VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
FAKULTA INFORMAČNÍCH TECHNOLOGIÍ

DOKUMENTACE K PROJEKTU DO PŘEDMĚTU KRY

Projekt 1

Bc. Lukáš Pelánek, xpelan03@stud.fit.vutbr.cz

12. dubna 2018

1 Úvod

Cílem tohoto projektu bylo prolomit neznámou synchronní proudovou šifru, kterou byly zašifrovány soubory, které jsme dostali jako součást zadání. Tyto soubory byly zašifrovány klíčem o délce 29 znaků ve formátu **KRY{*24 znaků ASCII textu*}**. Tato práce slouží jako dokumentace pro moje řešení.

2 První část řešení

V první části řešení bylo mým úkolem prolomit šifru ručně. Programovací jazyk byl stanoven na Python3. Zadaný archiv obsahoval 4 soubory:

- bis.txt – soubor obsahující plaintext
- bis.txt.enc – soubor obsahující zašifrovaný plaintext
- hint.gif.enc – zašifrovaný obrázek ve formátu gif
- super_cypher.py.enc – zašifrovaný skript

Prvním krokem k prolomení šifry bylo získat keystream, kterým byly soubory zašifrovány. Toho jsem docílil pomocí operace **xor** souborů bis.txt a bis.txt.enc. Tento keystream jsem následně využil pro rozšifrování souboru super_cypher.py.enc. Keystream byl kratší než obsah tohoto zašifrovaného skriptu, a tak se mi podařilo odkrýt pouze část tohoto souboru.

Rozšifrovaný soubor obsahoval důležité konstanty a algoritmus, kterým byl keystream generován. Toho jsem využil a použil prvních 32 bajtů známého keystreamu. Následně jsem byl schopen další části keystreamu generovat pomocí funkce **step** a rozšifrovat celý soubor včetně zašifrovaného obrázku. Bohužel rozšifrovaný obrázek mi nebyl nijak nápomocen, ale rozšifrovaný skript obsahoval algoritmus, kterým byly tyto soubory zašifrovány.

Posledním krokem bylo napsat inverzní funkci k funkci **step**, která z prvních 32 bajtů keystreamu odvodí klíč, který byl použit pro inicializaci keystreamu. Inverzní funkci jsem musel v cyklu vyvolat 128x, protože stejným způsobem byl keystream inicializován. Výsledkem je inicializační klíč, který v mém případě byl **KRY{xpelan03-5c56f895bf94372}**.

3 Druhá část řešení

V druhé části řešení bylo za úkol získat tajemství pomocí SAT solveru aplikovaného na správnou část šifry. K řešení jsem použil SAT solver Minisat[1]. K dispozici jsem měl stejné soubory jako v první části.

Prvním krokem bylo vytvořit si pole 256 proměnných, kde každý prvek reprezentoval jednotlivé bity keystreamu. Následně vhodně upravit funkci **step**, kterou jsem pojmenoval jako **get_expression**. Tato funkce přijímá jako parametr pole proměnných (bity reprezentující keystream) a vrací pole jednotlivých výrazů, které slouží pro vypočítání příslušných bitů. Funkce **get_expression** nejprve vloží nejméně významný bit na první pozici a nejvíce významný bit na poslední pozici. Následně volá funkci **rule**, která přijímá 3 parametry (bity) a můžeme ji vyjádřit následující pravdivostní tabulkou:

[1] http://minisat.se/

a	b	c	Y
0	0	0	0
0	0	1	1
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	0
1	1	0	1
1	1	1	0

Tabulka 1: Pravdivostní tabulka funkce rule

Funkce **rule** musí vrátit výraz dle této pravdivostní tabulky v konjunktivní normální formě, kterou SAT solver vyžaduje.

Následně nastavím počáteční keystream, který je prvních 32 bajtů již známého keystreamu. Poté v cyklu dojde k několika krokům. Složím finální výraz pro SAT solver tak, že pokud bit v keystreamu na x-té pozici je roven 0, tak daný výraz příslušející danému bitu v poli výrazů na x-té pozici vložím znegován, jinak jej vložím v původní formě. Následně nechám SAT solver ohodnotit jednotlivé proměnné. Následně otestuji výsledky pro jednotlivé bity. Pokud ohodnocení proměnné na pozici x je pravdivé, tak v novém keystreamu na pozici x bude nastaven bit 1, jinak 0. Tento cyklus se opakuje 128x. Na konci keystream obsahuje inicializační klíč, který je **KRY{xpelan03-5c56f895bf94372}**.

4 Závěr

V tomto projektu jsem si vyzkoušel prolomit neznámou synchronní proudovou šifru pomocí SAT solveru a klasickým způsobem. Prolomit šifru se mi podařilo oběma způsoby. Řešení pomocí SAT solveru zabere na mém počítači zhruba 8 vteřin, ale klasické řešení do vteřiny.