DOI: 10.58254/viti.7.2025.07.74

УДК 004.056.55

д-р техн. наук, професор Кузавков В. В. ORCID: 0000-0002-0655-9759 (ВІТІ ім. Героїв Крут)
Тлустий А. О. ORCID: 0000-0002-4777-9563 (ВІТІ ім. Героїв Крут)

АНАЛІЗ СТАТИСТИЧНИХ ТЕСТІВ ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ

У сфері інформаційної безпеки та криптографії оцінка характеристик (статистичної безпеки) згенерованих послідовностей є надзвичайно важливим завданням. Це питання відіграє ключову роль також у таких галузях, як комп'ютерна безпека, моделювання систем, мережева інформаційна безпека, статистика.

Інформаційна безпека в мережах забезпечується випадковістю створених ключів шифрування, токенів ідентифікації та цифрових підписів у мережевих протоколах. Ненадійні генератори можуть значно знижувати рівень захисту мережі.

У статистиці випадкових послідовностей і псевдовипадкових послідовностей для побудови тестових вибірок, перевірки гіпотез та проведення експериментів важливі генератори з низькою якістю можуть спотворити результати аналізу.

Комп'ютерна безпека забезпечується при застосуванні генераторів випадкових чисел і генераторів псевдовипадкових чисел шляхом створення паролів, токенів доступу, ідентифікаторів сесій тощо. Якщо генератор ненадійний, це може стати причиною вразливостей, що ставлять під загрозу безпеку інформації.

У моделюванні та симуляціях генератори використовуються для створення випадкових подій та наборів даних. Недоліки у їхній роботі можуть викликати похибки у моделюванні та призводити до неточних результатів.

У криптографічних системах випадкові послідовності і псевдовипадкові послідовності використовуються для генерації ключів, випадкових чисел та сольових значень. Низька якість генератора може створити вразливості, які дозволяють зловмисникам зламати криптографічні протоколи.

Таким чином, якість генераторів прямо впливає на ефективність і безпеку систем, що робить перевірку їхніх статистичних властивостей ключовим завданням.

Генератори псевдовипадкових чисел формують числові значення (послідовності), які здаються випадковими, хоча фактично мають певну структуру. Це робить їх затребуваними в різних галузях, зокрема для створення ключів, забезпечення конфіденційності, захисту цілісності даних і безпеки транзакцій. Тому надійність та захищеність таких послідовностей є основними характеристиками.

Робота присвячена аналізу інструментів для оцінки статистичних властивостей генераторів випадкових і псевдовипадкових послідовностей.

Ключові слова: *генератор псевдовипадкових послідовностей, криптозахист, статистичні тести, генерація ключів.*

V. Kuzavkov, A. Tlustyi. Analysis of statistical tests of pseudorando sequences

In the field of information security and cryptography, the assessment of the characteristics (statistical security) of generated sequences is an extremely important task. This issue also plays a key role in such areas as computer security, system modeling, network information security and statistics.

Information security in networks is ensured by the randomness of the generated encryption keys, identification tokens, and digital signatures in network protocols. Unreliable generators can significantly reduce the level of network security.

In statistics, random sequences and Pseudo-random sequences are important for constructing test samples, testing hypotheses, and conducting experiments. Low-quality generators can distort the results of the analysis.

Computer security is ensured when using Random Number Generators and Pseudo-Random Number Generators by generating passwords, access tokens, session identifiers, etc. If the generator is unreliable, this can cause vulnerabilities that threaten information security.

In modeling and simulations, generators are used to generate random events and data sets. Deficiencies in their operation can cause errors in the modeling and lead to inaccurate results.

In cryptographic systems, random sequences and Pseudo-random sequences are used to generate keys, random numbers, and salt values. Low quality of the generator can create vulnerabilities that allow attackers to break cryptographic protocols.

Thus, the quality of generators directly affects the efficiency and security of systems, which makes checking their statistical properties a key task.

Pseudo-random number generators generate numerical values (sequences) that appear random, although in fact they have a certain structure. This makes them in demand in various industries, in particular for key generation,

ensuring confidentiality, protecting data integrity, and transaction security. Therefore, the reliability and security of such sequences are the main characteristics.

The work is devoted to the analysis of tools for evaluating the statistical properties of random and pseudorandom sequence generators.

Keywords: *pseudorandom sequence generator, crypto protection, statistical tests, key generation.*

Постановка проблеми

Відомо, що генерація псевдовипадкових чисел (ПВЧ) значною мірою впливає на інформаційні технології у сферах як криптологія, аналітика, моделювання, медицина, симуляції та ін. Якість псевдовипадкових послідовностей (ПВП) безпосередньо впливає на стійкість шифрування, ефективність чисельних методів та надійність різних цифрових систем. Недостатня випадковість або наявність статистичних закономірностей у ПВП може призвести до їх передбачуваності та, як наслідок, до вразливостей у безпекових механізмах.

Тому необхідність дослідження тестових наборів обумовлена зростаючими вимогами до криптографічної стійкості та важливістю адаптації статистичних тестів до сучасних викликів. Основні проблеми, що потребують вирішення, включають: узагальнення критеріїв випадковості, оптимізацію існуючих тестових наборів, автоматизацію тестових процесів та як наслідок розробку нових тестових наборів. Значною мірою на якість ПВП впливають генератори псевдовипадкових чисел (ГПВЧ).

Аналіз публікацій за темою дослідження

Питанням надійності ГПВП займаються чимало вчених та дослідників. Так, у роботах [5; 6; 9] вчені Х. Махалінгам, В. Срірам, С. Поперешняк розглядають питання дослідження та оцінки тестових наборів і характеристик генераторів. Однак питанням порівняння широкої оцінки відомих тестів приділено недостатньо уваги, хоча це питання є достатньо необхідним для визначення надійності ПВП в умовах сучасних викликів.

У роботі [9] запропоновано методи оцінки бітових послідовностей із різноманітними шляхами використання тестових наборів, проте недостатньо формалізовано критерії вибору того чи іншого методу для конкретних випадків застосувань.

Мета статті. Провести порівняльний аналіз основних відомих тестових наборів псевдовипадкових та випадкових послідовностей: TestU01, NIST STS та DieHard як інструментів для оцінки статистичних властивостей генераторів, які формують псевдовипадкові та послідовності.

Виклад основного матеріалу

Тести NIST є стандартом у криптографії, пропонуючи широкий набір підходів для оцінки випадковості послідовностей. DieHard-тести спрямовані на глибокий аналіз складних статистичних характеристик і часто використовуються для виявлення детальних недоліків генераторів ПВП. TestU01 вирізняються більшою чутливістю, що дозволяє їм виявляти ширший спектр можливих недоліків ПВП.

Розглянемо особливості, недоліки і ключові області застосування згаданих механізмів перевірки.

Статистичні тести служать для перевірки того, чи забезпечують ГВП або ГПВП достатній рівень випадковості. Їхня мета полягає у зборі доказів, що генеровані числа мають властивості, притаманні істинно випадковим послідовностям [1].

Під час цих перевірок аналізуються згенеровані послідовності та порівнюються їхні статистичні характеристики з очікуваними властивостями ідеальних випадкових послідовностей. Це дозволяє оцінити якість генератора (наскільки його робота відповідає очікуванням) та виявити вразливості (знайти можливі закономірності чи систематичні відхилення), які можуть зробити генератор ненадійним для певних завдань.

Такі порівняння допомагають оцінити, чи є генератор придатним для криптографічних, аналітичних або застосування в інших галузях, де випадковість є ключовою вимогою.

Вирішення подібних завдань базується на перевірці гіпотез про властивості випадкових процесів (ВП) і випадкових вибірок (ВВ), які створюються генераторами. У ролі статистичної гіпотези може виступати будь-яке припущення щодо розподілу або характеристик випадкової величини. За допомогою методів математичної статистики це припущення перевіряється на істинність чи хибність: якщо результати підтверджують гіпотезу, вона приймається, якщо ні – відхиляється [2] .

Процес створення механізму перевірки статистичних гіпотез ґрунтується на створенні (висуванні) двох гіпотез: нульової (H0) та альтернативної (H1). Якщо нульова гіпотеза доводить, що послідовність дійсно випадкова, то альтернативна випадковість не є випадковою [3; 4; 12].

На сьогодні основна частина тестів сформована як програмне забезпечення. Це дає змогу ефективно та якісно виконувати тестування, заощаджує час і ресурси.

Тести необхідно провести для визначення безпечності генераторів та критеріїв їхньої випадковості. Важливо також звернути увагу на подальше застосування того чи іншого генератора при виборі набору тестів, оскільки це кардинально впливатиме на подальшу оцінку ГПВЧ.

Існує ряд тестів, які використовують різні інструменти та алгоритми. Найбільш відомим є набір тестів NIST Statistical Test Suite (NIST STS), до складу якого входить щонайменше 16 статистичних тестів, метою яких є визначення міри випадковості двійкової системи послідовностей, створених апаратними чи програмними генераторами [7].

Основний пакет тестів DieHard складається з 13 наборів окремих тестових перевірок [8; 9]. Тест DieHard був розроблений математиком Джорджем Магсаглією протягом шести років та вперше опублікований в 1995 році. В основі тесту лежить вимірювання якості множини випадкових чисел [8]. Він став одним із перших широко відомих інструментів для статистичного тестування генераторів випадкових чисел (ГВЧ) та криптографічних алгоритмів. Завдяки своїй новаторській природі та доступності він швидко здобув популярність у наукових та інженерних колах [10].

Всі тести спрямовані на виявлення можливих закономірностей, передбачуваності або структур, що можуть вказувати на слабкості ГВЧ. Результати тестів зазвичай використовуються для оцінки придатності генератора у криптографічних або інших чутливих до випадковості системах.

Однак, незважаючи на його історичну роль, набір тестів DieHard має свої недоліки:

суб'єктивність вибору тестів. Набір включає тести, які не завжди охоплюють усі аспекти випадковості, а їх добір вважається дещо неоднозначним;

складність інтерпретації. Результати деяких тестів можуть бути важкими для розуміння та аналізу, що ускладнює використання DieHard для неспеціалістів [11];

помилки в реалізації. Початкова програмна версія набору страждала від технічних недоліків, що могли впливати на точність тестів і знижувати їхню надійність.

Незважаючи на ці недоліки, DieHard послужив основою для створення більш сучасних і вдосконалених наборів тестів, таких як DieHarder і TestU01, які враховують недоліки оригінальної версії [13].

Test U01 – пакет статистичних тестів, реалізований на мові ZNSI C, який використовує набір утиліт для тестування ПВП. Остання версія тесту була представлена П'єром Л'Екуйе та Річардом Сімарде з Монреальского університету в 2007 році [2].

Перевагою TestU01 є:

− наявність трьох рівнів тестування: SmallCrush, Crush та BigCrush. Це дозволяє гнучко підходити до перевірки ГВЧ, починаючи з базових тестів і закінчуючи глибоким статистичним аналізом;

− висока чутливість до дефектів. Тести в TestU01 ґрунтуються на передових статистичних методах, що дозволяє виявити навіть незначні відхилення від ідеальної випадковості, які можуть бути критичними у криптографічних і симуляційних застосуваннях.

TestU01 дозволяє перевіряти як класичні ГВЧ (наприклад, лінійні конгруентні генератори), так і складніші криптографічні генератори. Він також підтримує різні параметри конфігурації для адаптації тестування до конкретних потреб [15];

− наявність вихідного коду дає змогу дослідникам аналізувати методи тестування, модифікувати алгоритми та інтегрувати TestU01 у власні проєкти.

Багато наукових публікацій і досліджень використовують TestU01 як стандартний інструмент для перевірки ГВЧ, що підтверджує його ефективність і надійність. При цьому відомі наступні недоліки TestU01:

− testU01 не є ідеальним для оцінки криптографічних ГВЧ, оскільки він не враховує деякі специфічні атаки та вимоги, що використовуються у криптографії;

− виконання повного набору потребує значного обсягу пам’яті та обчислювальної потужності, що може бути проблематичним для деяких застосувань [16];

− для ефективного застосування TestU01 необхідно мати глибокі знання в галузі статистики та теорії випадкових чисел, що може бути складним для дослідників, які не мають відповідної підготовки;

− обмежена підтримка багатьох мов програмування. TestU01 розроблений на мові C++ і може бути складним для інтеграції у середовища, що базуються на інших мовах програмування без додаткової адаптації.

Таким чином, TestU01 є потужним інструментом для оцінки псевдовипадкових генераторів, що широко використовується в наукових дослідженнях і практичних застосуваннях. Його головними перевагами є широка база тестів, відкритий код і висока чутливість до статистичних аномалій. Водночас його застосування вимагає значних обчислювальних ресурсів і спеціальних знань, що може бути бар’єром для нових користувачів.

Особливе місце серед інструментів оцінки ПВП займає пакет тестів NIST STS. У роботі проведемо аналіз наборів інструментів, які відсутні в DieHard та Test U01.

Тест перевірки накопичених сум призначений для виявлення довгострокових відхилень у випадковій послідовності. Він перевіряє, чи є загальна тенденція накопичених значень (сума бітів, інтерпретованих як +1 та –1) у послідовності узгодженою з випадковим розподілом.

Переваги тесту в тому що він виявляє довгострокові зміщення в послідовності та аналізує як локальні, так і глобальні аномалії у розподілі.

Обмеження зосереджені на чутливості до тривалих послідовностей, де навіть невеликі відхилення можуть накопичуватися, а також тест не враховує залежності між сусідніми бітами.

Тест перевірки накопичених сум використовується для аналізу якості генераторів випадкових чисел, криптографічних алгоритмів та інших систем, що вимагають високого рівня випадковості. Він є важливою складовою комплексного аналізу в пакеті тестів NIST STS, дозволяючи оцінювати як короткострокові, так і довгострокові характеристики випадковості.

Тест на перевірку серій є важливим етапом аналізу генераторів випадкових чисел (ГВЧ). Цей тест використовується для визначення того, чи кількість послідовних однакових бітів у виході ГВЧ відповідає статистичним очікуванням для випадкової послідовності.

Мета тесту полягає в перевірці частоти повторення однакових бітів (серій) заданої довжини та аналізу розподілу довжин серій 0 і 1, що відповідає теоретично очікуваному.

Тест працює наступним чином: спочатку обчислюється довжина серій (кількість послідовних 0 або 1) у бітах. Далі кількість серій порівнюється зі значеннями, які очікуються для випадкової послідовності. Заключним етапом є використання критерію хі-квадрат (x^2) для оцінки відповідності.

Теоретичні очікування:

− для випадкової послідовності частота серій довжиною *i* повинна зменшуватися експоненційно;

− формула теоретичної ймовірності для серії довжини і $P(i) = (1 – p)^2 \cdot p(i)$, де $p = 0{,}5$;

Результат роботи тесту:

− генерується *p*-значення (*p-value*), яке показує ймовірність того, що відхилення від очікуваного розподілу є випадковим;

− якщо $p \geq 0{,}01$, послідовність вважається випадковою для даного тесту.

Спектральний тест базується на дискретному перетворенні Фур'є (ДПФ). Він призначений для виявлення періодичних компонент у бітовій послідовності, які можуть свідчити про невипадковість.

Мета спектрального тесту полягає у виявленні синусоїдальних або періодичних компонент, що свідчать про наявність регулярності у послідовності.

Особливостями використання спектрального тесту є:

− чутливість до регулярностей та кореляцій у бітовій послідовності, особливо до періодичних компонент;

− необхідність використання достатньо довгих послідовностей, оскільки короткі послідовності можуть давати недостовірні результати.

Застосування універсального тесту Маурера обумовлено необхідністю визначитись, наскільки добре бітова послідовність стискається без втрати інформації. Якщо послідовність добре стискається, це може свідчити про наявність закономірностей, тобто невипадковість.

Тест базується на принципі інформаційної ентропії: чим більше випадковості в послідовності, тим складніше її стискати. Якщо послідовність має низький рівень ентропії, це свідчить про наявність повторюваних структур або передбачуваності [14].

До переваг даного тесту можна віднести: виявлення глобальних закономірностей у довгих послідовностях та якісну роботу для генераторів випадкових чисел, які мають тенденцію до повторюваності.

Тест перевірки шаблонів, які перекриваються, створений для оцінки якості ГВЧ. Він призначений для виявлення повторюваних шаблонів у послідовності бітів, які можуть свідчити про відхилення від випадковості.

Тест перевіряє, чи виявляються у бітовій послідовності повторювані шаблони заданої довжини *m*. Якщо таких повторень занадто багато або занадто мало порівняно з очікуваним рівнем випадковості, це може свідчити про невипадковість послідовності [14].

Основні етапи застосування тесту:

− вибір шаблону *m*: зазвичай вибирається короткий бітовий шаблон, наприклад, 111, 101 тощо;

− розбиття послідовності: бітова послідовність розбивається на блоки однакової довжини *N*, щоб аналізувати частоту появи шаблону в кожному блоці;

− обчислення частот: підраховується кількість повторень шаблону *m* в кожному блоці;

− порівняння з теоретичною моделлю: частоти повторення шаблону порівнюються з теоретичним розподілом Пуассона;

− обчислення *p*-значення: розраховується значення *p*, яке показує ймовірність того, що послідовність випадкова. Якщо *p*-значення менше порогового рівня (наприклад, 0,01 або 0,05), то послідовність вважається невипадковою.

Результат тесту наступний:

- якщо *p*-значення вище порогу, то послідовність вважається випадковою;
- якщо *p*-значення нижче порогу, то є ознаки невипадковості.

Тест на перевірку випадкових відхилень (Random Excursions Test) призначений для оцінювання, чи випадкова бітова послідовність поводиться як випадковий процес у термінах проходження через нуль (так звані «випадкові блукання»). Основна мета цього тесту – перевірити, наскільки часто певні стани (відхилення) виникають під час проходження послідовності.

Тест дозволяє проаналізувати, як часто послідовність повертається до стану 0, і перевіряє розподіл відхилень для певних значень. Якщо бітова послідовність є випадковою, частота появи станів повинна відповідати теоретично очікуваній для випадкового процесу.

Переваги тесту:

- наявність готової реалізації тесту, яка враховує його особливості;
- можлива реалізація на інших мовах (наприклад, MATLAB чи C++).

Перевірка випадкових відхилень (варіантний тест) є розширенням основного тесту випадкових відхилень. Цей тест аналізує кумулятивну суму випадкового блукання, але зосереджується на тому, як часто конкретні стани (від –9 до +9, крім 0) досягаються в послідовності. Метою є перевірка щодо відповідності частоти цих станів очікуваному розподілу для випадкового процесу.

Призначений для перевірки випадкових відхилень та визначає, чи є частоти появи певних станів (наприклад, −9, −8, …, 8, 9) у послідовності такими, що відповідають моделі випадкового блукання.

Перевірка шаблонів, які не перекриваються, використовується для виявлення конкретних бітових підпослідовностей (шаблонів) у випадковій послідовності. На відміну від тесту Overlapping Template Matching Test, тут шаблони не можуть перекриватися.

Тест перевіряє, чи з'являється заданий бітовий шаблон із частотою, яка відповідає очікуваній для випадкової послідовності, а також якщо шаблон з'являється занадто часто або занадто рідко, це може вказувати на наявність закономірностей у даних.

Загальні висновки

Набір тестів NIST STS (стандарт NIST SP 800-22) має низку переваг порівняно з TestU01 та DieHard. Цей набір використовується як загальноприйнятий стандарт для оцінки криптографічних генераторів випадкових чисел. Випробування спеціально підібрані для слабких місць, які можуть бути вставлені на безпеку системи. Для криптографічних продуктів NIST STS є оптимальним вибором завдяки спеціалізації на криптографічну стійкість та стандартизацію.

Слід зауважити, що комплексне використання згаданих тестів дає змогу досягти більш точної оцінки якості генераторів ПВП. Це є корисним для розробників, дослідників та інших спеціалістів, які працюють у цій галузі, а також сприятиме створенню більш надійних генераторів.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Almaraz Luengo E., Román Villaizán J. Cryptographically Secured Pseudo-Random Number Generators: Analysis and Testing with NIST Statistical Test Suite // Mathematics. 2023. № 11 (23). P. 4812. URL: https://doi.org/10.3390/math11234812.

2. Araki S., Wu J.-H., Yan J.-J. A Novel Design of Random Number Generators Using ChaosBased Extremum Coding // IEEE Access. 2024. № 12. P. 24039–24047. URL: https://doi.org/10.1109/ ACCESS.2024.3365638.

3. Нємкова О., Кіх М. Порівняльне дослідження тестів для оцінки статистичних характеристик генераторів випадкових та псевдовипадкових послідовностей // Електронне фахове наукове видання

«Кібербезпека: освіта, наука, техніка». 2024. № 4 (24). P. 115–132. URL: https://doi.org/10.28925/2663-4023.2024.24.115132.

4. Левин Б. Р. Теоретичні основи статистичної радіотехніки. Книга друга. Видавництво «Радянське радіо», 1968. С. 25–30.

5. Mahalingam H., Rethinam S., Janakiraman S., Rengarajan A. Non-Identical Inverter Rings as an Entropy Source: NIST-90B-Verified TRNG Architecture on FPGAs for IoT Device Integrity // Mathematics. 2023. № 11 (4). P. 1049. URL: https://doi.org/10.3390/math11041049.

6. Sriram V., Srikamakshi M., Jegadish Kumar K. J., Nagarajan K. K. Randomness Analysis of YUGAM-128 Using Diehard Test Suite // Innovative Data Communication Technologies and Application. 2020. № 46. P. 600–607. URL: https://doi.org/10.1007/978-3-030-38040-3_68.

7. Kim Y., Yeom Y. Accelerated implementation for testing IID assumption of NIST SP 800-90B using GPU // PeerJ Computer Science. 2021. № 7, e404. https://doi.org/10.7717/peerj-cs.404

8. Al-Daraiseh A., Sanjalawe Y., Al-E’mari S., Fraihat S., Bany Taha M., Al-Muhammed M. Cryptographic Grade Chaotic Random Number Generator Based on Tent-Map // Journal of Sensor and Actuator Networks. 2023. № 12 (5). P. 73. URL: https://doi.org/10.3390/jsan12050073.

9. Поперешняк С. В. Засіб для тестування бітової послідовності на випадковість // Вчені записки ТНУ ім. В. І. Вернадського. Серія: технічні науки. 2020. Том 31 (70). № 4. DOI: https://doi.org/10.32838/2663-5941/2020.4/16.

10. Abdulhameed H. A., Abdalmaaen H. F., Mohammed A. T., Mosleh M. F., Abdulhameed A. A. A Lightweight Hybrid Cryptographic Algorithm for WSNs Tested by the Diehard Tests and the Raspberry Pi. 2022 International Conference on Computer Science and Software Engineering (CSASE), P. 271–276. URL: https://doi.org/10.1109/CSASE51777.2022.9759589.

11. Ryan C., Kshirsagar M., Vaidya G., Cunningham A., Sivaraman R. Design of a cryptographically secure pseudo random number generator with grammatical evolution // Scientific Reports. 2022. № 12 (1). P. 8602. URL: https://doi.org/10.1038/s41598-022-11613-x.

12. Hussein S. N., Al-Alak S. M. Secret Keys Extraction Using Light Weight Schemes for Data Ciphering // Journal of Physics: Conference Series. 2021. № 1999 (1). 012114. URL: https://doi.org/ 10.1088/1742- 6596/1999/1/012114.

13. Semankiv M. Assessment of statistical properties random numbers // Bulletin of the National Technical University ‘KhPI’. A Series of «Information and Modeling». 2016. № 21. URL: https://doi.org/ 10.20998/2411-0558.2016.21.12.

14. NIST SP 800-22. A Statistic Test Suite for Random and Pseudorandom Number Generators for Cryptographic Application. April 2000. URL: http://csrc.nist.gov./publications/niatpubs//SP800-22rev1a.pdf.

15. L’Ecuyer P., Simard R. TestU01: A C Library for Empirical Testing of Random Number Generators // ACM Transactions on Mathematical Software. 2007. № 33 (4). P. 1–40.

16. Knuth D. E. The Art of Computer Programming, Volume 2: Seminumerical Algorithms. Addison-Wesley, 1997.