Використання генератора випадкових чисел з апаратним джерелом ентропії для задач симетричної криптографії

Маргарита Геннадіївна Долотій,
Павло Володимирович Мерзликін[0000-0002-0752-411X]

Криворізький державний педагогічний університет,
пр. Гагаріна, 54, м. Кривий Ріг, 50086, Україна
mdolotiy@gmail.com, linuxoid@i.ua

Анотація. Метою дослідження є перевірка можливості використання розробленого генератора випадкових чисел [1], який використовує як джерело ентропії шуми звукової карти, в алгоритмах симетричної криптографії.

Ключові слова: генератор випадкових чисел, джерело ентропії, статистичний критерій, закон розподілу, симетрична криптографія.

Using the random number generator with a hardware entropy source for symmetric cryptography problems

Marharyta H. Dolotii, Pavlo V. Merzlykin[0000-0002-0752-411X]

Kryvyi Rih State Pedagogical University, 54, Gagarin Ave., Kryvyi Rih, 50086, Ukraine
mdolotiy@gmail.com, linuxoid@i.ua

Abstract. The aim of the research is to test the possibility of using the developed random number generator [1], which utilizes the sound card noise as an entropy source, in the symmetric cryptography algorithms.

Keywords: random number generator, source of entropy, test statistic, probability distribution, symmetric cryptography.

1 Вступ

Генератори випадкових чисел найчастіше застосовують в криптографії, адже випадковість і криптографія дуже сильно взаємопов’язані. Складно знайти коректно розроблене криптографічне прикладне забезпечення, яке не

використовує випадкові числа. Вектори ініціалізації, модифікатори геш-функцій, унікальні параметри при роботі з цифровими підписами повинні прийматися випадковими [2]. При використанні генераторів випадкових чисел в криптографічних системах генератори випадкових чисел повинні відповідати наступним вимогам [3]:

– послідовність, що генерується повинна мати максимально великий період;
– послідовність, що генерується, не повинна мати прихованих періодичностей;
– послідовність, що генерується, повинна мати рівномірний спектр.

Для демонстрації статистичних властивостей генераторів випадкових чисел використовуються різні підходи до статистичного тестування. Найчастіше набір і методику тестування пропонував сам розробник генератора. Таким чином, склалася ситуація, яка характеризується тим, що неможливо було об'єктивно порівняти різні генератори з єдиних позицій. Щоб подолати дану ситуацію, необхідно використовувати деякий стандартний набір статистичних тестів, об'єднаних єдиною методикою розрахунку необхідних показників ефективності генератора й прийняття рішення про випадковість генерованих послідовностей.

2 Методологія

У 1999 році фахівцями NIST в рамках проекту AES (Advanced Encryption Standard) був розроблений набір статистичних тестів NIST Statistical Test Suite), а також запропонована методика проведення статистичного тестування генераторів випадкових чисел [4], які на даний момент найкраще відповідають потребам всіх зацікавлених сторін.

У даній роботі розглядаються критерії прийняття рішення про проходження послідовністю статистичного тесту, набір статистичних тестів NIST і наводяться результати експериментальних досліджень властивостей ГВЧ, описаного в роботі [1].

Порядок тестування окремої двійкової послідовності S виглядає наступним чином.

1. Висувається нульова гіпотезу H_0 – припускаємо, що дана двійкова послідовність S випадкова.
2. За послідовністю S розраховується статистику тесту $c(S)$.
3. З використанням спеціальної функції та статистики тесту розраховується значення ймовірності $P=f(c(S))$, $P\in[0; 1]$.
4. Значення ймовірності P порівнюється з рівнем значущості α, $\alpha\in[0{,}001; 0{,}01]$. Якщо $P\geq\alpha$, то гіпотеза H_0 приймається. В іншому випадку приймається альтернативна гіпотеза.

Обраний пакет статистичних тестів може використовуватися для вирішення наступних завдань:

– виявлення ГВЧ (ГПВЧ), які формують «погані» двійкові послідовності;

– розробка нових ГВЧ (ГПВЧ);
– перевірка коректності реалізації ГВЧ (ГПВЧ);
– вивчення генераторів, описаних в стандартах;
– дослідження ступеня випадковості реально використовуваних ГВЧ (ГПВЧ).

3 Обговорення результатів

Для перевірки було генеровано три випадкові послідовності в діапазоні [0; 255] розміром 10 000 елементів.

Пакет NIST STS було завантажено з офіційного сайту Національного інституту стандартів і технологій (National Institute of Standards and Technology) [5], також для інтерпретації отриманих результатів використовувалась офіційна інструкція із сайту [6].

Перед тим, як було протестовано кожну з трьох послідовностей, був побудований загальний графік розподілу для послідовностей, показаний на рис. 1. По осі x відкладено значення байту, а по осі y кількість байтів з таким значенням у послідовності.

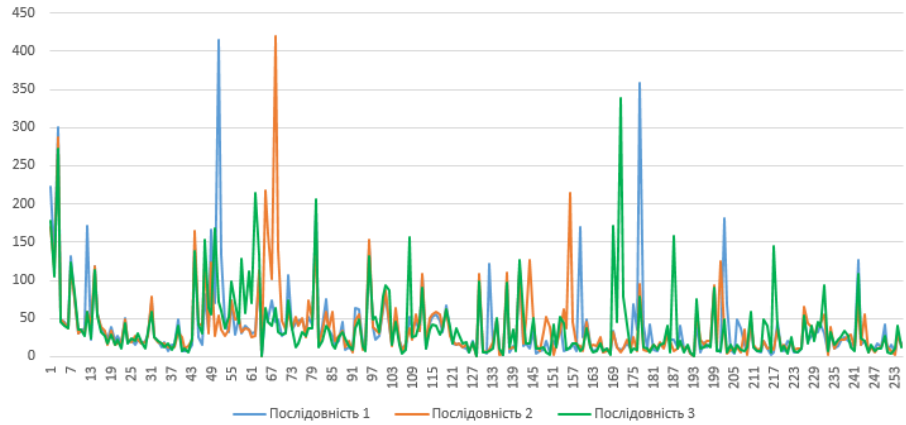

Рис. 1. Графік розподілів трьох генерованих послідовностей

З рис. 1 видно, що кожна із послідовностей має свої піки та особливості, тому вони підходять для перевірки пакетом NIST STS. Першим тестом, проведеним над генерованими послідовностями, був Binary Matrix Rank Test. Основну увагу в даному тесті приділяють рангу непересічної підматриці всієї послідовності. Метою цього тесту є перевірити лінійну залежність між підрядками фіксованої довжини оригінальної послідовності. Варто зауважити, що цей тест також використовуються в серії тестів DIEHARD.

Інтерпретація цього тесту: великі значення χ^2 (obs) вказують на те, що відхилення рангового розподілу від того, що відповідає випадковій послідовності, є значним. В результаті проходження даного тесту результатом, за

яким можна зазначати, що послідовність випадкова, є значення p. Якщо обчислене p<0,01, то вважається, що послідовність не випадкова. Інакше послідовність є випадковою.

Результати проходження тесту Binary Matrix Rank Test для першої послідовності показані на рис. 2.

Для даної послідовності значення p=0,203766, що свідчить про випадковість генерованої послідовності. Також значення χ^2=3,181562 є малим, тобто відхилення рангового розподілу від того, що відповідає випадковій послідовності, є незначним.

```
                                RANK TEST

---------------------------------------------
               COMPUTATIONAL INFORMATION:

---------------------------------------------
               (a) Probability P_32 = 0.288788
               (b)             P_31 = 0.577576
               (c)             P_30 = 0.133636
               (d) Frequency   F_32 = 5
               (e)             F_31 = 3
               (f)             F_30 = 1
               (g) # of matrices    = 9
               (h) Chi^2            = 3.181562
               (i) NOTE: 784 BITS WERE DISCARDED.

---------------------------------------------
SUCCESS        p_value = 0.203766
```

Рис. 2. Binary Matrix Rank Test для першої послідовності

Для послідовностей 2 і 3 значення p=0,374306 та p=0,648387, що також вказує на їх випадковість. Відповідні значення χ^2: 1,965364 та 0,866535 є досить малими, що знову свідчить про те, що відхилення рангового розподілу від того, що відповідає випадковій послідовності, є незначним.

При цьому варто відмітити, що третя генерована послідовність з отриманих результатів є більш «випадковою» через більше значення p та менше значення χ^2 у порівнянні із послідовностями 1 та 2.

Наступний тест, що використовувався для перевірки випадковості генерованих послідовностей, це Non-overlapping Template Matching Test. Основною для цього тесту є кількість випадків попередньо заданих цільових рядків. Мета цього тесту – виявлення генераторів, що виробляють забагато випадків даної аперіодичної моделі. Для цього тесту використовується m-бітне вікно, що шукає конкретний m-бітний шаблон. Якщо шаблон не знайдено, вікно пропускає одну бітну позицію. Якщо шаблон знайдено, вікно пропускає біт після знайденого шаблону, і пошук відновлюється.

Цей тест відкидає послідовності, що демонструють занадто багато або занадто мало випадків даної аперіодичної картини.

Тест можна інтерпретувати як відхилення послідовностей, що виявляють нерегулярні входження даної неперіодичної картини.

Перша генерована послідовність за тестом Non-overlapping Template Matching Test дала результуючий файл, показаний на рис. 3.

NONPERIODIC TEMPLATES TEST

COMPUTATIONAL INFORMATION

LAMBDA = 2.425781 M = 1250 N = 8 m = 9 n = 10000

	F R E Q U E N C Y										
Template	W_1	W_2	W_3	W_4	W_5	W_6	W_7	W_8	Chi^2	P_value	Assignment Index
000000001	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 0
000000011	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 1
000000101	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 2
000000111	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 3
000001001	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 4
000001011	0	0	0	0	0	0	0	0	19.944262	0.010549	SUCCESS 5

Рис. 3. Non-overlapping Template Matching Test для першої послідовності

Для кожного із 148-ми індексів значення p=0,010549, що дає підставу програмі видавати результат Assignment = SUCCESS (успіх), що означає успішне проходження тесту та підтверджує «випадковість» генерованої послідовності.

Для послідовності 2 і 3 значення p є аналогічним, як і помітки про успішне проходження даними послідовностями тесту Binary Matrix Rank.

Наступний тест, що використовувався для перевірки випадковості генерованих послідовностей, це Overlapping Template Matching Test. Основна увага тесту приділяється кількості випадків попередньо заданих цільових рядків. Даний тест використовує вікно m-біт для пошуку конкретного m-бітового шаблону. Якщо шаблон не знайдено, вікно пропускає бітну позицію. Коли шаблон знайдено, вікно пропускає лише один біт, перш ніж відновити пошук.

Цей тест відкидає послідовності, які показують занадто багато або дуже мало випадків m-пробілів, але може бути легко модифікований для виявлення нерегулярних випадків будь-якого періодичного малюнка.

Для першої генерованої послідовності відповідний результат продемонстровано на рис. 4:

OVERLAPPING TEMPLATE OF ALL ONES TEST

COMPUTATIONAL INFORMATION:

(a) n (sequence_length) = 10000
(b) m (block length of 1s) = 9
(c) M (length of substring) = 1032
(d) N (number of substrings) = 9
(e) lambda [(M-m+1)/2^m] = 2.000000
(f) eta = 1.000000

F R E Q U E N C Y								
0	1	2	3	4	>=5	Chi^2	P-value	Assignment
8	0	0	0	0	1	11.119950	0.049053	SUCCESS

Рис. 4. Overlapping Template Matching Test першої послідовності

З отриманих результатів ми бачимо значення p=0,049053 та значення χ^2=11,119950, що відповідають за успішне проходження генерованої послідовності даного тесту. Для другої та третьої послідовності значення p та χ^2 є аналогічними, що підтверджує їх випадковість.

Наступний тест, яким було перевірено генеровані послідовності, був Serial Test. Основна увага цього тесту сконцентрована на частотах всіх можливих перекриваючих m-бітних шаблонів по всій послідовності. Мета цього тесту полягає в тому, щоб визначити, чи є число входжень 2^m m-біт приблизно таким же, як було б очікувано для випадкової послідовності. Випадкові послідовності однорідні, тобто кожен m-бітний шаблон має такий же шанс з'являтись, як і будь-який інший m-розрядний шаблон. Варто звернути увагу, що для m=1 послідовний тест еквівалентний частотному тесту.

Серійний тест (генералізований) – це перелік процедур, що базується на тестуванні однорідності розподілу моделей заданих довжин.

У результаті проходження тесту першою послідовністю було отримано два значення p, та відповідні значення ψ (рис. 5).

```
                        SERIAL TEST
                -----------------------------------------
                 COMPUTATIONAL INFORMATION:
                -----------------------------------------
                (a) Block length    (m) = 16
                (b) Sequence length (n) = 10000
                (c) Psi_m               = -10000.000000
                (d) Psi_m-1             = -10000.000000
                (e) Psi_m-2             = -10000.000000
                (f) Del_1               = 0.000000
                (g) Del_2               = 0.000000
                -----------------------------------------
SUCCESS         p_value1 = 1.000000
SUCCESS         p_value2 = 1.000000
```

Рис. 5. Serial Test першої послідовності

З отриманих результатів можна зробити висновок, що послідовність є випадковою за Serial Test. Щодо послідовностей 2 і 3, то вони мають аналогічні показники значень, що також підтверджує випадковість розробленого генератора випадкових чисел.

Останнім тестом, що проходять генеровані послідовності, є Linear Complexity Test. Основна увага цього тесту приділена довжині регістру зсуву з лінійним зворотним зв'язком (LFSR). Метою цього тесту є визначення, чи є послідовність досить складною, щоб вважатись випадковою. Випадкові послідовності характеризуються більш тривалими LFSR. Якщо LFSR занадто короткий, то послідовність вважається невипадковою.

Для першої генерованої послідовності в результаті виконання даного тесту було отримано результати, показані на рис. 6.

```
-----------------------------------------------------
       L I N E A R  C O M P L E X I T Y
-----------------------------------------------------
       M (substring length)     = 500
       N (number of substrings) = 20
-----------------------------------------------------
       F R E Q U E N C Y
-----------------------------------------------------
  C0   C1   C2   C3   C4   C5   C6    CHI2     P-value
-----------------------------------------------------
       Note: 0 bits were discarded!
   0    0    4    7    5    4    0  9.101060 0.167974
```

Рис. 6. Linear Complexity Test першої послідовності

За отриманими результатами p=0,167974 та χ^2=9,101060 можна зробити висновок, що генерована послідовність є випадковою. Щодо інших послідовностей: для другої послідовності p=0,783217 та χ^2=3,201098; для третьої p=0,167974 та χ^2=9,101060. Тобто, значення першого і третього розподілу збігається, в той же час, значення другого розподілу відрізняється, проте, всі три послідовності є випадковими.

4 Висновки

1. Найкращі результати було продемонстровано в тестах Binary Matrix Rank та Linear Complexity, де кожна із послідовностей мала власні значення p та χ^2, що давало можливість порівняти роботу генератора випадкових чисел в однакових умовах, проте, при різних генерованих послідовностях. Не зважаючи на відмінні результати, всі три послідовності успішно пройшли тест. Загалом створений апаратний генератор випадкових чисел, згідно результатів тестування пакетом NIST STS, можна вважити таким, що задовольняє вимоги пройдених тестів.
2. Створена бібліотека генерації випадкових чисел може бути використана в проектах, які мають потребу в високоякісних послідовностях випадкових чисел. В майбутньому генератор випадкових чисел може бути вдосконалений за рахунок використання більш чистого звуку та реалізації можливості вибору користувачем розподілу, за яким генеруватимуться випадкові числа.

Список використаних джерел

1. Долотій М. Г. Генератор випадкових чисел з апаратним джерелом ентропії / Маргарита Геннадіївна Долотій, Павло Володимирович Мерзликін // Новітні комп'ютерні технології. – Кривий Ріг : Видавничий центр ДВНЗ «Криворізький національний університет», 2017. – Том XV. – С. 85-88.
2. Kelsey J. Cryptanalytic Attacks on Pseudorandom Number Generators / John Kelsey, Bruce Schneier, David Wagner, Chris Hall // Fast Software Encryption. 5th International Workshop, FSE'98. Paris, France, March 23-25, 1998 : Proceedings / Ed. : Serge Vaudenay. –Berlin, Heidelberg : Springer-Verlag, 1998. – P. 168-188. – (Lecture Notes in Computer Science, vol. 1372)
3. Долгих А. О. Генератор псевдо випадкових чисел [Електронний ресурс] / А. О. Долгих // Тези Всеукраїнської науково-практичної on-line конференції аспірантів, молодих учених та студентів, присвяченої Дню науки. 10-12 травня 2017 року // Конференції Житомирського державного технологічного університету : матеріали конференцій, проведених в ЖДТУ. – 2017. – Режим доступу : https://conf.ztu.edu.ua/wp-content/uploads/2017/06/123-1.pdf.
4. Soto J. Randomness Testing of the Advanced Encryption Standard Candidate Algorithms [Electronic resource] / Juan Soto, Jr. – Gaithersburg : National Institute of Standards and Technology, September 1999. – 10 p. – Access mode : https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir6390.pdf. – (NISTIR 6390)

5. NIST SP 800-22: Documentation and Software - Random Bit Generation | CSRC [Electronic resource]. – 2018. – Access mode : https://csrc.nist.gov/Projects/Random-Bit-Generation/Documentation-and-Software.
6. Bassham L. E. A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications [Electronic resource] / Lawrence E. Bassham III, Andrew L. Rukhin, Juan Soto, James R. Nechvatal, Miles E. Smid, Elaine Barker, Stefan D. Leigh, Mark Levenson, Mark Vangel, David L. Banks, Alan Heckert, James Dray, San Vo. – Gaithersburg : National Institute of Standards and Technology, September 16, 2010. – 131 p. – Access mode : https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=906762. – (Special Publication (NIST SP) - 800-22 Rev 1a)

References (translated and transliterated)

1. Dolotii, M.G., Merzlykin, P.V.: Henerator vypadkovykh chysel z aparatnym dzherelom entropii (The random number generator with hardware source of entropy). New computer technology. **15**, 85–88 (2017).
2. Kelsey, J., Schneier, B., Wagner, D., Hall, C.: Cryptanalytic Attacks on Pseudorandom Number Generators. In: Vaudenay, S. (ed.) Fast Software Encryption, 5th International Workshop, FSE'98. Paris, France, March 23-25, 1998. Lecture Notes in Computer Science, vol. 1372, pp. 168–188. Springer-Verlag, Berlin, Heidelberg (1998)
3. Dolhykh, A.O.: Henerator psevdo vypadkovykh chysel (Generator of pseudorandom numbers). In: The theses of the All-Ukrainian scientific and practical on-line conference of postgraduates, young scientists and students devoted to the Day of Science, ZhDTU, Zhytomyr, 10–12 May 2017. https://conf.ztu.edu.ua/wp-content/uploads/2017/06/123-1.pdf (2017). Accessed 12 Nov 2018
4. Soto, J.Jr.: Randomness Testing of the Advanced Encryption Standard Candidate Algorithms. National Institute of Standards and Technology, Gaithersburg, September 1999. (NISTIR 6390). https://nvlpubs.nist.gov/nistpubs/Legacy/IR/nistir6390.pdf (1999). Accessed 30 Nov 2018
5. NIST SP 800-22: Documentation and Software - Random Bit Generation | CSRC. https://csrc.nist.gov/Projects/Random-Bit-Generation/Documentation-and-Software (2018)
6. Bassham III, L.E., Rukhin, A.L., Soto, J., Nechvatal, J.R., Smid, M.E., Barker, E., Leigh, S.D., Levenson, M., Vangel, M., Banks, D.L., Heckert, A., Dray, J., Vo, S.: A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications. National Institute of Standards and Technology, Gaithersburg, September 16, 2010 (Special Publication (NIST SP) - 800-22 Rev 1a). https://ws680.nist.gov/publication/get_pdf.cfm?pub_id=906762 (2010). Accessed 30 Nov 2018