НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра математичних методів захисту інформації

«До захисту допущено»
В.о. завідувача кафедри
_____ Михайло САВЧУК
«__» _____ 2021 р.

Дипломна робота

на здобуття ступеня бакалавра

зі спеціальності: 113 Прикладна математика
на тему: **«Криптографічний аналіз системи розповсюдження ключів по відкритим каналам з використанням клітинних автоматів»**

Виконав: студент 4 курсу, групи ФІ-73
Гетьман Дмитро Олексійович

Керівник: доктор фіз.-мат. наук, в.о. завідувача кафедри ФТІ "КПІ ім. Ігоря Сікорського" Савчук М.М. _____
Консультант: _ _____

Рецензент: доцент, к.ф.-м.н. Южакова А.О. _____

Засвідчую, що у цій дипломній роботі немає запозичень з праць інших авторів без відповідних посилань.

Студент _____

Київ — 2021

НАЦІОНАЛЬНИЙ ТЕХНІЧНИЙ УНІВЕРСИТЕТ УКРАЇНИ
«КИЇВСЬКИЙ ПОЛІТЕХНІЧНИЙ ІНСТИТУТ
імені Ігоря СІКОРСЬКОГО»
ФІЗИКО-ТЕХНІЧНИЙ ІНСТИТУТ
Кафедра математичних методів захисту інформації

Рівень вищої освіти — перший (бакалаврський)
Спеціальність (освітня програма) — 113 Прикладна математика,
ОПП «Математичні методи криптографічного захисту інформації»

ЗАТВЕРДЖУЮ
В.о. завідувача кафедри
_____ Михайло САВЧУК
«__» _____ 2021 р.

ЗАВДАННЯ
на дипломну роботу

Студент: Гетьман Дмитро Олексійович

1. Тема роботи: «**Криптографічний аналіз системи розповсюдження ключів по відкритим каналам з використанням клітинних автоматів**»,

керівник: доктор фіз.-мат. наук, в. о. завідувача кафедри

ФТІ "КПІ ім. Ігоря Сікорського" Савчук М.М. затверджені наказом по університету №__ від «__» _____ 2021р.

2. Термін подання студентом роботи: «__» _____ 2021р.

3. Вихідні дані до роботи: клітинні автомати, системи розповсюдження ключів по відкритим каналам з використанням клітинних автоматів, статистичні тести псевдовипадкових послідовностей, критерії на відкритий текст.

4. Зміст роботи: дослідження однієї системи розповсюдження ключів по відкритим каналам з використанням клітинних автоматів на предмет ефективності та стійкості; проведення статистичних тестів псевдовипадкових послідовностей; побудова криптографічних атак та оцінка їх складності.

5. Перелік ілюстративного матеріалу (із зазначенням плакатів, презентацій тощо): Презентація доповіді.

6. Дата видачі завдання: 10 вересня 2020 р.

Календарний план

№ з/п	Назва етапів виконання дипломної роботи	Термін виконання	Примітка
1	Узгодження теми роботи із науковим керівником	01-15 вересня 2020 р.	Виконано
2	Огляд опублікованих джерел за тематикою дослідження	Вересень-жовтень 2020 р.	Виконано
3	Дослідження методів оцінки ефективності та стійкості криптосистем	Листопад-грудень 2020 р.	Виконано
4	Програмна реалізація протоколу, статистичних тестів та криптографічних атак	Січень-березень 2021 р.	Виконано
5	Аналіз отриманих результатів та їх оформлення	Квітень-травень 2021 р.	Виконано

Студент _____ Гетьман Д.О.

Керівник _____ Савчук М.М.

РЕФЕРАТ

Кваліфікаційна робота містить: 41 стор., 9 рисунків, 0 таблиць, 7 джерел.

Метою дослідження є перевірка стійкості та ефективності однієї системи розповсюдження ключів по відкритим каналам із використанням клітинних автоматів. Завданням є побудува та реалызація криптографічних атак та оцінка їх складнностей; дослідження статистичних властивостей отриманих псевдовипадкових послідовностей.

Об'єктом дослідження є інформаційні процеси в системах розповсюдження ключів по відкритим каналам із використанням клітинних автоматів.

Предметом дослідження є математичні моделі та алгоритми, використані для побудови системи розповсюдження ключів по відкритим каналам із використанням клітинних автоматів.

В ході досліджень встановлено, що запропонована система розповсюдження ключів по відкритим каналам із використанням клітинних автоматів не є криптографічно стійкою, запропоновано чотири атаки на цю систему, реалізовано дві атаки та оцінена складність двох атак; отриманні результати статистичних тестів, які свідчать про погані статистичні властивості ключів, отриманих в результаті застосування системи розповсюдження ключів по відкритим каналам, що використовує в якості одностороньої функції клітинний автомат.

КЛЮЧОВІ СЛОВА: КРИПТОГРАФІЯ, АСИМЕТРИЧНА КРИПТОГРАФІЯ, КРИПТОАНАЛІЗ, КЛІТИННІ АВТОМАТИ, ПРОТОКОЛ ДІФФІ-ГЕЛЛМАНА, СТАТИСТИЧНІ ТЕСТИ, ОЦІНКА СКЛАДНОСТІ КРИПТОГРАФІЧНИХ АТАК

ABSTRACT

Methods of advancing the reconversion of the efficiency and effectiveness of one system and the distribution of keys through visible channels from the list of key automata. Heads of staff є prompting and realizing cryptographic attacks and assessing their complexity; until the statistical power of the rimmed pseudo-bad messages.

About the informational processes in the systems for the distribution of keys through the visible channels from the contacts of the automatic machines.

The subject of advancing mathematical models and algorithms, vicories for prompting the system and the distribution of keys through the visible channels from the keys of the key automata.

In the course of the update, a system was installed and propagated for the distribution of keys via visible channels from the keys of the automatic machines not cryptographically, but the attacks on the whole system were propagated, and the folding of them was estimated; Rejected results of statistical tests, which indicate about the abominations of statistical validity of keys, rejected as a result of storing systems and distributing keys through visible channels, which is victorious in the capacity of one-way automatic functionality of the class.

KEY WORDS: CRYPTOGRAPHY, ASYMMETRIC CRYPTOGRAPHY, CRYPTOANALYSIS, CELLULAR MACHINES, DIFFY-HELLMAN PROTOCOL, STATISTICAL CYSTLISTIC

ЗМІСТ

ВСТУП

Актуальність дослідження.

Невід'ємною частиною людського соціуму є спілкування. Із розвитком технологій та способів комунікацій спілкування та обмін інформацією на великих відстанях стали доступними для всіх. Мережа Інтернет надала людству спосіб комунікації, який дозволив швидко передавати інформацію на великі відстані, і стала найбільшою мережою в історії людства, але з іншого боку, чим більше мережа, тим більше з'являється спроб порушення конфіденційності інформації, що передається. Основною причиною компрометації інформації у мережі Інтернет є злам хакерами ключа. Успіх зламу ключа скаладає від 20% до 40% від усіх ключів, що є у хакера [1]. Висококваліфікований хакер [1] може зламати 38% ключів за 4 години, за тиждень – 62%, за 5 тижнів 89%. Але ці дані стосуються ключів, що створила людина. *Для того, щоб уникати втрати конфіденційної інформації:*

1) Ключі мають буди випадковими (псевдовипадковими), тобто створенні не людьми.

2) Треба міняти ключі якомога частіше.

Найчастіше створювати спільні ключі доводиться в умовах відкритої мережі, коли у вдох чи дікількох людей не має можливості зустрітися та домовитися про створення секретного ключа для подальщого обміну інформацією на відстані. Одним із рішень цієї проблеми є алгоритм Діффі-Хелмана, який призначений для створення загальних спільних секретних ключів за допомогою відкритих каналів. Основою для створення ключів у протоколі Діффі - Геллмана є використання односторонньої функції з властивістю комутативності - піднесення до степеня за модулем. Перехопити ключ не знаючи секретів абонентів зловмисник може лише знайшовши зворотню функцію, тобто вирішивши задачу дискретного логарифмування. Оскільки розрахункова потужність

комп'ютерів, що використовуються для знаходження обернених функцій, з кожним роком стрімко збільшується, то постійно пропонуються нові криптосистеми або вдосконалення існуючих, ефективність та стійкість яких ще потрібно дослідити, тому оцінка та дослідження цих параметрів криптосистем є актульними.

Метою дослідження є оцінка однієї системи розповсюдження ключів по відкритим каналам із викристанням клітинних автоматів на предмет ефективності та криптографічної стійкості. Для досягнення мети необхідно розв'язати **задачу дослідження**, яка полягає в досліджені статистичних властивостей отриманих ключів та побудові криптографічних атак на цю систему, із подальшої оцінкою їх складності. Для розв'язання задачі необхідно вирішити такі завдання:

- провести огляд опублікованих джерел за тематикою дослідження;

- провести теоретичні дослідження щодо клітинних автоматів, систем розповсюдження ключів по відкритим каналам, статистичних тестів псевдовипадкових послідовностей;

- вибрати та реалізувати одну ситему розповсюдження ключів по відкритим каналам із використанням клітинних автоматів; дослідити отриманні псевдовипадкові послідовності на статистичні властивості

- запропонувати та реалізувати криптографічні атаки на цю систему; дослідити їх складність

Об'єктом дослідження є інформаційні процеси в системах розповсюдження ключів по відкритим каналам із використанням клітинних автоматів.

Предметом дослідження є математичні моделі та алгоритми, використані для побудови системи розповсюдження ключів по відкритим каналам із використанням клітинних автоматів).

При розв'язанні поставлених завдань використовувались такі *методи дослідження*: методи алгебри, методи оцінки складності агоритмів, методи прогрмної реалізації криптографічних систем, статистичні критерії згоди, теорія клітинних автоматів, машинні експеременти.

Наукова новизна отриманих результатів полягає у вперше проведеному дослідженні ефективності та криптографічної стійкості однієї системи розповсюдження ключів по відкритим каналах із використанням у якості одностороньої функції певного клітинного автомата. Вперше були побудовані атаки та оцінені їх складності на запропоновану криптосистему.

Практичне значення В результаті дослідження ефективності та криптографічної стійкості однієї системи розповсюдження ключів із використанням певного клітинного автомата можна знизити ризики при застосуванні в подальшому цієї криптосистеми на практиці.

1 НЕОБХІДНІ ВІДОМОСТІ ПРО КЛІТИННІ АВТОМАТИ, РОЗПОДІЛ КЛЮЧІВ, СТАТИСТИЧНІ ТЕСТИ ПСЕВДОВИПАДКОВИХ ПОСЛІДОВНОСТЕЙ ТА СТАТИСТИЧНІ КРИТЕРІЇ ПЕРЕВІРКИ НА ЗМІСТОВНИЙ ТЕКСТ

В данному розділі будуть представлені необхідні для подальшого дослідження теоретичні відомості щодо клітиних автоматів, розподілу ключів по відкритих каналах та статистичних тестів псевдовипадкових послідовнестей. Будуть розкриті можливі застосування клітиних автоматів у криптографії, важливість розповсюдження ключів по відкритим каналам та необхідність перевірки статистичних властивостей псевдовипадкових послідовностей для використання в криптографії. Також в цьому розділі будуть приведені статистичні критерії перевірки на змістовний текст.

1.1 Протокол Діффі — Геллмана

Проблема передачі ключів по відкритих каналах була однією з основних в XX столітті. Рішення цієї проблеми було важливим, тому що при обміні зашифрованими повідомлення було необхідно також якимось чином передавати секретний ключ розшифровки, який, як правило, був такого ж розміру як і саме повідомлення. Багато криптографів та математиків намагалися знайти спосіб створення спільного секретного ключа без обміну секретною інформацією по відкритому каналу. Першими, кому вдалося знайти рішення цієї проблеми стали Уітфілд Діффі (Whitfield Diffie) і Мартін Хеллман (Martin Hellman) та, незалежно від них, Ральф Меркле (Ralph Merkle). Вони запропонували першу

систему розповсюдження ключів по відкритим каналам, яка вподальшому стала називатися протокол Діффі-Хеллмана [2]. Уітфілд Діффі та Мартін Хеллман, взявши за основу концепцію, запропоновану Ральфом Меркле, виробили алгоритм, який дозволяє виключити передачу секрета по відкритому каналу. Цей винахід ознаменував початок асиметричної криптографії.

Опис алгоритму:

Більш детальний опис алгоритму можна знайти у першоджерелі [2]. Припустимо, що є два абонента Аліса (абонент A) та Боб (абонент B). Абоненти домовляються по відкритому каналу про два числа g і p, p - велике просте число, таке що $\frac{p-1}{2}$ - теж просте число, це необхідно для покращення безпеки, g - це первісний корінь по модулю p (також просте число). Ці параметри не секретні і можуть бути перехоплені будь-якими зацікавленими особами.

1) A та B, кожний, генерує велике число, яке буде секретом абонента, секретний ключ A - k_a, B - k_b.

2) A обчислює залишок від ділення: $K_a = g^{k_a} mod(p)$, та надсилає його абоненту B.

3) B обчислює залишок від ділення: $K_b = g^{k_b} mod(p)$, та надсилає його абоненту A.

4) A обчислює: $(K_b)^{k_a} mod(p) = (g^{k_b})^{k_a} mod(p) = g^{k_b k_a} mod(p) = K_{ba}$.

5) В обчислює: $(K_a)^{k_b} mod(p) = (g^{k_a})^{k_b} mod(p) = g^{k_a k_b} mod(p) = K_{ab}$.

6) Можна побачити, що $K_{ba} = K_{ab}$, тобто абоненти Аліса та Боб отримали спільний секретний ключ.

Як результат Аліса та Боб отримали спільний секретний ключ, використовуючи лише публічні параметри: g і p, та свої секрети: k_a та k_b. Як і кожна асиметрична криптосистема, протокол Діффі - Геллмана має піблічні ключі та секретні ключі. В данному випадку публічними ключами будуть числа p та g, а секретними - k_a та k_b.

Обгрунтування стійкості:

Криптографічна стійкість протоколу Діффі - Геллмана базується на

припустимій складності задачі дискретного логарифмування, тобто знаходження x, знаючи лише $A = y^x mod(m)$, де y та m - великі прості числа, а x - велике натуральне число.

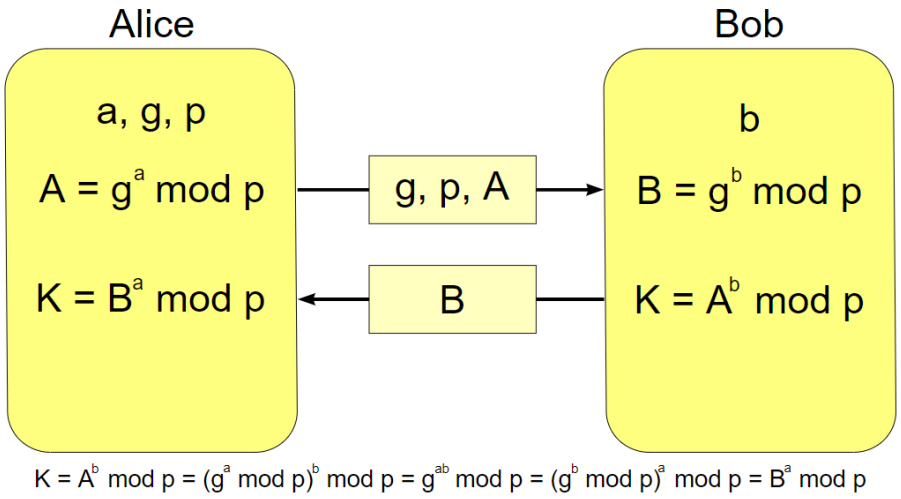

Рисунок 1.1 – Алгоритм Діффі — Геллмана, де K — спільний секрет

1.2 Клітинні автомати

Ідея створення універсального обчислювального середовища для побудови та аналізу алгоритмів була запропонована, незалежно один від одного, Джоном фон Нейманом та Артуром Бьорксом у середині XX століття [3]. Джон фон Нейман займався дослідженнями самовідтворюваних систем, його першою ідеєю було створення робота, який би збирал інших роботів.

В книзі [4] пропунується використовувати таке визначення клітинних автоматів. «Клітинні автомати є дискретними динамічними системами, поведінка яких повністю визначається в термінах локальних залежностей, в значній мірі так само як і стан справ для великого класу безперервних динамічних систем, визначених рівняннями в частинних похідних. В цьому сенсі клітинні автомати в інформатиці є аналогом фізичного поняття «поля». ... клітинний автомат можна представляти як

стилізований світ. Простір представлено рівномірної сіткою, кожна комірка якої, або клітинка, містить кілька бітів даних; час йде вперед дискретними кроками, а закони світу виражаються єдиним набором правил, скажімо невеликою довідкової таблицею, по якій будь-яка клітинка на кожному кроці обчислює свій новий стан по станам її близьких сусідів. Таким чином, закони системи є локальними і всюди однаковими.»

Тобто, узагальнивши, можна сказати, що клітинних автомат - це дискретна модель, яка являє собою решітку, де комірка містить в собі один стан із кінцевої множини станів, наприклад ноль чи одиницю. Для нормальної роботи клітинного автомата необхідно задати початковий стан та правила переходу комірок із одного стану в інший.

Класичним прикладом клітинного автомата є «Гра Життя», запропонований Джоном Конвеєм у 1970 році. Він вигадав наступні правила: місце дії гри — «всесвіт», який являє собою площину, поділену на комірки. Кожна комірка може представляти собою один з двох станів: бути живою або бути мертвою, тобто містити в собі одиницю чи ноль. Клітинка має вісім сусідів навколо себе, тобто радіус взаємодії комірок між собою - один. Розподіл комірок, які місять в собі одиницю, на початку гри називається першим поколінням. Кожне наступне покоління утворюється на основі попереднього за наведеними нижче правилами переходу станів:

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок дорівнює двом чи трьом, то вона лишається жити;

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок менше або дорівнює одиниці, то вона помирає від «самотності»;

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок більше трьох, то вона помирає від «перенаселення»;

- якщо у комірки, яка містить ноль, кількість сусідніх живих комірок дорівнює трьом, то вона «оживає»;

Гра закінчується тоді, коли виконується щось з наступного:

- всі живі комірки вмерли, тобто всі комірки містять лише нулі;

- стан клітинного автомата перестає змінюватися в наступних поколінням, тобто стан переходить у стійку конфігурацію;

- якщо конфігурація системи переходить у коливальний режим, тобто стани змінюються із певним сталим періодом;

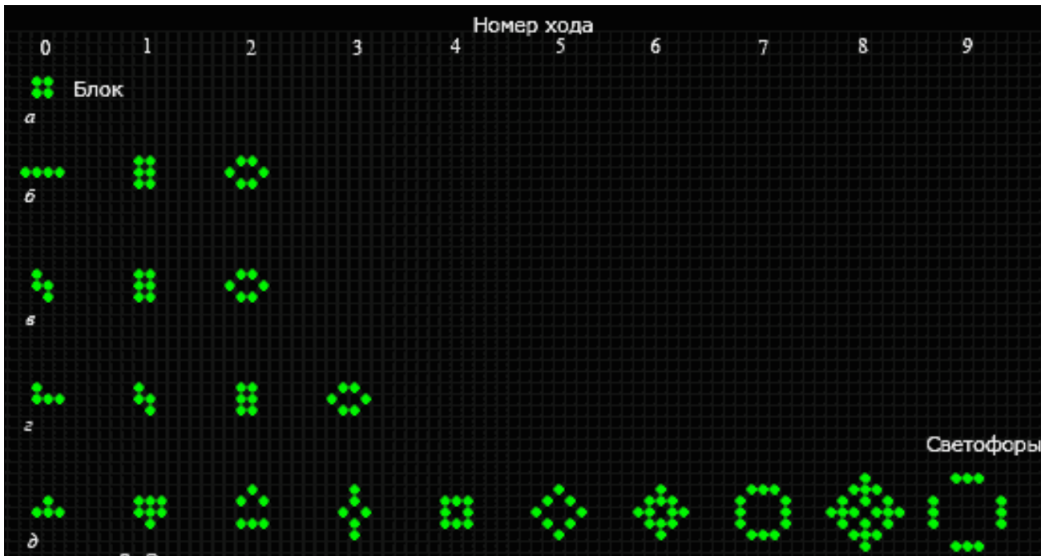

Рисунок 1.2 – Приклади переходу станів по крокам у «Грі Життя».

Застосування клітинних автоматів:

Клітинні автомати, через свою відносну простоту реалізації та універсальність, використовуються у багатьох напрямках, таких як: комп'ютерні процессори, моделювання фізичних явищ, фундаментальна фізика, але нас більше цікавить використання у криптографії. Наприклад, клітинний автомат «Правило 30» був запропонований як можлие покращення стійкості криптосистеми RSA [5]. Клітинні автомати були запропоновані для використання в криптосистемах із відкритим ключем, в даному випадку вони виступають в ролі односторонніх функцій, стійкість яких базується на неможливості обертання стану клітинного автомата, тобто неможливості однозначно отримати попередній стан системи, знаючи теперешній. Також клітинні автомати можна використовувати як генератори псевдовипадкових чисел, але статистичні властивості таких послідовностей будуть поганими для

використання в криптосистемах.

1.3 Статистичні тести псевдовипадкових послідовностей

Одним із етапів створення спільного секретного ключа, найчастіше, є вибір великого числа, представленого у бітовій чи у байтовій формі. Для цього існують генератори псевдовипадкових послідовностей, наприклад, генератор Джиффі, генератор Вольфрама, генератор «Бібліотекар». Але для використання у реальних криптосистемах нам потрібно, щоб сгенеровані послідовності мали певні статистичні властивості, такі як: рівноімовірність знаків, незалежність знаків, однорідність знаків. Це є необхідною умовою для підвищення стійкості криптосистеми. Для перевірки отриманих послідовностей на ці властивості використовують статистичні тести.

Далі будуть приведені три критерії, які перевіряють псевдовипадкові послідовності на наведенні вище статистичні властивості. Ці критерії є критеріями Пірсона для відповідно сформульованих гіпотез. Основна інформація щодо статистичних критеріїв була взята із книги [6].

Розглянемо послідовність $\{Y_j\}$, $j = 1, ..., m$, де кожна Y_j є випадковою величиною, що приймає набір значень із алфавіту A. Всі величини Y_j мають однаковий розподіл та розглядаються як вихідні значення деякого генератору.

Послідовність $\{Y_j\}$ задовольняє *умові рівноімовірності знаків*, якщо кожна Y_j розподілена рівноімовірно на A. Таким чином, кожне значення із A повинно зустрічатись у довільній реалізації даної послідовності однакову кількість разів. Тест на виконання умови рівноімовірності не відрізняється великою чутливістю, однак він доволі швидкий. В практичних задачах його рекомендується застосовувати в першу чергу, оскільки якщо послідовність не пройде цей тест, то немає рації застосовувати до неї інші тести.

Послідовність $\{Y_j\}$ задовольняє *умові незалежності знаків*, якщо імовірність прийняти деяке значення для Y_j не залежить від того, які

значення прийняли Y_1, Y_2, ..., Y_{j-1}. Однак перевірка такої умови зазвичай вкрай важка, тому часто розглядають більш послаблені вимоги – наприклад, значення Y_j не повинно залежати від значення Y_{j-1} (незалежність від попереднього знаку).

Послідовність $\{Y_j\}$ задовольняє *умові однорідності*, якщо для довільної реалізації вибірковий розподіл, одержаний на всій послідовності, буде співпадати із вибірковим розподілом, одержаним на довільній її підпослідовності достатньої довжини; іншими словами, на довільному фрагменті послідовність веде себе однаково. Зауважимо, що для виконання умови однорідності не важливо, який саме розподіл будуть мати Y_j. Зокрема, цей розподіл не обов'язково повинен бути рівноімовірним. Перевірка умови однорідності в повному обсязі також доволі складна, тому на практиці перевіряють більш слабкі форми даної умови – наприклад, розбивають послідовність на окремі інтервали та перевіряють, чи співпадають вибіркові розподіли на цих інтервалах.

Сформулюємо критерії Пірсона для кожної з наведених умов. Надалі ми вважаємо, що послідовність, яка перевіряється, представлена у вигляді байтової послідовності, тобто область значень кожної випадкової величини Y_j лежить між 0 і 255. Якщо генератор, який перевіряється, повертає послідовність бітів, то вихідні значення групуються по вісім біт.

1) *Критерій перевірки рівноімовірності знаків*

Крок 1. Розглядається байтова послідовність $\{Y_j\}$, $j = 1, ..., m$. Сформулюємо гіпотезу H_0, що полягає в тому, що всі байти послідовності рівноімовірні.

Крок 2. За значеннями послідовності $\{Y_j\}$, $j = 1, ..., m$ обчислюється статистика $\chi^2 = \sum_{j=0}^{255} \frac{(v_j - n_j)^2}{n_j}$, де v_j – число байтів j, що спостерігається у послідовності, n_j – очікуване число байтів j у послідовності за умови, що вірна гіпотеза H_0, тобто в даному випадку $n_j = \frac{m}{256}$ для всіх $j = 0, ..., 255$.

Крок 3. Обчислюється граничне значення $\chi^2_{1-\alpha}$, що відповідає рівню значимості α при $l = 255$, за формулою: $\chi^2_{1-\alpha} = \sqrt{2l}Z_{1-\alpha} + l$, де $Z_{1-\alpha}$ - (1-α)-квантиль стандартного нормального розподілу.

Крок 4. Якщо $\chi^2 \leqslant \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить експериментальним даним, у противному випадку гіпотеза H_0 відкидається.

2) *Критерій перевірки незалежності знаків*

Крок 1. Байти послідовності $\{Y_j\}$, $j = 1, ..., m$ розглядаються парами (Y_{2i-1}, Y_{2i}), $i = 1, ..., n$, де n – ціла частина $\frac{m}{2}$. Нехай H_0 – гіпотеза, що полягає в тому, що байти послідовності $\{Y_j\}$, $j = 1, ..., m$ незалежні від попереднього значення.

Крок 2. За значеннями послідовності за допомогою формули $\chi^2 = n(\sum_{ij=0}^{255} \frac{v_{ij}^2}{v_i \alpha_j} - 1)$ обчислюється значення статистики χ^2 за умови, що вірна гіпотеза H_0 про незалежність байтів, де v_{ij} – кількість появи пари (i,j) і $n = \sum_{ij=0}^{255} v_{ij}$, $v_i = \sum_{j=0}^{255} v_{ij}$ – кількість появи байта i на першому місці в парі, $\alpha_j = \sum_{i=0}^{255} v_{ij}$ – кількість появи байта j на другому місці в парі.

Крок 3. За формулою $\chi^2_{1-\alpha} = \sqrt{2l} Z_{1-\alpha} + l$ визначається граничне значення для розподілу χ^2, що відповідає рівню значимості α при $l = 255^2$.

Крок 4. Якщо $\chi^2 \leqslant \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить експериментальним даним, у противному випадку гіпотеза H_0 відкидається.

3) *Критерій перевірки однорідності послідовності*

Критерій перевірки однорідності аналогічний критерію для перевірки незалежності. Крок 1. Послідовність $\{Y_j\}$ розбивається на r відрізків довжиною $m' = [\frac{m}{r}]$, де m – загальне число байтів.

Формулюється гіпотеза H_0 про вибір байтів з того самого розподілу. Крок 2. За значеннями послідовності за допомогою формули $\chi^2 = n(\sum_{i=0}^{255} \sum_{j=0}^{r-1} \frac{v_{ij}^2}{v_i \alpha_j} - 1)$ обчислюється значення статистики χ^2 за умови, що вірна гіпотеза H_0 про вибір з одного розподілу, де де v_{ij} – кількість появи байта i у відрізку j, $v_i = \sum_{j=0}^{r-1} v_{ij}$, $\alpha_j = \sum_{i=0}^{255} v_{ij} = m'$.

Крок 3. За формулою $\chi^2_{1-\alpha} = \sqrt{2l} Z_{1-\alpha} + l$ визначається граничне значення для розподілу χ^2, що відповідає рівню значимості α при

$l = 255(r-1)$.

Крок 4. Якщо $\chi^2 \leqslant \chi^2_{1-\alpha}$, то гіпотеза H_0 не суперечить експериментальним даним, у противному випадку гіпотеза H_0 відкидається.

1.4 Статистичні критерії перевірки на змістовний текст

Під час криптоаналізу деяких криптосистем нам необхідно перебирати можливі ключі і перевіряти їх на корректність. Це можна зробити, наприклад, перехопиши деяку криптограму та розшифривовувати її доти, доки ми не отримуємо зашифроване повідомлення. Але коли кількість можливих кандидатів на ключ дуже багато, наприклад 100000, то самостійно перевіряти розшифрований текст на змістовність не доцільно. В такому випадку використовуються статистичні критерії перевірки на змістовний текст. Використання цих тестів дозволяє автоматизувати перевірку отриманих тестів на змістовність.

Існують статистичних критеріїв перевірки на змістовний текст, нижче будуть приведені декілька з них. Наступна інформаціє взята із книги [7].

Критерії заборонених $l-$грам.

1) *Загальний критерій заборонених $l-$грам.*

Ця група критеріїв орієнтується на такі комбінації $l-$грам, які у змістовному тексті зустрічаються дуже рідко або зовсім не зустрічаються. Будемо називати їх забороненими. Зрозуміло, що у випадковій послідовності будь-яка $l-$грама на фіксованих l позиціях зустрічається з ймовірністю $\frac{1}{m^l}$. Для побудови критерію необхідно знати частоту появи $l-$грам у змістовному тексті. Такі дані відомі для різних природних мов і різних сфер спілкування або типів повідомлень.

Прийняття рішення за критерієм заборонених $l-$грам.

1) В тексті X довжини L, який тестується, шукаємо заборонені $l-$грами.

2) Якщо серед знаків X не зустрілося ні однієї забороненої $l-$грами, то текст X оголошується як змістовний, тобто вірна гіпотеза H_0.

3) Якщо з'явилася хоча б одна заборонена $l-$грам, то текст X оголошується випадковою послідовністю, тобто вірна гіпотеза H_1.

Опишемо детальніше найбільш використовувані критерії заборонених знаків і заборонених біграм.

2) *Критерій заборонених знаків.*

Позначимо текст, який тестується X. Нехай він має довжину L знаків з алфавіту Z_m. Виділимо в алфавіті Z_m h знаків, які у змістовному тексті зустрічаються найбільш рідко. Будемо називати їх забороненими. Позначимо P ймовірність появи одного з цих знаків у змістовному тексті. Ймовірність появи будь-якого символу алфавіту Z_m в випадковому тексті $Q = \frac{h}{m}$. Очевидно, що ймовірність P значно менша за ймовірність Q.

Прийняття рішення за критерієм заборонених знаків.

1) В тесті X довжини L, який тестується, шукаємо заборонені знаки.

2) Якщо серед L знаків X не зустрівся ні один заборонений знак, то текст X оголошується як змістовний, тобто вірна гіпотеза H_0.

3) Якщо з'явився серед L хоча б один заборонений знак, то текст X оголошується випадковою послідовністю, тобто вірна гіпотеза H_1.

3) *Критерій заборонених біграм.*

Виділимо в алфавіті Z_m h біграм, які у змістовному тексті зустрічаються найбільш рідко. Будемо називати їх забороненими. Позначимо P ймовірність появи хоча б однієї з цих біграм у змістовному тексті. Ймовірність появи будь-якої біграми з алфавіту Z_m в випадковому тексті $Q = \frac{h}{m^2}$. Очевидно, що P ймовірність значно менша за ймовірність Q.

Прийняття рішення за критерієм заборонених біграм.

1) В тексті X довжини L, який тестується, шукаємо заборонені біграми.

2) Якщо серед знаків X не зустрілося ні однієї забороненої біграми, то текст X оголошується як змістовний, тобто вірна гіпотеза H_0.

3) Якщо з'явилася хоча б одна заборонена біграма, то текст X оголошується випадковою послідовністю, тобто вірна гіпотеза H_1.

Висновки до розділу 1

В цьому розділі були наведені необхідні теоретичні відомості щодо клітиних автоматів, розподілу ключів по відкритим каналах, статистичних тестів псевдовипадкових послідовнестей та про статистичні критерії перевірки на змістовний текст. Були розкриті можливі застосування клітиних автоматів у криптографії, необхідність передачі ключів по відкритим каналам та важливість статистичних властивостей псевдовипадкових послідовностей для використання в криптографії.

2 ОПИС ТА РЕАЛІЗАЦІЯ ОДНІЄЇ СИСТЕМИ РОЗПОВСЮДЖЕННЯ КЛЮЧІВ ПО ВІДКРИТИХ КАНАЛАХ З ВИКОРИСТАННЯМ КЛІТИННИХ АВТОМАТІВ

В цьому розділі буде запропонований клітинний автомат для використання в якості одностороньої функції у протоколі Діффі - Геллмана, буде наведена реалізація клітинного автомата та система розповсюдження ключів по відкритим каналах із використанням цього клітинного автомата. Також будуть проведені статистичні тести отриманих псевдовипадкових послідовностей та проаналізовані результати цих тестів. Прогрмна реалізація складається із п'яти модулів: index.js, protocol.js, attacks.js, tests.js та helpers.js. Кожний модуль відповідає за певний функціонал програми, що видно із назв деяких модулів. Наприклад, модуль index.js виконує роль головного, тому в ньому відбувається виклик усіх функцій та отримання результатів їх роботи; в модулі helpers.js містяться усі допоміжні функції.

2.1 Запропоновані модифікації кліитинного автомата «Гра Життя»

Для використання клітинного автомата в якості одностороньої функції у протоколі Діффі - Геллмана, необхідно провести деякі модифікації, для отримання більш надійних статистичних властивостей ключа. За основу беремо класичний приклад клітинного автомата - «Гра Життя», який використовує лише сталі параметри руху і розвитку. Ми удосконалимо правила народжуваності та продовження життя. Кожна комірка може приймати значення або нуль, або один. Початкове

розташування комірок, які містять один, на полі обирається випадково за допомогою генератора псевдовипадкових чисел, або початковим полем може бути заздалегіть підготовлена картинка.

Введемо деякі правила:

Введення правила замкненого простору

В класичній «Грі Життя» події відбувається на безмежному полі - «Всесвіті». У випадку моделювання поведінки реальної популяції поле має бути або чимось обмеженим, або замкнутим. В нашому випадку було вибрано другий варіант, оскільки тоді немає потреби вводити особливі правила для клітин на краях поля. Замикання відбувається шляхом з'єднання протилежних сторін та кутів. Така ситуація була б можлива, якби життєвий простір розміщався на торі. Така модифікація збільшує можливості щодо «виживання» комірок. Замикання реалізується створенням навколо масиву, що складає основне поле для моделі, додаткових частин поля, що відповідають протилежним краям основного поля. Програмно це було реалізовано у модулі helpers.js у вигляді функції $modifyField$, яка приймає як параметр поле, та повертає його модифікацію із додатковими частинами по краях, які відповідають протилежним сторонам масива.

Введення правила радіусу внутрішньопопуляційної взаємодії

В реальному житті взаємодія істот залежить від відстані, яку тварини можуть подолати за один життєвий цикл, щоб знайти собі подібних, тобто від радіусу взаємодії всередині популяції. У стандартній «Грі Життя» Джона Конвея кількість сусідніх клітин дорівнює 8, що відповідає радіусу взаємодії 1. В доопрацьованій моделі з правилом радіусу внутрішньопопуляційної взаємодії радіус може бути будь яким, наприклад (дозволяє мати 24 сусіда), або (максимум 48 сусідів). Для знаходження кількість сусідів клітини при довільному радіусі виведемо формулу: $n = (2R + 1)^2 - 1$, де R - це радіус внутрішньопопуляційної взаємодії, а n - це кількість сусідів. Ця модивікація вносить більшу

варіативність розвитку клітинного автомата із часом, та ускланює його обертання.

Перевірка працездотності запропонованої моделі

Модифікованні правила народжуваності та продовження життя:

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок більше або дорівнює 1 та менше або дорівнює 12, то вона лишається жити;

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок дорывнює 0, то вона помирає від «самотності»;

- якщо у комірки, яка містить одиницю, кількість сусідніх живих комірок більше 12, то вона помирає від «перенаселення»;

- якщо у комірки, яка містить ноль, кількість сусідніх живих комірок більше або дорівнює 1 та менше або дорівнює 11, то вона «оживає»;

Програмно запропонований клітинний автомат був реалізован у модулі helpers.js у вигляді функції $cellularAutomaton$, яка приймає як параметри: массив - початковий стан клітинного автомата та кількість кроків роботи автомата, та повертає кінцевий стан.

Працездатність моделі із радіусом внутрішньопопуляційної взаємодії - 2, та вище наведеними правилами була перевірена при таких параметрах: початковий стан було створенно за допомогою вбудованого в мову програмування генератора псевдовипадкових чисел, кількість кроків - 5000, розмір поля - 32 комірки.

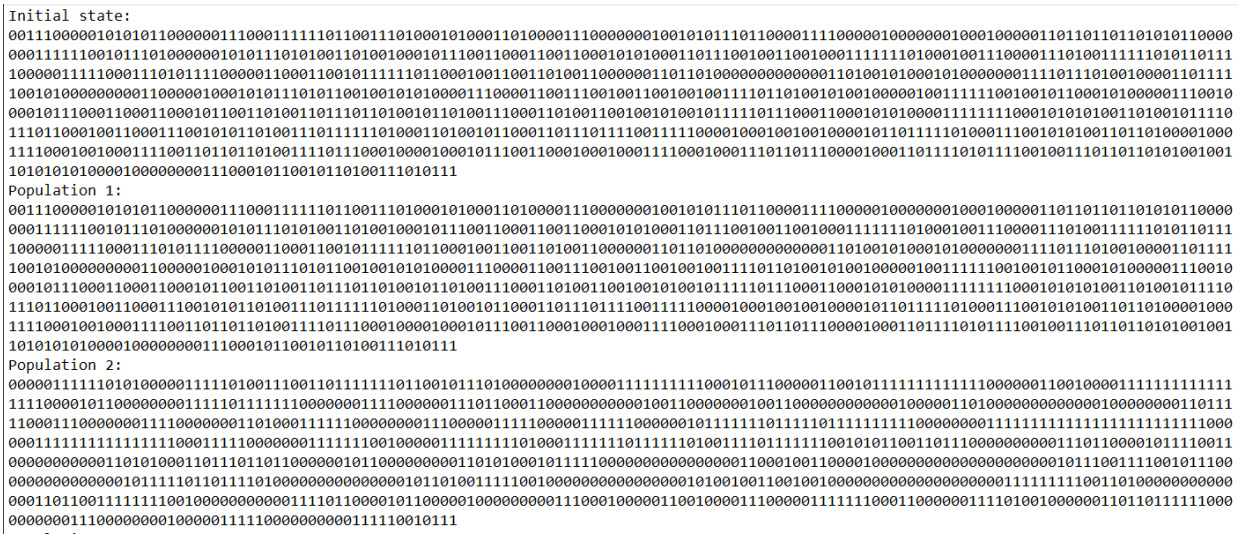

Рисунок 2.1 – Початковий стан та перші дві популяції клітинного автомата.

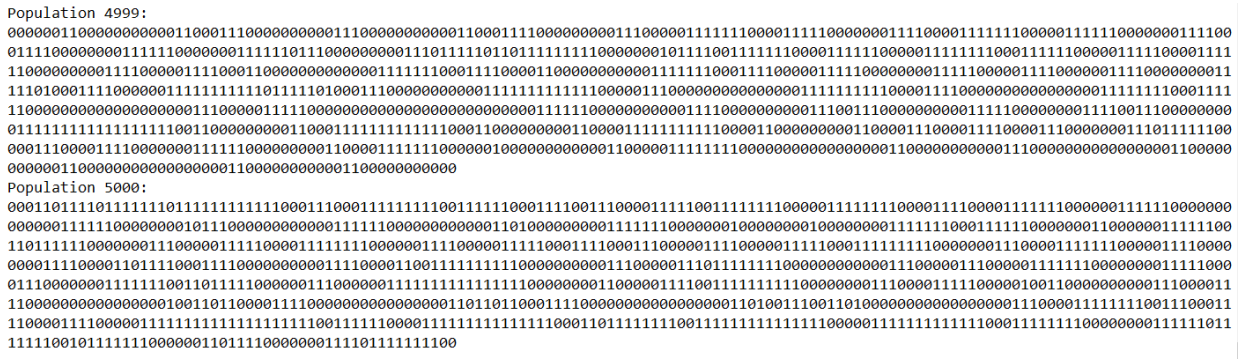

Рисунок 2.2 – Останні дві популяції клітинного автомата.

2.2 Система розповсюдження ключів по відкритим каналам із використанням модифікованого клітинного автомата

В якості ситеми розповсюдження ключів по відкритим каналам візьмемо протокол Діффі - Геллмана, але замінемо одностороннюю функцію - піднесення в степінь за модулем на клітинний автомат. В якості клітинного автомата будемо застосовувати наведену вище модифікацію «Гри життя».

Перепозначимо процедуру генерації спільного ключа використовуючи протокол Діффі - Геллмана на клітинних автоматах

Припустимо, що є два абоненти Аліса (абонент A) та Боб (абонент B). Абоненти домовляються по відкритому каналу про початковий стан та правила переходу станів клітинного автомата. Ці параметри не є секретними, та можуть бути перехоплені будь-якими зацікавленими особами.

1) Абонент A обирає свій секретний ключ - кількість кроків роботи клітинного автомата, позначимо його як k_a.

2) Абонент B обирає свій секретний ключ - кількість кроків роботи клітинного автомата, позначимо його як k_b.

3) Абонент A робить k_a кроків клітинного автомата, починаючи із початкового стану та отримує деякий стан K_a.

4) Абонент B робить k_b кроків клітинного автомата, починаючи із початкового стану та отримує деякий стан K_b.

5) Абонент A надсилає стан K_a по відкритому каналу абоненту B.

6) Абонент B надсилає стан K_b по відкритому каналу абоненту A.

7) Абонент A, починаючи зі стану K_b, продовжує роботу клітинного автомата на k_a кроків, та отримує стан K_{ba}.

8) Абонент B, починаючи зі стану K_a, продовжує роботу клітинного автомата на k_b кроків, та отримує стан K_{ab}.

Як можна побачити, Аліса та Боб зробили відпобідно $k_a + k_b$ та $k_b + k_a$ кроків. Так як початковий стан та правила переходу станів однакові, то як результат абоненти отримали один і той самий стан клітинного автомата - спільний секретний ключ.

Запропонований вище алгоритм був реалізован у модулі protocol.js у вигляді функція $DiffiHellmanProtocol$, яка приймає в якості параметрів: розмір поля та секрети обох абонентів, та повертає спільний секретний ключ.

Працездатність моделі було протестовано із такими параметрами: розмір поля - 32, секрет абонента A - 4391, секрет абонента B - 7611.

```
Secret subscriber A: 4391

Secret subscriber B: 7611

Secret key:
00000000011011001111111100000000000000000011111001111111111111100000000000111110011111111111111000000000000000000
00111111111100001111111100000000001110000111000111111111110000000011100000110011111001111110000001110000011011100
00000111100000011000000011100000000001111000111100000000011000000000111100011000000000000000000000000011100011100
00000000000000000000011100011000011110000000000000001111011110001111000000000000001111101111110111000000000000000
01111101110000001000000010000011111110011111000000011111111111111100000111111000000000011111111000000000111111101
11111111110000000000000111111111111111111111000001111100011111111000001111111000001111111000000001111111111110000111
11111000000011111111111111000011111111000000011111111111111000011111111000000111000111111110000011111111000000100000
11100111110111111110000001110001110011111111111110000000111001110001111111011111000000011111111000111111100000000
11111111111111100011111110000000111111110111110000111111111111001111111100000000000111111011111001111111100000000
```

Рисунок 2.3 – Результат роботи протоколу Діффі - Геллмана із використанням клітинного автомата.

2.3 Результати статистичних тестів отриманих псевдовипадкових послідовнотсей

Статистичні властивості псевдовипадкових послідовностей грають дуже важливу роль у криптографії. Існує безліч генераторів псевдовипадкових послідовностей, але далеко не всі використовується на практиці, так як сгенеровані послідовності повинні відповідати певним критеріям. В першу чергу треба сказати, що не існує дійсно випадкових генераторів, так як вихідний результат все одно буде від чогось залежати, будь-то такти роботи процессора чи час. Для того, щоб генератор був найбільш випадковим треба якнайбільш сильно завуалювати зв'язок між вхідними та вихідними даними.

Існує три найважливіших умов: умова рівноімовірності знаків, умові незалежності знаків, умова однорідності. Розглянемо потенційні проблеми при відсутності виконання кожної із цих умов.

Якщо ключі вибираються із множини усіх ключів неравноімовірно, то криптоаналітик має можливість здійснити направлений перебір, тобто він за рахунок неравноімовірності час атаки повного перебору скоротився.

Якщо у сгенерованих послідовностей спостерігаються наявність довгих підпослідовностей нулей чи одиниць, то це може привести до посилення ефективності алгебраїчних та кореляційних атак.

Якщо існує зв'язок між ключами, то можлива «атака із пов'язанизми ключами».

Якщо відсутня рівноімовірність знаків, то криптоаналітик може передбачити знаки, маючі інформацію про інші.

Статистичні тести були реалізовані у модулі tests.js у вигляді функцій $uniformityTestBin$ та $uniformityTestBytes$, які приймають як параметр послідовність бітів чи байтів, та повертають результат теста.

Для перевірки на статистичні властивості був сгенерований спільний секретний ключ із використанням протоколу Діффі - Геллмана на клітинних автоматах із параметрами: розмір поля - 32, секрет абонента A - 4391, секрет абонента B - 7611. Розмір отриманого ключа - 1024 біт або 128 байт.

```
Secret subscriber A: 4391
Secret subscriber B: 7611
Secret key:
00000111100111111111111111111111100000111111111000000000000000011100000111111111000000000000000011110000001111111000
11111111111111111111111111100011111111111111111111111111111111111000001101111100001111111111111000000000011110000
11110001100000000000000011110000111000000000000000000011111111000011100000000000000001111111100001111000000000000
00111111110000111110000000000000111000000000001111000000000000000011110000111101111000000000000001111111111111110
11110000000000000011111111111111101111000000000000001001111111110011100000000000001111110001110011110000000000000
00111111100001100001110000000001111111110000110111111000000000111111111000011111111110000011110111111100011111111
11111000001111111110000111111111111110000111111111000001111110111111111011110000111000011111111111111111011110000110
00111111111111111111011110000110000111111111111100111111100001110011111111001111111100001111110111111100000001111111
00000011110011111110111110011111110011111111111111111111111001111111111111111111110100111110001111111111111111111
```

Рисунок 2.4 – Результати статистичних тестів на рівноімовірність знаків.

Як можна побачити, результати бітового та байтового тестів на рівноімовірність знаків були незадовільні, що свідчить про погані статистичні властивості отриманого спільного секретного ключа у протоколі Діффі - Геллмана із використанням клітинного автомата. Так як тест на рівноімовірність знаків був провалений, то перевірку на інші статистичні властивості проводити не має сенсу, тому що сгенерований

ключ вже не можна використовувати у криптосистемах. Використання ключа із поганими статистичними властивості призводить до уразливості криптосистеми до деяких атак.

Висновки до розділу 2

У розділі запропонований клітинний автомат для подальшого використання в якості одностороньої функції у протоколі Діффі - Геллмана. Наведена реалізація клітинного автомата та системи розповсюдження ключів по відкритим каналах із використанням цього клітинного автомата. Наведені потенційні вразливості криптосистем, які використовують ключі із поганими статистичними властивостями. Реалізовані статистичні тести на рівноімовірність знаків псевдовипадкової послідовності, та застосовані на сгенерованих ключах. Результат цих тестів свідчить про погані статистичні властивості отриманого ключа. Також представлена структура програмної реалізації, та опис деяких функцій.

3 КРИПТОГРАФІЧНІ АТАКИ НА ПРОТОКОЛ ДІФФІ - ГЕЛЛМАНА ІЗ ВИКОРИСТАННЯМ КЛІТИННИХ АВТОМАТІВ

В данному розділі будуть запропоновані чотири криптографічні атаки на приведену в попередньому розділі систему розповсюдження ключів по відкритим каналам із використанням клітинних автоматів. Будуть проведені оцінки складності деяких з них. Атаки відрізняються між собою за способом та за наявною інформацією у криптоаналітика. Буде проведено аналіз цих атак, після чого можно буде зробити висновки щодо криптографічної стійкості запропонованої системи. Також слід зазначити, що при реалізації наступних атак береться до уваги, що криптоаналітик має доступ до публічних параметрів криптосистеми, тобто до правил работи клітинного автомата, начального стану та станів, що передаються між абонентами.

3.1 Атака «посередника» («Man in the middle» attack)

Як і у класичному алгоритмі Діффі - Геллмана, у алгоритмі із використанням в якості одностороньої функції клітинного автомата, має місце атака «посередника», так як змінюється лише інструмент створення спільного секретного ключа, ала не спосіб. Метою даної атаки є отримання доступу до конфеденційної інформації чи її спотворення. Ця атака має місце тому що, абоненти, які намагаються створити спільний секретний ключ, наприклад A та B, не можуть бути упевниними, що вони мають зв'язок із другим абонентом, а не з кимось іншим. Суть атаки полягає в тому, що зловмисник стає «посередником» між абонентами, та усі дані, які посилаються між абонентами, проходять через нього. Це дозволяє зловмиснику видавати себе абонентом A для абонета B, та

навпаки.

Розглянемо алгоритм атаки посередника на прикладі протоколу Діффі - Геллмана із використанням клітинного автомата: Припустимо є два абоненти A та B та деякий криптоаналітик M. Абоненти A та B використовують відкритих канал для зв'язку та обміном публічними ключами. Також по відкритому каналу абоненти домовляються про початковий стан клітинного автомата та правила його роботи, тому криптоаналітик M має знання про це.

1) Абонент A обирає свій секретний ключ - кількість кроків роботи клітинного автомата, позначимо його як k_a, та робить k_a кроків клітинного автомата, починаючи із початкового стану та отримує деякий стан K_a, та надсилає цей стан абоненту B.

2) Криптоаналітик M перехоплює це повідомлення, та замінює його на повідомлення K_m, тобто стан клітинного автомата після k_m кроків, та надсилає це повідомлення абоненту B. Також він продовжує роботу клітинного автомата, починаючи зі стану k_a на k_m кроків, отримуючи ключ k_{am}.

3) Абонент B обирає свій секретний ключ - кількість кроків роботи клітинного автомата, позначимо його як k_b, та робить k_b кроків клітинного автомата, починаючи із початкового стану та отримує деякий стан K_b, та надсилає цей стан абоненту A.

4) Криптоаналітик M перехоплює це повідомлення, та замінює його на повідомлення K_m, та надсилає це повідомлення абоненту A. Також він продовжує роботу клітинного автомата, починаючи зі стану k_b на k_m кроків, отримуючи ключ k_{bm}

5) Як результат ми отримали два канали зв'язку із криптоаналітиком M посередені, тобто в подальшому абонент A буде використовувати в якості ключа стан K_{ma}, абонент B - K_{mb}.

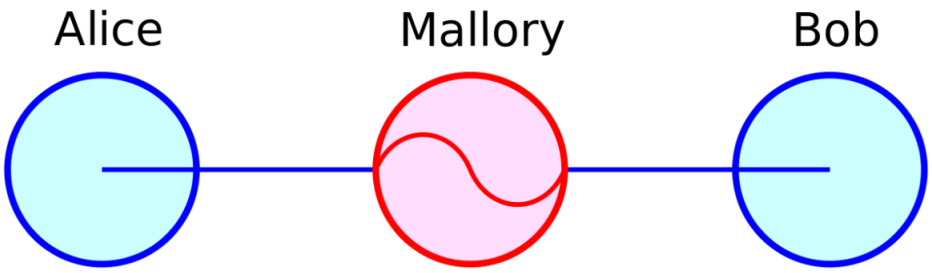

Рисунок 3.1 – Результат атаки «посередника»

Використання атаки «посередника» для підміни повідомлень:

1) Абонент A відправляє повідомлення m, зашифроване із використанням ключа K_{ma}, абоненту B.

2) Криптоаналітик M перехоплює це повідомлення, розшифровує його за допомогою ключа K_{am} та отримує відкритий текст. Далі він його може підмінити або спотворити. Після цього криптоаналітик зашифровую спотворений відкритий текст ключом k_{bm} та надсилає повідомлення m' абоненту B.

3) Такі ж самі дії криптоаналітик M проводить при передачі повідомлення від абонента B абоненту A.

3.2 Криптографічна атака із знанням одного секретного ключа

Запропонована в цьому підрозділі атака можлива, коли криптоаналітик якимось чином зміг дізнатися секрет одного із абонентів, наприклад, за допомогою соціальної інженерії. Також для реалізації даної атаки криптоаналітику потрібно перехопити лише деякі із публічних параметрів системи а саме: стани, які надсилають абоненти один одному.

Суть атаки полягає в тому, що криптоаналітик, маючи доступ до одного із секретів, виконує такі ж само дії, як і звичайний абонент, тобто починючи зі стану абонента, секретий ключ якого невідомий, продовжує роботу клітинного автомата на відомий секрет число кроків. Як результат криптоаналітик отримує секретний ключ. Слід зазначити, що ця атака має місце і в класичному варіанті протоколу Діффі - Геллмана.

Дана атака була реалізована в модулі attacks.js у вигляді функції $diffiHellmanAttack2$, яка приймає в якості параметрів: секрет одного із абонентів та стани, які передають абоненти один одному.

Працездатність атаки була перевірена із наступними параметрами системи: секрет абонета A - 7312, секрет абонента B - 3876, криптоаналітик зміг дізнатися секретний ключ абонента A.

Рисунок 3.2 – Результат криптографічної атаки із знанням секрета одного із абонентів

Були зроблені заміри часу роботи самого алгоритма та атаки, як можно побачити, атака займає набагато менше часу, а ніж сам алгоритм створення спільного секретного ключа.

3.3 Криптографічна атака із знанням тільки публічних параметрів

В даному підрозділі наведена криптографічна атака, при якій криптоаналітик має доступ тільки до відкритих параметрів системи - публічних ключів. Метою цієї атаки є отримання секретного ключа. Ця атака має місце через недолік протоколу Діффі - Геллмана із клітинним автоматом в якості одностороньої функції, так як клітинний автомат працює із кроком 1. Суть атаки полягаю в тому, що криптоаналітик, маючи початковий стан, стани абонентів, що передаються по відкритому каналу та правила переходу станів може зробити наступне:

1) Починаючи із початкового стану, послідовно, крок за кроком, перебирати стани клітинного автомата, порівнюючи проміжні стани із станами абонентів, що передаються по відкритому каналу.

2) Якщо проміжний стан дорівнює одному із станів абонентів, то кількість зроблений кроків - це секрет одного із абонентів.

3) Після знаходження першого секрета, продовжую роботу клітинного автомата для знаходження другого секрета аналогічним способом.

4) Після знаходження обох секретів, вибирається менший із них, та продовжується відкритий стан іншого абонента на менший секретний ключ кроків.

5) Як результат, криптоаналітик отримує секретний ключ.

Запропонована атака була реалізована у модулі attacks.js у вигляді функції $diffiHellmanAttack1$, яка приймає як параметри публічні ключі криптосистеми, та повертає секретний ключ.

Працездатність криптографічної атаки із знанням тільки публічних параметрів була перевірена із такими параметрами системи: секрет абонета A - 73120, секрет абонента B - 38765.

Були зроблені заміри часу створення спільного секретного ключа та

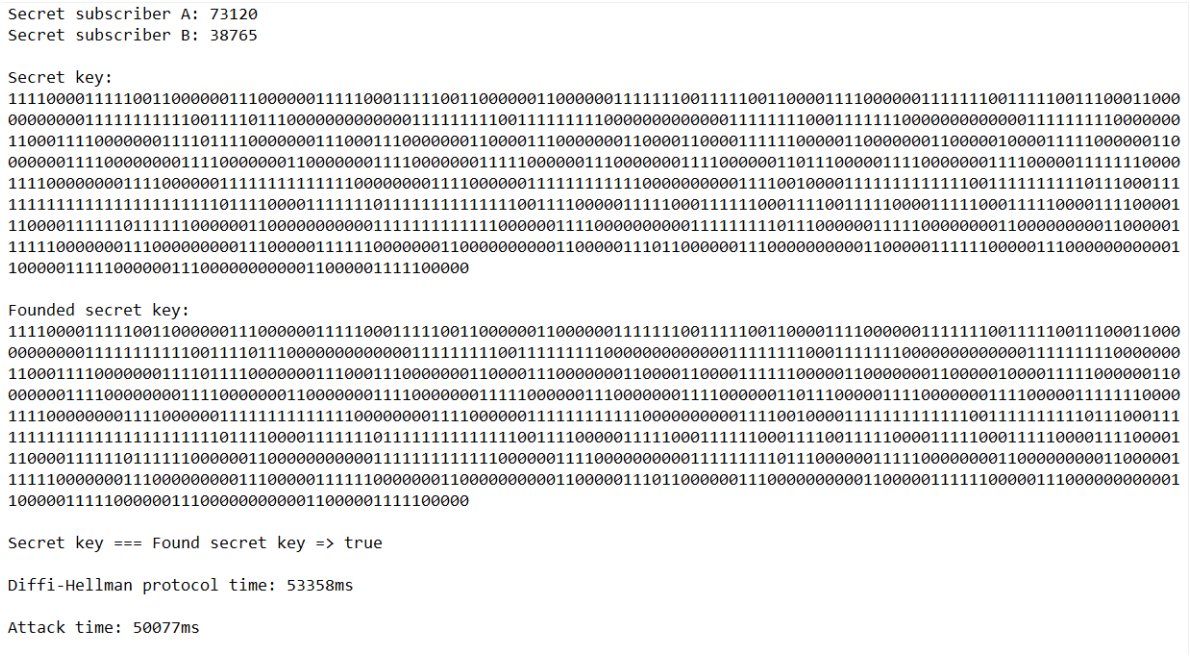

Рисунок 3.3 – Результат криптографічної атаки із знанням тільки публічних параметрів.

атаки. Як можна побачити, час атаки трохи менший за час роботи протокола. Такий результат виходить тому, що криптоаналітику необхідно зробити меншу кількість кроків роботи клітинного автомата. А такі операції як порівняння двох станів набагато швидші за отримання наступного стану клітинного автомата, тому вони не грають значної ролі.

Оцінка складності криптографічної атаки із знанням тільки публічних параметрів

Проведемо середню оцінку складності за часом:

Зробимо припущення, що секретні ключі абоненти вибирають випадково і незалежно із проміжку від 1 до N, де N - це максимально можливий секретний ключ абонента. В такому випадку складність атаки буде залежати від суми секретний ключів абонентів. Сума буде знаходитися в діапазоні від 2 до $2N$.

Обчислимо ймовірність того, що сумма буде дорівнювати деякому числу n:

$P(k_a + k_b = n) = \sum_{x=1}^{n-1} P(k_a = x)P(k_a = n - x) = \sum_{x=1}^{n-1} \frac{1}{N}\frac{1}{N} =$
$= \frac{1}{N^2}\sum_{x=1}^{n-1} 1 = \frac{n-1}{N^2}$

Середньою кількістю кроків, що робить криптоаналітик при атаці, буде математичне сподівання:

$M\xi = \sum_{n=2}^{2N} n\frac{n-1}{N^2} = \frac{1}{N^2}\sum_{n=2}^{2N} n(n-1) = \frac{1}{N^2}\sum_{n=2}^{2N} n^2 - n =$
$= \frac{1}{N^2}(\sum_{n=2}^{2N} n^2 - \sum_{n=2}^{2N} n) = \frac{1}{N^2}(\frac{(2N-1)2N4N}{6} - \frac{(2N-1)2N}{2}) =$
$= \frac{1}{N^2}(\frac{8N^2(2N-1)-6N(2N-1)}{6}) = \frac{1}{N^2}(\frac{16N^3-8N^2-12N^2+6N}{6}) = \frac{1}{N^2}(\frac{16}{6}N^3 + \frac{-20N^2+6N}{6})$
$= \frac{8}{3}N + \frac{-20N^2+6N}{6N^2} = \frac{8}{3}N - \frac{10}{3} + \frac{1}{N} \approx \frac{8}{3}N$

Отже, середня складність атаки за часом скаладає $O(N)$.

Проведемо оцінку складності за пам'ттю:

На кожному кроці криптоаналітику довотиться зберігати у пам'яті три стани клітинного автомата: ітераційний стан, стан абонента A та стан абонента B. Також необхідно виконувати операцію перехода стану клітинного автомата, при якій необхідно сберігати два масива: попередній стан та майбутній стан клітинного автомата.

3.4 Криптографічна атака із знанням деяких публічних параметрів та перехопленою криптограммою

В цьому підрозділі наведена криптографічна атака, в випадку коли криптоаналітику доступні не всі публічні параметри, а саме є невідомим початковий стан. Такий випадок може статися, наприклад, коли абоненти

заздалегіть домовилися по закритому каналу про використання деякого початкового стану. При такому розкладі, для успішної атаки, необхідно зробити припущення, що отриманий в результаті роботи алгоритма Діффі - Геллмана ключ, був використаний як ключ шифрування для деякої криптосистеми, симетричної або асиметричної, наприклад блоковий шифр DES. Якщо це припущення вірне, то криптоаналітик має змогу перехопити криптограму. Метою атаки є зменшення числа кандидатів на ключ шифрування, для подальшої атаки повного перебору (Brute-force attack). Суть атаки полягає в тому, що криптоаналітик, володіючи обома станами виконує наступні кроки:

1) Робить дві паралельні гілки розвитку клітинного автомата, починаючи із перехоплених станів абонентів A та B.

2) У кожній гілці робить по одному кроку, кожний раз порівнює стан у гілці із початковим станом іншої гілки, тобто із іншим перехопленим станом.

3) Коли знайдеться перше збіг, то це буде означати, що «менший стан» наздогнав «більший».

4) Починаючи із першого співпадіння усі наступні стани клітинного автомата можуть бути секретним ключем.

5) Після отримання масиву кандидатів на секретний ключ, проводиться послідовна розшифровка криптограми із використанням в якості ключа шифрування кандидата.

6) Проводиться перевірка отриманого текста на змістовність за допомогою критеріїв на відкритих текст.

7) Якщо текст змістовний, то був знайдений секретний ключ. Якщо отриманий текст - це випадкова послідовність, то вибирається наступний кандидат, та проводяться аналогічні дії.

Припустимо, що секретні ключі абонентів вибираються в деякому діапазоні, наприклад від 1 до 100000. Таке припущення має місце, тому що вибір дуже великого секретного ключа призведе до повільної роботи алгоритма. Якщо проводити атаку повного перебору одразу, то

доведеться перевірити усі стани клітинного автомата від одиниці до максимального секретного ключа помноженого на два. Наведений вище алгоритм дозволяє майже в два рази скоротити кількість можливих кандидатів на ключ шифрування.

Оцінка складності криптографічної атаки із знанням деяких публічних параметрів та перехопленної криптограми

Проведемо середню оцінку складності за часом:

У випадку даної атаки будемо проводити оцінку складності знаходження першого співпадіння, тобто моменту, коли «менший» стан наздоганяє «більший», так як складність подальшої розшифровки криптограми та перевірки його на змістовність буде залежати від криптосистеми, тому цю складність можна позначити як деяку константу c.

Зробимо припущення, що секретні ключі абоненти вибирають випадково і незалежно із проміжку від 1 до N, де N - це максимально можливий секретний ключ абонента. В такому випадку складність атаки буде залежати від різниці секретний ключів абонентів. Різниця буде знаходитися в діапазоні від 1 до $N-1$.

Обчислимо ймовірність того, що сумма буде дорівнювати деякому числу n:

$$P(|k_a - k_b| = n) = 2\sum_{x=1}^{N-n-1} P(k_a = x)P(k_a = n+x) = 2\sum_{x=1}^{N-n-1} \frac{1}{N}\frac{1}{N} = \frac{2}{N^2}\sum_{x=1}^{N-n-1} 1 = \frac{2(N-n-1)}{N^2}$$

Середньою кількістю кроків, що робить криптоаналітик при атаці, буде математичне сподівання:

$$M\xi = 2\sum_{n=1}^{N-1} n\frac{2(N-n-1)}{N^2} = \frac{4}{N^2}\sum_{n=1}^{N-1} Nn - n^2 - n = \frac{4}{N^2}\sum_{n=1}^{N-1} n(N-1) - n^2 = \frac{4}{N^2}(\sum_{n=1}^{N-1} n(N-1) - \sum_{n=1}^{N-1} n^2) = \frac{4(N-1)}{N^2}\sum_{n=1}^{N-1} n - \frac{4}{N^2}\sum_{n=1}^{N-1} n^2 = \frac{4(N-1)}{N^2}\frac{N(N-1)}{2} - \frac{4}{N^2}\frac{(N-1)N(2N-1)}{6} = \frac{12N(N^2-2N+1)-4N(2N^2-3N+1)}{6N^2} = \frac{12N^3-24N^2+12N-8N^3+12N^2-4N}{6N^2} = \frac{4N^3-12N^2+8N}{6N^2} = \frac{2}{3}N + \frac{12N^2+8N}{6N^2} = \frac{2}{3}N + 2 + \frac{4}{3N} \approx \frac{2}{3}N$$

Отже, середня складність атаки за часом скаладає $O(N)$.

Проведемо оцінку складності за пам'ттю:

На кожному кроці криптоаналітику доводиться зберігати у пам'яті три стани клітинного автомата ітераційний стан, стан абонента A та стан абонента B. Також необхідно виконувати операцію перехода стану клітинного автомата, при якій необхідно сберігати два масива: попередній стан та майбутній стан клітинного автомата.

Висновки до розділу 3

В данному розділі запропоновані та описані чотири криптографічні атаки на приведену в попередньому розділі систему розповсюдження ключів по відкритим каналам із використанням клітинних автоматів. Реалізовано та проведені дві атаки із заміром часу роботи. Наведені обчислення складності двох атак. Як видно із результатів оцінки складності атак та вимірів часу їх роботи, то вони мають майже таку ж складність, як і сам алгоритм, що в свою чергу, свідчить про криптографічну нестійкість запропонованої системи.

ВИСНОВКИ

В роботі наведені теоретичні відомості щодо клітиних автоматів, розподілу ключів по відкритим каналах, статистичних тестів псевдовипадкових послідовнестей та теоретична інформація про статистичні критерії перевірки на змістовний текст. Розкриті можливі застосування клітиних автоматів у криптографії, необхідність передачі ключів по відкритим каналам та важливість статистичних властивостей псевдовипадкових послідовностей для використання в криптографії. Також у роботі наведена реалізація запропонованого клітинного автомата, перевірена його працездатність; представлена реалізація протоколу Діффі - Геллмана із використанням клітинного автомата в якості одностороньої функції, перевірена працездатність.

Як результат, проведені статистичні тести отриманих ключей; запропоновані та описані чотири атаки, з яких дві були реалізовані, та одна була реалізована частково, тому що друга частини атаки залежить від криптосистеми. Наведені обчислення складностей двох атак. Складність атаки із знанням тільки публічних параметрів дорівнює $O(N)$, де N - це максимальний секретний ключ, який може вибрати абонент. Складність атаки із знанням деяких публічних параметрів та перехопленою криптограммою дорівнює $c \cdot O(N)$, де c - це константа, яка дорівнює складності розшифрування та перевірки тексту на змістовність. Результат перевірки отриманих псевдовипадкових послідовностей на статистичні властивості свідчить про погані характеристики ключа. Виходячи з результатів статистичних тестів отриманих послідовностей та оцінок складностей атак, можна зробити висновок про неефектиіність та криптографічну нестійкість запропонованої системи розповсюдження ключів по відкритим каналам з використанням клітинних автоматів.

Для використання криптосистем на практиці, вони повинні відповідати певним критеріям, таким як криптографічна стійкість,

швидкість роботи, універсальність, тощо. Із швидким ростом обчислювалтних потужностей новітніх комп'ютерних систем, пропонуються нові чи вдосконаленні криптосистеми, тому необхідно проводити їх криптографічний аналіз, перед використанням на практиці. У подальших дослідженнях необхідно проводити криптографічний аналіз інших криптосистем для виявлення уразливостей чи поганих статистичних властивостей.

ПЕРЕЛІК ПОСИЛАНЬ

1. Уийр М. DEFCON 17. Взлом 400 000 паролей, или как объяснить соседу по комнате, почему счет за электричество увеличился. Часть 1. [електронний ресурс] // Сайт Хабр - https://habr.com/ru/company/ua-hosting/blog/422731/

2. Diffie W. Whitfield, Hellman M. New directions in Cryptography / Whitfield Diffie, Martin Hellman, IREE TRANSACTIONS ON INFORMATION THEORY. VOL. IT-22, NO. 6, NOVEMBER 1976

3. Von Neumann, J. and A. W. Burks. Theory of self-reproducing automata / Urbana, University of Illinois Press(1966) - 408pp.

4. Тоффоли Т., Марголус Н. Машини клеточных автоматов / Тоффоли Т., Марголус Н. - Москва «МИР» 1991 - 284стр.

5. Kumar J.K.J. Improving Resistance against Attack of L2DCASKE Encryption Algorithm by using RCA Rule 30 based S-Box / International Journal of Computer Applications. – 2013. – vol. 70, № 16. – pp. 26–30.

6. Ивченко Г.И., Медведев Ю.И. Введение в математическую статистику / Ивченко Г.И., Медведев Ю.И. Москва: 2010.— 600 с.

7. Бабаш А.В., Шанкин Г.П. Криптография – / Бабаш А.В., Шанкин Г.П. Москва: СОЛОН-Р, 2002. – 512 с. - Гл.3