Таблица 10 – Относительная сложность алгоритма *Log Map* по сравнению с *Max Log Map*

	q			
m	2	3	4	5
2	1,8	1,7	1,7	1,6
3	1,9	1,8	1,7	1,6
4	1,9	1,8	1,7	1,7
5	1,9	1,8	1,7	1,7
6	1,9	1,8	1,7	1,7
7	1,9	1,8	1,8	1,7
8	1,9	1,8	1,8	1,7
9	1,9	1,8	1,8	1,7

V Выводы

- Для адекватного сравнения сложности реализации различных алгоритмов декодирования ТК алгебраические операции, необходимые для декодирования бита информации, должны быть представлены как элементарные для ЦСП.
- Сравнительный анализ алгоритмов декодирования показал, что самым сложным является *MAP* алгоритм, т. к., например, при $m = 3$ и $q = 3$ сложность этого алгоритма равна $\phi = 2620$ элементарных операций, и, как показал дальнейший анализ, он в 9,9 раза сложнее *Max Log Map* алгоритма и в 5,5 раза *Log Map* алгоритма.
- Полученные выражения целесообразно использовать для анализа сложности реализации кодера и декодера ТК на типовых ЦСП с целью выбора элементной базы для реализации ТК.

Литература: ***1.*** *Gracie K., Crozier S., Hunt A. Performance of a Low-Complexity Turbo Decoder with a Simple Early Stopping Criterion Implemented on a SHARC Processor // Communications Research Centre (*http://www.crc.ca/fec*).* ***2.*** *Gracie K., Crozier S., Hunt A. Performance of a Low-Complexity Turbo Decoder and its Implementation on a Low-Cost, 16-Bit Fixed-Point DSP // Communications Research Centre (http://www.crc.ca/fec).* ***3.*** *Руководство пользователя по сигнальным процессорам семейства SHARC ADSP-2106 x // Пер. с англ. Бархатов А. В., Коновалов А. А., Петров М. Н. ООО ЭЛТЕХ г. Санкт-Петербург. – 2002. С. 493 – 539.* ***4.*** *Srinivas K. Low-Cost Simd SHARC // Analog Devices, 2001. – P. 24 – 25 (http://www.analog.com/processors).* ***5.*** *Robertson P., Villebrun E., Hoeher P. Optimal and sub-optimal maximum a posteriori algorithms suitable for turbo decoding // Institute of communications technology. – Oberpfaffenhofen, Germany. – P. 4 – 9, 14.* ***6.*** *Malardel F. Simulation and Optimisations of the Turbo Decoding Algorithm // Signal Processing Research Institute – University of South Australia, 1996. – July-November. – P. 23 – 26.* ***7.*** *Золотарев В. В., Овечкин Г. В. Помехоустойчивое кодирование. Методы и алгоритмы: Справочник // Москва: Горячая линия – Телеком. – 2001. С. 104.* ***8.*** *Ливенцев С. П., Зайцев С. В., Горлинский Б. В. Анализ сложности Map, Max Log Map, Log Map алгоритмов декодирования турбокодов при декодировании бита информации // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – 2006. – № 1 (12). – С. 119 – 129.* ***9.*** *Ливенцев С. П., Алексеев Д. А., Зайцев С. В. Анализ характеристик перемежителей, используемых в турбокодах // Зв'язок. – 2005. – № 3. – С. 57 – 61.* ***10.*** *Фильчаков П. Ф. Справочник по высшей математике // "Наукова думка", Киев. – 1972. – 456 с.*

УДК 004.31, 004.056.55, 003.26

СТАТИСТИЧНІ МОДЕЛІ ДВОМІСНИХ ЛОГІЧНИХ ОПЕРАЦІЙ ДЛЯ ПРОВЕДЕННЯ ІНЖЕНЕРНО-КРИПТОГРАФІЧНИХ АТАК ЗА ПОБІЧНИМИ КАНАЛАМИ ВИТОКУ ІНФОРМАЦІЇ

Микола Карпінський*, Леся Коркішко

**Університет в Бєльську-Бяла, Польща, Тернопільський державний економічний університет*

Анотація: **Запропоновано статистичні моделі, досліджено їх властивості та запропоновано методику**

проведення інженерно-криптографічних атак за побічними каналами витоку інформації на комп'ютерні реалізації логічних операцій двох змінних.
Summary: **For realization of logical operations with two arguments the statistical models have been proposed. Their properties were investigated and a method for the side-channel attack has been proposed.**
Ключові слова: **Інженерно-криптографічні атаки, статистична модель, витік інформації, двомісна логічна операція, аналіз споживаної потужності.**

Вступ

Одним із можливих шляхів вирішення задачі захисту даних від неавторизованого доступу є криптографічні перетворення цих даних, зокрема шифрування, з використанням конфіденційної інформації – ключа. Загроза отримання несанкціонованого доступу до таких даних виникає за умов компрометування алгоритму криптографічного перетворення чи компрометування (отримання неавторизованого доступу) ключа. Сучасні алгоритми криптографічних перетворень піддаються всебічному математичному аналізу з метою виявлення можливих місць компрометування і отримання доступу до невідомого ключа. Альтернативним шляхом компрометування алгоритмів криптографічних перетворень є компрометування їх апаратної чи програмної реалізації за допомогою пасивного інженерно-криптографічного аналізу з використанням інформації, що отримується шляхом спостереження за роботою пристроїв, які реалізують криптографічні перетворення [1 – 8]. Часто такий вид аналізу називають інженерно-криптографічними атаками.

Для атакування засобів реалізації алгоритмів криптографічних перетворень використовують різні побічні канали витоку інформації (побічні канали). Такі побічні канали утворюються внаслідок залежності деяких параметрів засобів реалізації алгоритмів криптографічних перетворень від використаного ключа. Прикладами цих параметрів є: час виконання криптографічних операцій, споживана потужність пристрою, електромагнітне випромінювання пристрою [1 – 8]. Відповідні атаки, засновані на аналізі наведених параметрів, отримали назви “часовий аналіз”, “аналіз за споживаною потужністю”, “електромагнітний аналіз”. Результативність цих атак основана на залежності інформації, отриманої з побічних каналів, від ключів, які використовуються у криптографічних перетвореннях. Одним із джерел витоку такої інформації є сигнал про споживану потужність пристрою, який реалізує криптографічне перетворення. Тому актуальною задачею при розробці комп’ютерних пристроїв для реалізації криптографічних перетворень є мінімізація кількості інформації, доступної за побічними каналами її витоку, зокрема, інформації про статистичні відмінності споживаної потужності пристрою залежно від опрацьовуваних даних.

Виконання інженерно-криптографічного аналізу комп’ютерних засобів реалізацій алгоритмів криптографічних перетворень висвітлено в літературі, наприклад, DES [3, 4, 6], RSA [2, 5], AES [6], ГОСТ 28147-89 [9] тощо. Базою для проведення цього аналізу є інженерно-криптографічні атаки з використанням статистичних моделей складових операцій криптографічних перетворень та моделі витоку інформації з комп’ютерних засобів. В основі сучасних алгоритмів криптографічних перетворень використовуються як арифметичні операції, так і логічні операції перетворення даних [10]. Відомі роботи з побудови статистичних моделей деяких операцій. Наприклад, статистична модель для побітової операції додавання за модулем 2 є достатньо добре описаною та дослідженою в [7, 8], статистична модель операції додавання за модулем 2^N подана в [11]. Разом з тим, існують й інші двомісні логічні операції, які використовуються для обробки даних і ключів у комп’ютерних засобах реалізації алгоритмів криптографічних перетворень. Тому метою даної роботи є розв’язання задачі розробки статистичних моделей та атакування реалізацій додаткового переліку двомісних логічних операцій, зокрема логічного множення, логічного додавання, логічного інвертування, тощо.

Розроблені статистичні моделі базуються на лінійній моделі витоку інформації про Хемінгову вагу даних (які обробляються згідно з [8]), і дозволяють проводити пасивні інженерно-криптографічні атаки з використанням інформації про споживану потужність комп’ютерного пристрою, який реалізує логічні операції перетворення даних. Розроблені статистичні моделі очікуваного значення Хемінгової ваги результату виконання логічних операцій двох змінних можна використати для дослідження та сертифікування комп’ютерних засобів, які реалізують алгоритми криптографічних перетворень з використанням цих операцій [12].

I Модель витоку інформації для проведення пасивних інженерно-криптографічних атак

Сучасні комп'ютерні пристрої для виконання алгоритмів криптографічних перетворень реалізуються як програмно, так і апаратно на інтегральних мікросхемах. Найбільш поширеною технологією виготовлення цих мікросхем є КМОН (комплементарний метало-оксидний напівпровідник) технологія (англійський

відповідник – CMOS – complementary metal-oxide semiconductor). Особливістю роботи інтегральних мікросхем, виконаних за КМОН технологією, є залежність їх споживаного струму від значень, які записуються в елементи пам'яті (регістри) мікросхеми. При обробленні інформації з використанням алгоритмів криптографічних перетворень в елементи пам'яті записуються результати, отримані при виконанні деякої операції з використанням відомих даних і конфіденційних даних. Таким чином, зміна споживаного струму при записуванні такого результату несе в собі інформацію про результат виконання цієї операції. Зокрема, як було показано в [3, 8], значення споживаного струму залежить від Хемінгової ваги результату – пристрій споживає більший струм при обробці даних з більшою Хемінговою вагою.

З іншого боку, оскільки один аргумент операції є відомим (можна довільно маніпулювати відкритими даними), виконувана операція теж є відомою, то значення споживаного струму при записуванні результату буде залежати й від другого аргументу операції – конфіденційних даних. Таким чином, споживаний пристроєм струм при роботі з конфіденційними даними створює передумови для виникнення каналу витоку інформації про ці конфіденційні дані (невідомий аргумент операції) при відомому другому аргументі операції.

Пряме вимірювання споживаного струму не є зручним для проведення атак на комп'ютерні засоби реалізації криптографічних перетворень. Тому використовується вимірювання напруги на резисторі, який ставиться в розрив кола живлення комп'ютерного пристрою. Така техніка проведення атаки в літературі отримала назву "пасивної інженерно-криптографічної атаки за споживаною потужністю" [8].

Надалі, згідно з [8], приймемо, що:

- аргументами двомісних логічних операцій є таємні дані K і відкритий текст P;
- комп'ютерний пристрій, який реалізує алгоритм криптографічного перетворення з використанням двомісних логічних операцій, уможливлює витік інформації про Хемінгову вагу результату S;
- залежність споживаного струму від Хемінгової ваги наближено описується лінійною залежністю.

Нехай споживання потужності в момент часу j подана у вигляді $P[j]$. Тоді для опису залежності споживаної пристроєм потужності від Хемінгової ваги проміжних даних, які обробляються, чи результатів, скористаємося лінійною залежністю, запропонованою у [8]:

$$P[j] = \varepsilon \cdot d[j] + L + n, \tag{1}$$

де $d[j]$ репрезентує Хемінгову вагу результату, який отримується у момент часу j, ε – внесок у споживану потужність кожної одиниці Хемінгової ваги даних, L – споживана постійна загальна потужність, n – шум з нульовим середнім значенням.

II Атака на реалізацію логічної операції додавання за модулем 2

Нехай j позначає момент часу, коли виконується операція додавання за модулем 2. Тоді сума $S = K \oplus P$, де K – N-бітовий невідомий доданок, P – N-бітовий відкритий текст. Розглянемо запропоновану в [8] атаку на N-бітовий суматор за модулем 2, метою якої є визначення бітів K без відомостей про значення бітів S. Припустимо, що залежність між споживаною потужністю в момент часу j і Хеммінговою вагою результату, який отримується, описується виразом (1). Тоді узагальнений алгоритм атаки на реалізацію операції додавання за модулем 2 є таким:

```
Для і від 0 до N-1 {
      Для b=0 до 1 {
            Обчислити усереднене значення сигналу споживаної потужності Ab[j]
      {
            Встановити і-й біт P рівним b;
            Встановити решту бітів P у випадкові значення;
            Зібрати дані про споживану потужність пристрою;
            }
      }
     Обчислити диференційний сигнал T[j] = A0[j] – A1[j];
     Якщо T[j] > 0, то і-й біт K є "1", якщо T[j] < 0 то і-й біт K є "0";
}
```

Результативність цієї атаки базується на незалежності очікуваного значення Хемінгової ваги результату додавання за модулем 2 від позиції біту, який піддається аналізу. Тут очікуване значення E Хемінгової ваги d залежить лише від комбінації значень бітів k_i і p_i та розрядності N суматора за модулем 2 [8]:

$$E\left[d\middle|k_i \oplus p_i = 0\right] = \frac{N-1}{2}, \tag{2}$$

$$E\left[d\middle|k_i \oplus p_i = 1\right] = \frac{N+1}{2}. \tag{3}$$

Якщо $k_i = 0$, то для моменту часу *j** виконання операції додавання за модулем 2 вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і *P* з (1). З урахуванням (2) і (3) отримуємо:

$$A_0[j^*] \approx E\left[P\middle|k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 0\right] = \frac{N-1}{2} \cdot \varepsilon + L, \tag{4}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 1\right] = \frac{N+1}{2} \cdot \varepsilon + L. \tag{5}$$

Тоді значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon, \text{ за умови, що } k_i = 0. \tag{6}$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$. Тоді значення диференційного сигналу становитиме

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx \varepsilon, \text{ за умови, що } k_i = 1. \tag{7}$$

Отже, з виразів (6) і (7) випливає, що диференційний сигнал буде містити додатній пік за умови $k_i = 1$ і від'ємний пік за умови $k_i = 0$.

III Статистичні моделі та атаки на реалізації логічних операцій над двома змінними

Розглянуту вище атаку на реалізацію логічної операції додавання за модулем 2 можна використати для атакування інших двомісних логічних операцій. Результативність такого атакування, на відміну від [11], буде ґрунтуватися на незалежності очікуваного значення Хемінгової ваги результату від позиції біту результату, який піддається аналізу. Даний висновок зумовлений тим, що двомісні логічні операції над *N*-бітовими аргументами виконуються побітово і значення бітів результату залежать лише від значень відповідних бітів аргументів логічних операцій.

Для проведення атаки на двомісні логічні операції необхідні статистичні моделі очікуваного значення Хемінгової ваги результату виконання цих операцій, аналогічні до (2) – (5) з тим, щоб можна було побудувати вирази для обчислень диференційного сигналу, аналогічні до (6) і (7).

Серед двомісних логічних операцій оберемо ті, які використовуються при побудові обчислювальних пристроїв для реалізації криптографічних перетворень. Оскільки двомісна логічна операція додавання за модулем 2 досліджена в [8], розглянемо інші операції: логічного множення, логічного додавання, штрих Шеффера (інверсії логічного множення), стрілка Пірса (інверсії логічного додавання), імплікації та еквівалентності (інверсії додавання за модулем 2).

3. 1 Операція логічного множення

Для операції логічного множення очікуване значення *E* Хемінгової ваги *d* залежить лише від комбінації значень бітів k_i і p_i та розрядності *N* операційного пристрою логічного множення:

$$E\left[d\middle|k_i \cdot p_i = 0\right] = \frac{N-1}{4}, \tag{8}$$

$$E\left[d\middle|k_i \cdot p_i = 1\right] = \frac{N+3}{4}. \tag{9}$$

Якщо $k_i = 0$, то для моменту часу j* виконання операції логічного множення вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і P з (1). З врахуванням (8) і (9) отримуємо:

$$A_0[j^*] \approx E\left[P\middle|k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 0\right] = \frac{N-1}{4} \cdot \varepsilon + L, \tag{10}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 1\right] = \frac{N-1}{4} \cdot \varepsilon + L\,. \tag{11}$$

Тоді, підставивши значення (10) і (11) у (6), знайдемо значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx 0\,, \text{ за умови, що } k_i = 0\,. \tag{12}$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$:

$$A_0[j^*] \approx E\left[P\middle|k_i = 1, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 0\right] = \frac{N-1}{4} \cdot \varepsilon + L\,, \tag{13}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 1, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 1\right] = \frac{N+3}{4} \cdot \varepsilon + L\,. \tag{14}$$

Шляхом підстановки значень (13) і (14) у (7) отримуємо значення диференційного сигналу:

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon\,, \text{ за умови, що } k_i = 1\,. \tag{15}$$

Отже, з виразів (12) і (15) випливає, що диференційний сигнал не буде містити піку за умови $k_i = 0$ і від'ємний пік за умови $k_i = 1$.

3. 2 Операція логічного додавання

Для операції логічного додавання очікуване значення *E* Хемінгової ваги *d* залежить лише від комбінації значень бітів k_i і p_i та розрядності *N* операційного пристрою логічного додавання:

$$E\left[d\middle|k_i \vee p_i = 0\right] = \frac{3(N-1)}{4}\,, \tag{16}$$

$$E\left[d\middle|k_i \vee p_i = 1\right] = \frac{3N+1}{4}\,. \tag{17}$$

Якщо $k_i = 0$, то для моменту часу *j** виконання операції логічного додавання вирази для $A_0[j^*]$ і $A_1[j^*]$ можна записати через очікувані значення Хемінгової ваги суми і *P* з (1). З урахуванням (16) і (17) отримуємо:

$$A_0[j^*] \approx E\left[P\middle|k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 0\right] = \frac{3(N-1)}{4} \cdot \varepsilon + L\,, \tag{18}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 1\right] = \frac{3N+1}{4} \cdot \varepsilon + L\,. \tag{19}$$

Тоді, підставивши значення (18) і (19) у (6), отримаємо значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon\,, \text{ за умови, що } k_i = 0\,. \tag{20}$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$:

$$A_0[j^*] \approx E\left[P\middle|k_i = 1, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 0\right] = \frac{3N+1}{4} \cdot \varepsilon + L\,, \tag{21}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 1, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 1\right] = \frac{3N+1}{4} \cdot \varepsilon + L\,. \tag{22}$$

Шляхом підстановки значень (21) і (22) у (7) отримаємо диференційний сигнал

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx 0\,, \text{ за умови, що } k_i = 1\,. \tag{23}$$

Отже, з виразів (20) і (23) випливає, що диференційний сигнал міститиме від'ємний пік за умови $k_i = 0$ і не міститиме піку за умови $k_i = 1$.

3. 3 Операція штрих Шеффера (інверсія логічного множення)

Для операції штрих Шеффера (інверсії логічного множення) очікуване значення *E* Хемінгової ваги *d* залежить лише від комбінації значень бітів k_i і p_i, розрядності *N* операційного пристрою інверсії логічного множення та є аналогічним до очікуваного значення Хемінгової ваги для операції логічного додавання:

$$E\left[d\middle|k_i \mid p_i = 0\right] = \frac{3(N-1)}{4}, \tag{24}$$

$$E\left[d\middle|k_i \mid p_i = 1\right] = \frac{3N+1}{4}. \tag{25}$$

Однак, вигляд диференційних сигналів буде відрізнятися від вигляду диференційних сигналів для операції логічного додавання. Якщо $k_i = 0$, то для моменту часу *j** виконання операції інверсії логічного множення вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і P з (1). З врахуванням (24) і (25) отримаємо:

$$A_0[j^*] \approx E\left[P\middle|k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 0\right] = \frac{3N+1}{4} \cdot \varepsilon + L, \tag{26}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 1\right] = \frac{3N+1}{4} \cdot \varepsilon + L. \tag{27}$$

Тоді, підставивши значення (26) і (27) у (6), знайдемо значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx 0, \text{ за умови, що } k_i = 0. \tag{28}$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$:

$$A_0[j^*] \approx E\left[P\middle|k_i = 1, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 0\right] = \frac{3N+1}{4} \cdot \varepsilon + L, \tag{29}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 1, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 1, p_i = 1\right] = \frac{3(N-1)}{4} \cdot \varepsilon + L. \tag{30}$$

Тоді підставивши значення (29) і (30) у (7), отримаємо значення диференційного сигналу

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx \varepsilon, \text{ за умови, що } k_i = 1. \tag{31}$$

Отже, з виразів (28) і (31) випливає, що диференційний сигнал не буде містити піку за умови $k_i = 0$ і буде містити додатній пік за умови $k_i = 1$.

3. 4 Операція стрілка Пірса (інверсія логічного додавання)

Для операції стрілка Пірса (інверсії логічного додавання) очікуване значення E Хемінгової ваги d залежить лише від комбінації значень бітів k_i і p_i, розрядності N операційного пристрою інверсії логічного додавання та є аналогічним до очікуваних значень Хемінгової ваги для операції логічного множення (8) і (9):

$$E\left[d\middle|k_i \downarrow p_i = 0\right] = \frac{N-1}{4}, \tag{32}$$

$$E\left[d\middle|k_i \downarrow p_i = 1\right] = \frac{N+3}{4}. \tag{33}$$

Однак, вигляд диференційних сигналів буде відрізнятися від вигляду диференційних сигналів для операції логічного множення. Якщо $k_i = 0$, то для моменту часу *j** виконання операції інверсії логічного додавання вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і P з (1). З врахуванням (32) і (33) отримаємо:

$$A_0[j^*] \approx E\left[P\middle|k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 0\right] = \frac{N+3}{4} \cdot \varepsilon + L, \tag{34}$$

$$A_1[j^*] \approx E\left[P\middle|k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n\middle|k_i = 0, p_i = 1\right] = \frac{N-1}{4} \cdot \varepsilon + L. \tag{35}$$

Тоді, підставивши (34) і (35) у (6), отримаємо диференційний сигнал

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx \varepsilon, \text{ за умови, що } k_i = 0. \tag{36}$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$:

$$A_0[j^*] \approx E\left[P\middle|k_i=1, p_i=0\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=1, p_i=0\right] = \frac{N-1}{4}\cdot\varepsilon + L, \quad (37)$$

$$A_1[j^*] \approx E\left[P\middle|k_i=1, p_i=1\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=1, p_i=1\right] = \frac{N-1}{4}\cdot\varepsilon + L. \quad (38)$$

Шляхом підстановки значень (37) і (38) у (7) отримаємо диференційний сигнал

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx 0, \text{ за умови, що } k_i = 1. \quad (39)$$

Отже, з виразів (36) і (39) випливає, що диференційний сигнал буде містити додатній пік за умови $k_i = 0$ і не містити піку за умови $k_i = 1$.

3. 5 Операція імплікації

Для операції імплікації очікуване значення *E* Хемінгової ваги *d* залежить лише від комбінації значень бітів k_i і p_i та розрядності *N* операційного пристрою логічної імплікації та є аналогічним до очікуваного значення Хемінгової ваги для операції логічного додавання:

$$E\left[d\middle|k_i \to p_i = 0\right] = \frac{3(N-1)}{4}, \quad (40)$$

$$E\left[d\middle|k_i \to p_i = 1\right] = \frac{3N+1}{4}. \quad (41)$$

Вигляд диференційних сигналів буде аналогічним до вигляду диференційних сигналів для операції логічного додавання. Якщо $k_i = 0$, то для моменту часу *j** виконання операції логічного додавання вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і *P* з (1). З врахуванням (40) і (41) отримуємо:

$$A_0[j^*] \approx E\left[P\middle|k_i=0, p_i=0\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=0, p_i=0\right] = \frac{3N+1}{4}\cdot\varepsilon + L, \quad (42)$$

$$A_1[j^*] \approx E\left[P\middle|k_i=0, p_i=1\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=0, p_i=1\right] = \frac{3N+1}{4}\cdot\varepsilon + L. \quad (43)$$

Тоді, підставивши значення (42) і (43) у (6), знайдемо значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx 0, \text{ за умови, що } k_i = 0.$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$:

$$A_0[j^*] \approx E\left[P\middle|k_i=1, p_i=0\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=1, p_i=0\right] = \frac{3(N-1)}{4}\cdot\varepsilon + L, \quad (45)$$

$$A_1[j^*] \approx E\left[P\middle|k_i=1, p_i=1\right] = E\left[d\cdot\varepsilon + L + n\middle|k_i=1, p_i=1\right] = \frac{3N+1}{4}\cdot\varepsilon + L. \quad (46)$$

Тоді підставивши значення (45) і (46) у (7), отримаємо значення диференційного сигналу

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon, \text{ за умови, що } k_i = 1. \quad (47)$$

Отже, з виразів (44) і (47) випливає, що диференційний сигнал не буде містити піку за умови $k_i = 0$ і буде містити від'ємний пік за умови $k_i = 1$.

3. 6 Операція еквівалентності (інверсія додавання за модулем 2)

Для операції еквівалентності (інверсії додавання за модулем 2) очікуване значення *E* Хемінгової ваги *d* залежить лише від комбінації значень бітів k_i і p_i та розрядності *N* суматора і є аналогічним до очікуваного значення Хемінгової ваги для операції додавання за модулем 2:

$$E\left[d\middle|k_i \equiv p_i = 0\right] = \frac{N-1}{2}, \quad (48)$$

$$E\left[d\middle|k_i \equiv p_i = 1\right] = \frac{N+1}{2}. \quad (49)$$

Однак, вигляд диференційних сигналів буде відрізнятися від вигляду диференційних сигналів для

операції логічного додавання за модулем 2. Якщо $k_i = 0$, то для моменту часу *j** виконання операції інверсії додавання за модулем 2 вирази для $A_0[j^*]$ і $A_1[j^*]$ можна визначити через очікувані значення Хемінгової ваги суми і P з (1). З урахуванням (48) і (49) отримаємо:

$$A_0[j^*] \approx E\left[P \middle| k_i = 0, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n \middle| k_i = 0, p_i = 0\right] = \frac{N+1}{2} \cdot \varepsilon + L, \quad (50)$$

$$A_1[j^*] \approx E\left[P \middle| k_i = 0, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n \middle| k_i = 0, p_i = 1\right] = \frac{N-1}{2} \cdot \varepsilon + L. \quad (51)$$

Тоді, підставивши значення (51) і (52) у (6), отримаємо значення диференційного сигналу

$$T_0[j^*] = A_0[j^*] - A_1[j^*] \approx \varepsilon \text{, за умови, що } k_i = 0. \quad (52)$$

Аналогічно можна побудувати вирази для $A_0[j^*]$ і $A_1[j^*]$ для випадку $k_i = 1$.

$$A_0[j^*] \approx E\left[P \middle| k_i = 1, p_i = 0\right] = E\left[d \cdot \varepsilon + L + n \middle| k_i = 1, p_i = 0\right] = \frac{N-1}{2} \cdot \varepsilon + L, \quad (53)$$

$$A_1[j^*] \approx E\left[P \middle| k_i = 1, p_i = 1\right] = E\left[d \cdot \varepsilon + L + n \middle| k_i = 1, p_i = 1\right] = \frac{N+1}{2} \cdot \varepsilon + L. \quad (54)$$

Шляхом підстановки значень (53) і (54) у (7) отримаємо диференційний сигнал

$$T_1[j^*] = A_0[j^*] - A_1[j^*] \approx -\varepsilon \text{, за умови, що } k_i = 1. \quad (55)$$

Отже, з виразів (52) і (55) випливає, що диференційний сигнал буде містити від'ємний пік за умови $k_i = 1$ і додатній пік за умови $k_i = 0$.

Висновки

Інженерно-криптографічні атаки на реалізацію криптографічних алгоритмів є потужними методами визначення невідомої інформації, яка використовується для перетворення даних. В роботі розглянуто проведення такої атаки на реалізації двомісних логічних операцій: логічного множення, логічного додавання, штриха Шеффера (інверсії логічного множення), стрілки Пірса (інверсії логічного додавання), імплікації та еквівалентності (інверсії додавання за модулем 2) з використанням одного з можливих каналів витоку інформації – споживаної потужності пристрою. При цьому знайдено очікувані значення Хемінгової ваги результату для визначення значення кожного біту невідомого аргументу.

Встановлено, що використання алгоритму атаки на реалізації перелічених операцій, аналогічного до алгоритму атакування реалізації операції додавання за модулем 2, призводить до успішного визначення індивідуальних бітів невідомого аргументу розглянутих двомісних логічних операцій.

З метою однозначної ідентифікації бітів невідомого аргументу двомісних логічних операцій проведено дослідження та отримано вирази для оцінки залежності очікуваних Хемінгових ваг результату виконання логічних операцій над двома N-розрядними аргументами. На основі отриманих вище оцінок розраховано очікувані форми усередненого значення сигналу споживаної потужності. Встановлено, що форма диференційного сигналу при проведенні інженерно-криптографічної атаки дозволяє однозначно ідентифікувати значення бітів невідомого аргументу двомісних логічних операцій (табл. 1).

Таблиця 1 – Залежність форми диференційного сигналу від типу двомісної логічної операції і значення невідомого аргументу при проведенні інженерно-криптографічних атак

Невідомий аргумент	Двомісна логічна операція						
	$k_i \oplus p_i$	$k_i \cdot p_i$	$k_i \vee p_i$	$k_i \mid p_i$	$k_i \downarrow p_i$	$k_i \rightarrow p_i$	$k_i \equiv p_i$
$k_i = 0$	$-\varepsilon$	0	$-\varepsilon$	0	ε	0	ε
$k_i = 1$	ε	$-\varepsilon$	0	ε	0	$-\varepsilon$	$-\varepsilon$

Отримані результати підкреслюють необхідність, важливість та своєчасність створення, розвитку і використання спеціальних методів, засобів проектування і продукування комп'ютерних реалізацій криптографічних алгоритмів, стійких до проведення інженерно-криптографічних атак з використанням даних, отриманих з побічних каналів витоку інформації.

Запропоновані статистичні моделі очікуваного значення Хемінгової ваги результату виконання

логічних операцій двох змінних можна використати для дослідження та сертифікації комп'ютерних засобів, які реалізують алгоритми криптографічних перетворень з використанням цих операцій [12].

Література: ***1.*** *Kelsey J., Schneier B., Wagner D., Hall C.,* Side Channel Cryptanalysis of Product Ciphers // In 5th European Symposium on Research in Computer Security – ESORICS '98, vol. 1485 of *Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 1998. – P. 97 – 110.* ***2.*** *Clavier C., Coron J.-S., Dabbous N., Differential power analysis in the presence of hardware countermeasures // C. K. Koc, C.Paar, Eds., Cryptographic Hardware and Embedded Systems – CHESS 2000, vol. 1956 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 2000. – P. 252 – 263.* ***3.*** *Kocher P., Jaffe J., Jun B., Differential Power Analysis // In proceedings of International conference CRYPTO'99, vol. 1666 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 1999. – P. 388 – 397.* ***4.*** *Messerges T., Dabbish E., Sloan R., Eximining smart-card security under the threat of power analysis attack // IEEE Transactions on computers, Vol. 51, No 5, 2002, – P. 541 – 552.* ***5.*** *Messerges T., Dabbish E., Sloan R., Power analysis attacks of modular exponentiation in smartcards // C. K. Koc, C. Paar, Eds., Cryptographic Hardware and Embedded Systems – CHESS 1999, vol. 1717 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 1999. – P. 144 – 157.* ***6.*** *Akkar, M., Giraud, C. An implementation of DES and AES, secure against some attacks // In Proc. Cryptographic Hardware and Embedded Systems – CHES 2001, volume 2162 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 2001. – P. 309 – 318.* ***7.*** *Akkar M.-L., Bevan R., Dischamp P., Moyart D., Power analysis, what is now possible // T. Okamoto, Eds., International conference ASIACRYPT 2000, vol. 1976 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 2000. – P. 489 – 502.* ***8.*** *Messerges T., Using second-order power analysis to attack DPA resistant software // C. K. Koc, C.Paar, Eds., Cryptographic Hardware and Embedded Systems – CHESS 2000, vol. 1956 of Lecture Notes in Computer Science, Springer-Verlag Berlin Heidelberg, 2000. – P. 238 – 251.* ***9.*** *ГОСТ 28147-89. Система обработки информации. Защита криптографическая. Алгоритм криптографического преобразования. М.: Госстандарт СССР.* ***10.*** *Коркішко Т. А., Мельник А. О., Мельник В. А. Захист інформації в комп'ютерних і телекомунікаційних мережах: Алгоритми та процесори симетричного блокового шифрування. Львів: БАК, 2003. – 168 с.* ***11.*** *Коркішко Л. М., Васильцов І. В., Статистична модель операції додавання за модулем 2^N для проведення інженерно-криптографічних атак за побічними каналами витоку інформації // Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні. – 2004. – №8. – С. 115 – 121.* ***12.*** *Federal information processing standards publication. Security requirements for cryptographic modules. FIPS 140 – 2. National Institute of Standards and Technology (NIST), Gaithersburg, MD, 2001. – 68 p.*