https://doi.org/10.31891/2307-5732-2025-359-117
УДК 004.421.5+004.056.55

СЕМЕНКОВ МІХАЇЛ
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікоського»
https://orcid.org/0009-0008-0355-7548
e-mail: mih234bk@gmail.com
МАРЧЕНКО ОЛЕКСАНДР
Національний технічний університет України
«Київський політехнічний інститут імені Ігоря Сікоського»
https://orcid.org/0000-0002-4537-3420
e-mail: oleksandr.i.marchenko@gmail.com

СПОСІБ ПІДВИЩЕННЯ НАДІЙНОСТІ КЛЮЧІВ ШИФРУВАННЯ НА ОСНОВІ РОЗПОДІЛУ ГАУСА

У статті запропоновано спосіб генерації ключів шифрування, спрямований на підвищення їх надійності та стійкості до криптоаналітичних атак. Особливістю запропонованого способу є використання властивостей розподілу Гауса при генерації початкових випадкових послідовностей, після чого ці послідовності піддаються спеціалізованій обробці для генерації ключів шифрування. Такий спосіб дає змогу підвищити ентропію отриманих даних і знизити передбачуваність порівняно з низкою традиційних способів, зберігаючи при цьому задовільний для практичних застосувань рівень швидкодії.

У статті наведено теоретичне обґрунтування використання розподілу Гауса як джерела випадковості та описано загальні принципи перетворень, що застосовуються до згенерованих послідовностей з метою підвищення їх надійності та непередбачуваності. Також було проведено порівняльний аналіз, включаючи різні оцінки, такі як значення ентропії, передбачуваність, продуктивність і складність практичної реалізації. Аналіз показав збільшення характеристик випадковості та зменшення передбачуваності ключів шифрування при збереженні прийнятного часу генерації при використанні цього способу.

Запропонований спосіб може бути використаний в криптографічних системах, де одночасно необхідні висока якість випадковості ключів, обмежені обчислювальні ресурси та стабільна продуктивність. Результати дослідження підтверджують перспективність застосування підходу як у програмних, так і в апаратних реалізаціях, а також визначають напрямки подальшої роботи — розширене тестування на реальних платформах, більш глибокий криптоаналіз та стандартизоване оцінювання відповідно до сучасних вимог безпеки.

Ключові слова: *розподіл Гауса, шифрування, ключі шифрування, безпека даних, генерація ключів шифрування, захист від криптоаналітичних атак.*

SEMENKOV MIKHAIL, MARCHENKO OLEKSANDR
National Technical University of Ukraine "Igor Sikorsky Kyiv Polytechnic Institute"

A TECHINQUE FOR IMPROVING THE RELIABILITY OF ENCRYPTION KEYS BASED ON GAUSSIAN DISTRIBUTION

This article proposes a technique for generating encryption keys aimed at increasing their reliability and resistance to cryptanalytic attacks. The specific feature of the proposed technique is the use of Gaussian distribution properties in the generation of initial random sequences, after which these sequences undergo specialized processing to generate encryption keys. This approach is designed to increase the entropy of the input data and reduce predictability compared to a number of traditional techniques, while maintaining a satisfactory level of performance for practical applications.

The article provides a theoretical justification for using the Gaussian distribution as a source of variability and describes the general principles of transformations applied to generated sequences in order to increase the reliability and unpredictability of them. A comparative analysis was also conducted, including various assessments such as entropy values, predictability, performance, and complexity of practical implementation. The analysis noted an increase in randomness characteristics and a decrease in the predictability of key material while maintaining acceptable generation time when using this method.

The proposed technique can be used in cryptographic systems where high-quality randomness of keys, limited computing resources, and stable performance are simultaneously required. The research results confirm the promising application of the approach in both software and hardware implementations, and also determine the directions for further work — extended testing on real platforms, deeper cryptanalysis, and standardized evaluation in accordance with modern security requirements.

Keywords: *Gaussian distribution, encryption, encryption keys, data security, encryption key generation, protection against cryptanalytic attacks.*

Стаття надійшла до редакції / Received 22.10.2025
Прийнята до друку / Accepted 15.11.2025

Постановка проблеми у загальному вигляді та її зв'язок із важливими науковими чи практичними завданнями

В сучасній криптографії рівень безпеки інформації визначається не лише математичною стійкістю алгоритмів, але й якістю генерації випадкових чисел, що використовуються в якості ключів, тобто послідовностей, які дозволяють зашифрувати корисну інформацію і потім відтворити її в початковому вигляді. Надійна генерація послідовностей, які є криптографічно безпечними, тобто статистично рівномірними та стійкими до передбачення, має вирішальне значення, оскільки якщо послідовність відтворювана або передбачувана, то навіть найсучасніші алгоритми шифрування стають вразливими. Про це свідчать задокументовані випадки, коли недоліки генераторів випадкових чисел RNG (Random Number Generator) призвели до компрометації ключів та протоколів шифрування [1, 2].

Для генерації криптографічно безпечних послідовностей дуже важливо приділяти значну увагу ентропії та статистичній стійкості джерел випадковості, особливо у контексті можливої деградації RNG з часом або можливих атак на нього, які полягають у спробах зловмисника передбачити внутрішній стан RNG або зменшити ентропію його вихідних послідовностей. Цей аспект вважається критичним. Генератори ключів шифрування не можуть гарантувати безпеку лише на підставі статистичної якості, адже потенційне витікання стану RNG здатне підірвати всю безпеку [3, 4].

Особливо важливою є роль випадковості у забезпеченні семантичної безпеки криптосистем. Це означає, що для формування ключів та рандомізації шифрування необхідні джерела, що не допускають передбачуваність, навіть якщо текст шифрованих даних повторюється.

Водночас, у сфері прикладних досліджень криптографії, зокрема в протоколах розподілу квантових ключів (QKD, Quantum Key Distribution), було продемонстровано перевагу генерації випадкових чисел на основі розподілу Гауса через високу якість (ентропію) та швидкість. Наприклад, розробка RNG Гауса, реалізованого на FPGA для використання в QKD показала ефективність у генерації ключових послідовностей із високою статистичною якістю та продуктивністю [5].

На підставі викладеного сформулюємо наукову проблему. В той час, як традиційні способи генерації криптографічних ключів зазвичай обмежуються рівномірним розподілом і часто використовують псевдовипадкові алгоритми без гарантії ентропії та стійкості, генератори ключів на основі розподілу Ґауса забезпечують підвищену ентропію, статистичну непередбачуваність та криптоаналітичну стійкість у порівнянні з усталеними алгоритмами. Хоча деякі способи реалізації генераторів ключів на основі розподілу Ґауса вже були запропоновані, але цей напрямок має ще широке поле недосліджених можливостей. Тому дослідження підвищення надійності ключів шифрування на основі розподілу Ґауса є актуальними.

Аналіз останніх досліджень та публікацій

На сьогодні існує значна кількість способів генерації криптографічних випадкових послідовностей, кожен з яких має власні переваги та обмеження. Одні способи орієнтовані на забезпечення високої швидкодії, інші — на досягнення максимальної ентропії та, відповідно, підвищення стійкості згенерованих послідовностей до криптоаналітичних атак. Під терміном криптоаналітичні атаки у цьому контексті розуміють сукупність методів, спрямованих на зниження невизначеності (ентропії) ключів шифрування або на відновлення/передбачення вихідних значень генератора.

У криптографічних застосуваннях рівномірний розподіл є фундаментальним та еталонним способом, оскільки він забезпечує максимальну ентропію, симетричність й непередбачуваність для кожного значення. Для випадкової величини, що може набувати M різних значень, дискретний рівномірний розподіл має наступний вигляд

$$P(X = x)=\frac{1}{M},\ x \in \{0,1,\dots,M-1\}, \quad (1)$$

де усі значення x мають однакову ймовірність появи. З цього математичного означення безпосередньо випливає властивість рівноймовірності вихідних значень, яка гарантує рівномірне охоплення всього простору можливих ключів.

У такому випадку, ентропія випадкової величини досягає максимального значення:

$$H(X)=-\sum_{x=0}^{M-1}\frac{1}{M}\log_2 M. \quad (2)$$

Властивість рівномірності вихідних значень вважається обов'язковою для генераторів випадкових послідовностей, адже рівновірогідні розподіли мінімізують ймовірність криптоаналітичного вгадування, що забезпечує надійність ключів шифрування. Класичним прикладом використання рівномірно згенерованих ключів є шифр Вернама, який за умови рівномірного розподілу ключів та їх одноразового використання забезпечує абсолютну криптографічну стійкість [6].

У межах даного дослідження для порівняльного аналізу обрано три основні види генераторів (відповідно й способів генерації) випадкових чисел, які репрезентують як класичні, так і новітні підходи у розвитку теорії цього напряму:

1) генератори істинних випадкових чисел (True Random Number Generator – TRNG);
2) генератори псевдовипадкових чисел (PseudoRandom Number Generator – PRNG);
3) квантові генератори випадкових чисел (Quantum Random Number Generator – QRNG).

У контексті генераторів істинних випадкових чисел (TRNG), які ґрунтуються на фізичних джерелах ентропії, досягнення рівномірності розподілу випадкових чисел є нетривіальним завданням. Це підтверджується дослідженнями у межах пакета статистичних тестів NIST SP 800-22, який включає низку процедур, таких як тест на частоту (Frequency Test), тест на серії (Runs Test), тест на довгі серії однакових бітів (Long Run Test), що дозволяють виявити відхилення від ідеальної рівновірогідності та незалежності значень [7]. Дослідження показали, що навіть генератори на основі фізичного шуму мають тенденцію до відхилень, коли ймовірність появи «0» і «1» відрізняється від 0.5. Для усунення таких відхилень у сучасних TRNG застосовуються методи «відбілювання» (whitening), зокрема хешування (SHA-2, SHA-3) чи використання лінійних зсувних регістрів. Деякі практичні проблеми TRNG із рівномірним розподілом описані також в роботі [8], де показано, що в роботу генераторів на кільцевих осциляторах можна втрутитися за допомогою введення зовнішнього сигналу, що змінює статистику виходу та порушує рівномірність. В іншому дослідженні [9], прямо вимагається перевірка та післяобробка виходу TRNG, підкреслюючи, що фізичні джерела ентропії без додаткових перетворень рідко забезпечують ідеально рівномірний розподіл ймовірностей для вихідних значень.

Також важливо зазначити, що генератори істинних випадкових чисел зазвичай мають низьку швидкодію, оскільки їх продуктивність обмежена фізичними процесами, що лежать в основі генерації ентропії (наприклад, шум електронних компонентів чи радіоактивний розпад). Вони забезпечують високу якість випадковості, але, в умовах великих обсягів криптографічних операцій, їх обмежена швидкодія може істотно знижувати загальну ефективність обчислень.

Таким чином, на практиці TRNG завжди потребують моніторингу та корекції вихідних даних, щоб відповідати строгим криптографічним вимогам. Без цього навіть фізично коректні джерела випадковості стають вразливими до статистичних атак, що знижує їхню стійкість у криптосистемах.

Генератори псевдовипадкових чисел (PRNG) генерують послідовності чисел, що виглядають випадковими, але насправді є детермінованими. Вони використовують початкове значення та функцію генерації для створення послідовності:

$$x_{n+1} = f(x_n), \tag{3}$$

де x_0 — початкове значення, а f — функція генерації.

У криптографії застосовуються криптографічно безпечні генератори PRNG – CSPRNG (Cryptographically Secure PseudoRandom Number Generator), які повинні задовольняти двом ключовим вимогам:

- непередбачуваності згенерованих чисел — навіть знаючи попередні значення, неможливо визначити наступне без знання внутрішнього стану генератора;
- стійкості до відновлення стану — часткове розкриття внутрішнього стану не повинно дозволяти відтворення послідовності цілком.

Рівномірність вихідних значень у CSPRNG досягається комбінацією наступних способів:

- післяобробкою виходу із застосуванням криптографічних хеш-функцій або блочних шифрів;
- агрегацією ентропії з декількох джерел.

Сучасні реалізації CSPRNG ґрунтуються на вимогах стандарту NIST SP 800-90A Rev.1 та часто реалізуються за принципами генератора Fortuna. Такий підхід широко застосовується при розробці систем з високим рівнем безпеки [10, 11].

Втім, як показано в аналізі роботи [12], багато атак на RNG ґрунтуються саме на слабких або передбачуваних початкових значеннях, тобто даних, з яких ініціюється робота генератора випадкових чисел. Якщо таке значення можна вгадати чи відтворити, зловмисник отримує змогу повністю відновити згенеровану послідовність випадкових чисел. Це підкреслює важливість використання надійного джерела ентропії для формування початкових значень і, відповідно, для гарантування безпеки зашифрованих даних.

Генератори виду PRNG характеризуються високою швидкодією, оскільки їх реалізація виконується виключно програмно, причому з використанням оптимізованих арифметичних або криптографічних операцій. Вони здатні генерувати великі обсяги випадкових чисел у реальному часі, проте це відбувається ціною детермінованості, тобто залежності всієї послідовності від початкового значення.

Таким чином, генератори виду PRNG не створюють «істинної випадковості» згенерованих чисел, а лише масштабують та перетворюють випадкове початкове значення у швидкий потік псевдовипадкових чисел. Але такий процес генерації здатен забезпечити вимоги рівномірності та непередбачуваності, які є необхідними для створення безпечних криптосистем.

Квантові генератори випадкових чисел (QRNG) для створення істинно випадкових чисел використовують фундаментальні принципи квантової механіки. Це робить їх особливо цінними для криптографії, оскільки їхні виходи неможливо передбачити або відтворити навіть за повної інформації про криптографічну систему. Реалізація сучасних QRNG часто базується на явищах випромінювання одиничних фотонів, квантових флуктуацій вакууму або квантовій заплутаності. У криптографічному контексті це означає, що результати вимірювань таких процесів не можуть бути передбачені наперед, що забезпечує максимально можливу ентропію та стійкість отриманих ключів [13].

Незважаючи на фундаментальну непередбачуваність квантових процесів, QRNG не позбавлені практичних вразливостей. Дослідження показують, що атаки на джерела фотонів, що модулюють інтенсивність джерел світла, можуть знизити ентропію вихідної послідовності та, як наслідок, криптографічну стійкість генератора [14]. Ще один недолік генераторів виду QRNG полягає в тому, що їх якість залежить від апаратного забезпечення, оскільки для досягнення високої швидкості генерації потрібні спеціалізовані детектори та електронні пристрої. Це значно ускладнює реалізацію таких рішень у звичайних умовах і підвищує їхню вартість. Попри ці недоліки, сучасні QRNG демонструють високу статистичну якість та відповідають критеріям криптографічної безпеки, а їх реалізації досягають високої швидкості генерації (до 100 Гбіт/с), що робить їх придатними для створення високошвидкісних криптографічних протоколів, включно з TLS та іншими сучасними протоколами шифрування [15].

Як підсумок зазначимо, що загалом усі існуючі способи генерації випадкових чисел, що використовуються для підвищення надійності ключів шифрування, при їх реалізації передбачають пошук компромісу між швидкістю, надійністю ентропії та масштабованістю. Генератори виду TRNG забезпечують високу непередбачуваність, але низьку швидкість. Генератори виду PRNG забезпечують швидкість, але мають алгоритмічну передбачуваність. Генератори виду QRNG поєднують непередбачуваність із високою ентропією, але мають апаратні обмеження та складну інтеграцію.

Формування цілей статті

Метою роботи є: Запропонувати спосіб підвищення надійності згенерованих ключів шифрування, який дозволить поєднати високу ентропію та непередбачуваність, як у TRNG та QRNG, із відносною простотою програмної реалізації, як у PRNG.

Запропонований спосіб генерації ключів шифрування

Запропонований спосіб ґрунтується на комбінації властивостей генераторів істинних випадкових чисел TRNG, яким притаманний високий ступінь випадковості, з розподілом Гауса (нормальним розподілом), що дозволить поєднати високу ентропію та непередбачуваність, які характерні для фізичних джерел шуму, з ефективністю програмної реалізації, властивою для псевдовипадкових генераторів.

Основою цього способу є отримання випадкових значень з фізичного джерела, наприклад теплового шуму електронних компонентів, шуму фотодіодів або флуктуацій генераторів хаотичних сигналів. Ці значення мають високу ентропію, але їх розподіл часто є нерівномірним і вимагає нормалізації. Для виконання нормалізації пропонується використовувати перетворення до розподілу Гауса, що дозволяє підвищити стійкість генерованих ключів до статистичних атак.

Розглянемо спосіб, що пропонується більш детально. Нехай вхідні дані мають рівномірний розподіл на інтервалі $[0, 1]$. Для перетворення використаємо метод Бокса–Мюллера, який дозволяє отримати нормально розподілені випадкові величини з математичним очікуванням $\mu = 0$ та дисперсією $\sigma^2 = 1$. Зокрема, формули перетворення двох незалежних рівномірно розподілених випадкових змінних U_1 та U_2 до двох незалежних нормально розподілених змінних Z_1 та Z_2 мають вигляд:

$$Z_1 = \sqrt{-2\ln U_1} \cdot cos(2\pi U_2), \quad (4)$$

$$Z_2 = \sqrt{-2\ln U_1} \cdot sin(2\pi U_2). \quad (5)$$

Використання цього перетворення дає змогу не лише підвищити рівень непередбачуваності, але й забезпечити математичну коректність статистичних властивостей [16]. Важливо, що програмна реалізація на основі трансформації Бокса–Мюллера або її оптимізованих варіантів дозволяє досягти високої швидкодії без необхідності складного апаратного забезпечення, оскільки потребує лише обчислення стандартних математичних функцій — логарифма, квадратного кореня та тригонометричних функцій. На практиці програми, що реалізують цей спосіб, здатні працювати на швидкостях, порівнянних із сучасними псевдовипадковими генераторами. Більш того, якщо при його реалізації використовувати також і апаратні прискорювачі (наприклад, графічні процесори або спеціалізовані інструкції процесорів), то продуктивність може сягати мільйонів незалежних випадкових значень на секунду. Таким чином, запропонований спосіб є придатним для використання навіть у задачах з високими вимогами до продуктивності.

Далі отримані значення масштабуються до необхідного діапазону для генерації криптографічних ключів. Головна особливість способу полягає в тому, що рівномірність базового фізичного джерела гарантує високу ентропію, а нормальний розподіл у фінальній стадії створює додатковий захист проти атак, що ґрунтуються на виявленні закономірностей. Таким чином, запропонований спосіб поєднує апаратну непередбачуваність генераторів виду TRNG, строгість математичної обробки та ефективність програмної реалізації. Цей спосіб здатний одночасно забезпечити високу ентропію, стійкість до криптоаналітичних атак і практичність реалізації, що відповідає вимогам розробки сучасних систем захисту інформації.

Порівняльний аналіз

Розглянемо порівняння запропонованого способу з іншими способами (Таблиця 1).

На відміну від способів, які використовуються у псевдовипадкових генераторах, запропонований спосіб не залежить виключно від початкових умов і тому є більш захищеним у криптографічному контексті. У порівнянні зі способами, на яких ґрунтуються генератори істинних випадкових чисел, він є менш чутливим до якості апаратної реалізації, зокрема стабільності сенсорів, рівня шумів та зовнішніх фізичних факторів. Це пояснюється тим, що додаткові математичні перетворення компенсують статистичні похибки первинних даних, зменшуючи вплив апаратних недосконалостей на кінцеву якість випадкової послідовності. Порівнюючи зі способами, які лежать в основі квантових генераторів, запропонований спосіб забезпечує простішу та значно дешевшу реалізацію, зберігаючи при цьому високий рівень стійкості та практичну надійність.

Особливо відзначимо також порівняння за швидкодією, яка є важливою характеристикою. Завдяки тому, що основне реалізаційне навантаження у запропонованому способі переноситься на програмні обчислення, швидкодія обробки за цим способом є співставною зі швидкодією обробки у псевдовипадкових генераторах, які традиційно є найшвидшими. Таким чином, цей спосіб дозволяє одночасно досягти і високої швидкості, достатньої для систем з високими вимогами до швидкості генерації випадкових чисел, і високої якості випадковості, необхідної для безпечного формування ключів.

Таблиця 1

Порівняльна характеристика генераторів випадкових чисел

Характеристика	**Генератори істинних випадкових чисел (TRNG)**	**Псевдовипадкові генератори (PRNG)**	**Квантові генератори випадкових чисел (QRNG)**	**Генератори на основі запропонованого способу**
Джерело випадковості	Фізичні шуми (електричні, радіаційні, хаотичні)	Алгоритм, математична модель	Квантові процеси	Фізичний шум + математична трансформація
Якість випадковості (ентропія, передбачуваність)	Висока, залежить від якості джерела	Обмежена, залежить від початкового значення та безпеки стану	Дуже висока	Висока, стабілізована перетворенням
Простота реалізації	Потрібні спеціальні датчики шуму	Дуже проста, повністю програмна	Потрібна складна квантова апаратура	Потрібні тільки базові датчики шуму, решта — програмна реалізація
Швидкодія	Залежить від апаратури, зазвичай нижча, ніж у інших	Дуже висока	Невисока, обмежена оптичними системами	Висока: програмне перетворення дає продуктивність, близьку до PRNG
Придатність для криптографії	Так, але з ризиком дефектів джерела	Так, при забезпечені безпеки початкового значення	Так, але технологія дорога та складна	Так, оптимізовано під криптографічні ключі

Висновки і перспективи подальших розвідок у даному напрямі

У статті запропонований спосіб генерації ключів шифрування, що ґрунтується на поєднанні фізичного джерела шуму та математичного перетворення на основі розподілу Гауса. Цей спосіб демонструє збалансоване поєднання властивостей, необхідних для створення сучасних криптографічних систем. Завдяки використанню фізичного джерела значень гарантується достатній рівень ентропії та непередбачуваності, що робить спосіб стійким до відновлення або передбачення згенерованої послідовності. Додаткова програмна обробка, що передбачена цим способом, дозволяє згладити потенційні недоліки апаратного фізичного джерела, підвищуючи якість випадковості та забезпечуючи рівномірний розподіл значень.

Отже, запропонований спосіб може розглядатися як оптимізований компроміс між швидкодією, надійністю та складністю практичної реалізації. Він поєднує у собі кращі властивості відомих підходів, уникаючи їхніх ключових недоліків, і здатний забезпечити надійну основу для побудови захищених криптографічних систем без потреби у складному або дорогому обладнанні.

Напрямки подальших досліджень запропонованого способу генерації ключів шифрування пов’язані насамперед із його перевіркою на практиці у різних прикладних сценаріях. Доцільним є також проведення експериментів зі статистичною оцінкою якості отриманих послідовностей за допомогою загальновизнаних тестових наборів. Це дозволить підтвердити рівень ентропії та ступінь наближення розподілу до теоретично очікуваного.

Окремим напрямом подальшого дослідження може стати оптимізація реалізації алгоритмів запропонованого способу з точки зору швидкодії обробки. Попри те, що використання нормального розподілу не накладає значних обмежень на продуктивність, залишається актуальним питання масштабування алгоритму для високонавантажених систем, наприклад, серверів, що обслуговують велику кількість криптографічних операцій.

Література

1. Halprin R. Games for Extracting Randomness / Ran Halprin, Moni Naor // XRDS Crossroads The ACM Magazine for Students. – 2010. – С. 44–48. – DOI: https://doi.org/10.1145/1869086.1869101.

2. Kelsey J. Cryptanalytic Attacks on Pseudorandom Number Generators / J. Kelsey, B. Schneier, D. Wagner, C. Hall // Lecture Notes in Computer Science. – 1998. – С. – 168–188. – DOI: https://doi.org/10.1007/3-540-69710-1_12.

3. Bagini V. A Design of Reliable True Random Number Generator for Cryptographic Applications / V. Bagini, M. Bucci // Lecture Notes in Computer Scienc. – 1999. – С. 204–218. – DOI: https://doi.org/10.1007/3-540-48059-5_18.

4. CVE-2008-0166 [Електронний ресурс] // https://nvd.nist.gov. – Режим доступу: https://nvd.nist.gov/vuln/detail/cve-2008-0166 (дата звернення: 24.09.2025). – Назва з екрана.

5. Hu Y. Gaussian Random Number Generator: Implemented in FPGA for Quantum Key Distribution / Y. Hu, Y. Wu, Y. Chen, G. C. Chen, S. T. Mei // International Journal of Numerical Modelling Electronic Networks Devices and Fields. – 2019. – С. 2554. – DOI: https://doi.org/10.48550/arXiv.1802.07368.

6. Mulder V. Trends in Data Protection and Encryption Technologies. / V. Mulder, A. Mermoud, V. Lenders, B. Tellenbach // Springer. – 2023. – С. 3–6. – DOI: https://doi.org/10.1007/978-3-031-33386-6.

7. Rukhin A. A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications / A. Rukhin, J. Soto, J. Nechvatal, M. Smid // NIST Special Publication 800-22. – 2010. – DOI: https://doi.org/10.6028/NIST.SP.800-22r1a.

8. Beuchat J. Cryptographic Hardware and Embedded Systems / J. Beuchat, J. Detrey, N. Estibals, E. Okamoto, F. Rodríguez-Henríquez // Lecture Notes in Computer Science. – 2009. – T. 5747. – C. 225–239. – DOI: https://doi.org/10.1007/978-3-642-04138-9_17.

9. Turan M. Recommendation for the Entropy Sources Used for Random Bit Generation / M. Turan, E. Barker, J. Kelsey, K. McKay, M. Baish, M. Boyle // NIST Special Publication 800-90B. – 2018. – C. 84. – DOI: https://doi.org/10.6028/NIST.SP.800-90B.

10. Barker E. Recommendation for Random Number Generation Using Deterministic Random Bit Generators / E. Barker, J. Kelsey // NIST Special Publication 800-90A Rev. 1. – 2015. – C. 110. – DOI: https://doi.org/10.6028/NIST.SP.800-90Ar1.

11. Lee S. Mastering Fortuna PRNG for Secure Applications. [Електронний ресурс] / Sarah Lee // https://www.numberanalytics.com. – Режим доступу: https://www.numberanalytics.com/blog/mastering-fortuna-prng-secure-applications (дата звернення: 24.09.2025). – Назва з екрана.

12. Dodis Y. Security analysis of pseudo-random number generators with input: /dev/random is not robust / Y. Dodis, D. Pointcheval, S. Ruhault, D. Vergniaud, D. Wichs // Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. – 2013. – C. 647–658. – DOI: https://doi.org/10.1145/2508859.251665.

13. Mannalatha V. A comprehensive review of quantum random number generators: concepts, classification and the origin of randomness. / V. Mannalatha, S. Mishra, A. Pathak // Quantum Information Processing. – 2023. – C. 439 – DOI: https://doi.org/10.1007/s11128-023-04175-y.

14. Zhang B. Practical attack on a quantum random-number generator via injection of source-signal fluctuations / B. Zhang, L. Li, Y. Xu, Z. Tan, J. Shi, P. Huang, T. Wang, G. Zeng // American Physical Society. – 2025. – C. 13. – DOI: https://doi.org/10.1103/k3c9-ngt1.

15. Blanco-Romero J. Evaluating integration methods of a quantum random number generator in OpenSSL for TLS / J. Blanco-Romero, V. Lorenzo, F. Almenares, D. Díaz-Sánchez, C. García Rubio, C. Campo, A. Marín // Computer Networks. – 2024. – DOI: https://doi.org/10.1016/j.comnet.2024.110877.

16. Box G. Note on the Generation of Random Normal Deviates / G. Box, M. E. Muller // Annals of Mathematical Statistics. – 1958. – T. 29, – вип. 1. – DOI: https://doi.org/10.1214/AOMS%2F1177706645.

References

1. Halprin R. Games for Extracting Randomness / Ran Halprin, Moni Naor // XRDS Crossroads The ACM Magazine for Students. – 2010. – S. 44–48. – DOI: https://doi.org/10.1145/1869086.1869101.

2. Kelsey J. Cryptanalytic Attacks on Pseudorandom Number Generators / J. Kelsey, B. Schneier, D. Wagner, C. Hall // Lecture Notes in Computer Science. – 1998. – S. – 168–188. – DOI: https://doi.org/10.1007/3-540-69710-1_12.

3. Bagini V. A Design of Reliable True Random Number Generator for Cryptographic Applications / V. Bagini, M. Bucci // Lecture Notes in Computer Scienc. – 1999. – S. 204–218. – DOI: https://doi.org/10.1007/3-540-48059-5_18.

4. CVE-2008-0166 [Elektronnyi resurs] // https://nvd.nist.gov. — Rezhym dostupu: https://nvd.nist.gov/vuln/detail/cve-2008-0166 (Data zvernennia: 24.09.2025). – Nazva z ekrana.

5. Hu Y. Gaussian Random Number Generator: Implemented in FPGA for Quantum Key Distribution / Y. Hu, Y. Wu, Y. Chen, G. C. Chen, S. T. Mei // International Journal of Numerical Modelling Electronic Networks Devices and Fields. – 2019. – S. 2554. – DOI: https://doi.org/10.48550/arXiv.1802.07368.

6. Mulder V. Trends in Data Protection and Encryption Technologies. / V. Mulder, A. Mermoud, V. Lenders, B. Tellenbach // Springer. – 2023. – S. 3–6. – DOI: https://doi.org/10.1007/978-3-031-33386-6.

7. Rukhin A. A Statistical Test Suite for the Validation of Random Number Generators and Pseudo Random Number Generators for Cryptographic Applications / A. Rukhin, J. Soto, J. Nechvatal, M. Smid // NIST Special Publication 800-22. – 2010. – DOI: https://doi.org/10.6028/NIST.SP.800-22r1a.

8. Beuchat J. Cryptographic Hardware and Embedded Systems / J. Beuchat, J. Detrey, N. Estibals, E. Okamoto, F. Rodríguez-Henríquez // Lecture Notes in Computer Science. – 2009. – T. 5747. – S. 225–239. – DOI: https://doi.org/10.1007/978-3-642-04138-9_17.

9. Turan M. Recommendation for the Entropy Sources Used for Random Bit Generation / M. Turan, E. Barker, J. Kelsey, K. McKay, M. Baish, M. Boyle // NIST Special Publication 800-90B. – 2018. – S. 84. – DOI: https://doi.org/10.6028/NIST.SP.800-90B.

10. Barker E. Recommendation for Random Number Generation Using Deterministic Random Bit Generators / E. Barker, J. Kelsey // NIST Special Publication 800-90A Rev. 1. – 2015. – S. 110. – DOI: https://doi.org/10.6028/NIST.SP.800-90Ar1.

11. Lee S. Mastering Fortuna PRNG for Secure Applications. [Elektronnyi resurs] / Sarah Lee // https://www.numberanalytics.com. – Rezhym dostupu: https://www.numberanalytics.com/blog/mastering-fortuna-prng-secure-applications (Data zvernennia: 24.09.2025). – Nazva z ekrana.

12. Dodis Y. Security analysis of pseudo-random number generators with input: /dev/random is not robust / Y. Dodis, D. Pointcheval, S. Ruhault, D. Vergniaud, D. Wichs // Proceedings of the 2013 ACM SIGSAC conference on Computer & communications security. – 2013. – S. 647–658. – DOI: https://doi.org/10.1145/2508859.251665.

13. Mannalatha V. A comprehensive review of quantum random number generators: concepts, classification and the origin of randomness. / V. Mannalatha, S. Mishra, A. Pathak // Quantum Information Processing. – 2023. – S. 439 – DOI: https://doi.org/10.1007/s11128-023-04175-y.

14. Zhang B. Practical attack on a quantum random-number generator via injection of source-signal fluctuations / B. Zhang, L. Li, Y. Xu, Z. Tan, J. Shi, P. Huang, T. Wang, G. Zeng // American Physical Society. – 2025. – S. 13. – DOI: https://doi.org/10.1103/k3c9-ngt1.

15. Blanco-Romero J. Evaluating integration methods of a quantum random number generator in OpenSSL for TLS / J. Blanco-Romero, V. Lorenzo, F. Almenares, D. Díaz-Sánchez, C. García Rubio, C. Campo, A. Marín // Computer Networks. – 2024. – DOI: https://doi.org/10.1016/j.comnet.2024.110877.

16. Box G. Note on the Generation of Random Normal Deviates / G. Box, M. E. Muller // Annals of Mathematical Statistics. – 1958. – T. 29, – vyp. 1. – DOI: https://doi.org/10.1214/AOMS%2F1177706645.