https://doi.org/10.31891/2307-5732-2025-359-21
УДК 004.005:004.9

ДОБРОВОЛЬСЬКИЙ ЮРІЙ
Чернівецький національний університет ім. Ю. Федьковича
https://orcid.org/0000-0002-1248-3615
e-mail: y.dobrovolsky@chnu.edu.ua
ПРОХОРОВ ПАВЛО
Чернівецький національний університет ім. Ю. Федьковича
https://orcid.org/0009-0008-0965-5771
e-mail: prokhorov.pavlo@chnu.edu.ua
ПАВЛЮЧЕНКО ОЛЕГ
Чернівецький національний університет ім. Ю. Федьковича
e-mail: oleh.pavl@gmail.com
МАКАРУК ОЛЕГ
Чернівецький національний університет ім. Ю. Федьковича
e-mail: makaruk.oleh@chnu.edu.ua

ВІДМОВОСТІЙКА РОЗПОДІЛЕНА СИСТЕМА ДЛЯ ІНІЦІАЛІЗАЦІЇ ГЕНЕРАТОРІВ ПСЕВДОВИПАДКОВИХ ЧИСЕЛ ЗА ДОПОМОГОЮ ХЕШУВАННЯ ВІДЕОДАНИХ

У цій статті розглядаються переваги та методи розробки розподілених систем для генерації псевдовипадкових величин сідів (seeds) за допомогою зовнішніх пристроїв відеозапису та/або відеозйомки. Дослідження проводиться на існуючих прототипах, зокрема на прототипі LavaRand, який розглядається як нерозподілена версія системи, що фіксує різні стани хаотичної системи з групи стаціонарних лавових ламп. Описано як оригінальну реалізацію, згадану в патенті 1998 року, так і більш досконалу. Також розглянуто особливості реалізації з точки зору теорії розподілених систем, зокрема, поряд з відмовами вузлів з технічної точки зору, вводиться поняття логічної відмови, при якій результати, отримані з вузла, можуть бути незадовільними, а також критерії такої відмови та механізми заміни такого вузла на інший. Крім того, надається інформація про методи перетворення кадрів у псевдовипадкові числа, описуються проблеми існуючих рішень та наводяться додаткові аргументи на користь побудови розподіленої системи з урахуванням специфіки методів генерації з зображень. Наведено обґрунтування, переваги та недоліки використання розподіленої системи порівняно зі стаціонарним підходом. Окреслено підходи до реалізації в контексті використання технологій контейнеризації та оркестрації контейнерів. Описано питання, пов'язані з побудовою мережі вузлів та їх загальною взаємодією, включаючи випадок, коли вузли можна розділити на групи за здатністю до фіксації, здатністю до генерації сідів та здатністю до виконання обох операцій на одному обчислювальному пристрої. Таким чином, ця стаття може бути корисною для реалізації додаткового та/або резервного джерела ентропії за допомогою розподіленої системи відео-сенсорів.

Ключові слова: *розподілені системи; генерація ентропії; програмна інженерія; контейнеризація; надійність програмного забезпечення*

DOBROVOLSKY YURIY
PROKHOROV PAVLO
PAVLIUCHENKO OLEH
MAKARUK OLEG
Chernivtsi National University named after Yu. Fedkovicha

RELIABLE DISTRIBUTED SYSTEM FOR INITIALIZATION OF PSEUDORANDO NUMBER GENERATORS USING VIDEO DATA HASHING

Current article examines the advantages and methods of distributed systems development for pseudorandom variables (seeds) generation through external video recording and/or video shooting devices. An investigation is carried out on existing prototypes, specifically the LavaRand prototype, which is considered as a non-distributed version of the system that captures different states of a chaotic system from a group of stationary lava lamps, both the original implementation mentioned in the 1998 patent and a more advanced one are described. Also, implementation features are examined from the distributed systems theory point of view, in particular, along with node failures from a technical point of view, the concept of a logical failure is introduced, in which the results obtained from a node could be unsatisfactory, as well as the criteria for such a failure and the mechanisms for replacing such a node with another one. In addition, information is provided on the methods for converting frames into pseudorandom numbers, the problems of existing solutions are described, and additional arguments are given in favor of building a distributed system taking into account the specifics of the methods of generation from images. The rationale, advantages, and disadvantages of using a distributed system compared to a stationary approach are given. The implementation approaches in the context of the utilization of containerization and container orchestration technologies, are outlined. Issues related to the construction of a network of nodes and their general interaction are described, including the case when nodes could be split into groups of ability to capture, ability of seed ganaration and ability to perform both of operations on the same node. Thus, this article can be useful in implementing an additional and/or backup source of entropy using a distributed system of video sensors.

Keywords: *distributed system; entropy generation; software engineering; containerization (computing); software reliability.*

Стаття надійшла до редакції / Received 03.10.2025
Прийнята до друку / Accepted 15.11.2025

Постановка проблеми

Багато сучасних задач криптографії та моделювання вимагають використання випадкових чисел. Водночас, в умовах забезпечення безпеки великих обсягів даних та/або великої кількості обчислювальних пристроїв може виникнути потреба у великих обсягах таких випадкових чисел, що генеруються з певною швидкістю. Незважаючи на те, що існує величезна кількість апаратних рішень [1], рішень на рівні операційних

систем [2] та поєднуючих два попередні підходи, реалізація додаткового/резервного джерела ентропії на основі запису та обробки деяких природних явищ залишається актуальною. Актуальність виникає через те, що вищезазначені рішення не завжди прозорі та у разі їх компрометації може просто не бути доступних альтернативних джерел ентропії. Крім того, бувають випадки, коли функціональність, що дозволяє генерувати такі значення, ще не реалізована на апаратному та/або системному рівні, наприклад, при розробці будь-яких нових архітектур процесорів, що також призводить до необхідності зовнішніх джерел ентропії. Таким чином, розробка систем, що дозволяють генерувати ентропію із зовнішніх явищ, є актуальною темою.

Існує досить велика кількість різних реалізацій джерел ентропії заснованих на природних явищах, таких як коливання в електричних колах, радіоактивний розпад ядра, атмосферні явища тощо. Одним з найпростіших у використанні є підхід відомий як LavaRand. Система, описана в патенті 1998 року [3], базується на відеозаписі хаотичної системи у вигляді стаціонарної стійки з лавовими лампами. Хаотичний характер записуваного середовища досягається завдяки фізичним властивостям самих лавових ламп, в результаті чого, використовуючи пристрої відеозапису, спрямовані на цю стійку, можна отримати послідовність різних між собою кадрів.

Незважаючи на відносну простоту розгортання та прозорість роботи, ця система має й свої недоліки, такі як залежність від спеціальних пристроїв у вигляді лавових ламп та необхідність розміщення всієї системи в одному стаціонарному просторі, а також складність масштабування такої системи. У якості рішення зазначених вище недоліків пропонується система, яка також використовує послідовність відеокадрів, але архітектурно є розподіленою.

Аналіз останніх джерел

Система LavaRand описується у оригінальному патенті [3]. Вичерпний огляд усіх відомих способів генерувати псевдовипадкові числа апаратним чином на момент написання статті наведено у [1], більш стабільні та швидкодіючі рішення, засновані на коливаннях струму у електронних ланцюгах наведено у [4]. Приклад підходу до генерування псевдовипадкових чисел на рівні операційної системи розглянуто у [2].

Основний теоретичний матеріал по теорії випадкових та псевдовипадкових генераторів випадкових чисел, зокрема їх побудові та підходів то оцінок їх статистичних характеристик викладено у [5, 6].

Теорія розробки та проектування розподілених систем достатньо широко розглядається у [7, 8], основні практичні підходи при реалізації подібних систем наведено у [9, 10].

Сучасний розвиток технологій контейнеризації та оркестрації контейнерів дозволяє реалізовувати відмовостійкі та легко масштабовані розподілені системи, які можуть ефективно працювати з великою кількістю сенсорів. Крім того, використання цих технологій також дозволяє реалізовувати підходи, що використовують різні типи сенсорів, а також їх додавання або поєднання з відкритими даними, такими як відкриті відеотрансляції, трансляції з вуличних веб-камер та інші.

Водночас, згадані вище технології дозволяють нівелювати деякі недоліки розподіленої версії такої системи. Наприклад, очевидно, що, на відміну від стаціонарної версії, при використанні зовнішніх веб-камер або відеосенсорів контроль над середовищем що фіксується є набагато меншим. На практиці це призводить до того, що, залежно від умов освітлення, погоди та інших факторів, різниця в оброблюваних кадрах може бути недостатньою для вимог випадковості при такій генерації. Через ці обставини виникає необхідність у конфігурації певних елементів розподіленої системи, а саме налаштування розподіленої системи таким чином, щоб керуючий вузол відключав вузол у разі його логічного збою та динамічно замінював його іншим, або реалізації логіки розподілення вузлів на групи за здатністю генерування та обробки відеоданих. Таким чином, використовуючи алгоритми розподілених систем, можна оптимально використовувати доступні обчислювальні ресурси різних типів.

Метою роботи є: дослідження архітектурних підходів та алгоритмів програмної інженерії до вирішення задач побудови розподілених систем генерації ентропії.

Виклад основного матеріалу

У рамках аналізу оригінального патенту LavaRand [3], що датовано 1998 роком, детально розглянемо алгоритм його роботи (рис. 1):

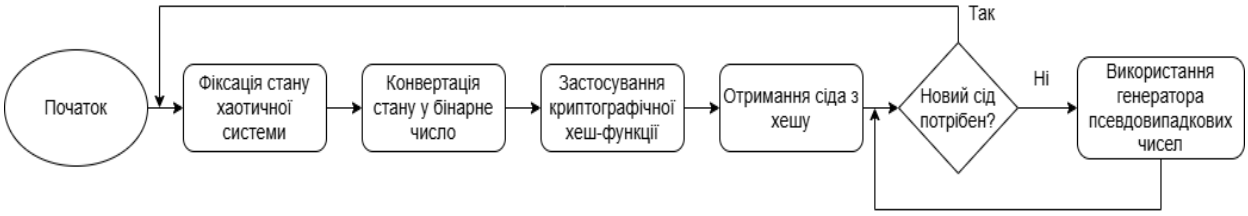

Рис. 1. Алгоритм роботи LavaRand

1. Фіксація стану хаотичної системи.

За допомогою веб-камери чи будь-якого іншого засобу відеофіксації робляться знімки стійки з лавовими лампами. При цьому в оригінальному документі не вказуються ані формат зберігання зображення кадру, ані частота кадрів зйомки, тому будемо вважати що LavaRand фіксує 1 кадр за 1 секунду.

2. Конвертація стану у бінарне число

Цей крок полягає у таких діях (рис. 2):

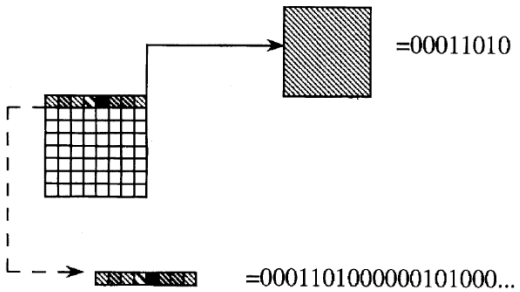

Рис 2. Метод конвертації фіксованого зображення у бінарний рядок (зображення з оригінального патенту [3])

• Кожен піксель представлений у 24-х бітному форматі (формат RGB, по 8 біт на кожен з 3-х кольорів) розкладається у один рядок з бінарних значень методом конкатенації.

• Попередню дію зробити з кожним пікселем зображення починаючі з верхнього лівого кута та рухаючись зліва направо, згори вниз.

У результаті отримується вихідний бінарний рядок. Оскільки у оригінальному документі не дається ніяких даних про параметри зображення та його параметри розподільної здатності, то кінцева довжина бінарного рядку буде дорівнювати 24 х висота зображення х ширина зображення у пікселях.

3. Застосування криптографічної хеш-функції.

До згенерованого бінарного рядку застосовується хеш-функція (у оригінальному документі пропонується використовувати NIST SHS-1, що скоріш за все є драфтом SHA-0, або використати вже розроблений на той час стандарт MD5). Також у оригінальному документі опосередковано описується лавинний ефект, ніяких даних про статистичні характеристики генеруємих сідів не надається.

1. Отримання сіда (seed) хешу

З отриманого у результаті хешування бінарного рядка обираємо якусь частину (або, якщо потрібно, зберігаємо цей результат для подальшої конкатенації з новим згенерованим сідом), за допомогою буде відбуватись ініціалізація використовуваного генератора псевдовипадкових чисел.

Очевидно що з моменту створення оригінального документа у 1998 році відбулися технологічні зміни, що суттєво впливають на результати застосування такої системи, наведемо основні з них:

1. Обчислювальна здатність середнього пристрою або комп'ютера суттєво зросла, а розмір обчислювальних пристроїв суттєво зменшився, завдяки чому виконувати функції відеофіксації та розрахунку сідів можна навіть за допомогою відносно дешевих одноплатних комп'ютерів з веб-камерою.

2. Було розроблено достатньо велику кількість нових стандартів хеш-функцій, хоча розроблені на момент написання патенту функції MD5 та SHA-1 мають найкращі показники з загального лавинного ефекту навіть на момент написання статті [11].

3. Розвиток технологій передачі інформації взагалі та розвиток технологій передачі даних мережею Internet, а також розвиток технологій бездротової передачі даних зробили можливими швидку передачу відносно об'ємних файлів, що дає змогу налаштовувати мережу з достатньо великої кількості пристроїв що можуть швидко взаємодіяти між собою обмінюючись даними.

Вказані вище фактори відкривають нам можливість розробки розподільної системи, аналогічної до оригінальної LavaRand. Назвемо розподілену систему що пропонується як Distributed LavaRand (далі DLR).

Одною з головних особливостей DLR є те що за її використання не дається великого контролю над середовищем що фіксується, особливо якщо використовуються зовнішні веб-камери. Тому, у порівнянні з LavaRand, пропонується ввести додатковий крок, що буде відповідати за вибір зображень, що підлягають хешуванню. Реалізувати це можна за допомогою великої кількості різноманітних алгоритмів, у свою чергу пропонується підхід заснований на використанні «середнього зображення» та попереднього калібрування. Полягає він у наступному:

1. Взяти початкову множину кадрів (наприклад усіх що було зафіксовано за перші 15 секунд зйомки) і розрахувати з неї «середнє зображення», шляхом розрахування середнього значення для кожного із відповідних пікселів серії зафіксованих кадрів. Цей крок і буде попереднім калібруванням.

2. Кожен наступний зафіксований кадр має бути порівняно з попереднім, і у випадку якщо різниця між ними більше або дорівнює різниці по певній метриці (наприклад кількість пікселів значення яких відрізняються у відповідних позиціях більше або дорівнює 10), то цей кадр можна використовувати для подальшого хешування.

3. Якщо певний кадр виявився не якісним то він пропускається і відбувається перехід до наступного. Якщо наприклад генерується 5 неякісних кадрів підряд, то відбувається випадок «логічної відмови» пристрою що фіксує кадри, за якої цей пристрій має бути виключено з генерації та замінено на новий або переорієнтовано на нове середовище з більшою варітивністю кадрів.

4. Через певний проміжок часу (наприклад 15 хвилин), процес калібрування повинен буде зроблений наново, у результаті чого отримується нове референсне «середнє зображення», після чого кроки 1-3 повторюються до моменту «логічної відмови» або відключення пристрою.

Рис 3. Алгоритм роботи системи генерації сідів та псевдовипадкових чисел у розподіленій системі

За допомогою наведеного алгоритму (рис. 3) можна одночасно використовувати велику кількість публічних трансляцій та веб-камер, зберігаючи варіативність сідів що генеруються такою системою загалом.

З точки зору хешування процес залишається ідентичним оригінальному, при цьому можуть бути використані SHA-1 або MD5 як такі що мають достатньо гарних лавинний ефект як вже було попередньо вказано.

З точки зору загальної архітектури DLR наведемо найпростіший варіант, що засновано на технологіях контейнеризації. Принцип роботи такої системи полягає у наступному:

Розгорнуто доступний для користувача (не обов'язково зовнішнього) HTTP API інтерфейс, за допомогою якого користувач може здійснити запит на сід або набір випадкових чисел у вигляді бінарного рядка.

У залежності від запиту користувача застосунок відправляє сід з бази даних напряму, або ініціалізує сідами з бази генератор псевдовипадкових чисел та відправляє згенеровані ним значення.

База даних сідів наповнюється за допомогою контейнерів-генераторів сідів (на рис. 4 такий контейнер-генератор один, хоча їх може бути будь-яка кількість) за допомогою доступу до наявних онлайн-трансляцій через мережу Internet.

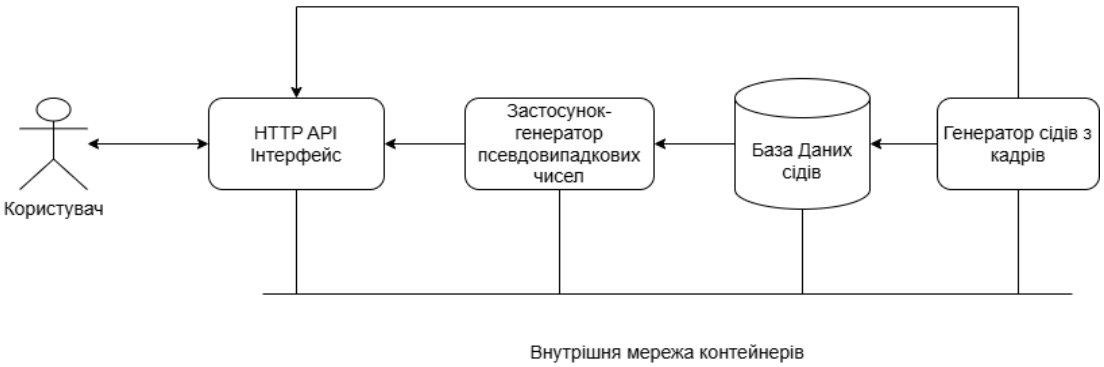

Рис. 4. Архітектура системи DLR

Таким чином показана реалізована система дозволяє користувачу отримувати по запиту ентропію або набір випадкових чисел.

Висновки

Продемонстрована система Distributed LavaRand дозволяє відносно швидко розгорнути джерело ентропії із зовнішніх публічних відеоданих, що може бути корисним у задачах криптографії та моделювання, особливо у випадках коли більш стабільні та швидкісні варіанти не є доступними для використання.

У подальших дослідженнях планується покращити архітектуру системи за допомогою інструментів оркестрації контейнерів, що дозволить наприклад збільшити загальну відмовостійкість системи через контроль над генеруючими контейнерами, їх динамічну заміну та інше.

Література

1. Classification of random number generator applications in IoT: A comprehensive taxonomy / Kübra Seyhan, Sedat Akleylek / Journal of Information Security and Applications – 2022 – Volume 71, 103365. DOI: https://doi.org/10.1016/j.jisa.2022.103365.

2. Documentation and Analysis of the Linux Random Number Generator / Federal Office for Information Security of Germany / 2024 – ver 5.8. Internet resource: https://www.bsi.bund.de.

3. US Patent # US5732138A: Method for seeding a pseudo-random number generator with a cryptographic hash of a digitization of a chaotic system / Landon Curt Noll, Robert G. Mende, Sanjeev Sisodiya ; 1998-03-24 -

Publication of US5732138A; 2016-01-29 - Anticipated expiration – 12 pages. Google Patents: https://patents.google.com/patent/US5732138A/en.

4. Very low cost chaos-based entropy source for the retrofit or design augmentation of networked devices / Sergio Callegari, Mattia Fabbri, Ahmad Beirami. // Analog Integrated Circuits and Signal Processing – 2016 – 87(2). DOI: https://doi.org/10.1007/s10470-015-0631-y.

5. Random Numbers Generators - Principles and Practices. / David Johnston. // De|G Press – 2018. DOI: https://doi.org/10.1515/9781501506062.

6. Random Numbers and Computers. / Ronald T. Kneusel. // Springer Cham – 2018. DOI: https://doi.org/10.1007/978-3-319-77697-2.

7. Distributed systems. / Maarten van Steen, Andrew S. Tanenbaum // Independent online publication – 2025 – ver. 4.03.

8. Fault-Tolerant Message-Passing Distributed Systems: An Algorithmic Approach. / Michel Raynal. // Springer Cham – 2018. DOI: https://doi.org/10.1007/978-3-319-94141-7.

9. Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies. / Ali Sunyaev. // Springer Cham – 2024 – 2nd edition. DOI: https://doi.org/10.1007/978-3-031-61014-1.

10. Patterns of Distributed Systems / Unmesh Joshi // Pearson Education – 2024. ISBN-13: 978-0-13-822198-0

11. Investigating the Avalanche Effect of Various Cryptographically Secure Hash Functions and Hash-Based Applications. / Darshana Upadhyay, Nupur Gaikwad, Marzia Zaman, Srinivas Sampalli. // IEEE Access – 2022 – Volume 10, pp. 112472 – 112486. DOI: https://doi.org/10.1109/ACCESS.2022.3215778.

References

1. Classification of random number generator applications in IoT: A comprehensive taxonomy / Kübra Seyhan, Sedat Akleylek / Journal of Information Security and Applications – 2022 – Volume 71, 103365. DOI: https://doi.org/10.1016/j.jisa.2022.103365.

2. Documentation and Analysis of the Linux Random Number Generator / Federal Office for Information Security of Germany / 2024 – ver 5.8. Internet resource: https://www.bsi.bund.de.

3. US Patent # US5732138A: Method for seeding a pseudo-random number generator with a cryptographic hash of a digitization of a chaotic system / Landon Curt Noll, Robert G. Mende, Sanjeev Sisodiya ; 1998-03-24 - Publication of US5732138A; 2016-01-29 - Anticipated expiration – 12 pages. Google Patents: https://patents.google.com/patent/US5732138A/en.

4. Very low cost chaos-based entropy source for the retrofit or design augmentation of networked devices / Sergio Callegari, Mattia Fabbri, Ahmad Beirami. // Analog Integrated Circuits and Signal Processing – 2016 – 87(2). DOI: https://doi.org/10.1007/s10470-015-0631-y.

5. Random Numbers Generators - Principles and Practices. / David Johnston. // De|G Press – 2018. DOI: https://doi.org/10.1515/9781501506062.

6. Random Numbers and Computers. / Ronald T. Kneusel. // Springer Cham – 2018. DOI: https://doi.org/10.1007/978-3-319-77697-2.

7. Distributed systems. / Maarten van Steen, Andrew S. Tanenbaum // Independent online publication – 2025 – ver. 4.03.

8. Fault-Tolerant Message-Passing Distributed Systems: An Algorithmic Approach. / Michel Raynal. // Springer Cham – 2018. DOI: https://doi.org/10.1007/978-3-319-94141-7.

9. Internet Computing: Principles of Distributed Systems and Emerging Internet-Based Technologies. / Ali Sunyaev. // Springer Cham – 2024 – 2nd edition. DOI: https://doi.org/10.1007/978-3-031-61014-1.

10. Patterns of Distributed Systems / Unmesh Joshi // Pearson Education – 2024. ISBN-13: 978-0-13-822198-0

11. Investigating the Avalanche Effect of Various Cryptographically Secure Hash Functions and Hash-Based Applications. / Darshana Upadhyay, Nupur Gaikwad, Marzia Zaman, Srinivas Sampalli. // IEEE Access – 2022 – Volume 10, pp. 112472 – 112486. DOI: https://doi.org/10.1109/ACCESS.2022.3215778.