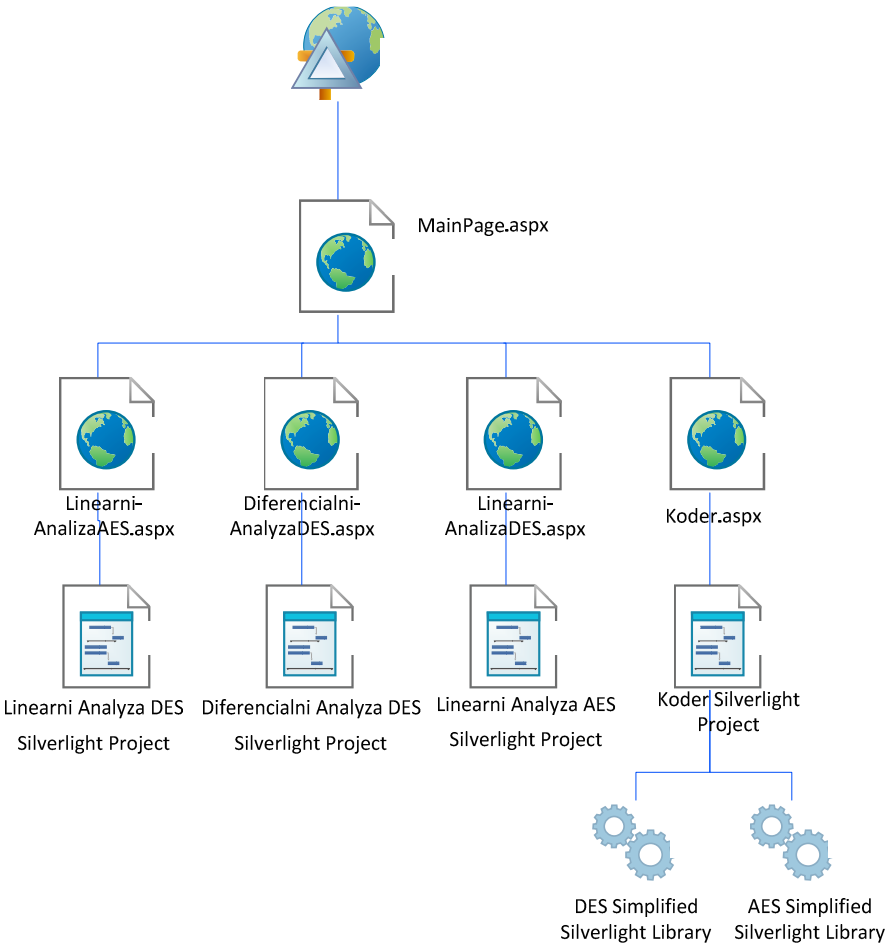
Silverlight poskytuje grafický systém podobný Windows Presentation Foundation (WPF) a integruje multimédia, grafiku a animace do jedné softwarové platformy. Byl navržen pro práci s XAML a jazyky .NET Framework. XAML pak slouží k označení stránek, které používají vektorové grafiky a animace. Následně pak text, který obsahuje aplikace Silverlight, není k dispozici pro vyhledávače, jelikož se nekompiluje, ale je k dispozici v podobě XAML. Další výhodou této technologie je, že bez problémů přehrává formáty WMV, WMA, MP3 a další multimediální prvky.

Navíc k této technologii bylo využito volně dostupné rozšíření, Silverlight Toolkit. To obsahuje komponenty, které napomáhají vytvářet více funkcionální aplikace a efektivně napomáhají k tvorbě bohatších aplikaci.

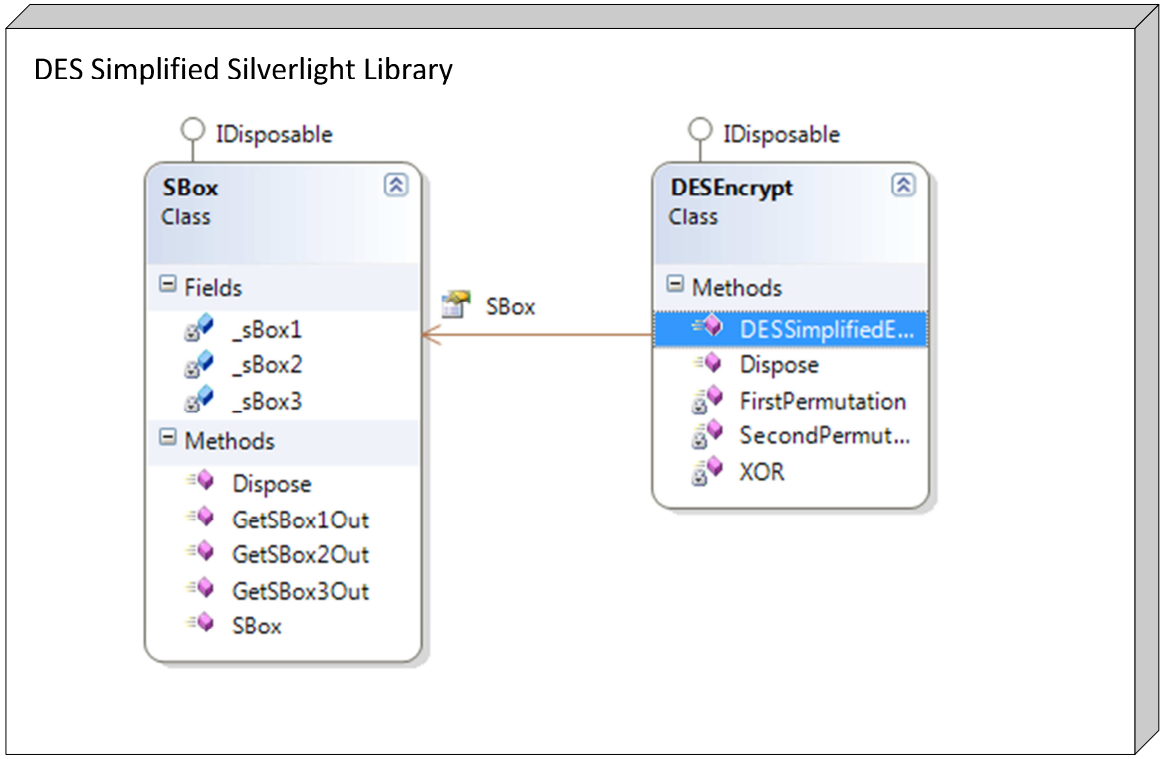
Pro vytvoření zjednodušené verze šifrovacích algoritmů DES a AES byla použita technologie C#. C# (C Sharp) je vysokoúrovňový objektově-orientovaný programovací jazyk rovněž vyvíjený společností Microsoft spolu s platformou .NET Framework. Jazyk C# byl zvolen hlavně pro jeho moderní, mnohoúčelové a robustní vlastnosti, které programátorovi umožňují vytvářet jakékoli aplikace a algoritmy prakticky bez omezení.

4.2. Návrh architektury

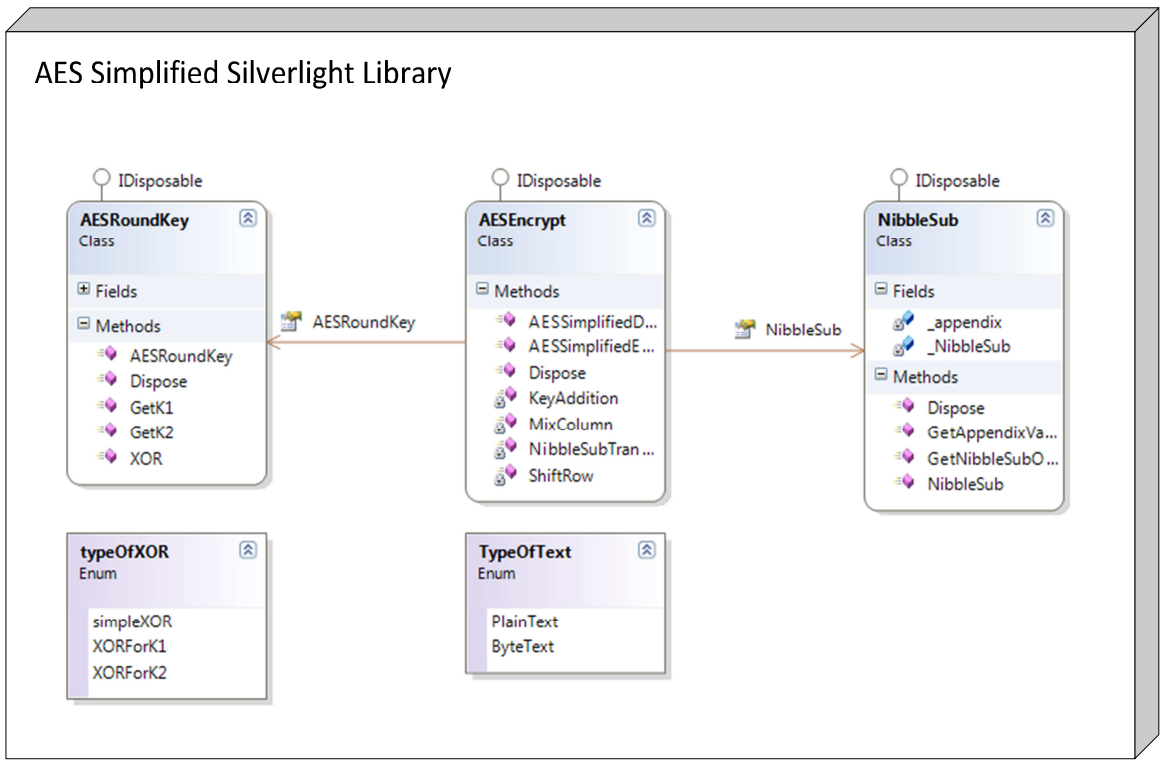
V rámci diplomové práce byly realizovány animované praktické ukázky lineární kryptoanalýzy S-AES a S-DES a diferenciální kryptoanalýzy S-DES. Následně byly implementovány zjednodušené verze šifrovacích algoritmů S-AES a S-DES. Níže uvedené obrázky názorně popisují architekturu vytvořené aplikace.



Obrázek 6 - Architektura aplikace

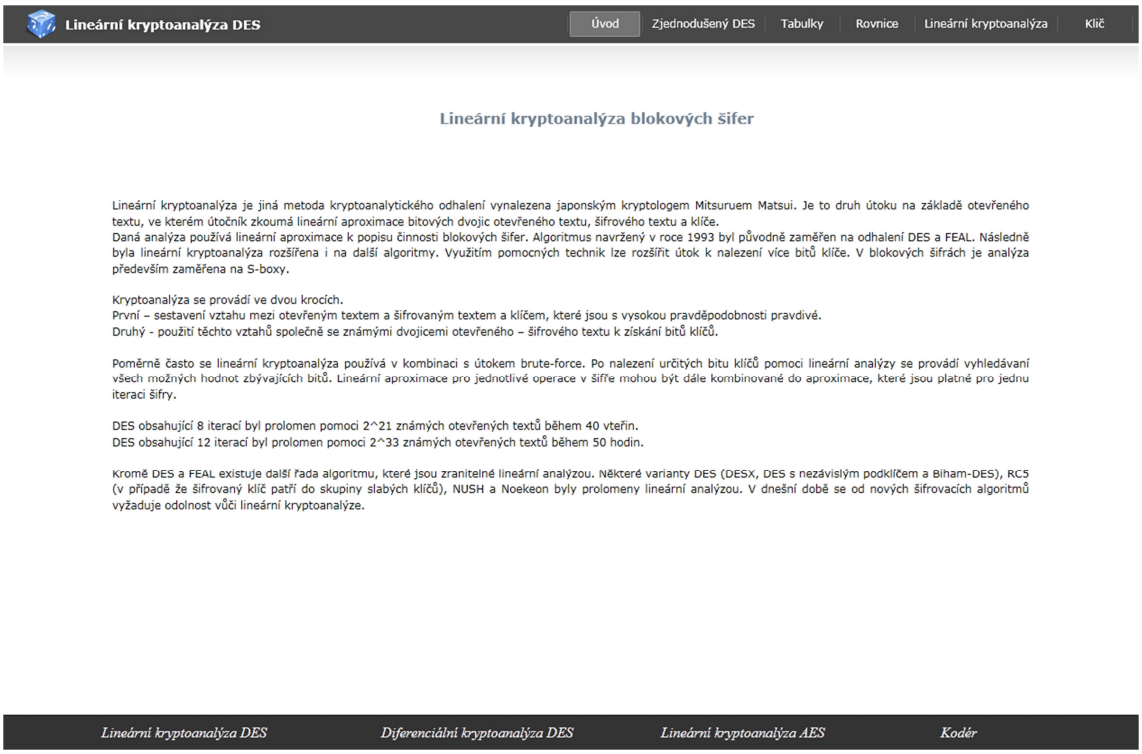


Obrázek 8 - Struktura S-DES



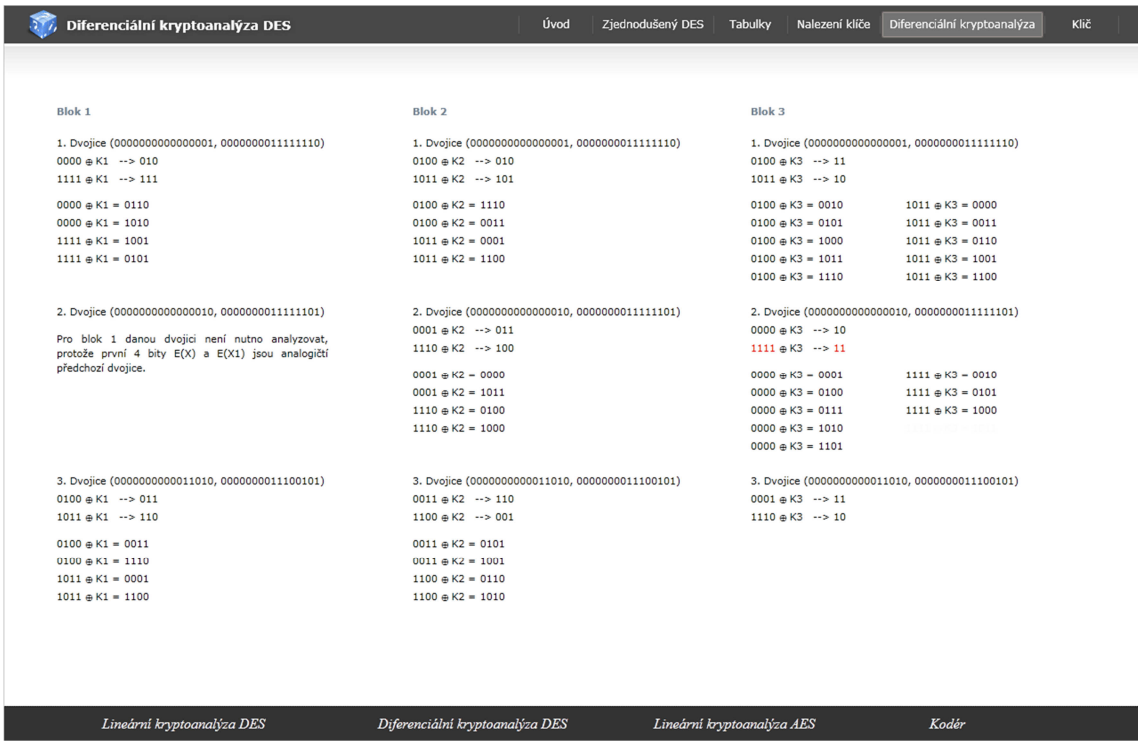
Obrázek 7 - Struktura S-AES

Na obrázku číslo 8 zobrazen prakticky přiklad lineární kryptoanalýzy. V jednotlivých sekcích představeny: zjednodušena verze algoritmu, tabulky k analýze, podle kterých navrženy lineární rovnice a postupy k naležení klíčů.



Obrázek 9 - Lineární kryptoanalýza S-DES

Na obrázku číslo 9 zobrazen prakticky přiklad diferenciální kryptoanalýzy. V jednotlivých sekcích představeny: zjednodušena verze algoritmu, tabulky k analýze, diferenciální kryptoanalýza a postupy k naležení klíčů.



Obrázek 10 - Diferenciální kryptoanalýza S-DES

5. Závěr

Cílem této diplomové práce bylo nastudovat důkladně problematiku kryptologie (kryptografie a kryptoanalýzy) a vytvořit aplikaci reprezentující praktickou ukázku kryptoanalytických metod. Aplikace byla realizována za použití nejnovější technologie Microsoft Silverlight, a to na konkrétních blokových šifrách. Aplikace přehledně a demonstruje tři rozsáhlé příklady metod kryptoanalýzy (S-AES, S-DES).

Práce by pak mohla být dále rozšířena o využití dalších metod kryptoanalýzy, o další šifry či v neposlední řadě pak také může být rozšířeno ovládání o další prvky pro lepší interakci z pohledu uživatelů.

6. Literatura

[1] American National Standards Institute, Triple Data Encryption Algorithm Modes of Operation, ANSI X9.52, 1998.

[2] Daniel J. Bernstein, Cache-timing attacks on AES, Department of Mathematics, Statistics, and Computer Science, The University of Illinois at Chicago, Chicago, IL 60607-7045, 2005.

[3] E. Biham, A. Biryukov, O. Dunkelman, E. Richardson, and A. Shamir,"Initial observations on skipjack: Cryptanalysis of skipjack-3xor," in Selected Areasin Cryptography, SAC 1998 (S. E. Tavares and H. Meijer, eds.), vol. 1556 of Lecture Notesin Computer Science, pp. 362-376, Springer-Verlag, 1999.

[4] E. Biham and A. Shamir, "Differential Cryptanalysis of DES-like Cryptosystems," Advances in Cryptology – CRYPTO'90, Springer-Verlag, 2–21, 1990, URL: http://citeseer.ist.psu.edu/index

[5] E. Biham and A. Shamir, "Differential Cryptanalysis of the Full 16-round DES," Advances in Cryptology – CRYPTO'92, Springer-Verlag, 487–496, 1992, URL: http://citeseer.ist.psu.edu/index

[6] E. F. Brickell, J .H.Moore and M. R. Purtill, "Structure in the S-Boxes of DES," Advances in Cryptology –CRYPTO'86, 3–7, 1986.

[7] Daemen, J. and V. Rijmen, "The design of Rijndael: AES –The Advanced Encryption Standard", Springer-Verlog, Berlin, 2002.

[8] M. Matsui, "Linear Cryptanalysis Method for DES Cipher," Advances in Cryptology – EUROCRYPT'93, 386–397, 1994.

[9] Dr. Mohammed Mrayati, Dr. Yahya Meer Alam, Dr. M. Hassan at-Tayyan al-Kindi's Treatise on Cryptanalysis, KFCRIS & KACST, ISBN 9960-890-08-2, 2003.

[10] National Bureau of Standards, Data Encryption Standard, U.S. Department of Commerce, FIPS pub. 46, January 1977.

[11] Bruce Schneier, Wiley, Applied Cryptography: Protocols, Algorithms, and Source Code in C, Second Edition, ISBN 0471117099, 1995.

[12] William Stallings, Cryptography and Network Security: Principles and Practice, Prentice Hall, 2010. ISBN-10: 0136097049

[13] URI: http://www.wikipedia.com

[14] URL: http://schneier.com/blog/archives/2012/03/can_the_nsa_bre.html

[15] D. Wagner, "The boomerang attack," in Fast Software Encryption, FSE'99 (L. R. Knudsen, ed.), vol. 1636 of Lecture Notes in Computer Science, pp. 156-170, Springer-Verlag, 1999.

Seznam příloh

Příloha A – pomocné tabulky

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

Tabulka 1: Pomocná tabulka pro transformace MixColumns (S-AES)

	1	2	3	4	5	6	7
1	9	9	6	10	9	5	8
2	7	9	8	4	9	11	8
3	9	11	8	4	7	5	12
4	9	9	10	8	7	7	6
5	7	11	6	8	9	9	6
6	9	7	8	6	9	7	10
7	7	9	12	10	7	5	6
8	7	9	8	8	9	7	8
9	9	11	8	8	7	9	4
10	11	7	10	6	7	7	8
11	5	9	10	10	9	5	8
12	9	7	8	10	1	11	10
13	7	9	12	6	7	9	6
14	13	9	6	8	7	7	6
15	11	3	10	8	9	9	6

Tabulka 2: Analýza prvního bloku (S-DES)

	1	2	3	4	5	6	7
1	8	11	9	9	7	6	6
2	10	9	9	3	7	6	12
3	8	9	11	9	7	4	8
4	8	7	9	9	7	6	10
5	6	7	11	7	7	12	6
6	8	9	7	9	7	8	8
7	10	5	9	11	7	6	8
8	8	7	9	7	9	8	8
9	6	7	11	9	5	10	8
10	8	9	7	11	5	6	10
11	10	5	9	9	9	8	6
12	6	7	7	9	13	6	8
13	8	3	9	7	9	8	12
14	2	9	9	9	9	8	10
15	8	9	3	11	5	10	10

Tabulka 3: Analýza druhého bloku (S-DES)

	1	2	3
1	7	8	9
2	9	8	7
3	7	10	7
4	7	8	9
5	9	6	9
6	7	10	7
7	5	8	11
8	9	8	7
9	7	10	7
10	9	6	9
11	11	8	5
12	7	10	7
13	5	8	11
14	11	8	5
15	5	14	5

Tabulka 4: Analýza třetího bloku (S-DES)

ΔC / ΔA	000	001	010	011	100	101	110	111
0001	0	0	8	0	0	2	2	4
0010	0	0	2	2	0	6	0	6
0011	4	2	2	0	0	4	0	4
0100	0	6	2	4	0	0	4	0
0101	0	4	0	4	6	0	0	2
0110	0	2	0	2	6	2	4	0
0111	0	2	2	4	4	0	4	0
1000	0	6	0	6	0	0	2	2
1001	0	4	0	4	4	2	2	0
1010	0	2	2	0	8	0	4	0
1011	2	0	0	6	4	0	4	0
1100	6	2	4	0	0	2	0	2
1101	4	0	4	0	0	2	2	4
1110	2	0	4	2	0	0	0	8
1111	4	2	2	0	0	8	0	0

Tabulka 5: Závislost ΔC od ΔA v prvním bloku (S-DES)

ΔC / ΔA	000	001	010	011	100	101	110	111
0001	0	0	8	2	0	2	4	0
0010	0	2	0	0	2	6	2	4
0011	0	2	2	2	2	2	0	6
0100	0	4	2	4	0	2	2	2
0101	4	4	0	6	0	2	0	0
0110	0	0	4	2	6	0	2	2
0111	0	2	0	0	6	2	6	0
1000	0	6	0	4	0	0	4	2
1001	2	2	0	6	2	4	0	0
1010	2	6	2	2	4	0	4	2
1011	2	2	2	0	4	2	4	0
1100	6	0	2	2	2	2	2	0
1101	4	0	6	0	2	0	4	0
1110	0	4	2	0	2	8	0	0
1111	2	0	2	2	0	0	0	10

Tabulka 6: Závislost ΔC od ΔA v druhem bloku (S-DES)

ΔC / ΔA	00	01	10	11
0001	0	4	6	6
0010	0	4	6	6
0011	8	4	2	2
0100	0	4	6	6
0101	8	4	2	2
0110	8	4	2	2
0111	4	0	6	6
1000	0	4	6	6
1001	8	4	2	2
1010	8	4	2	2
1011	4	0	6	6
1100	8	4	2	2
1101	4	0	6	6
1110	4	0	6	6
1111	6	10	0	0

Tabulka 7: Závislost ΔC od ΔA v třetím bloku (S-DES)

Dvojice otevřeného a šifrového textu:

0000000000001000 – 1001100100001000
0000000010011000 – 1101011010011000
0000000010001001 – 0110011110001001
0000000001011001 – 1101011101011001
0000000001001011 – 1000111101001011
0000000010011011 – 1001111110011011
0000000010001100 – 1010101110001100
0000000001011100 – 1001101101011100
0000000001001110 – 1100011101001110
0000000011011101 – 1110010011011101
0000000010001110 – 1010011110001110
0000000000011110 – 1111010100011110
0000000001001111 – 1000111001001111
0000000001101111 – 0111110001101111
0000000010010000 – 1101001110010000
0000000000100000 – 0110000100100000
0000000001010001 – 1101101001010001
0000000011100001 – 1001101011100001
0000000010010010 – 1101101110010010
0000000011100010 – 1001101111100010
0000000001010011 – 1101001101010011
0000000011100011 – 0101001111100011
0000000010010100 – 1101111110010100
0000000000100100 – 0110110100100100
0000000001010101 – 1101001101010101

0000000000100110 – 0110100000100110
0000000010010110 – 1101101010010110
0000000011100101 – 0101001111100101
0000000001010111 – 1101111101010111
0000000011100111 – 0101111111100111