УДК 004.056.5:004.421.5

АТАКА АПАРАТНИХ ЗБОЇВ НА ПРОРІДЖУЮЧИЙ ГЕНЕРАТОР ПСЕВДОВИПАДКОВИХ ЧИСЕЛ

Богдан Карпінський

Тернопільський державний економічний університет, кафедра безпеки інформаційних технологій

Анотація: **Запропоновано алгоритм диференційної криптоатаки збоїв на базову компоненту сучасних потокових шифрів – регістр зсуву з лінійними зворотними зв'язками. Розроблено програмне забезпечення, що імітує проведення даної атаки, на основі якого показано її ефективність. Запропонована модифікація даної атаки на конструкцію проріджуючого генератора.**

Summary: **The algorithm of differential fault attack on the linear feedback shift register – basic component of modern stream ciphers, was proposed. The software tool for simulation of this attack was developed. The efficiency of the proposed algorithm was shown by this software tool. The modification of attack was proposed and applied on the construction of shrinking generator.**

Ключові слова: **Атаки побічних каналів, диференційна криптоатака збоїв, ГПВЧ, регістр зсуву з лінійними зворотними зв'язками, потокові шифри.**

I Вступ

До недавнього часу атаки на криптографічні криптоалгоритми базувалися на аналізі відкритих текстів та криптотекстів. Проте, на даний час розвинулися нові методи криптографічних атак, що спрямовані саме на апаратну реалізацію криптоалгоритмів. Їх ще називають атаками на побічні канали витоку інформації – Side-Channel Analysis (SCA) [1 – 6]. Найбільш популярними серед них є: аналіз енергоспоживання (Power Analysis), часовий аналіз (Timing Analysis), атака на основі внесення збоїв (Fault Insertion Attacks) та атака на основі аналізу електромагнітного випромінювання (Electro Magnetic Emissions).

Модель цих атак передбачає, що криптоаналітик має безпосередній доступ до криптографічного пристрою (наприклад, смарткартки) і має можливість [2]:

- ✓ задавати будь-які дані на вхід криптопристрою;
- ✓ отримувати та аналізувати дані на виході аналізованого пристрою;
- ✓ вимірювати та аналізувати часові залежності виконуваних пристроєм дій;
- ✓ вносити збої в роботу пристрою під час виконання ним криптографічних перетворень;
- ✓ вимірювати електромагнітні випромінювання під час роботи пристрою.

Така модель є достатньо реалістичною в багатьох прикладних застосуваннях. Яскравим прикладом може бути випадок із запропонованими атаками на реалізацію на смарт-картки [2].

Внаслідок аналізу інформації про особливості роботи криптографічного пристрою зловмисник може отримати інформацію про значення секретного ключа, тобто зламати таку криптосистему. Таку інформацію ще можна назвати як отриману із побічних каналів витоку інформації.

Особливого інтересу серед множини атак побічних каналів набула атака на основі внесення апаратних збоїв. Вона базується на основі аналізу реакції криптопристрою на внесені збої під час виконання ним криптографічних перетворень чи під час переходу криптографічного пристрою із одного внутрішнього стану в інший.

Boneh, DeMillo і Lipton в [6] представили криптоатаку, базовану на внесенні апаратних збоїв на криптопристрій, що реалізовував асиметричний криптоалгоритм. Перевагою запропонованої атака є те, що зловмиснику немає необхідності реалізовувати збій у конкретному біті чи в декількох бітах – результат атаки все рівно буде успішним.

Також, Biham і Shamir в [4] представили алгоритм проведення такої атаки на симетричний блочний криптоалгоритм DES, який вимагає 200 пар дані-криптотекст та дані-криптотекст зі збоями для визначення ключа шифрування.

Алгоритми криптоаналізу на основі внесення апаратних збоїв на апаратно-реалізовані симетричні та асиметричні криптоалгоритми отримали значний розвиток. Проте про застосування такого виду атак на симетричні потокові шифри у відкритій літературі не було згадок. Вперше припущення про можливість проведення таких атак на потокові шифри були висловлені в [1], проте безпосередні дослідження були проведені тільки в [7].

У даній статті автором запропоновано узагальнений алгоритм атаки на базову компоненту потокових шифрів – регістр зсуву із лінійними зворотними зв'язками, реалізовано програмне забезпечення, що імітує проведення даної атаки. Також запропоновано алгоритм проведення диференційної атаки збоїв на

конструкцію проріджуючого генератора.

II Внесення апаратних збоїв

На даний час розвинулося багато методів внесення апаратних збоїв, базованих на зовнішніх дестабілізуючих факторах, якими можуть виступати: електромагнітне випромінювання, температура, нестабільність енергоживлення та нестабільність тактової частоти (over-clocking). Зовнішні дестабілізуючі фактори можуть призвести до неалгоритмічної роботи криптопристрою, що в деяких випадках дасть криптоаналітику інформацію про внутрішню роботу пристрою чи „секретні" дані (наприклад ключі) [3].

Усі види примусового зовнішнього випромінювання (електромагнітного, іонізуючого, інфрачервоного, ультрафіолетового, світло спалаху) індукують завади, абсолютні значення яких є більшими за порогові рівні базових логічних елементів в нормальному режимі роботи. Це призводить до хибних спрацювань атакованого пристрою. Слід відмітити, що для деяких типів випромінювання необхідно забезпечити доступ до кристалу мікросхеми [3].

В загальному випадку збої можуть різнитися як за направленістю, так і за типом. За направленістю їх можна розділити на

- ✓ збої на біти комірок пам'яті (наприклад, де зберігається ключ чи підключ);
- ✓ збої на внутрішню систему керування (перемикання між станами роботи пристрою);
- ✓ збої на шини передачі даних чи команд.

За типом збої можна поділити на:

- ✓ встановлення в логічне значення $'1'$;
- ✓ встановлення в логічне значення $'0'$;
- ✓ інверсія біта ($'1' \to '0'$, або $'0' \to '1'$).

Окрім того, для атаки апаратних помилок можна використовувати відмови – довготривалі збої, що викликають незворотні перетворенні в структурі пристрою [4]. Недоліком таких атак є втрата працездатності пристрою у нормальному режимі роботи, що вказує на певне обмеження у застосуванні таких атак.

III Потокові шифри

Основу будь-якого потокового шифру складає генератор псевдовипадкових чисел (ГПВЧ), де для виконання шифрування вихідна псевдо випадкова гама накладається операцією XOR на біти відкритого тексту.

Більшість сучасних ГПВЧ побудовані на основі регістрів зсуву, наприклад LFSR, що дозволяє досягати високих показників продуктивності при апаратній реалізації. До них відносять A5, що успішно використовується в стандарті GSM, Panama – в платному цифровому телебаченні, E0 – для захисту комерційних пакетів даних і т. д. [8, 9].

На рис. 1 наведено базову загальну структуру регістру зсуву з лінійними зворотними зв'язками.

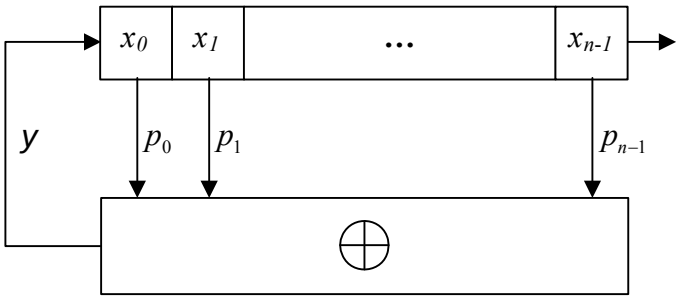

Рисунок 1 – Базова структура LFSR

На рис. 1 прийнято наступні позначення: $\{x_0, ..., x_{n-1}\}$ – заповнення регістру зсуву, $\{P_0, ..., P_{n-1}\}$ – зворотні зв'язки, $\oplus$ – операція виключного АБО (XOR), y – генерований псевдовипадковий біт.

У більшості сучасних реалізацій ГПВЧ, що використовуються в потоковому шифруванні, зворотні зв'язки є фіксованими, а тому секретним ключем виступає тільки початкове заповнення регістрів зсуву. Наприклад, в алгоритмі A5 використовують поліноми $(18,7,0)$, $(21,2,0)$, $(22,1,0)$ [8]. Даний факт спрощує реалізацію аналітичних криптографічних атак на такі криптопристрої, а також, спрощує проведення

криптоатаки аналізу енергоспоживання, базованого на вагах Хемінга, проведеної в [10] на потоковий шифр, побудований з використанням LFSR.

Застосування атак побічних каналів на базові компоненти ГПВЧ, наприклад LFSR, розширить можливості криптоаналітика, тобто дасть змогу зламувати складніші конструкції, а, отже, підвищить успіх при проведенні атак на сучасні реалізації потокових шифрів.

VI Алгоритм диференційної криптоатаки помилок на LFSR

Атака буде спрямована на узагальнену конструкцію, наведену на рис. 1. Модель атаки передбачає, що криптоаналітик має змогу вносити збої у визначені біти регістру зсуву – на певній ітерації роботи встановлювати значення комірки в ‘1’. Для успішного проведення запропонованої атаки криптоаналітику відомі наступні дані:

- ✓ розмірність регістру n та відповідний поліном зворотних зв’язків P;
- ✓ l – біт гами y, де l – максимальна відстань між двома сусідніми зворотними зв’язками в аналізованому LFSR.

Хід атаки полягає у виконанні наступних кроків:

1 організовуємо цикл по $i = 0, \ldots, l - 1$;

2 під час кожної ітерації циклу по i організовується цикл по, $j = 0, \ldots, m - 1$; де m – кількість зворотних зв’язків поліному;

3 кожної ітерації внутрішнього циклу проводимо наступні дії:

- ✓ якщо значення j-тої комірки регістру вже відоме, то пропустити всі операції кроку 3 для даної комірки регістру;
- ✓ “встановлюємо” (вносимо збій) значення j-тої комірки регістру (відповідна комірка регістру, значення якої йде у зворотний зв’язок - впливає на обчислення наступного біту гами);
- ✓ отримуємо $y'(i)$ – біт гами зі збоєм; $y'(i) = y(i)$;
- ✓ порівнюємо значення біту гами у нормальному режимі роботи пристрою $y(i)$ із отриманим значенням режиму роботи зі збоями $y'(i)$. Якщо значення рівні, то відповідне значення комірки регістру $x(j)$ було рівне '1', інакше $x(j)$='0';
- ✓ проводимо перезапуск пристрою, що аналізується (LFSR).

Для успішного проведення атаки криптоаналітик повинен мати вихідну гаму довжиною щонайменше l біт, а також провести n збоїв, отримати в загальному випадку n спотворених однобітних вихідних гам і провести відповідно n порівнянь із відомим еквівалентом. Отже обчислювальна складність (набір операцій кроку 3 запропонованого алгоритму) проведених обчислень буде становити n.

Для дослідження розробленого алгоритму автором розроблено програмне забезпечення (ПЗ), яке імітує проведення даної криптографічної атаки. Дане ПЗ може використовуватися як при навчанні студентів, так і для спеціалістів – при проведенні реальних криптоатак на основі збоїв.

За допомогою такого ПЗ проведено експериментальні дослідження шляхом комп’ютерного моделювання. В табл. 1 наведено приклад роботи запропонованого алгоритму криптоатаки. Для унаочнення роботи алгоритму розмірність регістра зсуву свідомо прийнята малою. На вхід програмного забезпечення подавалися наступні відомості про аналізований LFSR:

1 розмірність регістру зсуву: 4 біти;

2 поліном зворотних зв’язків: „1011”;

3 результуюча гама розмірністю 2 біти: „01”;

Шуканим є заповнення регістру зсуву reg () на будь-якому з кроків його роботи.

Всі позначення в таблиці використано відповідно до наведених вище в алгоритмі атаки, а під кроком операції розуміється послідовність дій кроку 3 із наведеного алгоритму диференційної атаки збоїв на аналізований LFSR.

Відповідно до наданого ПЗ поліному, 0-ва, 2-га та 3-тя комірки регістру впливають на обчислення результату, а, отже, на них має проводитися вплив для внесення збою (встановлення значення відповідної комірки в ‘1’).

Оскільки на кроці операції 4, в змінній reg (), що є шуканим регістром зсуву, всі елементи відомі, то алгоритм завершено, що свідчить про те, що складність проведення атаки рівна розмірності регістру, тобто 4.

Таблиця 1 – Приклад роботи запропонованого алгоритму

i номер такту роботи пристрою	Значення реального регістру зсуву на *i*-му кроці роботи	$y(i)$	j	Значення шуканого регістру зсуву reg ()	$y'(i)$	крок операції	Операції алгоритму
0	1001	0	0	‘1’ X X X	X	1	Внести збій у reg (0) і отримати $y'(i)$
				1 X X X	0		Оскільки $y'(i)=y(i)$, то при нульовому такті роботи reg (0) = ‘1’, а reg = “1XXX”
			1	1 X ‘1’ X	1	2	Внести збій у reg (2) і отримати $y'(i)$
				1 X 0 X	1		Оскільки $y'(i)\neq y(i)$, то при нульовому такті роботи reg (2) = ‘0’ а reg = “1X0X”
			2	1 X 0 ‘1’	0	3	Внести збій у reg (3) і отримати $y'(i)$
				1 X 0 1	0		Оскільки $y'(i)=y(i)$, то при нульовому такті роботи reg (3) = ‘1’ а reg = “1X01”
1	0010	1	0	‘1’ 0 1 0	0	4	Внести збій у reg (0) і отримати $y'(i)$
				0 0 1 0			Оскільки $y'(i)=y(i)$, то при першому такті роботи reg (0) = ‘1’ а reg = “0010”

Як було зазначено вище, обчислювальна складність рівна розмірності аналізованого LFSR. Це підтверджують отримані результати симуляції атаки за допомогою розробленого ПЗ на 12 LFSR розмірностей 8, 32, 19, 24 біти, що наведені в табл. 2.

Якщо під ефективністю розуміти мінімум відомих даних криптоаналітику про аналізований LFSR для реалізації атаки, то ефективність алгоритму атаки буде вищою у тому випадку, коли ми маємо меншу максимальну відстань між двома сусідніми зворотними зв’язками у поліномі аналізованого LFSR, оскільки це безпосередньо впливає на мінімально-необхідний розмір вихідної гами роботи пристрою (табл. 2).

Таблиця 2 – Результати атаки на LFSR довільної розмірності

Варіант	Поліном Кількість біт / десятковий еквівалент	Максимальна відстань між двома сусідніми зворотними зв’язками	Кількість зворотних зв’язків	Розмір відомої (необхідної) гами (біт)	Відома гама	Обчислювальна складність
1	8 / 153	3	4	3	011	8
2	8 / 141	4	4	4	1101	8
3	8 / 135	5	4	5	01000	8
4	32 / 2147483657	28	3	28	000000000000101 0101100011001	32
5	32 / 2147483713	25	3	25	110111010010110 0011011110	32
6	32 / 2147483777	24	3	24	000100000001111 111101010	32
7	32 / 2147491841	18	3	18	110111011111011 100	32
8	32 / 2966973503	5	17	5	11100	32
9	19 / 262183	13	5	13	0010000011001	19
10	19 / 263169	10	3	10	0011100001	19
11	24 / 8388635	19	5	19	000101101111100 0000	24
12	24 / 9472915	5	10	5	11110	24

V Визначення поліному зворотних зв'язків

У випадку, коли криптоаналітику невідома структура зворотних зв'язків, дана атака відповідно ускладнюється.

Для проведення процедури визначення поліному зворотних зв'язків у базовому регістру зсуву з лінійними зворотними зв'язками, криптоаналітику необхідно мати наступні відомості:

- ✓ розмірність аналізованого LFSR n ;
- ✓ один біт результуючої гами y.

У такому випадку алгоритм процедури визначення поліному зворотних зв'язків за допомогою механізмів диференційної атаки помилок містить наступні кроки:

1 організовується цикл по; $i = 0, \ldots, n - 1$

2 кожної ітерації роботи циклу криптоаналітик виконує:

- ✓ внесення збою в $i-$ ту комірку регістру зсуву (у даному випадку збій має носити характер інверсії);
- ✓ генерування поточного псевдовипадкового біту регістром зсуву із збоєм y' ;
- ✓ якщо y та y' еквівалентні, то це свідчить про те, що даний біт регістру зсуву не впливає на обчислення біту результату, а отже від даної комірки регістру зсуву немає зворотного зв'язку;
- ✓ провести перезапуск пристрою.

Для успішного проведення даного алгоритму визначення поліному зворотних зв'язків криптоаналітику достатньо мати відомості про розмірність регістру зсуву, один біт вихідної гами (для порівняння) та провести n експериментів по внесенню збою та порівняння.

Перевагою даного методу є те, що для визначення поліному зворотних зв'язків криптоаналітику необхідно мати тільки один біт вихідної гами роботи аналізованого LFSR, в той час як інший відомий метод – алгоритм Берлекампа-Мессе – вимагає вихідну послідовність довжиною $2 \cdot n$.

VI Диференційна атака збоїв на проріджуючий ГПВЧ

Один LFSR не використовується для генерування псевдовипадкової гами в потоковому шифруванні, оскільки вихідна послідовність не задовольняє вимозі непередбачуваності. В такому випадку для внесення нелінійності використовують спеціальні конструкції. Яскравим прикладом простого за конструкцією (відповідно у апаратній реалізації) та криптографічно стійкого до аналітичних атак є проріджуючий генератор [9, 11], що за певних умов володіє достатньо високою доказуваною криптографічною стійкістю.

В проріджуючому ГПВЧ використовують два базові LFSR, що тактуються одночасно. На вихід генератора подається результуючий біт першого LFSR тоді і тільки тоді, коли вихідний біт другого LFSR рівний '1'. Узагальнена схема такого генератора наведена на рис. 2.

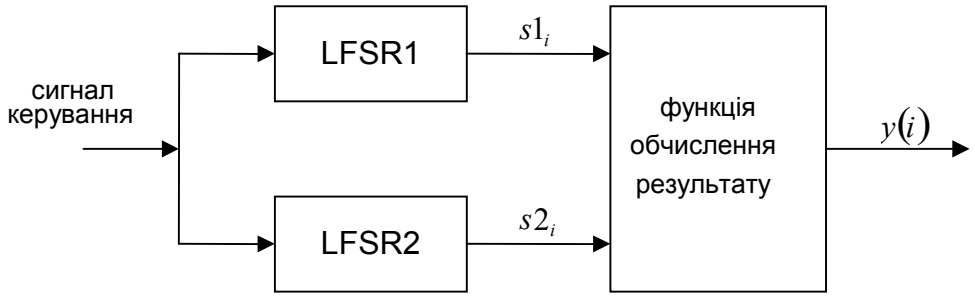

Рисунок 2 – Узагальнена схема проріджуючого ГПВЧ

Атака внесення збоїв може бути успішно застосована проти такої конструкції. Для реалізації атаки користувачу необхідно мати наступні відомості:

- ✓ довжини $n1$ та $n2$ – розмірності базових LFSR1 та LFSR2, що використовуються в проріджуючому ГПВЧ;
- ✓ поліноми зворотних зв'язків LFSR1 та LFSR2;
- ✓ вихідна послідовність y_0, K, y_x цілого ГПВЧ, довжина якої рівна x – розмірності більшого із регістрів ($x = \max(n1, n2)$

Шуканим є ключ – початкове заповнення обох LFSR, що використовується в ГПВЧ. Атака складається

з двох частин. Перша призначена для отримання початкового заповнення регістру LFSR1 і містить наступні кроки:

1 організовуємо цикл по $i = 0, \ldots, n1 - 1$;

2 кожної ітерації циклу вноситься збій, який встановлює вихідний біт LFSR2 в значення ‘1’;

3 отримуємо $s1(i)$ – вихідний біт ГПВЧ, який при успішному виконанні кроку 2 буде рівний вихідному біту із регістру LFSR1;

4 по завершенню циклу провести пере запуск пристрою.

В результаті проведення $n1$ тактів роботи пристрою та, відповідно, $n1$ внесених збоїв у останній біт регістру LFSR2 отримується послідовність, $s1_0$, …, $s1_{n1-1}$, яка рівна початковому заповненню LFSR1 на початку роботи пристрою.

Отже, складність процедури визначення початкового заповнення LFSR1 становить $n1$.

Друга частина запропонованої атаки призначена для визначення початкового заповнення регістру LFSR2. Для її реалізації необхідно:

- ✓ всі відомості про LFSR1: розмірність $n1$, початкове заповнення, поліном зворотних зв’язків; це дозволить повністю прогнозувати його роботу;
- ✓ розмірність LFSR2 – $n2$.

Шуканим є початкове заповнення LFSR2 – $s2$. Для початку перевіряємо яке співвідношення одиниць та нулів може дати LFSR1 протягом $n2$ кроків роботи. Якщо більше одиниць, то в алгоритмі атаки у всіх випадках, коли вихідний біт LFSR1 буде мати значення ‘0’, то збій буде вноситися з метою переводу значення комірки в ‘1’ і навпаки. Нехай, протягом $n2$ кроків роботи LFSR1 більше на виході буде значень типу ‘1’, тоді друга частина атаки містить наступні кроки:

1 організовуємо цикл по $i = 0, \ldots, n2 - 1$;

2 кожного кроку циклу перевіряється умова

якщо $s1(i) = '0'$, де $s1(i)$ - вихідний біт у LFSR1

тоді внести збій у вихідний біт LFSR1 – встановити його в ‘1’;

3 отримати вихідний біт гами ГПВЧ $y'(k)$; $k = k + 1$.

У такому випадку, після завершення алгоритму, отримаємо послідовність $y'(\ )$, що буде містити набір одиничних бітів кількістю k. Виходячи з цього:

- ✓ значення k буде рівне кількості одиничних бітів у початковому заповненні LFSR2 перед початком роботи алгоритму; а відповідно, виконується нерівність $0 \le k < n2$.
- ✓ отже $k0 = n2 - k$, де $k0$ - кількість нульових бітів у початковому заповненні LFSR1.

Складність реалізації запропонованого алгоритму у термах кількості операцій буде рівна $n2$, проте кількість необхідних при цьому збоїв рівна приблизно $n2/2$. В результаті виконання наведеного вище алгоритму криптоаналітик отримує кількість нулів та одиниць в початковому заповненні LFSR2, тобто вагу Хемінга заповнення регістру зсуву. Як відомо, загальну кількість розміщень в $n2$ комірках k бітів, що мають значення ‘1’ можна обчислити за формулою:

$$C_{n2}^{k} = \frac{n2!}{k! \cdot (n2 - k)!} \tag{1}$$

Зрозуміло, що кількість таких можливих варіантів внутрішнього заповнення LFSR2 буде максимальною у тому випадку, коли буде рівномірний розподіл значень типу ‘0’ і ‘1’. Проте, обчислювальна складність запропонованого алгоритму є значно меншою, ніж при використанні прямого перебору можливого заповнення регістру.

Наприклад, якщо використовується LFSR2 розмірності 64 біти, то при повному переборі внутрішнього заповнення криптоаналітик повинен переглянути 2^{64} можливих варіантів внутрішнього заповнення, а при використанні запропонованого методу необхідно максимум 2^{61} варіантів.

В такому випадку, складність проведення двох частин атаки на основі внесення збоїв на конструкцію проріджуючого ГПВЧ буде рівна $n1 + n2$. Також необхідно отримати C_{n2}^{k} наборів можливого заповнення LFSR2 та за допомогою відомої відкритої гами остаточно визначити внутрішнє заповнення LFSR2.

У [9, 11] показано, що при невідомих поліномах зворотних зв’язків найкращий із відомих методів

криптоаналізу конструкцій проріджуючих генераторів вимагає проходження $O\left(2^{n2}\cdot n1^3\right)$ операцій. При невідомих поліномах зворотних зв'язків складність атаки значно зростає. Також є підхід криптоаналізу, базований на лінійній складності проріджуючого ГПВЧ (незалежно від того, чи є відомості про поліноми зворотних зв'язків), що також вимагає проходження $O\left(2^{n2}\cdot n1^3\right)$ операцій, проте криптоаналітик повинен мати $n1\cdot 2^{n2}$ послідовних біт вихідної послідовності. Запропонований в даній роботі підхід криптографічної атаки на конструкцію проріджуючого ГПВЧ має набагато меншу обчислювальну складність, про що свідчать дані, наведені в табл. 3 та на рис. 3, а також вимагає значно меншу довжину вихідної гами генератора.

Таблиця 3 – Результати атаки на LFSR довільної розмірності

№	Розмір LFSR1, n1	Розмір LFSR2, n2	Кількість '1' в початковому заповненні LFSR2, k	Складність реалізації традиційної криптоатаки, O()	Складність реалізації запропонованого алгоритму атаки, O'()
1	16	64	8	7,55579E+22	4426165368
2	24	64	12	2,55008E+23	3,28421E+12
3	32	64	16	6,04463E+23	4,88527E+14
4	40	64	20	1,18059E+24	1,96197E+16
5	48	64	24	2,04006E+24	2,50649E+17
6	56	64	28	3,23954E+24	1,11877E+18
7	64	64	32	4,8357E+24	1,83262E+18

На рис. 3 по осі y наведено обчислювальну складність атак; по осі x для традиційної атаки O() наведено відносно варіанту розмірності LFSR1, а для запропонованої O'() – відносно ваги Хемінга в початковому заповненні LFSR2.

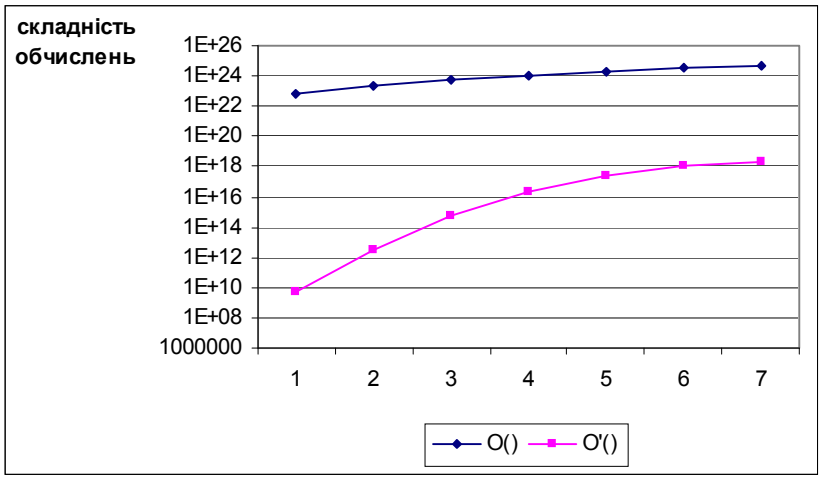

Рисунок 3 – Графік залежності обчислювальної складності проведення традиційної та запропонованої атак

VII Висновки

Запропоновані алгоритми атаки на базову компоненту LFSR та на конструкцію проріджуючого ГПВЧ, що є актуальним як для розробників криптографічних засобів захисту інформації так і для криптоаналітиків. Також розроблене програмне забезпечення, що імітує проведення диференційної атаки збоїв на базовий LFSR.

Ефективність запропонованого алгоритму атаки буде вищою, якщо аналізований LFSR матиме не проріджений поліном зворотних зв'язків, з невеликою максимальною відстанню між двома сусідніми зворотним зв'язками у поліномі, оскільки розмір необхідної результуючої гами для проведення атаки відповідно зменшується.

Вагомою перевагою запропонованого алгоритму диференційної атаки збоїв на базовий LFSR є те, що для визначення поліному зворотних зв'язків криптоаналітику необхідно мати тільки один біт вихідної гами роботи аналізованого LFSR, в той час як інший відомий метод – алгоритм Берлекампа-Мессе –

вимагає вихідну послідовність довжиною $2n$.

Запропонований алгоритм атаки на конструкцію проріджуючого ГПВЧ в деякій мірі є адаптацією і загальненням запропонованих ідей алгоритму диференційної атаки збоїв на базовий LFSR. Це свідчить про те, що диференційна атака на основі внесення збоїв може бути перенесена на більшість конструкцій потокових шифрів.

Основним недоліком запропонованих атак є те, що модель атаки є надзвичайно строгою. Крім основної вимоги, що накладається атаками побічних каналів – безпосередній доступ до пристрою, криптоаналітик також повинен вміти направлено вносити збої у конкретну комірку аналізованого регістру зсуву. Така строгість вносить певну додаткову складність у практичну реалізацію таких атак, проте отримані в ході дослідження науково-практичні результати можуть слугувати базисом для розробки диференційних атак на основі збоїв на реальні криптографічні пристрої, що реалізують потокове шифрування.

*Література:**1.** J. Kelsey, B. Schneier, D. Wagner, and C. Hall, "Side Channel Cryptanalysis of Product Ciphers," in Proceedings of ESORICS '98 , Springer- Verlag, September 1998, pp. 97- 110. **2.** James Alexander Muir. Techniques of side channel cryptanalysis // University of Waterloo. Dept. of Combinatorics and Optimization, 2001. **3** Sergei P. Skorobogatov, Ross J. Anderson. Optical Fault Induction Attacks // Proceedings of the 4th International Workshop Redwood Shores, CA, USA, August pages 2-12. **4.** Eli Biham , Adi Shamir, Differential Fault Analysis of Secret Key Cryptosystems // Proceedings of the 17th Annual International Cryptology Conference on Advances in Cryptology, p.513-525, August 17-21, 1997. **5.** P. Kocher. Timing attacks on implementations of Diffie-Hellman, RSA, DSS and other systems. In N. Koblitz (ed.), Advances in Cryptology — Crypto '96, Lecture Notes in Computer Science, vol. 1109, pages 104-113, Springer-Verlag, 1996. **6.** D. Boneh, R. DeMillo, and R. Lipton, "On the importance of checking cryptographic protocols for faults", Proceedings of Eurocrypt, Lecture Notes in Computer Science, Springer-Verlag, LNCS 1233, pp. 37_51, 1997. **7.** Jonathan J. Hoch, Adi Shamir, Fault Analysis of Stream Ciphers, Lecture Notes in Computer Science, Volume 3156, Jul 2004, Pages 240 – 253. **8.** Schneier B. Applied cryptography. – N.Y.: John Wiley & Songs Inc., 1996. – 757 p. **9.** Menezes A., Oorshot P., Vanstone S. Handbook of applied cryptography. – N.Y: CRC Press Inc., 1996. – 816 p. **10.** Valery Shyrochyn, Ihor Vasyltsov, Bohdan Karpinskij. Hemming Weight Power Analysis of LFSR-based Stream Ciphers // Матеріали VIII міжнародної науково-технічної конференції "Досвід розробки та застосування приладо-технологічних САПР в мікроелектроніці" CADSM 2005, 23-26 лютого, 2005, Львів-Поляна, Україна, ст. 168-171. **11.** http://ssl.stu.neva.ru/psw/crypto/ potok/str_ciph.htm.

УДК 681.3.06

МЕТОД ПОДНЯТИЯ ЭЛЛИПТИЧЕСКОЙ КРИВОЙ ПРИ ПОСТРОЕНИИ ЕЁ ПАРАМЕТРОВ С ИСПОЛЬЗОВАНИЕМ МОДУЛЯРНОГО ПОЛИНОМА

Ольга Илясова

Харьковский национальный университет радиоэлектроники

Анотація: **Розглянуто процес підняття еліптичної кривої за допомогою модулярного полінома. Доведено, що складність підняття еліптичної кривої поліноміальная.**

Summary: **The process of lifting elliptic curve with the modular polinomial was considered. At his been proved that the complication of lifting the elliptic curve is polinomial.**

Ключевые слова: **Эллиптическая кривая, порядок эллиптической кривой, поднятие кривой, модулярный полином.**

Введение

Развитие информационных технологий, способствовало возникновению и развитию электронного бизнеса. Два из наиболее важных условий существования электронного бизнеса – информационная безопасность и аутентификация. Задачи проверки целостности документа и подлинности его автора решаются при использовании ассиметричных криптосистем [1].

В настоящее время криптосистемы RSA и Эль Гамаля и их различные модификации исчерпывают свои возможности [2, 3]. Альтернативой этим системам являются криптосистемы, основанные на преобразованиях в группе точек эллиптической кривой. Такие системы в последнии годы получили широкое распространение в электронной комерции. Это связано с тем, что эти криптосистемы