Іванущак Наталія Михайлівна, кандидат технічних наук, асистент,

Чернівецький національний університет імені Юрія Федьковича, м. Чернівці

ORCID: 0009-0001-3747-9903

Комаришин Тарас Ігорович, студент

Чернівецький національний університет імені Юрія Федьковича, м. Чернівці

ORCID: 0009-0007-1178-8146

ДОСЛІДЖЕННЯ РІЗНОВИДІВ ГЕНЕРАТОРІВ ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ: ПРИНЦИПИ РОБОТИ, ЗАСТОСУВАННЯ ТА ОБМЕЖЕННЯ

Сьогодні створення псевдовипадкових послідовностей [1] набуває особливої ваги. Ці послідовності знаходять широке застосування у багатьох сферах: від криптографії та захисту інформації до комп'ютерного моделювання та ігрової індустрії. Їх генерація стає ключовим завданням для фахівців різних галузей. **Генератор псевдовипадкових послідовностей** (ГПВП) - це механізм, який використовує джерело ентропії, що базується на фізично випадкових явищах, і генерує випадкові послідовності, які є незалежними одна від одної. Основними вимогами для генерації псевдовипадкових послідовностей є непередбачуваність, статична безпека та незалежність, відсутність слабких особистих ключів, складність генерації, захищеність від відомих силових та потенційно можливих криптоатак.

Ентропія є ключовим фактором для забезпечення випадковості в ГПВП. Це міра невизначеності або випадковості в системі, яка визначає наскільки непередбачуваними є генеровані дані. Висока ентропія означає високий рівень випадковості, що робить результати менш передбачуваними та більш захищеними від потенційних атак. Для забезпечення високої ентропії використовують різні джерела, такі як фізичні процеси, системні показники, користувацьку активність тощо.



Рис. 1. Cхема алгоритму генератора псевдовипадкових послідовностей

До переваг генераторів псевдовипадкових послідовностей можна віднести швидкість та ефективність. Вони зазвичай працюють дуже швидко і можуть генерувати великі об'єми даних за короткий час, що є важливим для застосувань у реальному часі. Також вони мають простоту реалізації, оскільки багато алгоритмів ГПВП є простими для впровадження та інтеграції в різні системи. У деяких випадках відтворюваність також можна віднести до переваг. Завдяки детермінованому характеру, результати можуть бути відтворені за умови знання початкового стану. Це особливо корисно в тестуванні та налагодженні різноманітних систем.

Генератори псевдовипадкових послідовностей також мають ряд недоліків. До них належать циклічність послідовності, через невелику довжину періоду послідовності випадкових чисел генератор зациклюється і послідовність починає повторюватися. Також існує залежність між числами, які не є по-справжньому випадковими. Якщо алгоритм або його внутрішній стан буде відомий, послідовність може бути передбачена. Крім того, не варто забувати про безпеку, оскільки у криптографічних додатках, де висока ентропія є критичною, використання ГПВП може бути недостатнім, оскільки вони не можуть забезпечити справжню випадковість.

Досить часто відбуваються атаки [2] на генератори псевдовипадкових послідовностей, які можуть бути спрямовані на розкриття їх параметрів для подальшого передбачення вихідних даних. Для захисту від таких атак використовуються хеш-функції [3] та блокові шифри, щоб приховати або перетворити вихідні значення, використовуються різноманітні джерела ентропії та їх хешування різними значеннями, періодична зміна внутрішнього стану. Це дозволяє підвищити рівень невизначеності та забезпечити більш надійний і безпечний процес генерації випадкових чисел.

У наш час існує розмаїття методів генерації послідовностей з різним ступенем випадковості. Проте на практиці виявляється, що значна частина цих генераторів продукує послідовності, які не відповідають строгим критеріям випадковості. Показовим прикладом є генератори псевдовипадкових послідовностей, вбудовані у стандартні бібліотеки багатьох мов програмування. Числа, отримані за допомогою подібних функцій, нерідко демонструють помітні закономірності, що суперечить поняттю справжньої випадковості.

Генератори випадкових чисел відіграють ключову роль у криптографічних застосунках, створюючи ключі для шифрування та розшифрування інформації. Однак парадоксально, що саме ці генератори часто стають ахіллесовою п'ятою систем шифрування. Програмні генератори за своєю природою є детермінованими, що призводить до певної міри передбачуваності та відтворюваності отриманих послідовностей.

**Апаратні генератори випадкових послідовностей** — це особливі пристрої, що створюють ряди випадкових чисел, спираючись на вимірювання хаотично змінних параметрів фізичних процесів. У теорії ці процеси вважаються цілком непередбачуваними, що мало б забезпечувати істинну випадковість.

Однак практика вносить свої корективи: отримані таким чином числові послідовності все ж потребують ретельної перевірки. Для цього застосовують спеціально розроблені статистичні тести, які допомагають оцінити якість та ступінь випадковості згенерованих чисел.

Незважаючи на більш високу ступінь випадковості, апаратні генератори мають такі недоліки:

1. високі часові і матеріальні витрати на конструювання, встановлення та налаштування в порівнянні з ГПВП;
2. низька швидкість генерації випадкових чисел в порівнянні з ГПВП;
3. неможливість відтворення раніше згенерованої послідовності чисел, що в деяких випадках є небажаним.

**Програмні генератори**, також відомі як генератори псевдовипадкових послідовностей (ГПВП), функціонують на основі чітко визначених алгоритмів. Характерною особливістю послідовностей, створених ГПВП, є наявність періоду повторення. Через обмеженість ресурсів, кожен ГПВП неминуче потрапляє у цикл, відтворюючи одну й ту саму послідовність чисел. Тривалість цього періоду залежить від специфіки генератора та його налаштувань.

Більшість простих арифметичних генераторів хоча і володіють великою швидкістю, але мають багато серйозних недоліків:

1. занадто короткий період;
2. послідовні значення не є незалежними;
3. нерівномірний розподіл;
4. можливість відтворити вхідні дані.

**Програмно-апаратні генератори** поєднують переваги апаратних та програмних підходів до генерації випадкових чисел. Їх функціонування може відбуватися двома основними способами:

1. Генерація потоку випадкових шумів з подальшим перетворенням їх у числові послідовності.
2. Гібридний метод, де "зерно" (початкові дані для алгоритму шифрування) створюється апаратним генератором, а кінцева послідовність формується програмним шляхом. Цей підхід особливо ефективний, враховуючи зазвичай невеликий розмір "зерна".

Цікаво, що до категорії програмно-апаратних генераторів можна віднести деякі компоненти звичайного комп'ютера. Вони здатні використовувати фізичні процеси та програмні алгоритми для створення випадкових чисел. Зокрема, джерелом випадкової послідовності можуть бути:

1. шуми пристроїв комп’ютера (наприклад, процесора);
2. системний час;
3. тимчасові інтервали між натисканнями клавіш, руху миші;

Як правило, послідовності, отримані в результаті таких процесів, потребують додаткової обробки. До того ж швидкість їх отримання є досить низькою.

 Вимоги до якісного генератору випадкових чисел:

1. **Рівномірність розподілу**: згенеровані числа повинні мати рівномірний розподіл у заданому діапазоні.
2. **Незалежність**: кожне згенероване число повинно бути незалежним від попередніх та наступних чисел у послідовності.
3. **Непередбачуваність**: неможливість передбачити наступне число в послідовності, навіть знаючи попередні числа та алгоритм генерації.
4. **Великий період повторення**: для псевдовипадкових генераторів важливо мати дуже довгий період перед повторенням послідовності.
5. **Відтворюваність** (для псевдовипадкових генераторів): можливість відтворити ту саму послідовність при заданому початковому значенні.
6. **Швидкодія**: генератор повинен працювати швидко, особливо для застосувань, що вимагають великої кількості випадкових чисел.
7. **Стійкість до статистичних тестів**: генератор повинен успішно проходити різноманітні статистичні тести на випадковість.
8. **Криптографічна стійкість** (для криптографічних застосувань): неможливість передбачити попередні чи наступні числа, навіть знаючи частину послідовності.
9. **Стійкість до зовнішніх впливів**: для апаратних генераторів - нечутливість до зовнішніх факторів, які могли б вплинути на випадковість.

У реальних умовах досягнення повної відповідності всім ідеальним критеріям генерації випадкових послідовностей зазвичай виявляється недосяжним. Тому ключовим завданням стає пошук оптимального балансу між різними вимогами, враховуючи специфіку конкретної задачі.

Практика показує, що найефективніші рішення часто народжуються на стику різних підходів до генерації випадкових послідовностей. Показовим прикладом такої синергії є гібридний метод, де початкові дані отримують за допомогою апаратного генератора, а потім ці дані слугують вхідною інформацією для програмного генератора, який і формує кінцеву послідовність. Цей підхід дозволяє скористатися перевагами обох методів: непередбачуваністю фізичних процесів, що лежать в основі апаратних генераторів, та гнучкістю й ефективністю програмних алгоритмів.

Література

1. Гулак Г. Різні підходи до визначення випадкових послідовностей. / Гулак Г., Ковальчук Л. // Науково-технічний збірник "Правове, нормативне та метрологічне забезпечення системи захисту інформації в Україні". – 2001. – №3 – С. 127–133
2. Шапочка Н.В. Аналіз атак на генератори випадкових бітів / Шапочка Н.В. // Радіоелектронні і комп’ютерні системи. – 2010. – С. 99-105.
3. Шапочка Н.В. Перспективний генератор випадкових бітів на геш - функціях та його властивості / Шапочка Н.В. // Молодіжний форум "Радіоелектроніка і молодь у ХХІ ст.". – Харків – 2009.