УДК 004.312.26

© 2014 р. Д.О. Вацек, С.Е. Остапов

Чернівецький національний університет імені Юрія Федьковича, Чернівці

РЕАЛІЗАЦІЯ ТА ДОСЛІДЖЕННЯ БЛОКОВОГО ШИФРУ НА ОСНОВІ КЛІТИННИХ АВТОМАТІВ

В цій роботі реалізовано в програмному коді, досліджено статистичні характеристики та швидкодію блокового шифру на основі двовимірних клітинних автоматів з використанням околу Мура. Результати свідчать про перспективність подальших досліджень цього алгоритму шифрування.

This paper deals with the realization and investigation of the statistical and speed characteristics of the block cipher on the base of two-dimensional cellular automata. The intercellular interaction rules are used the Moore neighbourhood. The obtained results show that the cipher is very prospective for further investigation.

Вступ

XX століття ознаменувало собою створення, розвиток та впровадження комп'ютерних технологій у всіх сферах людської діяльності. Зокрема, із розвитком комунікації та поширенням комп'ютерних мереж особливо актуальним стало питання захисту даних, що передаються каналами зв'язку. При цьому збільшення можливостей обчислювальної техніки та зростання обсягу даних, що передаються каналами зв'язку, спричиняє постійне зростання вимог до надійності та швидкості роботи засобів криптографічного захисту інформації. Одним із перспективних напрямів розвитку сучасних криптосистем є криптографія на клітинних автоматах.

Ідею клітинних автоматів незалежно сформулювали в 40-х роках минулого століття Станіслів Улам [1], при вивченні росту кристалів та Джон фон Нейман при розробці саморепродуктивних машин, праці якого у даному напрямку пізніше були систематизовані та видані А. Берксом [2]. Це породило хвилю багаточисленних теоретичних та прикладних досліджень в багатьох галузях науки та техніки на базі клітинних автоматів.

У 80-х роках минулого століття Стівеном Вольфрамом опубліковано серію статей з аналізу поведінки та складності клітинних автоматів [3-5]. Ним же висловлено ідею застосування клітинних автоматів до генерування бінарних псевдовипадкових послідовностей для криптографічних цілей [6]. З тих пір дослідниками отримано вагомі результати в цьому напрямку.

Також є спроби побудови блокових шифрів на основі клітинних автоматів [7], одному з таких шифрів присвячена дана стаття.

Шифр двовимірних клітинних автоматів з околом Мура

В роботі [8] розроблено блоковий шифр на основі двовимірних клітинних автоматів, особливостями якого є перетворення, залежні від ключа; немає обмежень на довжину ключа та блоку відкритого тексту; оригінальність алгоритму перетворень. Автори розглядали бітовий та байтовий варіанти шифру. В першому вхідна інформація обробляється побітово, а в другому — побайтово. Перетворення вхідного тексту виконуються з використанням простих логічних операцій.

Однак, в роботі не виконано аналіз криптостійкості запропонованого шифру, його статистичних, розсіювальних та швидкісних характеристик. В даній роботі ми, до деякої міри, спробували з'ясувати ці питання. Ми розглядали лише бітовий варіант шифру.

Початковий стан. Розглянемо вхідний текст як послідовність бітів, отриманих, наприклад, з ASCII-кодів літер або з будь-яких інших таблиць заміни. Цим потоком бітів заповнюється квадратна матриця, розміри якої розраховуються, виходячи з довжини послідовності.

Оскільки майбутній стан кожної клітини має залежати від поточних станів її найближчих сусідів, то необхідно визначити, які саме сусідні клітини будуть входити в цей окіл. У двовимірному масиві кожна клітина має 8 сусідів, за ви-

нятком клітин, що знаходяться на межі матриці. Отже оптимально буде називати околом клітини вісім її безпосередніх сусідів, тобто тих клітин, з якими вона має спільну вершину, а не тільки спільну сторону. Такий окіл в теорії клітинних автоматів називається околом Мура.

Ключ та процес шифрування. Використаємо як ключ випадкову послідовність символів, кожному з яких в кодуванні ASCII відповідає 8 бітів. Якщо перевести всі символи ключа в двійкову форму, можна створити наступну залежність: номер розряду елемента ключа буде відповідати певному сусідові кожної клітини, а саме: перший розряд – «північно-західному» (для зручності, при вказівці сторін клітини будемо користуватися термінами сторін світу) сусідові, другий – «північному», третій – «північно-східному», четвертий – «східному» і т. д.

Пн. Зх.	Пн.	Пн. Сх.	Сх.	Пд. Сх.	Пд.	Пд. Зх.	Зх.
1	0	1	1	1	0	0	1

Пн. Зх.	Пн.	Пн. Сх.
Зх.		Сх.
Пд. Зх.	Пд.	Пд. Сх.

Рис.1. Принцип відповідності номеру біта в байті ключа – просторовому розташуванню сусідів клітини

Тоді буде проводитися операція «виключаючого АБО» (XOR) між значенням кожної клітини поля (0 або 1) і значенням тих сусідніх комірок, у яких відповідний їм біт в елементі (байті) ключа дорівнює одиниці. Тобто, якщо в першому розряді елемента ключа стоїть одиниця, то операція буде проведена між значенням поточної клітини і значенням її «північно-західного» сусіда. Якщо, до того ж, одиниця стоїть ще й, наприклад, в четвертому розряді елемента ключа, то отриманий результат буде брати участь в операції «виключаючого АБО» зі значенням «східного» сусіда клітини.

Можна сказати, що елемент (байт) ключа буде вказувати кожній клітині, з якими сусідами з її околу їй слід провести операцію XOR.

При цьому потрібно враховувати, що граничні елементи матриці мають менше восьми сусідніх клітин, у зв'язку з чим виникає питання визначення для них околу Мура. Способом вирішення цього питання без втрати в криптостійкості та необхідності збереження додаткових даних є замикання площини в тороподібну поверхню, тобто для граничних клітин, в якості сусідніх, яких не вистачає, беруться клітини з протилежного боку масиву.

Таким чином, створено клітинний автомат, в якому майбутній стан кожної клітини залежить від поточного стану свого околу, і на кожному кроці його роботи правило, що визначає цю залежність, буде змінюватися відповідно до введеного ключем. Кожен крок – нове правило.

Алгоритм шифру

Генерація ключів

1. Генерується випадкова послідовність символів (алфавіт містить 255 символів).

2. Кожному символу введеної ключової послідовності ставиться у відповідність унікальне число (ASCII – код символу).

3. Всі отримані числа подаються у двійковому представленні (результат – послідовність з нулів та одиниць довжиною 8K, де K- кількість символів у ключі).

Процес шифрування

1. Обчислюється розмірність квадратної матриці як $\sqrt{8N}$, якщо $\sqrt{8N}$ – ціле число; і $\left[\sqrt{8N}\right]+1$, в протилежному випадку (де N – кількість символів шифрованого тексту).

2. Кожному символу повідомлення ставиться у відповідність унікальне число (ASCII – код цього символу), що перетворюється до двійкового вигляду і записується в поточні 8 клітин поля.

3. В будь-якому блоковому шифрі існує проблема неповного останнього блоку, для вирішення якої доводиться додавати деякі зайві біти. У даному випадку ми дозаповнюємо останній блок нулями.

4. Виконується перетворення значення поточної клітини: проводиться операція XOR між значенням поточної клітини поля і значеннями тих сусідніх клітин, у яких відповідний їм біт в елементі ключа дорівнює одиниці. Результат взаємодії записується в поточну клітину.

Відповідність номера розряду елемента ключа наступне: перший розряд відповідає «північно-західному» сусідові, другий – «північному», третій – «північно-східному», четвертий – «східному», п'ятий – «південно-східному», шостий – «південному», сьомий – «південно-західному», восьмий – «західному».

Шифрування виконується із замиканням

площини в тороподібну поверхню, тобто в якості відсутніх сусідніх клітин для граничних елементів матриці беруться клітини з протилежного боку масиву.

Якщо довжина ключової послідовності менше довжини бітового представлення тексту, виконуємо повторення ключової послідовності для досягнення потрібної довжини.

5. Запис (вивід) шифротексту: з масиву беруться значення восьми поточних клітин, які перетворюються в десяткове значення; отримане число замінюється символом, який відповідає цьому числу (по таблиці ASCII), і записується в шифротекст.

Алгоритм розшифрування повністю ідентичний алгоритму шифрування, за винятком таких моментів:

– елементи ключа беруться в зворотному порядку;

– елементи матриці обробляються в зворотному порядку.

Результати статистичного тестування

Для дослідження статистичних властивостей результатів ми використовували пакет статистичного тестування NIST STS v.1.8, розроблений Агентством національної безпеки США. Пакет включає в себе 16 груп статистичних тестів, розроблених для перевірки гіпотези про випадковість двійкових послідовностей довільної довжини. Кожна група тестів спрямована на виявлення різноманітних статистичних дефектів. Оскільки тести запускаються кілька разів з різними параметрами, загальна їх кількість становить 189.

Тестування виконувалося за методикою, рекомендованою NIST США, яка стала вже стандартом для таких досліджень. На вхід пакету подається бінарна послідовність, загальною довжиною 10^8 бітів. Послідовність розбивається на 100 однакових частин по 10^6 бітів, які й піддаються тестуванню. Деталі методики можна прочитати в [9], однак спрощено можна сказати, що тестування відбувається в такий спосіб. Припускається, що послідовність, яка тестується, є випадковою. За тестуванням послідовності розраховується статистика тесту. З використанням спеціальної функції та статистики тесту розраховується значення ймовірності, яке порівнюється з рівнем значущості. Якщо розрахована ймовірність більша за рівень значущості, гіпотеза про випадковість приймається. Якщо ж ні — вважається, що послідовність не випадкова.

Мінімальна ймовірність, потрібна для успішного проходження конкретного тесту за цією методикою, становить 0,960150 для 100 підпослідовностей (частин) двійкового потоку.

Таким чином, у результаті тестування двійкової послідовності формується вектор значень ймовірностей $P = (P_1, P_2, \ldots, P_{189})$. Аналіз складових P_i цього вектора дозволяє зробити висновки про конкретні дефекти отриманих послідовностей.

Результати статистичного тестування наочніше усього подати у вигляді діаграми, яку показано на рис.2. Тут по вісі абсцис позначено номер тесту, а по вісі ординат — ймовірність його проходження.

Як бачимо, алгоритм демонструє непогані статистичні характеристики: всього пройдено тестів — 187 (98,9%); на рівні 1,00 — 79 тестів (41,8%); на рівні 0,99 — 52 (27,5%); 0,98 — 39 (20,6%); 0,97 — 14 (7,4%); 0,96 — 2 (1,06%). Отже, на рівні не менше 0,98 пройдено 170 зі 189 тестів, що становить 89,9%. Не пройдено лише два тести на наявність неперіодичних шаблонів.

Що стосується швидкодії алгоритму, зокрема його програмної реалізації в середовищі Delphi, маємо наступні результати: шифрування (розшифрування) тексту розміром 13475967 символів (107807736 бітів) на комп'ютері з процесором Intel Celeron CPU G1620(2,7 GHz) та 4 ГБ ОЗП не перевищує 2 хв., тобто приблизно зі швидкістю 112,3 КБ/сек. або майже 900 Кбіт за секунду. Якщо вважати, що програмна реалізація не була оптимізована за швидкодією (однопотокова реалізація), можна зробити висновок про потенційно досить високу швидкість алгоритму.

Суттєвою перевагою даного криптографічного алгоритму є те, що є ключ не бере участі в перетворенні повідомлення, він лише вказує кожному біту повідомлення, з якими саме сусідами виконувати операцію XOR. Тому при криптоаналізі неможливо буде виявити ніяких слідів взаємодії бітів ключа з бітами повідомлення, оскільки такої взаємодії взагалі немає.

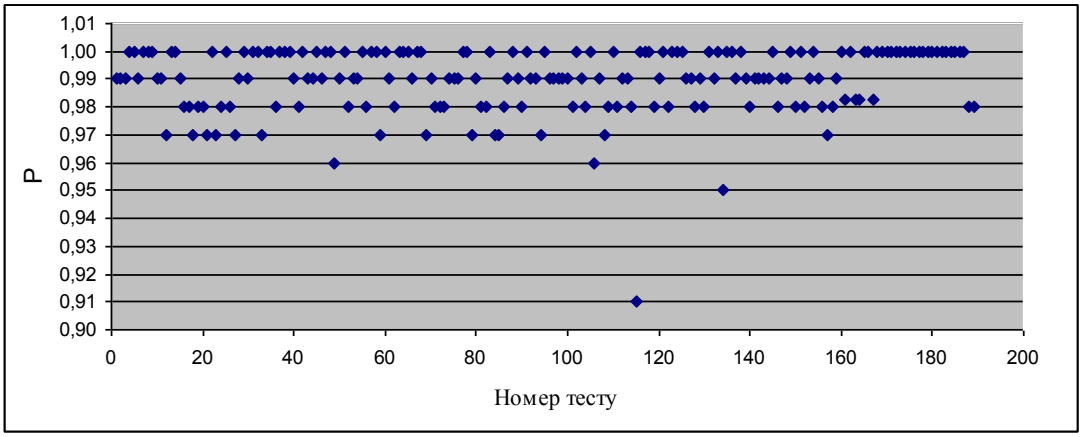

Рис.2. Результати статистичного аналізу

Висновки

Підсумовуючи все вищесказане, можна зробити висновок про те, що розглянутий алгоритм, який виявив непогану криптостійкість за даними тестування NIST STS, та швидкодію, є перспективним в плані подальших досліджень підвищення криптостійкості та ефективності застосування до нього методів криптоаналізу.

Подальші дослідження слід зосередити на покращенні статистичних характеристик, дослідженні розсіювання та оптимізації швидкісних показників шифру.

СПИСОК ЛІТЕРАТУРИ

1. *Stanislav Ulam.* Random Processes and Transformations. // Proceedings of the International Congress on Mathematics, 1952. – Vol. 2. – pp. 264–275.
2. *John Von Neumann.* Theory of Self-Reproducing Automata, A.W. Burks, ed., University of Illinois Press, Urbana, Illinois, 1966. – 388 p.
3. *Wolfram S.*. Statistical mechanics of cellular automata // Rev. Mod. Phys. – 1983. – V.55. – P. 601-644.
4. *Wolfram S.* Universality and Complexity in Cellular automata // Physica D. – 1984. – V.10. – P. 1-35.
5. *Wolfram S.* Computation Theory of Cellular Automata // Commun. Math. Phys. – 1984. – V.96 – P. 15-57.
6. *Wolfram S.* Random sequence generation by cellular automata // Advances in Applied Mathematics. – 1986. – V.7. – P. 123-164.
7. *Marcin Seredynski, Pascal Bouvry.* Block Cipher based on Reversible Cellular Automata, Proceedings of the 2004 IEEE Congress on Evolutionary Computation, vol. 2, IEEE Press, 2004. – pp. 2138-2143.
8. *Росошек С.К., Боровков А.А., Евсютин О.О.* Криптосистемы клеточных автоматов. // Прикладная дискретная математика. – 2008. – №1. – С.43-49.
9. *Потій О.В.* Методика статистичного тестування NIST STS та математичне обгрунтування тестів /Потій О.В., Лєншин А.В., Ізбенко Ю.А.// Технічний звіт ІІТ – 001-2004. – Інститут інформаційних технологій. – 2004. – 62 с.

D.O Vatsek, S. Ostapov. **The realization and research implementation of cellular automata based on block cipher**

Д.О. Вацек, С.Э. Остапов. **Реализация и исследование блочного шифра на основе клеточных автоматов**

Работа посвящена программной реализации, исследованию статистических и скоростных характеристик блочного шифра на основе двумерных клеточных автоматов с использованием окрестности Мура. Полученные результаты свидетельствуют о перспективности дальнейших исследований этого алгоритма шифрования.