Міністерство освіти і науки України

Харківський національний університет імені В. Н. Каразіна

Факультет комп’ютерних наук

Спеціальність 125 «Кібербезпека»

Освітня програма «Кібербезпека»

«Допущено до захисту»

В.о. завідувача кафедри БІСТ

Ольга МЕЛКОЗЬОРОВА

_____

« » 2024 р.

Пояснювальна записка

до кваліфікаційної роботи бакалавра

на тему: «Дослідження та аналіз застосування постквантового алгоритму асиметричного шифрування Crystals-Kyber для зашифрування інформації, що вбудовується у зображення та аудіофайли»

оцінка « » Керівник _____ к.т.н. Єсіна М. В.

Голова ЕК

Лемешко О. В. _____ Рецензент _____ к.т.н. Бобух В. А.

Виконавець: студент групи КБ-41

_____ Бодня М. О.

Харків – 2024

РЕФЕРАТ

Пояснювальна записка до проєкту бакалавра містить 55 сторінок, 48 рисунків, 13 таблиць, 12 лістингів, 4 додатки, 22 джерел посилань.

Мета роботи полягає в аналізі та дослідженні: постквантового алгоритму Crystals-Kyber, проблематики, на якій ґрунтується Kyber, обчислювальних операцій та математичних перетворень методу Kyber, сфери його експлуатації в інформаційно-технічних системах, можливості комбінування зашифрованих повідомлень з техніками стеганографічного захисту інформації та огляду рівня якості та надійності класичних алгоритмів стеганографії.

Об’єкт дослідження – методики векторів криптографії та стеганографії для забезпечення інформаційної безпеки в інформаційно-телекомунікаційних системах.

Предмет дослідження – специфікація постквантового алгоритму Crystals-Kyber, особливості, спектр математичних перетворень, стан та області інтеграції в системи комунікації, аналізування прийомів монтування інформації в різноманітні мультимедійні об’єкти, оцінка стану захищеності та якості стеганосистем.

Основними методами дослідження є експериментальний та порівняльний аналіз алгоритмів за різноманітними критеріями надійності та стійкості, елементи теоретичного аналізу на базі відкритих джерел. Для проведення аналізу використовувалися власні джерела, перелік яких наведений в додатку А.

У роботі досліджено: особливості алгоритму Kyber, спектр математичних операцій, які в ньому застосовуються, загальна специфікація алгоритму, степінь якості та надійності різних стеганосистем, а також потенційні сфери їх експлуатації.

На основі досліджених специфікацій стеганографічних алгоритмів побудована програмна реалізація системи зв’язку. Дана програмна реалізація дозволяє симулювати систему передачі даних з експлуатацією модулів стеганографічного кодування та декодування. Результати тестування розробленого застосунку представлені в додатку Б. Окремі фрагменти коду функціональних модулів програмної реалізації наведені в додатку В. Створену програмну реалізацію можна

використовувати для дослідження параметрів безпеки та надійності стеганографічних систем, заснованих на експлуатації цифрових нерухомих зображень та аудіопотоків. Програмна реалізація містить 3 блоки реалізованих стеганографічних систем. Тестування програмного застосунку дозволяє оцінити параметри системи передачі. Гістограми емпіричних даних, отриманих в рамках досліджень каналу передачі наведені в додатку Г.

Результати роботи можуть застосовуватися у варіативних наукових виданнях, а також для отримання базових знань в сфері постквантової криптографії та векторів стеганографії.

Ключові слова: АУДІОКОНТЕЙНЕРИ, ІНФОРМАЦІЙНИЙ КОНТЕНТ, КОНТЕЙНЕР, КРИПТОАНАЛІЗ, НЕРУХОМІ ЗОБРАЖЕННЯ, ПОСТКВАНТОВА КРИПТОГРАФІЯ, ПРОСТОРОВА ОБЛАСТЬ, РОБАСТНІСТЬ, СЕКРЕТНИЙ КЛЮЧ, СТЕГАНОАНАЛІЗ, СТЕГАНОГРАМА, СТІЙКІСТЬ.

ABSTRACT

The explanatory note for the bachelor's project contains 55 pages, 48 figures, 13 tables, 12 listings, 4 appendices, 22 references. The aim of the work is to analyze and investigate the post-quantum algorithm Crystals-Kyber, the problems on which Kyber is based, the computational operations and mathematical transformations of the Kyber method, its areas of application in information and technical systems, the possibilities of combining encrypted messages with steganographic information protection techniques, and review of the level of quality and reliability of classical steganography algorithms.

Object of the research – methods of cryptographic and steganographic vectors for ensuring information security in information and telecommunication systems.

Subject of the research – specification of the post-quantum algorithm Crystals-Kyber, its features, range of mathematical transformations, state and areas of integration into communication systems, analysis of methods for embedding information into various multimedia objects, and assessment of the security and quality of steganographic systems.

The main research methods are experimental and comparative analysis of algorithms based on various reliability and robustness criteria, as well as elements of theoretical analysis using open sources. Own sources were used for the analysis, the list of which is provided in Appendix A.

The work investigates: the features of the Kyber algorithm, the range of mathematical operations it employs, the overall specification of the algorithm, degree of quality and reliability of various steganosystems, as well as their potential areas of application.

Based on the investigated specifications of steganographic algorithms, a software implementation of the communication system was developed. This software implementation allows for the simulation of a data transmission system using steganographic encoding and decoding modules. The results of testing the developed application are presented in Appendix Б. Specific code fragments of the functional modules of the software implementation are provided in Appendix B. The created software implementation can be used to study the security and reliability parameters of steganographic systems based on the

use of digital still images and audio streams. The software implementation includes 3 blocks of realized steganographic systems. Testing the software application allows for the evaluation of the transmission system parameters. The histograms of empirical data obtained within the framework of channel transmission studies are presented in Appendix Г.

The results of the work can be applied in various scientific publications and also for obtaining basic knowledge in the field of post-quantum cryptography and steganography vectors.

Keywords: AUDIO CONTAINERS, INFORMATION CONTENT, CONTAINER, CRYPTOANALYSIS, STILL IMAGES, POST-QUANTUM CRYPTOGRAPHY, SPATIAL DOMAIN, ROBUSTNESS, SECRET KEY, STEGANALYSIS, STEGANOGRAM, RESILIENCE.

ЗМІСТ

ПЕРЕЛІК ПОЗНАЧЕНЬ І СКОРОЧЕНЬ

БУОК – блок урахування особливостей контейнеру

ДПФ – дискретне перетворення Фур’є

ЗДПФ – зворотне дискретне перетворення Фур’є

ЗСЛ – зорова система людини

ІТС – інформаційно-телекомунікаційні системи

КЗ – канал зв’язку

НЗБ – метод модифікації найменш значущого біту

НСД – несанкціонований доступ

ПВІ – метод псевдовипадкового інтервалу

ПВП – метод псевдовипадкової перестановки

ПЧ – поріг чутливості

ССЛ – слухова система людини

AKE – протокол обміну ключами з автентифікацією

BMP – bitmap picture

ECC – elliptic curve cryptography

JPEG – joint photographic experts group

KEM – механізм інкапсуляції ключів

LSB – least significant bit

LWE – проблема навчання з помилками

MLWE – проблема модульного навчання з помилками

NIST – національний інститут стандартів і технологій

RGB – червоний, зелений, синій

UAKE – односторонній протокол обміну ключами з автентифікацією

ВСТУП

Еволюція інформаційно-телекомунікаційних систем (ІТС) суттєво спростила доступ до інформації та проведення комунікації, розширивши людські можливості. Інтернет став технічною платформою для обміну інформацією, спілкування, співпраці та інновацій. Сучасна інфраструктура мереж володіє великими потужностями, які дозволяють передавати масивні потоки даних за короткі проміжки часу. Глобальна мережа Інтернету надає широкий спектр можливостей в різних сферах життя. Однак, виникнення глобальної мережевої системи стало початковою точкою для розвитку кібератак. Потенційні кіберзловмисники використовують вразливості інформаційних систем [1] для реалізації сценаріїв атак, направлених на витік інформаційного контенту або пошкодження системи, що може супроводжуватися повною її відмовою. Більшість атак направлені на компрометацію чутливої інформації для реалізації зловмисних цілей. Наслідками реалізації кіберзагроз [1, 2] є отримання несанкціонованого доступу (НСД) до конфіденційних даних, порушення властивостей інформації та інформаційно-обчислювальних систем, фальсифікація даних тощо. Зловмисна сторона може використовувати скомпрометовану інформацію як інструмент впливу на потенційну жертву [1].

Виникнення потенційних загроз в цифровому середовищі спонукало до створення технік інформаційної безпеки. Наразі інтенсивно актуалізується питання забезпечення безпеки інформації як методу протидії можливим загрозам в інформаційному просторі. Фахівці кібербезпеки та комп’ютерної інженерії оновлюють стандарти безпеки та відкривають інноваційні механізми захисту інформації та надійності електронних ресурсів. Прогресивними векторами безпеки [2], які експлуатуються в парадигмі інформаційної безпеки є криптографія та стеганографія. Криптографія [2] – це наука, яка дозволяє забезпечити безпеку інформації шляхом перетворення її у нерозбірливий вигляд для сторонніх користувачів. В свою чергу, стеганографія дозволяє приховувати факт передавання інформаційного контенту, інкапсулюючи його в так звані контейнери (мультимедійні

об’єкти, графічні зображення, мережеві пакети тощо) [2, 3]. З появою машинних комплексів, заснованих на квантовій інформатиці, з’явилися нові виклики для індустрії кіберзахисту. Це зумовлено вразливістю класичних криптографічних схем перед методами квантового криптоаналізу. Розвиток варіативних криптографічних моделей та криптоперетворень сприяло породженню нового покоління криптографічних шифрів в галузі криптографії. Оновлені методики криптографічних алгоритмів постквантового періоду здатні реагувати на загрози з боку квантових комп’ютерів. На даний момент, постквантові алгоритми криптографії синхронізуються із мережевими протоколами [4] для забезпечення безпеки сеансів обміну даними та комунікаційних взаємовідносин. Поширеною практикою є гібридизація методів захисту [2] для забезпечення надійності системи інформаційної безпеки.

Метою даної роботи є аналіз особливостей та досвіду експлуатації постквантових механізмів на прикладі фіналісту конкурсу NIST PQC – Crystals-Kyber, варіативних стандартизованих технік стеганографії в парадигмі інформаційної безпеки. Розглянуті аспекти безпеки застосовуються в модулях комплексних систем захисту інформації, програмно-апаратних комплексів систем захисту, стандартах та стратегіях безпеки розроблених Державною службою спеціального зв’язку та захисту інформації України. Підвищується рівень інтересу до алгоритмів стеганографічного захисту інформації у воєнних галузях. Важливість даного підходу полягає в можливості створення прихованого каналу зв’язку (КЗ) для обміну секретною інформацією.

1 АНАЛІЗ ПОСТКВАНТОВОГО АЛГОРИТМУ CRYSTALS-KYBER

1.1 Основні положення шифру Crystals-Kyber

Поява нового напрямку загроз, пов'язаного з квантовою інформатикою, поставило під загрозу стійкість традиційних криптосистем. У зв'язку з цим NIST ініціював у 2017 році процедуру створення нових криптографічних алгоритмів з відкритими ключами [5], здатних протидіяти методам квантового криптоаналізу. Один з фіналістів 3-го раунду конкурсу NIST PQC [6], який отримав позитивну оцінку в області забезпечення захисту інформаційних систем – Crystals-Kyber.

Шифр Crystals-Kyber представляє сімейство алгоритмів асиметричного шифрування. Узагальнена модель роботи асиметричних криптографічних систем проілюстрована на рис 1.1.

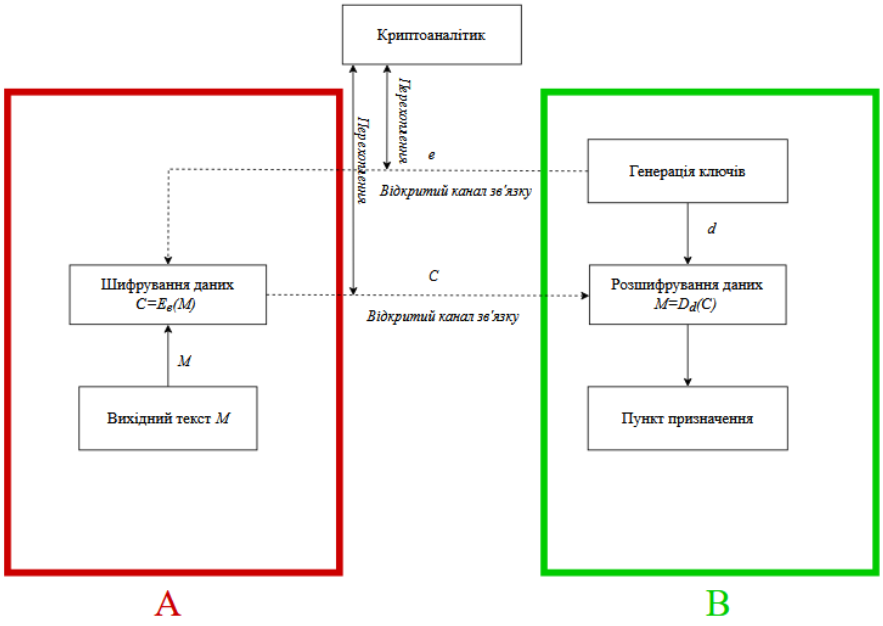


Рисунок 1.1 – Узагальнена модель роботи асиметричної криптосистеми

Інструментами управління процедурами зашифрування/розшифрування є пара відкритий/секретний ключ [2]. Через відкриті КЗ можливе поширення лише відкритих ключів для шифрування інформації. Секретний ключ є базовим атрибутом для розшифрування конфіденційних даних. Компрометація секретних ключів може призвести до НСД до чутливої інформації. Сучасні криптосистеми в мережевих

протоколах ґрунтуються на криптографічних моделях, які викликають у криптоаналітика складності відносно розв’язання. Чим складніше завдання визначення загальносистемних параметрів алгоритму, тим важче зловмиснику витягнути правильні значення ключів.

Crystals-Kyber має відношення до оновленої криптографії постквантового періоду, який враховує всі критичні виклики кібербезпеці. Crystals-Kyber [5, 6] є механізмом інкапсуляції ключів (key-encapsulation mechanism, KEM), який дозволяє згенерувати спільний секрет між сторонами електронної комунікації в просторах цифрового середовища. Алгоритм Kyber експлуатується в протоколах безпечного обміну ключами для узгодження криптографічних ключів у мережі. Функції методу Kyber на вхід приймають фіксований блок повідомлення розміром 32 байти [6], над яким виконуються асиметричні перетворення при передачі по відкритих КЗ. Обмеженість розміру повідомлення спонукає розділити його на блоки для шифрування. З одного боку, це не зовсім зручно, оскільки втрата одного блоку призведе до порушення цілісності інформаційних даних. Однак, з професійної точки зору, такий підхід є необхідним для забезпечення ефективного та безпечного обміну даними в мережевих протоколах.

Важливим атрибутом криптографічних алгоритмів є стійкість проти потенційних атак криптоаналізу. Криптоалгоритм Crystals-Kyber спроможний забезпечити 1, 3, 5 рівні криптографічної стійкості [6] згідно з моделлю рівнів безпеки NIST. Кожний рівень визначає значення комплексу загальносистемних параметрів, які реалізують конкретний рівень стійкості.

Безпека методу Crystals-Kyber ґрунтується на складності розв’язання проблеми навчання з помилками (LWE) через модульні решітки [5-7]. Ідея гіпотези LWE заснована на складності знаходження параметра *s*, використовуючи (A, *t*) з виразу $t=As+e$, де A – випадкова матриця, s – випадковий вектор секретного ключа, значеннями якого є коефіцієнти поліному у визначених границях, e – вектор з випадковими невеликими коефіцієнтами, які вибираються з певного розподілу [6, 7]. Дослідження показали [7], що не є обов’язковим випадковим чином визначати

параметр *s* у контексті проблеми LWE. Незалежно від того, чи вибраний вектор *s* з того ж самого обмеженого розподілу [6, 7], що і помилка *e*, проблема залишається складною для вирішення. Концепція алгоритму Crystals-Kyber дозволяє протидіяти технікам сучасного криптоаналізу. Вирішення проблеми модульного навчання з помилками (MLWE) є дуже складним завданням [6] навіть для квантових програмно-апаратних комплексів. Цим пояснюється високий рівень популяризації даного алгоритму.

Алгоритм шифрування Kyber належить до криптосистеми з відкритим ключем. Як і будь-який алгоритм асиметричного шифрування, Crystals-Kyber включає три ключові процедури [2]: генерацію ключової пари, шифрування та розшифрування. Потрібно зробити акцент, на важливості етапу формування ключової пари, оскільки розмір та структура криптографічних ключів впливає на стійкість криптосистеми. Поняття стійкості визначається здатністю криптографічного алгоритму залишатися невразливим до потенційних атак в цифровому середовищі. Пара відкритий/секретний ключ є важливими атрибутами для контролю процесами шифрування та розшифрування. В контексті криптографії, процедури шифрування та розшифрування інформації управляються різними ключами [2]. Як система KEM, Kyber включає два додаткові прийоми [7] інкапсуляції та декапсуляції криптографічних ключів. Такі прийоми застосовуються в мережевих протоколах обміну ключами. Схематичні процедури в постквантовому примітиві Crystals-Kyber продемонстровано на рис. 1.2.

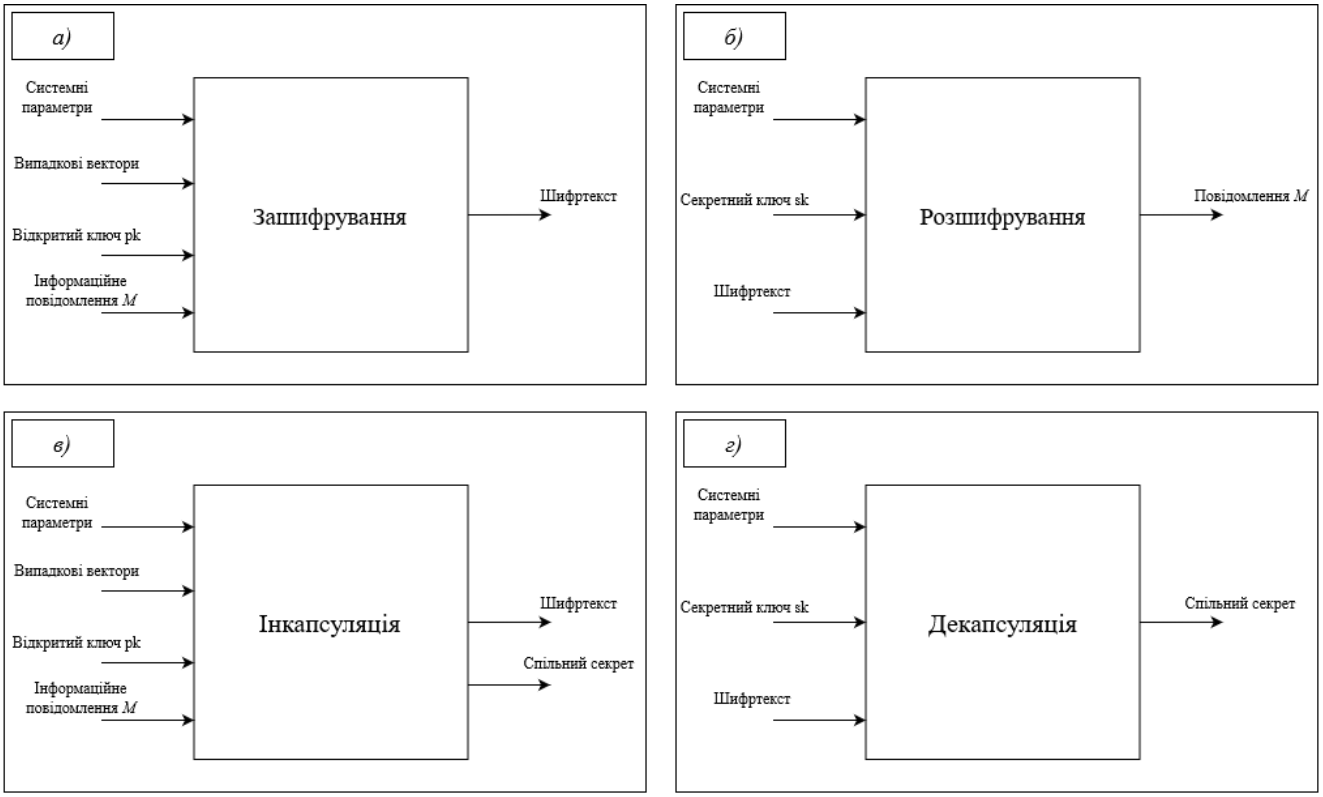


Рисунок 1.2 – Спрощені моделі процедур зашифрування (а), розшифрування (б), інкапсуляції (в), декапсуляції (г) в постквантовому алгоритмі асиметричного шифрування Crystals-Kyber

1.2 Експлуатація Crystals-Kyber в протоколах обміну ключами

Постквантовий криптопримітив Kyber використовується для захисту криптографічних ключів у мережевому просторі. Після визнання NIST ефективності алгоритму Kyber в контексті захисту інформації, він був інтегрований в сучасні мережеві протоколи. В системі обміну ключами на основі KEM використовуються механізми гешування. Ці функції породжують унікальне геш-значення або «дайджест» [7, 8] для переданих даних, і це геш-значення є фактичними підтвердженням образу сторони комунікаційного процесу. При породженні спільних ключів у парадигмі KEM [7] залишається «внесок» кожного з учасників. В парадигмі генерації спільного секрету існують схеми KEM (рис. 1.3, а) та Діффі-Хеллмана (рис. 1.3, б).

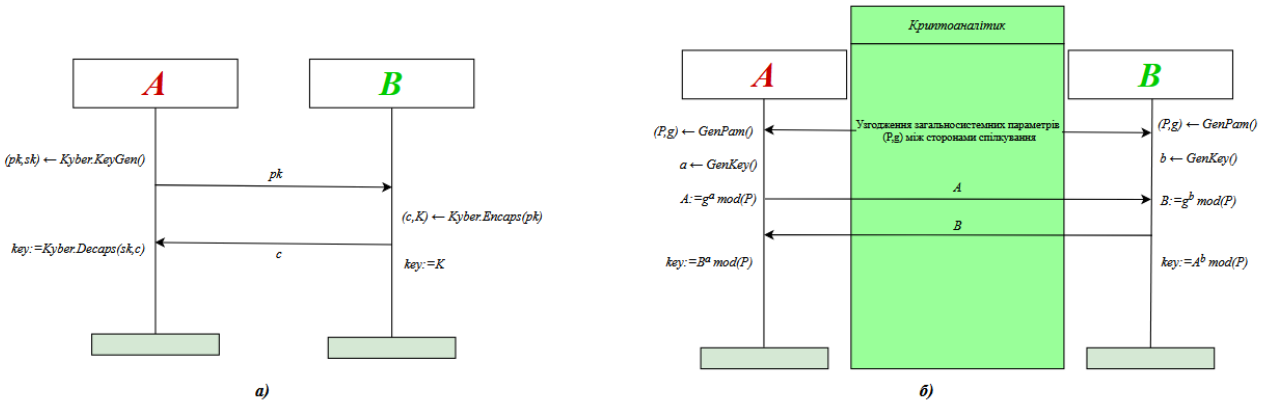


Рисунок 1.3 – Моделі безпечного обміну ключами KEM (а), Діффі-Хеллмана (б)

Інтернет надає безмежний спектр можливостей, але дотичний з серйозними загрозами. Прогресивними техніками інформаційної безпеки є протоколи обміну ключами, які дозволяють встановлювати безпечний комунікаційний ланцюжок між мережевими вузлами. Одним з розповсюджених представників цієї категорії є схема Діффі-Хеллмана, для якої рекомендується використовувати еліптичні криві для підвищення безпеки. KEM використовує асиметричне шифрування для встановлення загального секрету між сторонами за певною схемою. Він забезпечує гнучкість системи, оскільки не прив’язаний до конкретного математичного методу [7, 9]. NIST рекомендує [7, 10] впроваджувати майбутні стандарти обміну ключами саме за допомогою KEM. Така криптографічна конструкція включає три основні підалгоритми [10]: генерацію ключів (*KeyGen*), інкапсуляцію (*Encaps*) і декапсуляцію (*Decaps*). *Encaps* застосовує відкритий ключ (*pk*) для створення шифртексту (*c*) та загального секрету (*s*) [10]. *Decaps*, у свою чергу, використовує секретний ключ (*sk*) та зашифрований текст (*c*) для обчислення відповідного загального секрету (*s*) [10]. Розглянута схема KEM на рис. 1.3 (а) дозволяє реалізувати захист від атак пасивних сценаріїв, але вона є вразливою до атак [7, 8] типу «людина посередині». В парадигмі KEM щодо Kyber виділяють класифікацію [7] протоколів обміну ключами з автентифікацією. Дані схеми створені для забезпечення надійності системи захисту на тлі потенційних загроз.

Розглянемо специфікацію протоколів з експлуатацією алгоритму Kyber [7]: односторонній протокол обміну ключами з автентифікацією (unilateral authenticated key exchange protocol, UAKE) та безпосередньо протокол обміну ключами з автентифікацією (authenticated key exchange protocol, AKE). Нехай визначено [7] функцію $H:\{0,1\}^{*}\to\{0,1\}^{256}$ – геш-функція. Основна відмінність між UAKE та AKE полягає у степені обізнаності сторін сеансу спілкування. В UAKE, лише сторона A володіє інформацією про статичні ключі сторони B [7], а в AKE навпаки обидві сторони знають статичні ключі один одного. Особливість таких конструкцій лежить в формулі породження спільного ключа. Інакше кажучи, значення генерованого спільного ключа s визначається тимчасовими та статичними ключами (pk_i) та криптограмами (c_i) [7]. Принцип роботи схем UAKE та AKE представлено на рис. 1.4, 1.5. З аналізу структури наведених схем можна сказати, що обидві моделі протоколів обміну ключами (UAKE, AKE) є вдосконаленими варіантами схеми KEM (рис 1.3, а). Експлуатація механізму гешування гарантує унікальність виробленого ключа певної довжини, який можна застосовувати для проведення обміну інформацією.

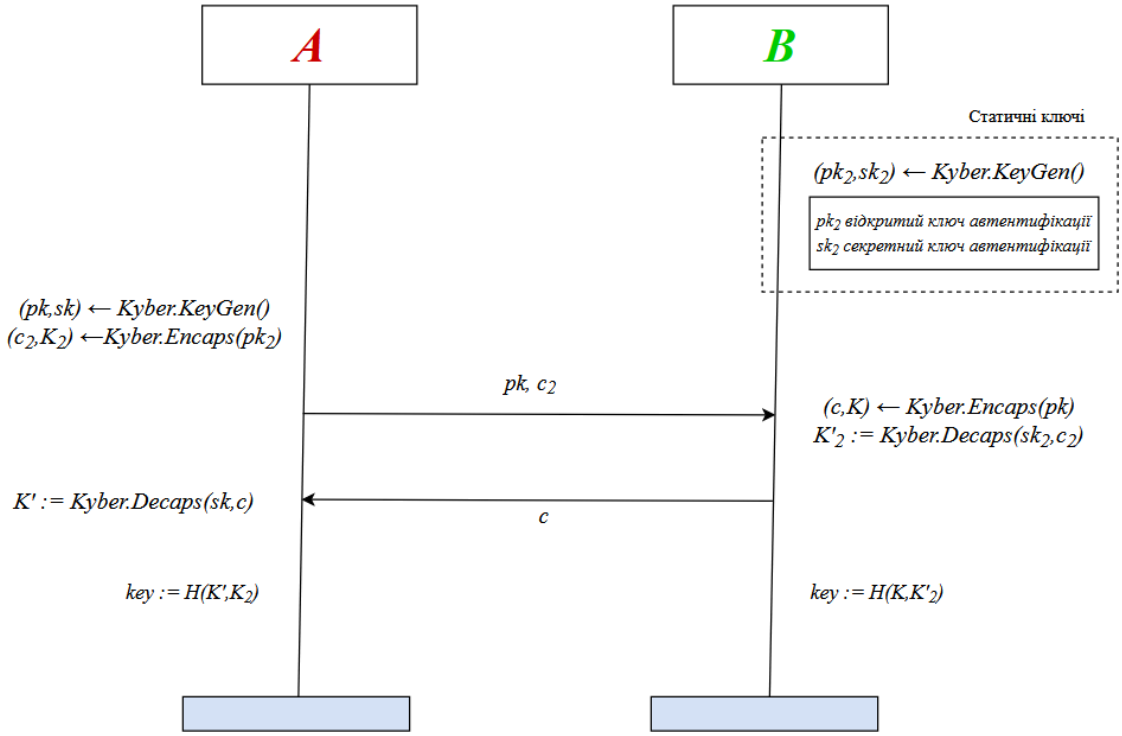


Рисунок 1.4 – Схема Kyber.UAKE

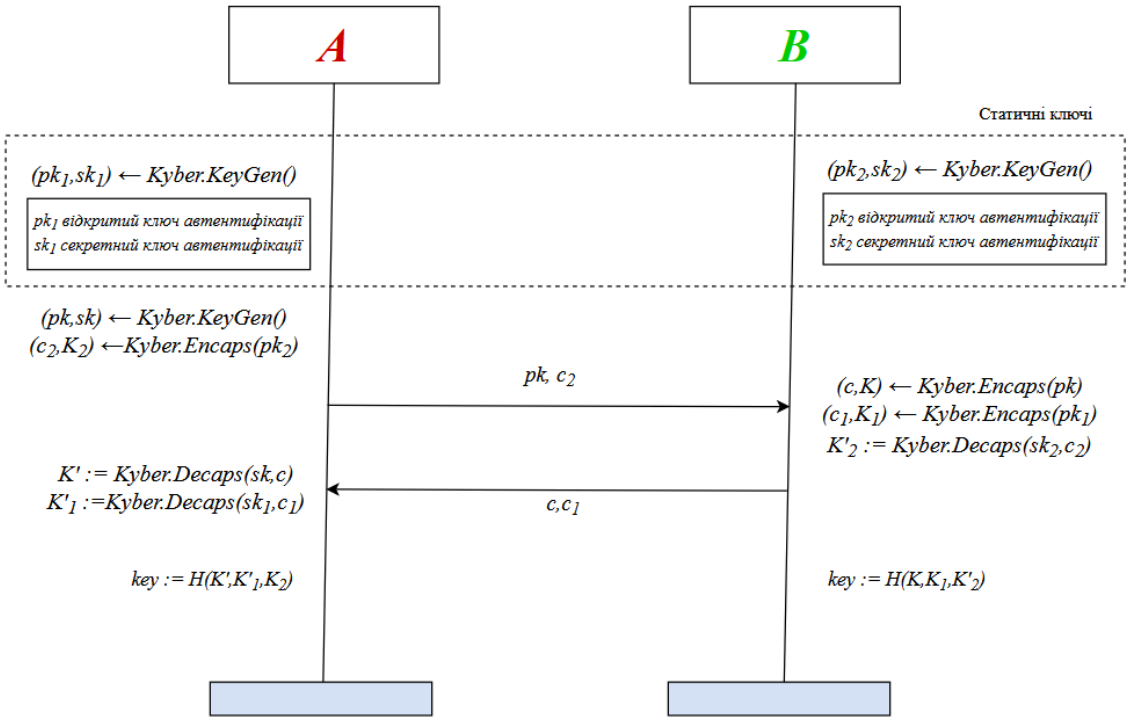


Рисунок 1.5 – Схема Kyber.AKE

1.3 Проблематика модульного навчанням з помилками

Основна проблематика, на якій побудована ідея алгоритму Kyber – MLWE. Абстрагуємо дану задачу до вигляду звичайної системи лінійних рівнянь [11]. Маємо наступну систему лінійних рівнянь:

$$\begin{cases} a_{11}s_1 + a_{12}s_2 + a_{13}s_3 + a_{14}s_4 = b_1 \\ a_{21}s_1 + a_{22}s_2 + a_{23}s_3 + a_{24}s_4 = b_2 \\ a_{31}s_1 + a_{32}s_2 + a_{33}s_3 + a_{34}s_4 = b_3 \\ a_{41}s_1 + a_{42}s_2 + a_{43}s_3 + a_{44}s_4 = b_4 \end{cases}, \quad (1.1)$$

де $A = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix}$ – квадратна матриця 4x4 коефіцієнтів лінійної системи рівнянь та компонента відкритого ключа;

$s = \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix}$ – вектор секретного ключа;

$b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix}$ – вектор відкритого ключа.

З рівняння (1.1) можемо виділити формулу для розрахунку відкритого ключа:

$$b = As$$

$$b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \times \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix} = \begin{pmatrix} a_{11}s_1 + a_{12}s_2 + a_{13}s_3 + a_{14}s_4 \\ a_{21}s_1 + a_{22}s_2 + a_{23}s_3 + a_{24}s_4 \\ a_{31}s_1 + a_{32}s_2 + a_{33}s_3 + a_{34}s_4 \\ a_{41}s_1 + a_{42}s_2 + a_{43}s_3 + a_{44}s_4 \end{pmatrix}. \quad (1.2)$$

Для вирішення системи рівнянь (1.1) можна використовувати матричний спосіб. Найпоширенішими для цього є методи Гауса та оберненої матриці. Наведені техніки розповсюджені в експлуатації при чисельних розрахунках. Сучасні обчислювальні потужності процесорів дозволяють легко знайти *s*, використовуючи методи математичних операцій. Модифікуємо систему рівнянь (1.1), додавши вектор помилки *e*. При цьому система рівнянь (1.1) набуде наступної конфігурації:

$$\begin{cases} a_{11}s_1 + a_{12}s_2 + a_{13}s_3 + a_{14}s_4 + e_1 = b_1 \\ a_{21}s_1 + a_{22}s_2 + a_{23}s_3 + a_{24}s_4 + e_2 = b_2 \\ a_{31}s_1 + a_{32}s_2 + a_{33}s_3 + a_{34}s_4 + e_3 = b_3 \\ a_{41}s_1 + a_{42}s_2 + a_{43}s_3 + a_{44}s_4 + e_4 = b_4 \end{cases}, \quad (1.3)$$

де $e = \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{pmatrix}$ – вектор помилок, з невеликими цілими числами.

Можливий варіант збільшення кількості рівнянь [11] в (1.3) для підвищення складності обчислювальної задачі. При цьому для обчислювальної системи – це стане складним завданням, оскільки значення вектору *s*, повинно задовольняти співвідношенню:

$$b = As + e$$

$$b = \begin{pmatrix} b_1 \\ b_2 \\ b_3 \\ b_4 \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{pmatrix} \times \begin{pmatrix} s_1 \\ s_2 \\ s_3 \\ s_4 \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ e_3 \\ e_4 \end{pmatrix} = \begin{pmatrix} a_{11}s_1 + a_{12}s_2 + a_{13}s_3 + a_{14}s_4 + e_1 \\ a_{21}s_1 + a_{22}s_2 + a_{23}s_3 + a_{24}s_4 + e_2 \\ a_{31}s_1 + a_{32}s_2 + a_{33}s_3 + a_{34}s_4 + e_3 \\ a_{41}s_1 + a_{42}s_2 + a_{43}s_3 + a_{44}s_4 + e_4 \end{pmatrix}. \quad (1.4)$$

Знаходження *s* зі співвідношення (1.4) буде суттєво важким завданням для програмно-апаратних комплексів. Ускладнюється процес вилучення вектору *s* [11] із суміші *b* за рахунок супроводжуючих модульних обчислень. Комп'ютеру необхідно перебрати всі можливі комбінації значень вектору $s = (s_1, s_2, \ldots, s_n)$ з урахуванням вектору помилок $e = (e_1, e_2, \ldots, e_n)$, використовуючи модульну арифметику. Додавання вектору похибки *e* ускладнює експлуатацію матричних схем вирішенням систем рівнянь, оскільки ці методики не передбачають додаткових значень. Дана ситуація створює додатковий шар складності та може призвести до непередбачуваних наслідків. Основна ідея завдання MLWE може бути передана через вираз:

$$b = As + e \pmod q$$

$$b = \begin{pmatrix} b_1 \\ b_2 \\ \cdots \\ b_n \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & \cdots & a_{1n} \\ a_{21} & a_{22} & \cdots & a_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ a_{n1} & a_{n2} & \cdots & a_{nn} \end{pmatrix} \times \begin{pmatrix} s_1 \\ s_2 \\ \cdots \\ s_n \end{pmatrix} + \begin{pmatrix} e_1 \\ e_2 \\ \cdots \\ e_n \end{pmatrix} \pmod q = \begin{pmatrix} a_{11}s_1 + a_{12}s_2 + \cdots + a_{1n}s_n + e_1 \pmod q \\ a_{21}s_1 + a_{22}s_2 + \cdots + a_{2n}s_n + e_2 \pmod q \\ \cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots\cdots \\ a_{n1}s_1 + a_{n2}s_2 + \cdots + a_{nn}s_n + e_n \pmod q \end{pmatrix}, \quad (1.5)$$

вважається, що навіть апаратам з квантовими процесорами складно виділити вектор секретного ключа *s* зі співвідношення (1.5). Таким чином, розробляються алгоритми генерації ключової пари в криптографічних системах постквантового періоду з урахуванням задачі MLWE.

1.4 Принцип роботи алгоритму Crystals-Kyber

В алгоритмі Kyber використовується обчислення над поліномами [9-11] в кільці поліномів $Z_p[X]/(X^n + 1)$. Базисний поліном кільця $X^n + 1$, який використовується [5-7] в шифрі Kyber – $x^{256} + 1$. Ключові пари відкритий/секретний ключ представляються у вигляді векторів, компоненти, яких є поліноми. Основні поліноміальні форми ключових атрибутів методу Kyber [11]:

- $A(x)$ – поліноміальна конструкція матричного виду, до складу якої входять поліноми з поліноміального кільця $Z_p[X]/(X^n+1)$, де всі коефіцієнти з Z_p;

- $e(x)$ та $s(x)$ – поліноми з поліноміального кільця $Z_p[X]/(X^n+1)$, з невеликими коефіцієнтами;

- $b(x)=A(x)s(x)+e(x)$ – співвідношення, яке визначає відкритий ключ, що є також математичною конструкцією поліноміального виду з кільця поліномів $Z_p[X]/(X^n+1)$, де всі коефіцієнти з Z_p.

Розглянемо на прикладі з малими числами принцип роботи алгоритму асиметричного шифрування Kyber. Використаємо наступні значення загальносистемних параметрів k=2, q=17 та n=4. Першим базовим етапом в будь-якому алгоритмі шифрування або генерації підпису є породження ключової пари. Структура та розмір ключів є важливими атрибутами стійкості криптографічної системи. Секретний ключ s в схемі Kyber є вектором з k поліномів, кожен з яких може мати n членів. Поліноміальна форма складається з суми членів, які є добутком константи та змінної, зведеної в деякий ступінь, що не перевищує n. Коефіцієнтами поліномів секретного ключа [9-11] є випадкові невеликі числа. Нехай секретний ключ має наступну конфігурацію:

$$s=(-x^3-x^2+x,-x^3-x).$$

Відкритий ключ складається з двох складових: квадратної матриці поліномів A розміром nxn та поліноміального вектору t. Поліноми з матриці A мають коефіцієнти [8], які не перевищують модуль q. Вектор t може бути обчислений з виразу t=As+e, але для цього необхідно визначити вектор похибки e, який містить невеликі коефіцієнти. Для розглянутого прикладу визначимо вищевказані параметри для розрахунку компоненти відкритого ключа t.

$$A=\begin{pmatrix} 6x^3+16x^2+16x+11 & 5x^3+3x^2+10x+1 \\ 9x^3+4x^2+6x+3 & 6x^3+x^2+9x+15 \end{pmatrix},$$

$$e=(x^2,x^2-x) \qquad t=(16x^3+15x^2+7,10x^3+12x^2+11x+6).$$

Після виконання всіх необхідних криптографічних операцій маємо пару ключів відкритий/секретний ключ. Секретний ключ тримається власником у секреті в захищеному хмарному середовищі, а відкритий ключ (A, *t*) розповсюджується в загальний доступ іншим користувачам у мережі. В алгоритмі Kyber включені функції пакування [6], які перетворюють вектори поліномів у байтовий рядок. Основною метою таких перетворень є представлення інформації у зручному форматі для обробки, зберігання та передачі. Розглянемо процедуру зашифрування інформації в криптографічній схемі Kyber. Для цього візьмемо невелике повідомлення $m=11_{10}=1011_2$. Для проведення шифрування інформаційних бітів повідомлення *m*, потрібно привести його бінарне представлення до поліноміальної форми. Специфікація методу Kyber визначає фіксований розмір повідомлення [6, 7] на вході блоку шифрування, що складає 32 байти (256 бітів). Поліноміальне представлення включає 256 коефіцієнтів [6] кожен, з яких позначає біт інформаційного повідомлення. Після перетворення бітів інформації у поліном, необхідно перемножити [11] кожний його коефіцієнт на $\lfloor q/2 \rceil$, тобто ціле число найближче до $q/2$. В контексті розглянутого прикладу маємо наступну послідовність перетворень для конвертації повідомлення *m*:

$$\begin{aligned} m_b &= 1x^3 + 0x^2 + 1x^1 + 1x^0 = x^3 + x + 1 \\ m &= 9 \times m_b = 9x^3 + 9x + 9 \end{aligned}.$$

Для виконання подальших математичних перетворень необхідно 3 вектори *r*, e_1, e_2, елементами яких є поліноми з випадково-генерованими малими коефіцієнтами. Визначимо для вказаних векторів такий вміст:

$$\begin{aligned} r &= \left(-x^3 + x^2, x^3 + x^2 - 1\right) \\ e_1 &= \left(x^2 + x, x^2\right) \\ e_2 &= -x^2 - x \end{aligned}.$$

Процес шифрування інформації здійснюється за допомогою відкритого ключа (*A*, *t*). В процедурі зашифрування [11] обчислюються два значення *u* та *v*:

$$\begin{aligned} u &= A^T r + e_1 \\ v &= t^T r + e_2 + m \end{aligned}. \qquad (1.6)$$

У випадку представленого прикладу маємо наступні значення елементів u та v:

$$u = \left(11x^3 + 11x^2 + 10x + 3.4x^3 + 4x^2 + 13x + 11\right)$$
$$v = \left(7x^3 + 6x^2 + 8x + 15\right)$$

Розглянемо процес розшифрування – вилучення конфіденційної інформації з криптограми (u, v). Проведення цієї процедури вимагає наявність секретного ключа s. На вхід модуля розшифрування надходять криптограма (u, v) та секретний ключ s. Для отримання вихідного секрету необхідно виконати наступні перетворення:

$$m_n = v - s^T u \,. \tag{1.7}$$

Однак вилучений секрет представлений в шумовому вигляді. Повідомлення, яке містить частинки шуму, поєднані з вихідним секретом, характеризується співвідношенням:

$$m_n = e^T r + e_2 + m + s^T e_1 \,. \tag{1.8}$$

У контексті нашого прикладу маємо наступну структуру інформаційного контенту:

$$m_n = 7x^3 + 14x^2 + 7x + 5 \,.$$

Щоб позбутися шумового ефекту у вилученому повідомленні, потрібно зіставити кожний коефіцієнт шумового варіанту повідомлення з найближчим цілим до $q/2$. В проаналізованому прикладі необхідно перевірити, до якої ближче числової границі лежить кожний коефіцієнт [11] чи ближче до $\lfloor q/2 \rceil$=9, чи до 0 (або q). Порівняємо окремо кожний коефіцієнт m_n за визначеним критерієм:

- 7 розташовано ближче до 9, ніж до 0 або q, тому приводимо коефіцієнт до 9;

- 14 розташовано ближче до q, ніж до 9, тому присвоюємо даному коефіцієнту значення 0, що тотожно q;

- 7 розташовано ближче до 9, ніж до 0 або q, тому приводимо коефіцієнт до 9;

- 5 розташовано ближче до 9, ніж до 0 або q, тому приводимо коефіцієнт до 9.

Після ліквідації шумової складової, отримуємо масштабовану версію нашого вихідного повідомлення. Щоб отримати первісну версію вихідного повідомлення,

потрібно кожний член поліноміального представлення масштабованого повідомлення помножити на $\frac{1}{9}$.

$$m_b = \frac{1}{9}\left(9x^3 + 0x^2 + 9x + 9\right) = x^3 + 0x^2 + x + 1.$$

Після виконання математичних операцій отримуємо початковий варіант інформаційного повідомлення в поліноміальній формі. З отриманого полінома можна виділити секрет, виписавши всі бінарні коефіцієнти зліва направо. Конвертуємо отриману послідовність в десяткову форму $(1011)_2=11_{10}$. Отриманий результат повністю співпадає з вихідним повідомлення, що вказує на ефективність роботи алгоритму. При цьому сторони спілкування мають спільний секрет, який вони можуть застосовувати для проведення безпечного сеансу зв'язку.

1.5 Дослідження безпеки та швидкодії Crystals-Kyber

Безпека алгоритму асиметричного шифрування Crystals-Kyber визначається набором загальносистемних параметрів [7]. Значення параметрів алгоритму (табл. 1.1) визначається його криптографічною стійкістю [6, 7] згідно з моделлю рівнів безпеки NIST.

Таблиця 1.1 – Характеристика параметрів для криптоалгоритму Kyber

Позначення	Сутність	Значення
λ	Рівень криптографічної стійкості	1, 3, 5
n	Степінь полінома x^n+1	256
q	Модуль	3329
k	Розмірність вектору, елементи якого є поліном(и)	2 для λ=1 3 для λ=3 4 для λ=5
η_1	Встановлює межі для коефіцієнтів поліномів s, e	3 для λ=1 2 для λ=3 2 для λ=5
η_2	Встановлює межі для коефіцієнтів поліномів e_1, e_2	2 для λ=1 2 для λ=3

Продовження таблиці 1.1

		2 для λ=5
(d_u, d_v)	d_u – позначає кількість бітів, які необхідні для кодування коефіцієнту полінома при перетворенні вектору поліномів у рядок байтів d_v – позначає кількість бітів, які необхідні для кодування елементу при перетворенні поліному в рядок байтів	10, 4 для λ=1 10, 4 для λ=3 11, 5 для λ=5
m_len	Довжина інформаційного повідомлення, що шифрується	32

Для кожного набора параметрів визначено два варіанти алгоритму Kyber [6]:

– алгоритм за замовчуванням;

– алгоритм, що позначається як KYBER_90S.

Відмінність між цими двома варіантами алгоритму Kyber [6] полягає в експлуатації криптографічних функцій та процедур (табл. 1.2).

Таблиця 1.2 – Відмінності між алгоритмом Kyber за замовчуванням та KYBER_90

Вид алгоритму	Гешування	Потокове шифрування
За замовченням	H – SHA3-256 G – SHA3-512	XOF – SHAKE-128 KDF – SHAKE-256 PRF – SHAKE-256
KYBER_90S	H – SHA-256 G – SHA-512	XOF – AES-256 в режимі CTR KDF – SHAKE-256 PRF – AES-256 в режимі CTR

Розмірність геш-значень для функцій гешування складають [6]: H_len=32, G_len=64 байтів. Результати експериментальних досліджень показали меншу обчислювальну складність алгоритму [6] за замовчуванням порівняно з KYBER_90, тому надалі планується застосування цього режиму. Ключова пара, яка управляє криптографічними процедурами алгоритму Kyber породжується на основі змісту загальносистемних параметрів (табл. 1.1). Фундаментальним атрибутом стійкості є довжина ключів, тому, чим більший рівень безпеки алгоритму, тим надійніша система захисту. На вибір алгоритму шифрування впливають ряд факторів: особливості алгоритму, складність математичної задачі, на якій заснована суть даного методу, довжина ключової пари, швидкодія і т. д. При цьому важливо враховувати обчислювальну потужність та особливості системи, для якої планується створити захисний механізм. Порівняння розмірів ключової пари (*pk* – відкритий ключ, *sk* –

секретний ключ), шифртекстів *ct* для алгоритму Kyber та інших KEM, та альтернативних постквантових механізмів наведено у табл. 1.3.

Таблиця 1.3 – Порівняння Kyber з альтернативними методами постквантової криптографії

Алгоритм	Ідентифікатор	Рівень безпеки	Розмір *pk*, байти	Розмір *sk*, байти	Розмір *ct*, байти
Kyber	Kyber512(-90 s)	1	800	1632	768
	Kyber768(-90 s)	3	1184	2400	1088
	Kyber1024(-90 s)	5	1568	3168	1568
Скеля	Скеля-3-192	3	1432	152	1464
	Скеля-5-256	5	2102	200	2166
Classic McEliece	Classic_McEliece-348864(f)	1	261120	6492	128
	Classic_McEliece-460896(f)	3	524160	13608	188
	Classic_McEliece-6688128(f)	5	1044992	13932	240
	Classic_McEliece-6960119(f)	5	1047319	13948	226
	Classic_McEliece-8192128(f)	5	1357824	14120	240
NTRU	NTRU-HPS-2048-509	1	699	935	699
	NTRU-HPS-2048-677	3	930	1234	930
	NTRU-HRSS-701	3	1138	1452	1138
	NTRU-HPS-4096-821	5	1230	1592	1230
	NTRU-4096-1229	5	1842	2366	1842
	NTRU-HRSS-1373	5	2401	2983	2401
SABER	LightSaber	1	672	1568	736
	Saber	3	992	2304	1088
	FireSaber	5	1312	3040	1472
BIKE	BIKE-L1	1	1541	5223	1573
	BIKE-L3	3	3083	10105	3115
	BIKE-L5	5	5122	16494	5154
FrodoKEM	FrodoKEM-640	1	9616	19888	9720
	FrodoKEM-976	3	15632	31296	15744
	FrodoKEM-1344	5	21520	43088	21632
HQC	HQC-128	1	2249	2289	4481
	HQC-192	3	4522	4562	9026
	HQC-256	5	7245	7285	14469
NTRU PRIME	sntrup653	1	994	1518	897
	sntrup857	3	1322	1999	1184
	sntrup1277	5	2067	3059	1847
	ntrulpr653	1	897	1125	1025
	ntrulpr857	3	1184	1463	1312
	ntrulpr1277	5	1847	2231	1975

З гістограм (рис. 1.6-1.8) спостерігається відставання різновидів Kyber від багатьох досліджуваних криптосистем постквантового періоду. Рекордсменом по довжині відкритих ключів став алгоритм Classic McEliece. Найвищі результати з породження секретних ключів та криптограм продемонстрував алгоритм FrodoKEM. Зі зростанням рівня безпеки, збільшується розмірність ключової пари та інкапсульованих даних. Найнижчі результати по генерації відкритого ключа показав LighSaber. Порівняльний аналіз показав, що секретний ключ, згенерований за алгоритмом «Скеля» [6] значно менший у порівнянні з Kyber, а також є найнижчим відносно інших алгоритмів. Це може мати негативний вплив на стійкість алгоритму проти потенційних атак, бо об'ємний розмір секретного ключа вважається надійнішим у протидії кібератак, направлених на компрометацію ключів. В свою чергу, Classic McEliece програє іншим алгоритмам по довжині шифртекстів.

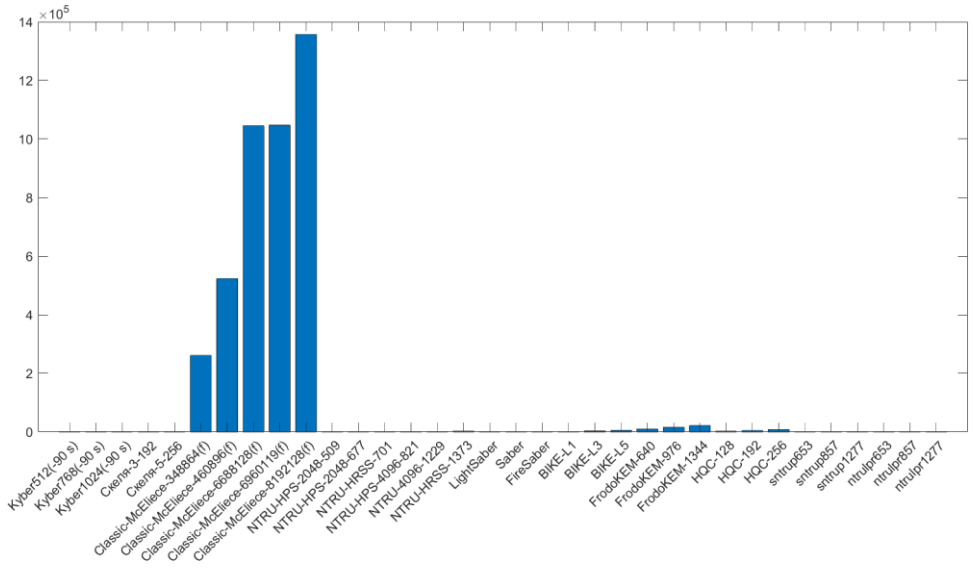


Рисунок 1.6 – Гістограма довжини відкритих ключів для постквантових алгоритмів

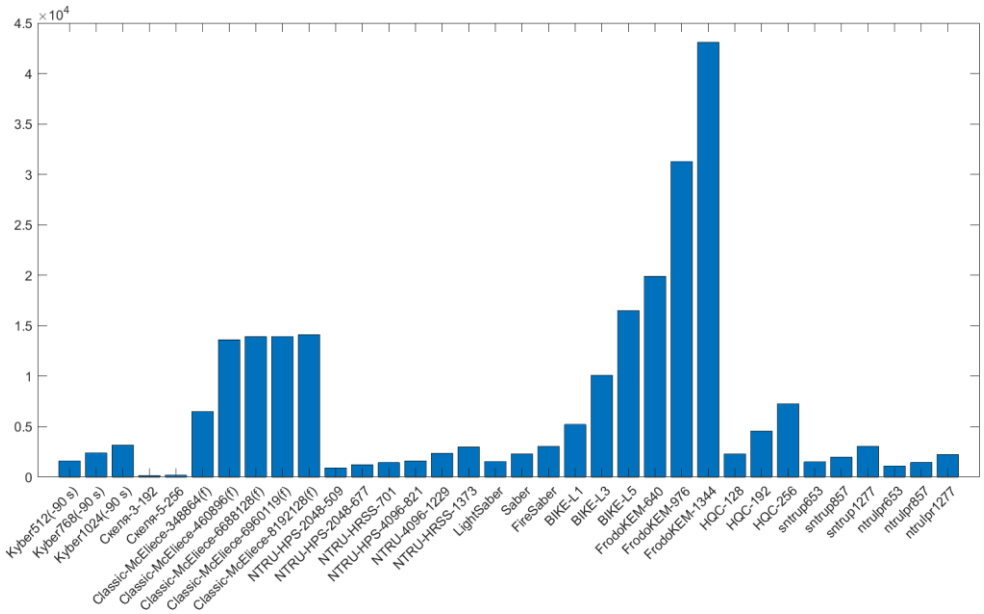


Рисунок 1.7 – Гістограма довжини секретних ключів для постквантових алгоритмів

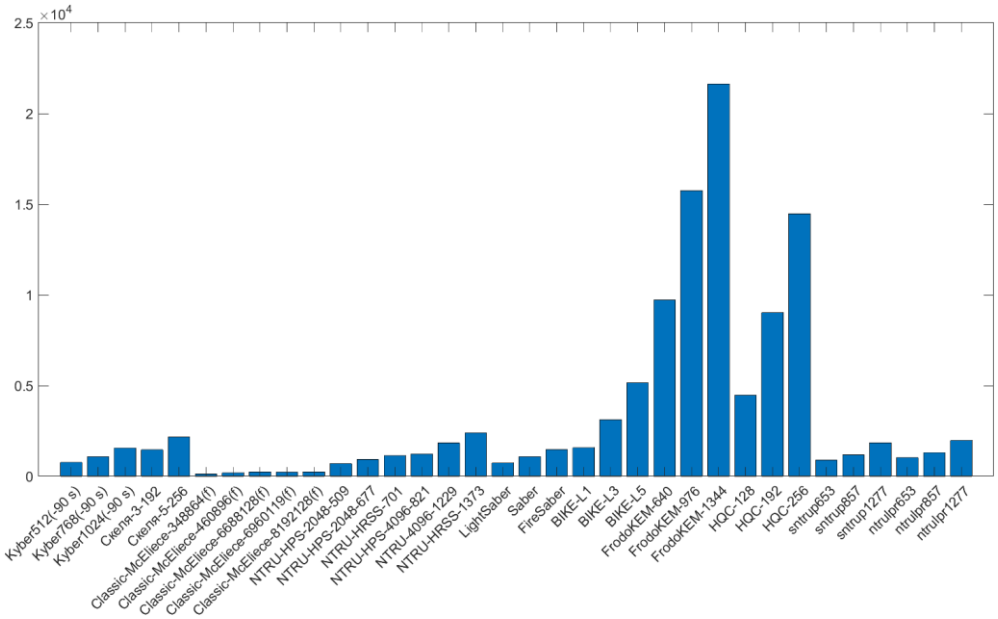


Рисунок 1.8 – Гістограма шифртекстів, вироблених постквантовими примітивами

Проведемо оцінку ключової пари та шифртекстів для алгоритму Kyber та традиційних алгоритмів асиметричного шифрування RSA та ECC (Elliptic Curve Cryptography). У цьому випадку, метод Kyber має перевагу, оскільки належить до оновленого покоління шифрів та включає складні математичні перетворення. Наразі криптографічні вектори асиметричного шифрування, які популяризовані в експлуатації в мережевих системах та протоколах є RSA та ECC. Оскільки постквантові примітиви перебувають в процесі інтеграції в парадигму інформаційних

систем, що супроводжується адаптацією цифрових механізмів до модернізованих стандартів шифрування. У табл. 1.4 наведено порівняння розмірів ключової пари (відкритий/секретний ключ) та шифртекстів для алгоритму Kyber та стандартизованих методів криптографії з відкритим ключем (RSA, ECC). Вказані проблеми, на яких засновані порівняні криптографічні методи та рівні безпеки. У сукупності кількість рівнів безпеки складає 3 рівні AES, кожен з яких визначає набір параметрів та особливості моделей стійкості кожного варіанту алгоритму.

Таблиця 1.4 – Порівняння Kyber з RSA та ECC

Версія	Рівень безпеки	Проблема, що лежить в основі	Розмір відкритого ключа, байти	Розмір секретного ключа, байти	Розмір шифрованого тексту, байти
Kyber512	AES128	MLWE	800	1632	768
Kyber768	AES192		1184	2400	1088
Kyber1024	AES256		1568	3168	1568
RSA3072	AES128	проблема факторизації чисел	384	384	384
RSA15360	AES256		1920	1920	1920
ECC256	AES128	проблема дискретного логарифмування в групі точок ЕК	32	32	387
ECC521	AES256		64	64	525

Алгоритм Kyber переважає за розмірами ключової пари та довжині шифртексту відносно інших досліджуваних стандартизованих криптосистем в табл. 1.4. Розглянемо гістограму на рис. 1.9, яка зображає результати досліджень розмірності ключів та криптограм. Класичні криптосистеми мають значно менші значення ключів та шифртекстів за розміри асиметричних ключів та вироблених шифртекстів для алгоритму Kyber. На фоні попередніх досліджень Kyber програє за розміром ключових атрибутів та шифртексту більшості алгоритмам KEM. Однак, навіть при суттєво низьких показниках криптографічних параметрів може забезпечити високий рівень стійкості. З огляду на рівень безпеки, RSA15360 має більші розміри за певними досліджуваними параметрами за розміри для варіантів алгоритму Kyber. Потрібно зробити акцент на необхідності наявності об’ємних блоків пам’яті в комп’ютерних системах для організації зберігання породжених асиметричних ключів та шифрованих

текстів. Класичні криптосистеми можуть показати хороші результати щодо генерації криптографічних атрибутів, але є вразливими до атак постквантового криптоаналізу. Тому при виборі криптографічної техніки важливо враховувати фактор розвитку методів криптоаналізу для коректного підходу щодо вибору криптографічної системи захисту інформації. Еволюція комп'ютерних систем спонукає до переходу до модернізованих стандартів шифрування.

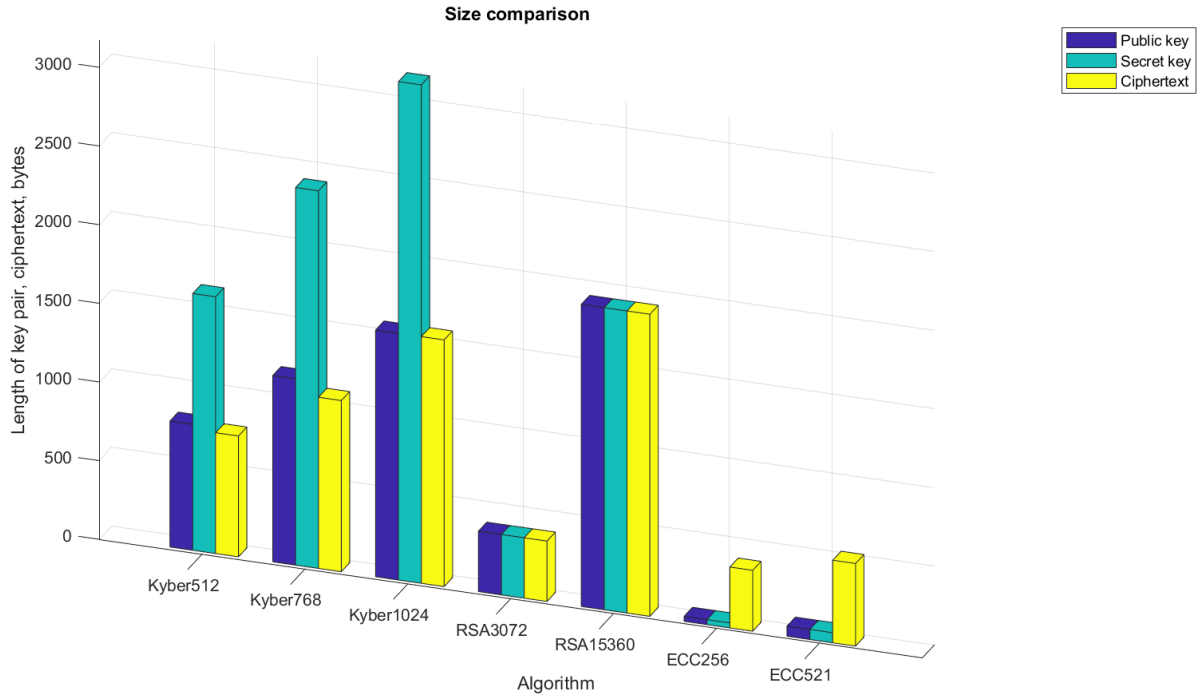


Рисунок 1.9 – Гістограма розмірів ключів та шифртекстів для Kyber, RSA та ECC при різних рівнях безпеки

Важливим фактором при виборі алгоритму є продуктивність криптографічного алгоритму. Оцінка швидкодії постквантової системи Kyber здійснювалася на базі існуючої програмної реалізації алгоритму Kyber [12]. Програмне забезпечення реалізовано на мові програмування C та дозволяє протестувати всі програмні модулі постквантової системи KEM Crystals-Kyber. Результати оцінки продуктивності наведені в табл. 1.5, для досліджень використовувалося ядро процесора Intel Core i7-4770K (Haswell) [10]. Результати досліджень наведені в машинних циклах. Позначення «ref» позначає універсальну версію реалізації методу Kyber, «AVX2» вказує на застосування векторних команд AVX2.

Таблиця 1.5 – Оцінка продуктивності алгоритму Kyber

Версія алгоритму	Криптографічні параметри	Розміри, байти	Процедура	Haswell цикли (ref)	Haswell цикли (AVX2)
Kyber512	відкритий ключ	800	зашифрування	154524	45200
	секретний ключ	1632	генерація	122684	33856
	шифртекст	768	розшифрування	187960	34572
Kyber512-90s	відкритий ключ	800	зашифрування	249084	28592
	секретний ключ	1632	генерація	213156	21880
	шифртекст	768	розшифрування	277612	20980
Kyber768	відкритий ключ	1184	зашифрування	235260	67624
	секретний ключ	2400	генерація	199408	52732
	шифртекст	1088	розшифрування	274900	53156
Kyber768-90s	відкритий ключ	1184	зашифрування	432764	40140
	секретний ключ	2400	генерація	389760	30460
	шифртекст	1088	розшифрування	473984	30108
Kyber1024	відкритий ключ	1568	зашифрування	346648	97324
	секретний ключ	3168	генерація	307148	73544
	шифртекст	1568	розшифрування	396584	79128
Kyber1024-90s	відкритий ключ	1568	зашифрування	672644	56556
	секретний ключ	3168	генерація	636380	43212
	шифртекст	1568	розшифрування	724144	44328

На основі емпіричних даних побудуємо гістограми (рис. 1.10) та залежності (рис. 1.11) для криптографічних процедур. Досліджено продуктивність модулів зашифрування, розшифрування та генерації ключів для різних версій алгоритму асиметричного шифрування Kyber. Результати досліджень показали, що зі збільшенням розміру ключа, підвищується кількість машинних циклів процесору для виконання відповідних криптографічних процедур. Тобто затрачується більше часу та ресурсів процесору. Варіанти методу Kyber за замовчуванням та KYBER_90S суттєво різняться за показниками продуктивності. Для виконання алгоритмів та процедур другого варіанту Kyber витрачається більше часу у порівнянні зі стандартною версією. Емпіричні дані експериментальних досліджень вказують на меншу обчислювальну складність первісної версії методу Kyber, що пояснює причину його найчастішої експлуатації. Перша версія алгоритму Kyber є менш ресурсозатратною у порівнянні з альтернативним варіантом даного алгоритму.

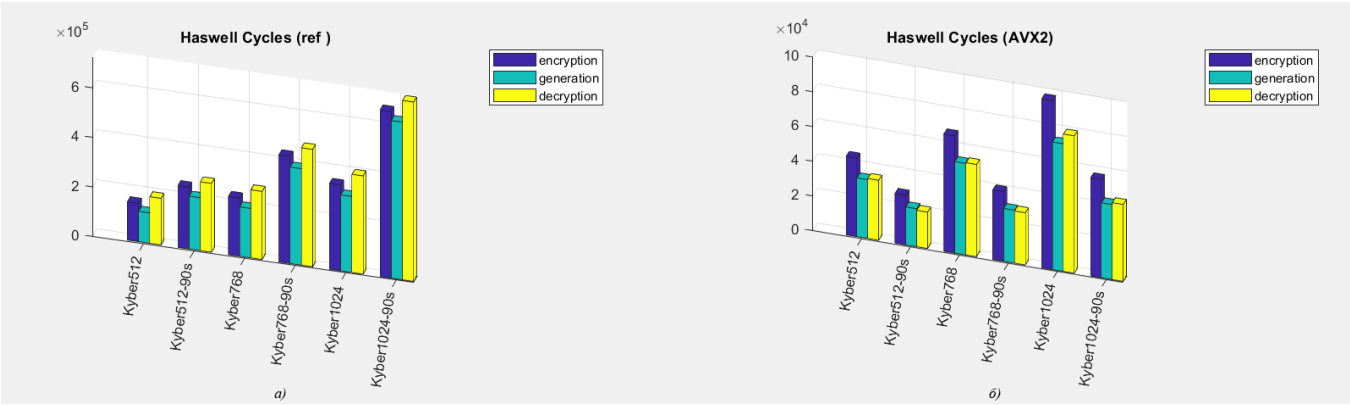


Рисунок 1.10 – Гістограма швидкодії криптографічних модулів Kyber у режимі ref (а) та AVX2 (б)

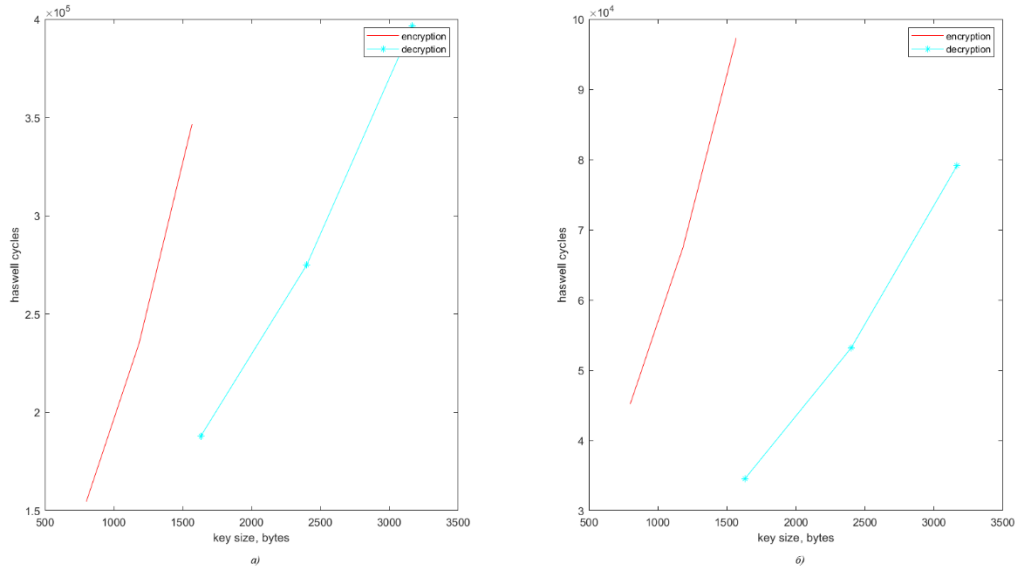


Рисунок 1.11 – Залежність продуктивності виконання процедур зашифрування та розшифрування від розміру ключів для режимів ref (а) та AVX2 (б)

2 ОЦІНКА МЕТОДІВ ПРИХОВУВАННЯ ДАНИХ В ЦИФРОВИХ ЗОБРАЖЕННЯХ ТА АУДІОЗАПИСАХ

2.1 Огляд специфікації методу LSB

Поширеною практикою стеганографічного приховування інформації є вбудовування інформаційних даних за рахунок маніпуляції над кольоровими характеристиками цифрових нерухомих зображень. Алгоритми стеганографічного приховування інформації варіюються за складністю перетворень, але мають одну схожість. Вона полягає в експлуатації природної збитковості [13, 14] зображення, яку можна використати для приховування інформаційного контенту. При цьому така модифікація не буде впливати на вихідну конфігурацію зображення. Найпростішим методом, який використовується в стеганографії є метод модифікації найменш значущого біту (НЗБ) [13-15]. Метод НЗБ [2, 15] заснований на 1-й низькорівневій властивості зорової системи людини (ЗСЛ) – низька чутливість до незначної зміни яскравості. Вбудовування здійснюється шляхом модифікації початкових бітів значень інтенсивностей, якими кодуються кольорові компоненти пікселів. У сукупності кожний піксель включає три основні кольорові компоненти (R – червона, G – зелена, B – синя) [2], які формують всю відому ієрархію кольорів. Загальна модель, яка відповідає за характеризування всіх можливих кольорів за допомогою цифрових значень рівнів яскравості трьох ключових кольорів (червоного, зеленого та синього) називається моделлю RGB. Принцип роботи алгоритму НЗБ наведено на рис. 2.1.

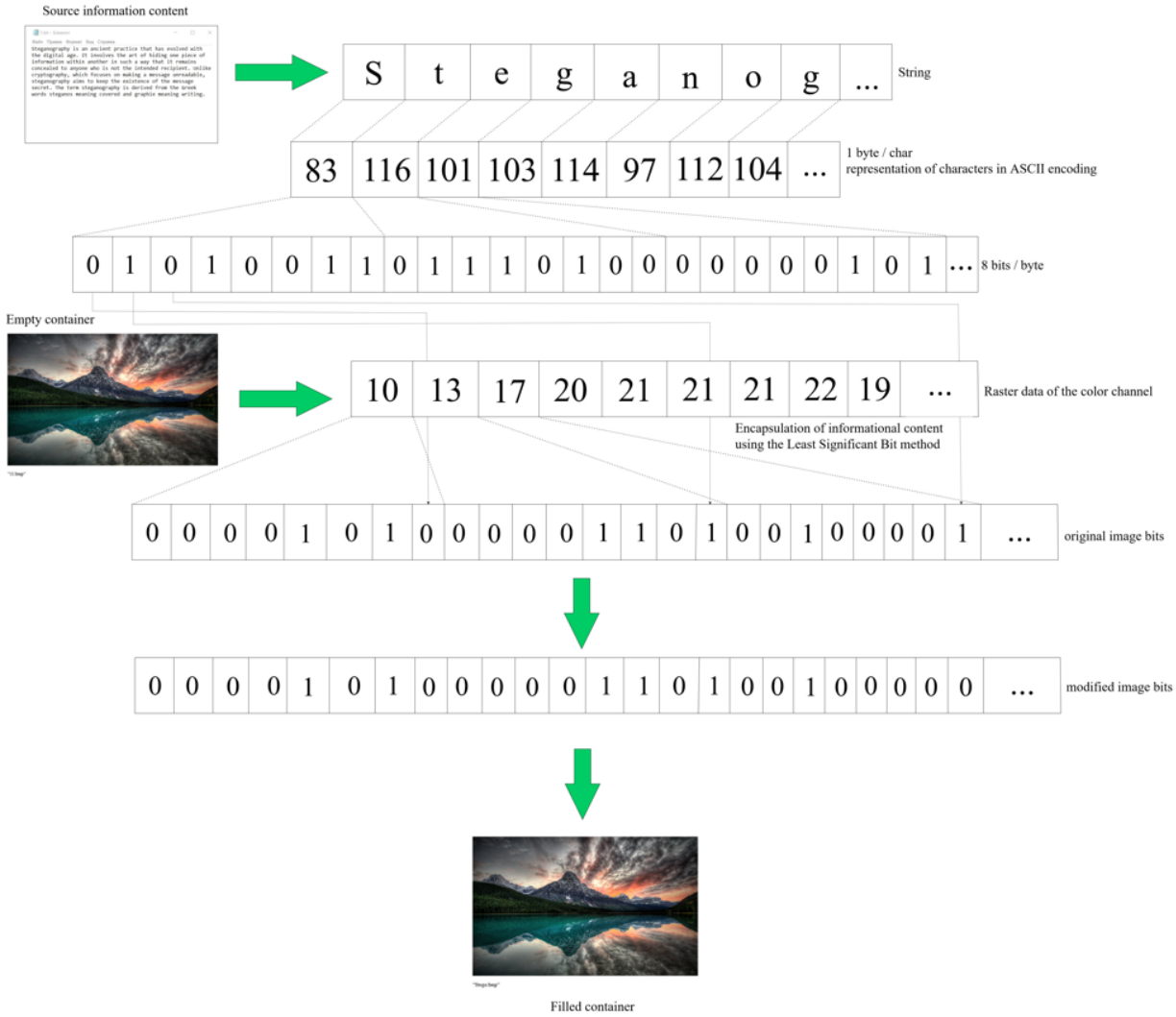


Рисунок 2.1 – Модель вбудування даних методом НЗБ

Степінь видимості спотворень після застосування методу НЗБ до цифрового контейнера оцінюється за допомогою порогу чутливості (ПЧ) ЗСЛ [13-15]. Якщо величина спотворень окремого пікселя не перевищує граничне значення ПЧ, тоді ЗСЛ не здатна буде виявити зміну властивостей модифікованого зображення [2]. Експериментальні дослідження показали [13-15], що ПЧ ЗСЛ до незначної модифікації рівнів яскравості растрових даних зображення становить 2–3%. Оцінимо ПЧ ЗСЛ до незначної модифікації яскравості [15] зображення, використовуючи графічну картинку формату *bmp24. Формат BMP (Bitmap Picture) визначає різні довжини бітів (1, 4, 8, 16, 24 і 32) для кодування кольорових відтінків [13-15] матриці пікселів. Модель формату BMP-24 визначає $2^{8=}256$ рівнів квантування яскравості для

окремого пікселя [13-15]. Оцінити ПЧ ЗСЛ до незначної зміни яскравості [13] кольорових каналів пікселів графічного зображення можна з виразу:

$$ПЧ = \frac{\Delta}{256} \cdot 100\%, \quad (2.1)$$

де Δ – величина спотворень (число рівнів квантування), які будуть внесені до пікселів.

Результати досліджень ПЧ при застосуванні алгоритму НЗБ подані на рис. 2.2.

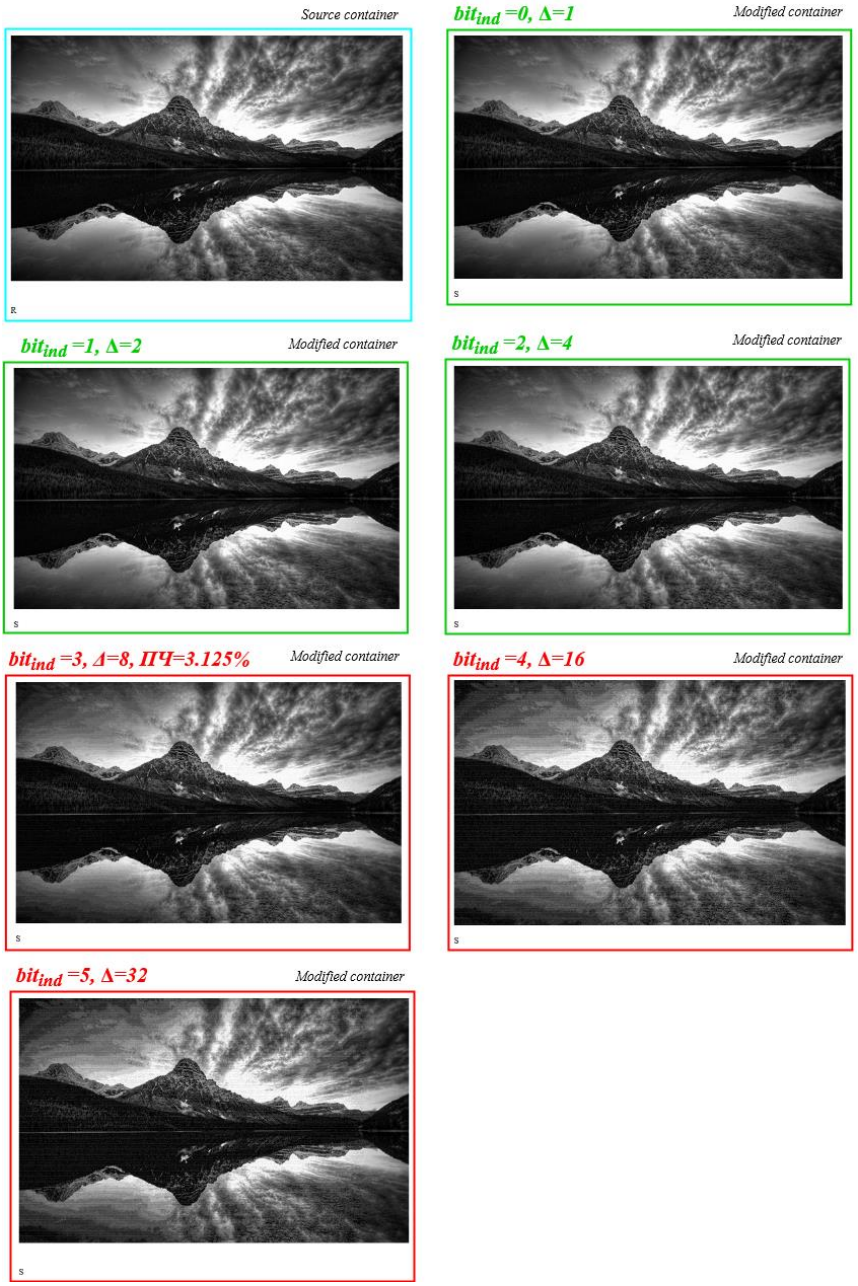


Рисунок 2.2 – Порогові можливості методу НЗБ

Досліджуване зображення представлено в градації сірого кольору. Розрахунки ПЧ з отриманих емпіричних даних відповідають теоретичних даним [13, 15]. З рис. 2.2 та залежності на рис. 2.3 спостерігається збільшення рівня освітленості зображення при послідовній експлуатації бітів від молодших до старших розрядів кольорових атрибутів пікселів зображення.

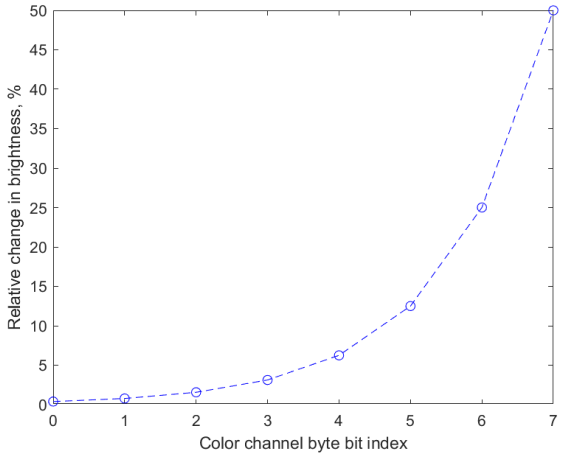


Рисунок 2.3 – Залежність варіації яскравості від розряду модифікованого біту

Метод LSB (Least Significant Bit) є ефективними алгоритмом для приховування масивного потоку даних в мультимедійних об’єктах. Даний алгоритм в спрощеному режимі дозволяє інкапсулювати інформаційний контент [2] в контейнер, мінімізуючи витрати ресурсів процесора. У лістингу 2.1 наведена реалізація модуля вбудування інформаційних повідомлень в графічні зображення.

Лістинг 2.1 – Метод інкапсуляції даних методом LSB

```
public int[][] embed(){
        int [][] S = new int[redChannel.length][redChannel[0].length];
        for (int i=0;i<redChannel.length;i++){
            for (int j=0;j<redChannel[i].length;j++){
                S[i][j] = redChannel[i][j];
            }
        }
        int i,j;
        for (int l=0;l<M_b.length;l++){
            i = (int)(Math.floor(l/redChannel[0].length));
            j = l - i* redChannel[0].length;
            int V[] = D_B(redChannel[i][j]);
            V[bit] = M_b[l];
            S[i][j] = B_D(V);
        }
        return S;
    }
```

Проведено тестування програмного модуля вбудування даних методом LSB, задля перевірки якості приховування [2, 16] інформаційного контенту. Для верифікації модуля використовувалися нерухомі зображення формату BMP-24 як контейнери. Пара зображень, які використовувалися для проведення експериментальних досліджень методу LSB, наведена на рис. 2.4. Первісне зображення 11.bmp є порожнім контейнером (рис 2.4, а), наступне зображення (2.4, б) – сформована стеганограма, яка містить біти інформаційного повідомлення.

а) *б)*

Рисунок 2.4 – Нерухоме зображення-контейнер (а) та стеганограма (б)

Видимих спотворень в заповненому контейнері немає, що вказує на ефективність функції вбудування. Візуально виявити відмінності між оригінальним та модифікованим зображенням неможливо. Розмір файлу-зображення становив 622 134 байтів, після інкапсуляції інформаційного контенту, розмір BMP-файлу залишився незмінним. При проведенні експериментів були взяті інші графічні контейнери 12.bmp та 13.bmp на рис. 2.5, 2.6, дані досліджень показали аналогічні результати.

а) *б)*

Рисунок 2.5 – Нерухоме зображення-контейнер (а) та стеганограма (б)

а) *б)*

Рисунок 2.6 – Нерухоме зображення-контейнер (а) та стеганограма (б)

Відсутність візуальних спотворень [16] вказує на коректність роботи програмного модуля. Програмний метод LSB показав хороші результати швидкодії (тестування проводилося на персональному комп'ютері [16], з операційною системою Windows 10 та тактовою частотою процесора 1.80 Ггц). Процедура інкапсуляції та вилучення інформаційних бітів методом LSB не потребує значних ресурсів та займає невеликі проміжки часу. Оцінка породженого зображення співпадає з оригінальним зображенням, що вказує, що рівень спотворень [15] лежить нижче ПЧ ЗСЛ до модифікації освітленості цифрового зображення. Дані досліджень по тестуванню функцій приховування даних методом LSB над графічними зображеннями наведені в табл. 2.1.

Таблиця 2.1 – Дані досліджень модуля LSB по BMP-зображенням

Назва зображення контейнеру, bmp	Розмір файлу до модифікації, байти	Розмір файлу після вбудування даних, байти	Розмір інформаційного контенту, що приховується, байти	Візуальна оцінка якості приховування даних	Час інкапсуляції та вилучення інформації, мс
11.bmp	622 134	622 134	470	Відмінна	11
12.bmp	584 694	584 694	470	Відмінна	10
13.bmp	622 134	622 134	470	Відмінна	7.5

Сучасні мультимедійні платформи мають програмні модулі, які здатні аналізувати об’єкт на наявність прихованого інформаційного контенту. Метод LSB є типовим методом [14] приховування інформації в файлових об’єктах. Популяризація методу НЗБ (LSB) пов’язана з простотою його реалізації [13-15], що дозволяє приховувати великі масиви даних у файлових об’єктах. Експлуатація даного методу в організації таємного стеганографічного каналу не рекомендується, в силу його вразливості [2, 15] до різноманітних атак (геометричні перетворення, маніпуляції над кольоровими властивостями зображення). Стандартний алгоритм НЗБ (LSB) дозволяє реалізувати відкриту (безключову) стеганосистему [17]. Безключова стеганографічна система не потребує ключів [14, 17] для інкапсуляції та декапсуляції інформаційних повідомлень. Приховування інформації потребує лише комплекс стеганографічних операцій. Підхід до організації відкритих стегосистем [17] підвищує ризик НСД до чутливої інформації. Тому модель безпеки в стеганографічних системах вимагає застосування стегоключів для захисту даних.

Досліджено стан інформаційних даних після приховування за допомогою методу НЗБ та реакцію даних на атаку, що заснована на алгоритмі стиснення JPEG (Joint Photographic Experts Group). На рис. 2.7 представлені контейнери-зображення (оригінальне зображення, стеганограма та стиснута стеганограма-зображення за допомогою алгоритму JPEG), разом із відповідними гістограмами.

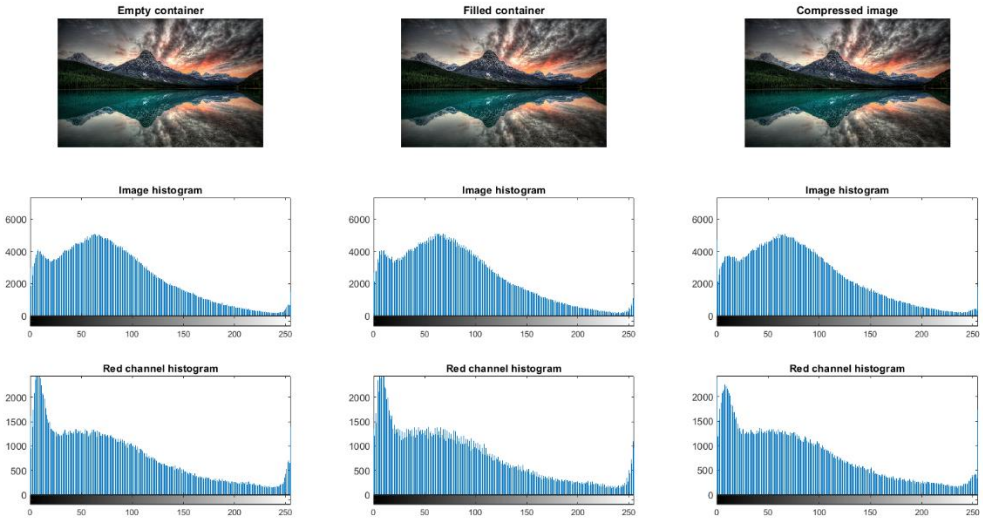


Рисунок 2.7 – Варіанти зображення-контейнеру з супровідними гістограмами

Важливим атрибутом стеганографічних систем є робастність [13, 14] – здатність зберігати цілісність інформації при впливі зовнішніх факторів (природні шуми, атаки зловмисника і т.д.). Проведено дослідження надійності стегаканалу, створеного за допомогою методу НЗБ, результати представлені в табл. 2.2.

Таблиця 2.2 – Емпіричні дані вилученої інформації

Представлення контейнеру	Ймовірність правильного вилучення даних	Помилка вилучених даних
Стеганограма	1	0
Стиснуте зображення	0.502	0.498

Емпіричні дані показують безпомилкове вилучення інформації зі стеганограми у випадку відсутності зовнішнього впливу. Імітація атаки стиснення контейнеру-зображення показує не ефективність методу НЗБ в контексті забезпечення цілісності та автентичності інформації [15, 17]. Модифікація стеганограми шляхом стиснення призвела до викривлень інформаційних бітів. Алгоритм НЗБ є нестійким до внесення похибок до графічних контейнерів, що супроводжується фрагментарним або повним знищенням прихованого інформаційного контенту [2, 3]. Статистичні дані інформації до та після модифікації стеганограми наведені на рис. 2.8.

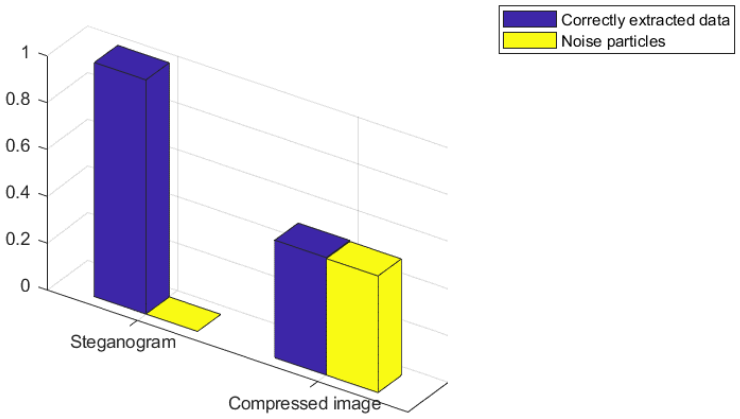


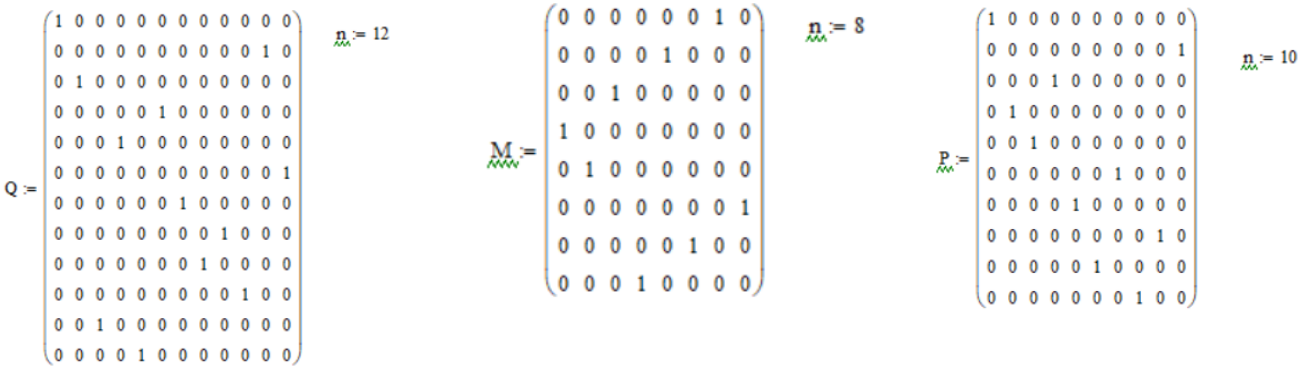
Рисунок 2.8 – Статистичні дані вилученої інформації

Методами протидії ймовірним спотворення є застосування завадостійких кодів та додаткове копіювання [2, 17] в границях стегоканалу.

2.2 Методи псевдовипадкової перестановки та псевдовипадкового інтервалу

Вразливість методу НЗБ призвела до створення похідних методів, які включають додаткові захисні механізми. Реалізувати модель безпеки стеганографічного каналу можна при експлуатації секретних ключів. Процедури вбудування та вилучення інформації управляються через стегоключ [17], який використовується при приховуванні інформаційних бітів. Стегоключ повинен зберігатися абонентами в таємниці [17] для забезпечення безпеки стегоканалу. Ключ гарантує секретність стегоканалу між сторонами спілкування для випадкових спостерігачів [2]. Застосування стегоключа забезпечує секретність існування [17] інформаційного контенту в межах об’єкту-контейнера. Першою похідною версією методу НЗБ є метод псевдовипадкової перестановки (ПВП). Особливість даного методу полягає в комбінаційному застосуванні технік криптографії та стеганографії. Сутність методу виражається [15] в розбитті бітів інформаційного повідомлення на блоки розміром *n*. Кожний окремий блок шифрується шляхом множення на квадратну матрицю, яка задає правило перестановки [13-15]. Розмір матриці еквівалентний розміру бітового блоку та представляє собою стегоключ. Бітові елементи блоку

зберігаються, але перемішуються між собою для трансформації вихідного блоку повідомлення у нерозбірливий вигляд (зашифрування) [2]. Приклади конструкції ключових матриць наведені на рис. 2.9.



а) *б)* *в)*

Рисунок 2.9 – Можливі варіанти ключів для методу ПВП

Перетворення для зашифрування [13-15] бітового блоку інформації:

$$M_i^{'} = M_i * P = (M_1, M_2, \ldots, M_n) * \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix}, \quad (2.2)$$

де M_i – блок бітів оригінального інформаційного повідомлення розміром n;

P – стегоключ, квадратна матриця розміром $n \times n$, яка встановлює правила перестановки.

Наступна частина алгоритму [15] передбачає вставку зашифрованих блоків бітів у цифровий контейнер за допомогою методу НЗБ. Перетворення для відновлення вихідного інформаційного повідомлення виконуються симетрично в оберненому порядку [15]: вилучення методом НЗБ та поблокове розшифрування за допомогою стегоключа. Вираз для розшифрування блоку бітів [13-15] має наступну математичну конструкцію:

$$M_i = M_i^{'} * P^{-1} = \left(M_1^{'}, M_2^{'}, \ldots, M_n^{'}\right) * \begin{pmatrix} p_{11} & p_{12} & \cdots & p_{1n} \\ p_{21} & p_{22} & \cdots & p_{2n} \\ \cdots & \cdots & \cdots & \cdots \\ p_{n1} & p_{n2} & \cdots & p_{nn} \end{pmatrix}^{-1}, \quad (2.3)$$

де $M_i^{'}$ – зашифрований блок бітів розміром n;

P^{-1} – зворотна матриця до матриці P.

Перетворення методу ПВП дозволяють зберегти пропускну спроможність стегоканалу [18, 19] та забезпечити додатковий шар безпеки. Еволюція методології стеганографічних методів включає класи гібридних методів, які дозволяють забезпечити додатковий рівень стійкості, мінімізуючи потенційні ризики. Наступною модифікацією методу НЗБ [15] є метод псевдовипадкового інтервалу (ПВІ). Даний алгоритм, також використовує секретний ключ для інкапсуляції чутливої інформації абонентів спілкування в нерухоме зображення. Ключовою особливістю цього методу є застосування неповної просторової області контейнеру [14, 15] для вставки бітів повідомлення. Такий підхід ускладнює детектування інформаційного повідомлення, але негативно позначається на пропускній здатності [13] стегаканалу, тобто на кількості інформації, яка може бути передана через такий стегоканал. Структура стегоключа представлена псевдовипадковою послідовністю чисел [13, 15], розміром n, що відповідає кількості стовпців матриці пікселів зображення. Сутність методу полягає в послідовній вставці бітів у зображення-контейнер за допомогою алгоритму НЗБ. Індекс біту відповідає індексу стовпця контейнера-зображення, а номер кортежу, куди буде здійснюватися вбудування визначається поточним числом секретного ключа [13-15]. Вилучення інформації здійснюється за аналогічним правилом методом LSB. Комбінаторика стегоключа методу ПВІ залежить від кількості рядків [15] аналізованого зображення-контейнеру. На основі визначеної специфікації методів ПВП та ПВІ виконано порівняльний аналіз з базовим методом НЗБ. Порівняльна характеристика [13-15] наведена в табл. 2.3.

Таблиця 2.3 – Порівняльний опис алгоритмів групи LSB

Назва методу	Властивість ЗСЛ, яка застосовується	Переваги	Недоліки	Коментарі
Метод НЗБ (LSB)	1-а низькорівнева властивість (слабка чутливість до незначної зміни яскравості)	Висока пропускна спроможність стегоканалу, фактично означає експлуатацію щонайменше 1/8 ємності контейнеру. Зі збільшенням числа використаних бітів підвищується значення пропускної здатності стегоканалу (до 2/8=1/4 або навіть до 3/8). Простота практичної реалізації, що позитивно впливає на швидкодію вставки та вилучення.	Вразливість алгоритму до методів стеганоаналізу, що обумовлене відсутністю стегоключа. Низький рівень стійкості та надійності стеганографічної системи. Вразливість до геометричних трансформацій та маніпуляцій над освітленістю зображення. Випадкові модифікації контейнеру тягнуть за собою часткове або повне зашумлення повідомлення, що порушує властивість доступності інформації.	Метод LSB є класичним методом стеганографії, який дозволяє організувати прихований стегаканал. Даний метод спирається, на ідеї того, що випадковий спостерігач з апріорними знаннями не зможе виявити та довести існування прихованого контенту. Слабкий рівень стійкості негативно впливає на безпеку прихованої інформації.
Метод ПВП		Комбінаційна структура методу підвищує шар стійкості перед атаками стеганоаналізу. Супроводжуючі операції шифру ніяк не впливають на пропускну здатність стегаканалу.	Вразливість до геометричних трансформацій та маніпуляцій над освітленістю зображення. Випадкові модифікації контейнеру тягнуть за собою часткове або повне зашумлення повідомлення.	Гібридизація методу дозволяє забезпечити подвійний рівень безпеки інформації. Введення стегоключа гарантує забезпечення секретності прихованих даних.

Продовження таблиці 2.3

Назва методу	Властивість ЗСЛ, яка застосовується	Переваги	Недоліки	Коментарі
Метод ПВІ	1-а низькорівнева властивість (слабка чутливість до незначної зміни яскравості)	Введення секретного ключа, що регламентує процедури вставки та вилучення ускладнює детектування прихованого контенту зловмисником. Простота практичної реалізації позитивно впливає на швидкодію вставки та вилучення.	Пропускна здатність стегоканалу суттєво зменшується. Система показує низьку стійкість до атак геометричного характеру та маніпуляцій над кольоровими властивостями зображення. Випадкові модифікації контейнеру тягнуть за собою часткове або повне зашумлення повідомлення.	Метод ПВІ дозволяє створити закриту стеганосистему, що обумовлено інтеграцією стегоключа в специфікацію алгоритму. Сутність методу синхронізована з методом НЗБ, але пониження області вбудування ускладнює механізм виявлення даних повідомлень.

Пропускна спроможність [18, 19] інтерпретується як відношення обсягу інкапсульованої інформації N, до загально обсягу контейнеру V:

$$C = \frac{N}{V}. \tag{2.4}$$

Досліджено пропускну здатність стегаканалу для алгоритмів LSB-1, LSB-2, LSB-4, ПВП та ПВІ. Проаналізовані контейнери представлені на рис. 2.10.

Експериментальні дані оцінки пропускної спроможності [18] з формули (2.4) наведені в табл. 2.4. Назва контейнеру відповідає графічному обрису зображення-контейнеру на рис. 2.10. У діаграмі на рис. 2.11 представлено графічне подання результатів досліджень.

а)

б)

в)

г)

ґ)

Рисунок 2.10 – Досліджувані контейнери-зображення

Таблиця 2.4 – Оцінка пропускної спроможності для контейнерів різних розмірів

Назва об'єкту контейнеру	Розмірність, пікселі	Пропускна здатність				
		LSB-1	LSB-2	LSB-4	ПВП	ПВІ
11.bmp *(а)*	575 x 360	0.125	0.125	0.125	0.125	0.00035
12.bmp *(б)*	541 x 360	0.125	0.125	0.125	0.125	0.00035
13.bmp *(в)*	576 x 360	0.125	0.125	0.125	0.125	0.00035
14.bmp *(г)*	620 x 354	0.125	0.125	0.125	0.125	0.00035
15.bmp *(ґ)*	662 x 372	0.125	0.125	0.125	0.125	0.00034

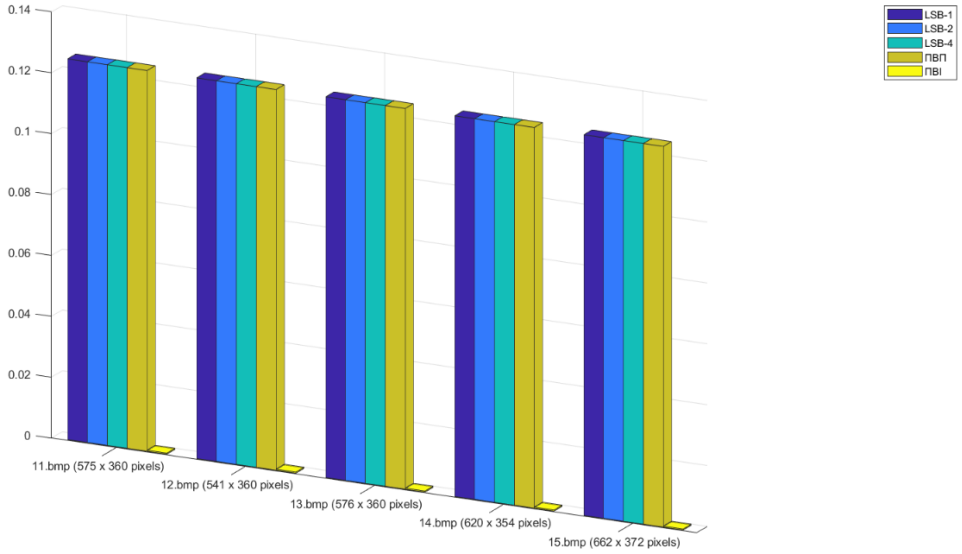


Рисунок 2.11 – Гістограма оцінки пропускної здатності

Проведено експериментальні дослідження реакції вилученої інформації на модифікацію геометричних та кольорових властивостей нерухомого зображення. Реалізовані трансформації (*а)* – перенесення, *б*) – поворот, *в)* – зсув, *г)* – масштабування, *ґ)* – зміна насиченості, *д)* – накладення шуму) представлені на рис. 2.12.

а) *б)* *в)*

г) *ґ)* *д)*

Рисунок 2.12 – Прийоми зміни властивостей зображення

Емпіричні дані досліджень стану прихованих даних після трансформації нерухомого зображення формату BMP-24 зазначено в табл. 2.5. Графічна статистика експериментальних даних подана на рис. 2.13. Експеримент проводився для методів НЗБ (рис. 2.13, а), ПВП (рис. 2.13, б), ПВІ (рис. 2.13, в).

Таблиця 2.5 – Експериментальні дані реакції даних на обробку контейнеру

Назва перетворення	Алгоритм НЗБ		Метод ПВП		Метод ПВІ	
	Правильно вилучені дані	Помилка вилучених даних	Правильно вилучені дані	Помилка вилучених даних	Правильно вилучені дані	Помилка вилучених даних
Перенесення	0.501	0.499	0.499	0.501	0.489	0.511
Поворот	0.491	0.509	0.51	0.49	0.522	0.478
Зсув	0.5	0.5	0.501	0.499	0.497	0.503
Масштабування	0.497	0.503	0.501	0.499	0.515	0.485
Зміна насиченості	0.51	0.49	0.488	0.512	0.52	0.48
Накладення шуму	0.502	0.498	0.5	0.5	0.534	0.466

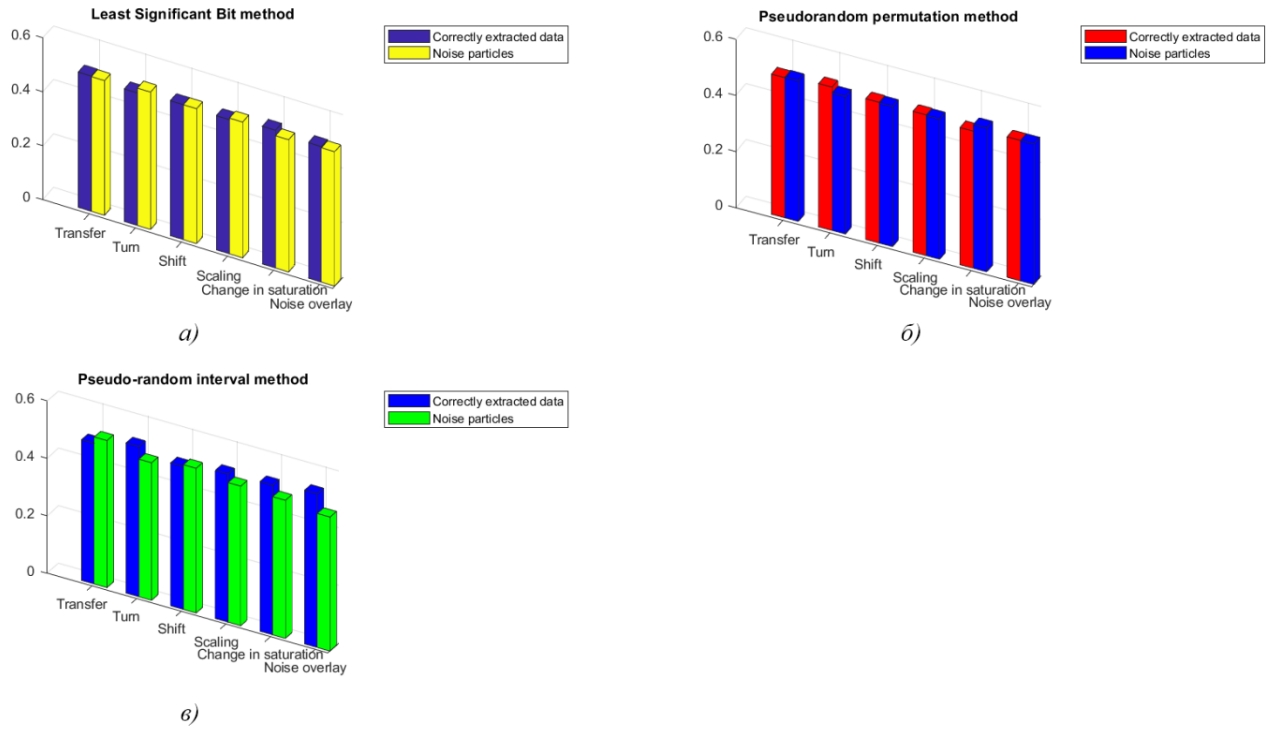


Рисунок 2.13 – Гістограма оцінки алгоритмів НЗБ (а), ПВП (б), ПВІ (в)

2.3 Дослідження показників безпеки стегоалгоритмів

Концепція безпеки стеганографічних алгоритмів базується на структурі стегоключа. Ключ дозволяє забезпечити факт таємниці ведення комунікації між абонентами мережі, забезпечуючи секретність існування інформаційного контенту [13, 17] в об'єкті-контейнері. Процедури вставки та вилучення інформаційних даних з стеганограми управляються стегоключем [17]. В процесі стеганографічного кодування абонентам комунікації повинен бути відомий секретний ключ [14, 17], який заздалегідь узгоджений або переданий через безпечний КЗ. Зазвичай в стеганографічних системах використовується один секретний ключ для забезпечення узгодженості [2] процедур вбудування та вилучення інформації. Можливий сценарій експлуатації асиметричних ключів, що є ефективною основою для встановлення комунікаційних взаємовідносин. Даний підхід дозволяє організувати безпечний механізм спілкування, якщо сторони не довіряють одна одній. Надійні стегосистеми повинні відповідати наступному ряду вимог [17]:

– схема стеганографічного перетворення включає загальнодоступний алгоритм та секретний стегоключ;

– техніка стеганографічного кодування повинна забезпечувати цілісність та автентичність прихованої інформації [14];

– при сценарії обізнаності атакуючого про зберігання інформаційного контенту у файловому об’єкті, це не повинно стати обґрунтуванням для третьої сторони [17] або компрометації інформаційних повідомлень без володіння секретним ключем;

– жодний зі спостерігачів не повинен знайти аргументів присутності інформаційного контенту на підставі статичних та ймовірнісних показників без знання секретного ключа.

Модель захищеності стеганографічної системи оцінюється конфігурацією секретного ключа. Розглянемо ключові показники стійкості секретних стегоключів. Обсяг даних у секретному ключі (число бітів) задається формулою [19]:

$$\ell_{key} = \log_2(|K|), \tag{2.5}$$

де K – потужність ключового простору.

Оцінку стійкості стеганографічного алгоритму [19] можна дослідити шляхом розрахунку оберненої величини до потужності ключової множини. Вона розглядається як імовірнісний показник підбору секретного ключа [13, 19]:

$$W = \frac{1}{|K|} = 2^{-l_{key}}. \tag{2.6}$$

Ентропія джерела ключів задається наступною формулою [20]:

$$H(K) = -\sum_{i=1}^{K} P(K_i)\log_2 P(K_i), \tag{2.7}$$

де $P(K_i)$ – вірогідність породження ключа K_i джерелом.

Якщо $P(K_i) = \frac{1}{K}$ для всіх K_i, то формула ентропії в (2.7) спрощується до виду:

$$H(K) = -\sum_{i=1}^{K} \frac{1}{K}\log_2 \frac{1}{K}. \tag{2.8}$$

Потрібно підкреслити, що формула (2.8) тотожна формулі Шеннона для рівномірного розподілу, де всі значення випадкової величини мають однаку ймовірність. У такому випадку, вираз (2.8) можна значно спростити до:

$$H(K) = -\sum_{i=1}^{K} \frac{1}{K} \log_2 \frac{1}{K} = -K * \frac{1}{K} * \log_2 \frac{1}{K} = -\log_2 \frac{1}{K} = \log_2 K \,. \qquad (2.9)$$

У табл. 2.6 наведені формули для обчислення потужності ключів [15, 18] для різноманітних стеганографічних алгоритмів. Вираз розрахунку потужності простору ключів може варіюватися в залежності від специфікації ключа та стеганографічної системи.

Таблиця 2.6 – Визначення потужності для стеганографічних методів

Назва алгоритму	Потужність ключової множини
ПВП	$n!$
ПВІ	n
Метод квантування	2^n

Параметр n визначає розмірність стеганографічного ключа, який дозволяє налаштовувати алгоритм стеганографічного кодування. Чим більше значення довжини ключа n, тим більше комбінацій секретного ключа, що підвищує стійкість алгоритму перед потенційними атаками. Рівень надійності та ефективності стеганографічного захисту залежить від структури стегоключа та його розмірності. Складність конструкції та масштабна множина комбінаторики ключа робить складним задачу компрометації секретного ключа. Ідея стеганографічного захисту заснована на тому, що третя сторона [17] не спроможна відновити та довести факт присутності таємного КЗ без відповідного стегоключа. У таблиці 2.7 наведені дані досліджень [13, 18] параметрів безпеки класичних стеганоалгоритмів.

На основі емпіричних даних в табл. 2.7 побудовані гістограми на рис. 2.14 для досліджуваних ключових стеганографічних систем.

Таблиця 2.7 – Емпіричні дані оцінки ключів стеганоалгоритмів

Назва алгоритму	Тип стегоключа	Розмірність ключа n	Потужність простору ключа, K	Імовірнісний показник підбору секретного ключа	Ентропія джерела ключів $H(K)$, біти
ПВП	Квадратна матриця	2	2	0.5	1
		3	6	0.167	2.58
		4	24	0.042	4.58
		5	120	$8.333*10^{-3}$	6.9
		6	720	$1.389*10^{-3}$	9.49
		7	5040	$1.984*10^{-4}$	12.3
		8	40320	$2.48*10^{-5}$	15.3
		9	362880	$2.756*10^{-6}$	18.47
ПВІ	Масив псевдовипадкових десяткових чисел	100	100	0.01	6.64
		200	200	$5*10^{-3}$	7.64
		320	320	$3.125*10^{-3}$	8.32
		450	450	$2.222*10^{-3}$	8.81
		550	550	$1.818*10^{-3}$	9.1
		600	600	$1.667*10^{-3}$	9.23
		650	650	$1.538*10^{-3}$	9.34
		700	700	$1.429*10^{-3}$	9.45
Метод квантування	Таблиця різниць інтенсивностей пікселів	511	$6.704*10^{153}$	$1.49*10^{-154}$	511

Отримані емпіричні дані вказують на залежність показника ймовірності підбору секретного ключа $P(K_i)$ (рис. 2.15, а) та ентропії джерела ключів (рис. 2.15, б) від параметра потужності ключової множини K. Спостерігається зменшення ймовірності підбору стегоключа зі збільшенням потужності ключової множини K, що підвищує стійкість стеганографічної системи. В свою чергу, підвищення параметра потужності збільшує значення ентропії ключового генератора $H(K)$. Збільшення варіації ключів обумовлює більш випадковий розподіл ймовірностей вибору секретного ключа, тобто більшої різноманітності інформації в ключовій множині. Ускладнюється процес вгадування або підбору секретного ключа зловмисником через великий вектор варіантів ключів.

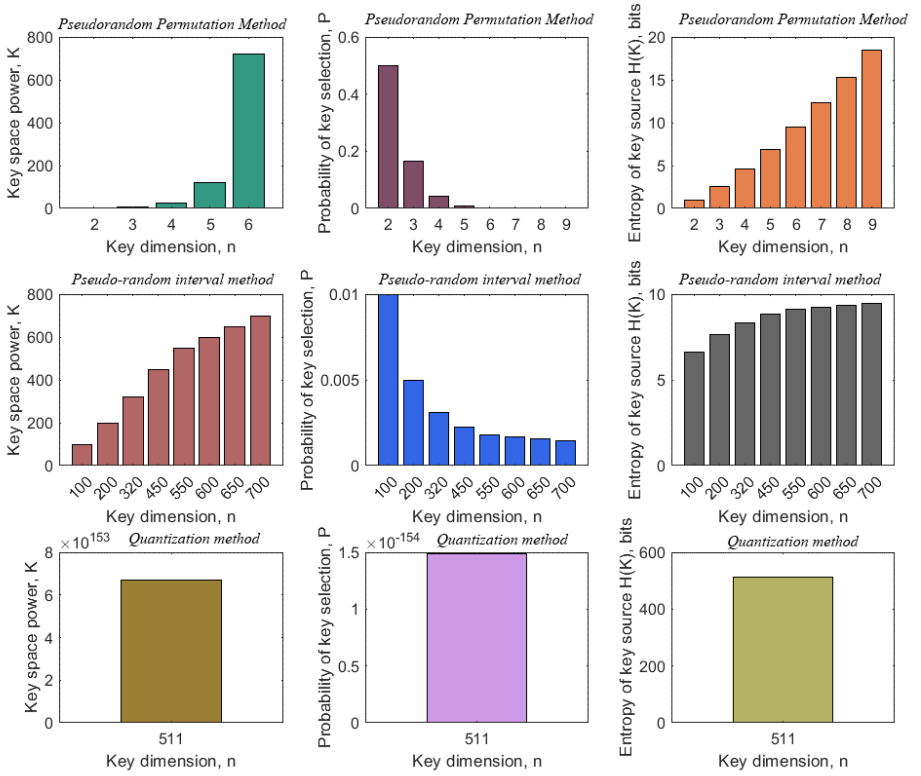


Рисунок 2.14 – Гістограми оцінки показників безпеки стегоключів

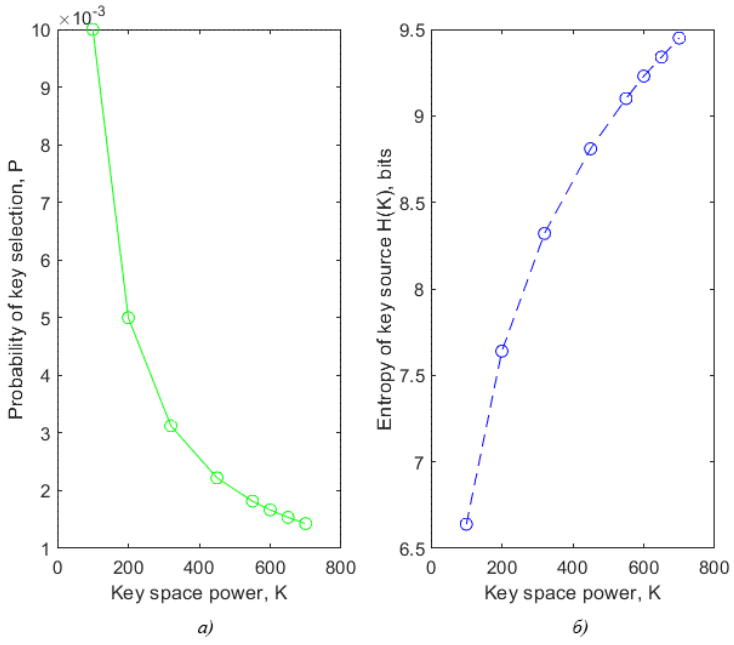


Рисунок 2.15 – Залежність показника ймовірності підбору стегоключа (а) та ентропії джерела ключів (б) від величини потужності простору ключів

2.4 Аналіз методу кодування початкових фаз

Прогресивним вектором стеганографії є техніка приховування даних в аудіозаписах. Фізіологічні властивості [21] слухової системи людини (ССЛ) експлуатуються для розробки нових методів стеганографічного захисту в аудіопарадигмі. Один з ефективних методів приховування інформації в аудіоконтейнерах – метод кодування початкових фаз [14, 21]. Цей алгоритм ґрунтується на експлуатації 4-ї низькорівневій властивості ССЛ (несприйнятливість ССЛ до зміни абсолютної фази аудіосигналу) [21, 14]. Основна концепція алгоритму кодування фаз [21] реалізується через заміну фази початкового аудіосегменту на базову фазу, в якій закодовані біти чутливої інформації. Збереження фазових різниць досягається через узгодження [21] між еталонною фазою та фазами аудіосегментів вихідного сигналу. ССЛ не помічає незначну модифікацію фази, що дозволяє вбудовувати дані в аудіооб'єкт. Алгоритм кодування фаз є найбільш ефективним в контексті відношення сигнал/шум [21]. Використання цього способу стеганографічного кодування дозволяє мінімізувати компоненту шуму у сигналі, який ССЛ не спроможна розпізнати у модифікованій версії. Це пояснюється особливостями сприйняття звуку, які сформувалися в ході еволюції людини. Прийоми фазового кодування інформаційного контенту у аудіоконтейнер спрямовані на забезпечення непомітності присутності стороннього шуму в об'єкті-контейнері. Методологія інтеграції інформаційного фрагмента у аудіосигнал через кодування фаз включає наступні етапи [13, 21]:

1) Потік звукового сигналу $C(1 \le i \le I)$ розділяється на спектр N сегментів $Seg_n(1 \le n \le N)$.

2) Кожний блок Seg_n перетворюється за допомогою K-точкового дискретного перетворення Фур'є (ДПФ), де $K = \frac{I}{N}$. Математична процедура ДПФ обчислюється за формулою [22]:

$$X_k = \sum_{n=0}^{N-1} x_n e^{-\frac{2\pi i}{N} kn} = \sum_{n=0}^{N-1} x_n e^{-iw_k t_n} = \sum_{n=0}^{N-1} x_n \cdot \left[\cos\left(\frac{2\pi kn}{N}\right) - i \cdot \sin\left(\frac{2\pi kn}{N}\right) \right], \; k = 0,1,\ldots,N-1, \qquad (2.10)$$

де N – кількість значень сигналу, виміряних за період T, а також кількість компонент розкладу;

$x_n = x(t_n),\ n = 0,1,\ldots,N-1$ – дискретні відліки з номерами $n = 0,1,\ldots,N-1$, тобто вимірювані значення вихідного сигналу [22] в дискретних часових точках $t_n = \frac{n}{N}T$, дискретні вимірювання є вхідними даними для ДПФ та вихідними для зворотного дискретного перетворення Фур'є (ЗДПФ);

$X_k,\ k = 0,1,\ldots,N-1$ – комплексні амплітуди синусоїдальних сигналів;

k – індекс частоти;

$w_k = 2\pi f_k = \frac{2\pi k}{T}$ – кругова частота (частота обертання); $w_k t_n = \frac{2\pi k}{T}\cdot\frac{n}{N}T = \frac{2\pi}{N}kn$;

i – уявна одиниця, тобто це число квадрат якого дорівнює -1.

Застосування формули (2.10) до кожного відліку аудіофрагмента трансформує його у комплексну амплітуду:

$$X_k = \mathrm{Re}(X_k) + i\,\mathrm{Im}(X_k). \tag{2.11}$$

Математична конструкція (2.11) отримана з формули (2.10) дозволяє сформувати масиви [14, 21] амплітуд $A_n(X_k)$ та фаз $\varphi_n(X_k)$ для $1 \le k \le K$. Математичні вирази для обчислення амплітуд та фаз мають вигляд:

$$\begin{aligned} A_k &= \sqrt{\mathrm{Re}^2(X_k) + \mathrm{Im}^2(X_k)} \\ \varphi_k &= \arg(X_k) = \arctan\left(\frac{\mathrm{Im}(X_k)}{\mathrm{Re}(X_k)}\right). \end{aligned} \tag{2.12}$$

3) Зберігається різниця фаз [14] між кожними дотичними сегментами для $1 \le n \le N$:

$$\Delta\varphi_n(X_k) = \varphi_n(X_k) - \varphi_{n-1}(X_k),\quad \Delta\varphi_1(X_k) = 0. \tag{2.13}$$

4) Правила кодування інформаційних бітів повідомлення у фазах передбачає створення послідовності компоненти, якої $\varphi_{data} = \frac{\pi}{2}$ та $\varphi_{data} = -\frac{\pi}{2}$, що представляє стан біту «1» або «0» відповідно, $\varphi_1'(X_k) = \varphi_{data}$.

5) Враховуючи фазові різниці, будується новий вектор фаз:

$$\left|\begin{array}{l}\varphi'_1(X_k)=\varphi_{data}\\ \varphi'_2(X_k)=\varphi'_1(X_k)+\Delta\varphi_2(X_k)\\ \quad\cdots\\ \varphi'_n(X_k)=\varphi'_{n-1}(X_k)+\Delta\varphi_n(X_k)\\ \quad\cdots\\ \varphi'_N(X_k)=\varphi'_{N-1}(X_k)+\Delta\varphi_N(X_k)\end{array}\right|. \quad (2.14)$$

6) Для відновлення аудіосигналу [21] використовується ЗДПФ до масивів оригінальних амплітуд та масивів змінених фаз.

Обчислення дискретного сигналу здійснюється за допомогою ЗДПФ, яке має вигляд [22]:

$$x_n=\frac{1}{N}\sum_{k=0}^{N-1}X_k e^{\frac{2\pi i}{N}kn}=\frac{1}{N}\sum_{k=0}^{N-1}X_k e^{iw_k t_n}=\frac{1}{N}\sum_{k=0}^{N-1}X_k\cdot\left[\cos\left(\frac{2\pi kn}{N}\right)+i\cdot\sin\left(\frac{2\pi kn}{N}\right)\right],\ n=0,1,\ldots,N-1. \quad (2.15)$$

Процедура вилучення інформації з аудіосигналу включає наступні етапи:

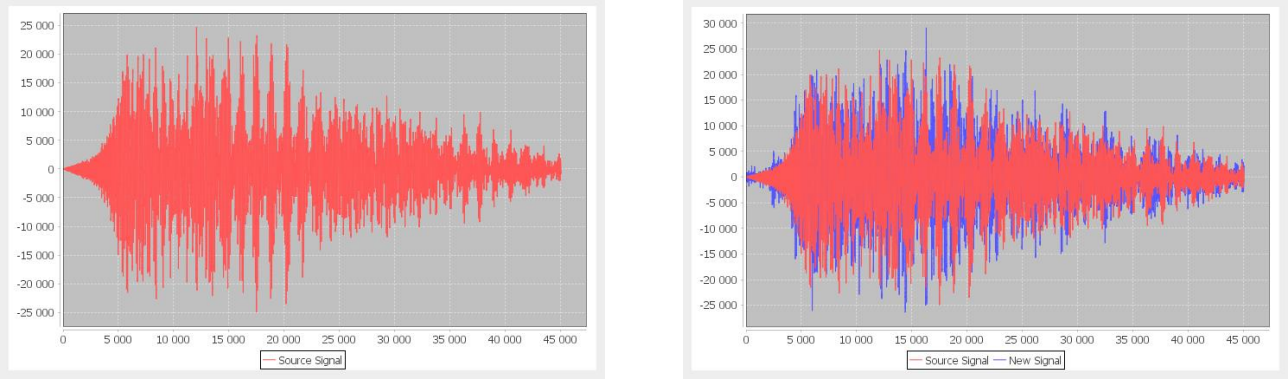
1) Вилучення першого сегмента $G(1\le k\le K)$ з аудіостеганограми $C1(1\le i\le I)$.

2) Застосування ДПФ до G та виділення масиву амплітуд $A_1(X_k)$ та фаз $\varphi_1(X_k)$.

3) Декодування інформаційних бітів на підставі правила:

$$\begin{cases}M\_b_j\leftarrow 0 & if\ \varphi_k\le 0\\ M\_b_j\leftarrow 1 & if\ \varphi_k>0\end{cases}. \quad (2.16)$$

Графічна форма первісного (рис. 2.16, а) та модифікованого сигналу (рис. 2.16, б) за допомогою алгоритму кодування фаз наведено на рис. 2.16.



а) *б)*

Рисунок 2.16 – Оригінальний (а) та новий (б) звуковий сигнал

Загалом, новий сигнал імітує початковий звуковий потік [21], що вказує на коректність вбудування інформаційних бітів. Присутні незначні спотворення модифікованого сигналу [14, 21], але вони не становлять вагомого значення для оцінки отриманого аудіопотоку. Порівняння двох сигналів на слух не показало суттєвих розходжень у звучанні, що підтверджує якість методу кодування фаз. Важливим нюансом є вплив *K*-точкового ДПФ на довжину спектральної форми дискретного сигналу. Дискретний сигнал розмірністю *K* при трансформації за допомогою часткового ДПФ набуває довжини *K/2+1*, що позначається на пропускній спроможності стегоканалу [14, 18].

Досліджено поведінку пропускної здатності на зміну розмірності аудіофрагмента. Аналізований контейнер має такі характеристики:

– частота дискретизації $f_д$ = 44100 Гц;

– число каналів N_K = 1 (моно сигнал);

– кількість семплів numSamples = 45070;

– кількість бітів в одному семплі Q = 16 біт.

Результати експериментальних досліджень наведені в табл. 2.8.

Таблиця 2.8 – Результати оцінки аудіостеганосистеми

Кориговані довжини аудіопотоку, семпли	Сукупна кількість серій аудіопотоку	Довжина сегменту, семпли	Розмір перетвореного спектра за допомогою *K*-точкового ДПФ, комплексні амплітуди	Пропускна спроможність стегоканалу, *C*
45056	44	1024	513	0.00071
45056	22	2048	1025	0.00142
45056	11	4096	2049	0.00284
40960	5	8192	4097	0.00625
32768	2	16384	8193	0.01563
32768	1	32768	16385	0.03125

Аналіз даних показав, що застосування *K*-точкового ДПФ для обробки сегменту аудіопотоку призводить до скорочення розмірності сегменту більше, ніж удвічі. Цей фактор пасивно впливає на обсяг контейнеру, який можна використовувати для приховування набору даних. Збільшення довжини сегменту призводить до підвищення пропускної здатності стегаканалу. Варіативні розміри блоків викликають

необхідність реконструювати сигнал, тобто обрізання частини його дискретних відділків для поділу на рівні фрагменти. Переформування структури сигналу є необхідною процедурою для коректного монтування даних в аудіозапис. Інтенсивний вплив на параметри контейнеру впливають на його звукову якість. Викривлення звучання аудіопотоку можуть розкрити факт існування прихованого зв'язку. Криві проілюстровані на рис. 2.17 візуалізують взаємозв'язки атрибутів аудіостеганосистеми.

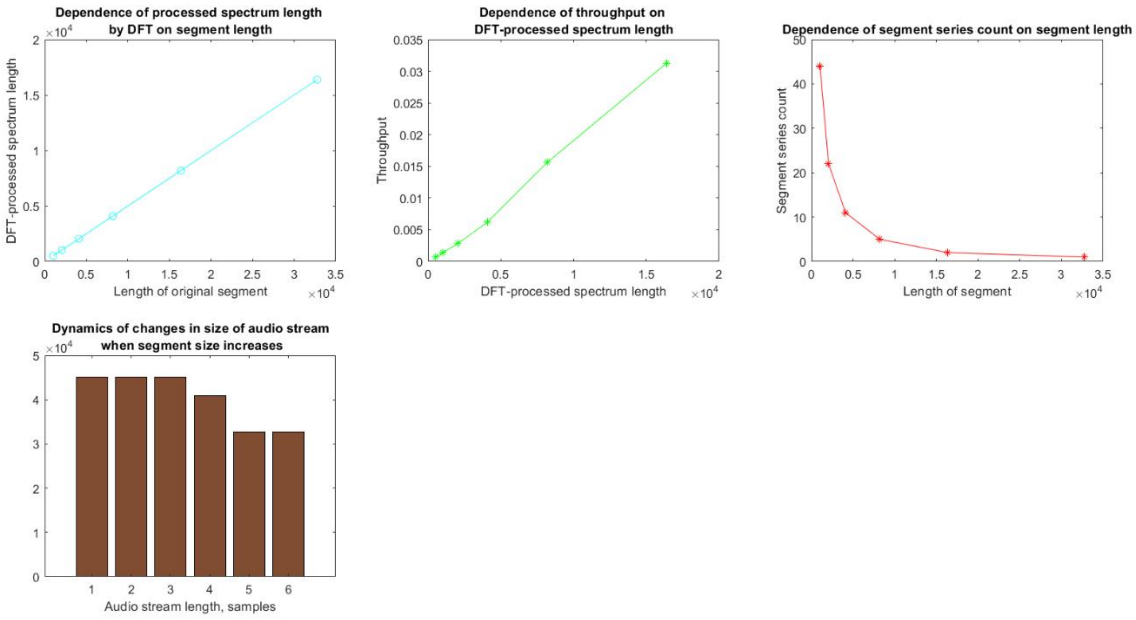


Рисунок 2.17 – Залежності характеристик аудіостеганограми

Фазове кодування дозволяє використовувати збитковість аудіоконтейнеру, зміна якої непомітна для ССЛ. Метод кодування фаз демонструє якість та непомітність монтування даних в аудіопотік, але обсяг інформації, який можна інкапсулювати обмежений розміром перетвореного спектра за допомогою ДПФ. Даний метод можна використовувати для приховування невеликих повідомлень. Застосування неповної просторової області аудіосигналу може в позитивному аспекті вплинути на цілісність та автентичність інформації. Величина стороннього шуму [21], яка вноситься до аудіосигналу не суттєва, що значно не впливає на вихідну структуру аудіозапису і не руйнує його якість.

ВИСНОВКИ

Активізація інтеграції технічних систем у людську діяльність стимулювала розвиток варіативних технологій комунікації та зв’язку. Сучасна програмно-апаратна інфраструктура спілкування, яка є базисом теперішнього Інтернету розширює спектр можливостей соціуму, забезпечуючи якість представлених послуг. Разом з цим, Інтернет як глобальна технічна медіаплатформа комунікаційної взаємодії дотична з великим спектром кіберзагроз. Векторами протидії, які інтегруються в автоматизовані системи та комплекси встановлення зв’язку є криптографічні та стеганографічні методики інформаційної безпеки. Виникнення квантових систем, спонукало розробити новий підхід до криптографічну захисту. Постквантова криптографія є новим прогресивним розділом криптографії, який активно впроваджується в мережеві протоколи. Традиційні криптосистеми вразливі до атак квантового криптоаналізу, що мотивує до оновлення стандартизації в галузі криптографії. Безперечно, постквантові алгоритми продемонстрували якість та надійність захисту проти кібератак. Проблематики, на яких базуються постквантові криптосистеми є дуже складними завданнями навіть для квантових обчислювальних процесорів. Підвищується інтерес до прийомів та методів стеганографії, що можливо реалізувати на програмно-апаратних комплексах, у випадку неможливості застосування криптопротоколів. Поширеним феноменом є гібридизація методик захисту, наприклад, криптографії та стеганографії, задля впровадження подвійного рівня інформаційної безпеки. Основою моделі безпеки криптосистем та стеганосистем є секретний ключ, яким управляються процеси інкапсуляції та декапсуляції чутливої інформації [2]. Вважається, що зловмисник не спроможний отримати доступ до вмісту конфіденційних даних без знання ключа.

Сучасні стеганографічні системи не володіють достатнім рівнем надійності та якості, що є сигналом для модернізації стеганографічних методик. Тестування стеганографічних систем заснованих на традиційних стеганоалгоритмах показали вразливість стеганограми до зовнішнього впливу. Найменша модифікація контейнеру

[2, 13] провокує часткове або повне знищення інформаційного контенту. Важливою властивість стеганографічних систем є забезпечення непомітності та секретності факту існування КЗ [14]. Сучасні інформаційні платформи містять модулі аналізу файлових об’єктів на присутність супроводжуючої інформації. Оцінка відбувається на базі статичних та ймовірнісних показників. Специфікація стеганографічного алгоритму повинна включати механізм [13], при якому обсяг внесеного шуму суттєво не впливає на природні показники цифрового контейнеру.

Концепція безпеки алгоритмів криптографії та стеганографії базується на структурі секретного ключа. Великий простір комбінацій ключів дозволяє суттєво зменшити ймовірність підбору правильного варіанту ключа, що забезпечує стійкість системи проти атак направлених на отримання НСД до ключа. Збільшення потужності ключової множини розширює різноманітність ключової інформації, що породжується ключовим генератором. Кожна версія секретного ключа [20] повинна мати властивості випадковості та бути незалежною одна від одної. Виключається ймовірність компрометації таємної інформації [17], якщо зловмисна сторона не володіє потрібним екземпляром секретного ключа.

Програмні реалізації алгоритмів стеганографії дозволяють в прискореному режимі монтувати інформаційні дані в контейнер, не затрачуючи багато обчислювальних ресурсів системи. Інтеграція стеганографічних систем є необхідною мірою безпеки, яку можна використовувати у військовій галузі, секторі державної політики та сфері інтелектуальної власності. Об’єднання прийомів захисту інформації може гарантувати високий рівень захищеності інформації від кіберзлочинців.

На тлі появи нових різновидів кіберзагроз, розгортання векторів безпеки є актуальною задачею. Оновлена стандартизація в галузі кібербезпеки дозволить створити безпечний механізм для комунікаційної взаємодії.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Богданова Є., Чорна Т., Малахов С. Огляд поточного стану загроз, що обумовлені впливом експлойтів. Computer science and cybersecurity. 2022. № 2(22). URL: https://periodicals.karazin.ua/cscs/article/view/21039/19745 (дата звернення: 08.04.2024).

2. Микита Бодня Дослідження можливостей застосування стеганографічних та криптографічних алгоритмів для приховування інформації / Микита Бодня, Марина Єсіна, Володимир Пономар // Computer science and cybersecurity. – Харківський національний університет імені В. Н. Каразіна, 2023, Випуск 2(24) – С. 43–57. – Режим доступу: https://periodicals.karazin.ua/cscs/article/view/23121/21124 (дата звернення: 08.04.2024).

3. Кузнецов О. О., Кононченко А. В. Стеганографічні методи в векторній графіці. Радіотехніка. 2021. № 2(205). URL: http://rt.nure.ua/article/view/239515/237998 (дата звернення: 08.04.2024).

4. Єсіна М. В., Кандій С. О., Острянська Є. В., Горбенко І. Д. Генерація загальносистемних параметрів для схеми електронного підпису Rainbow для 384 та 512 біт безпеки. Радіотехніка. 2017. № 191. URL: http://rt.nure.ua/article/view/238514/237125 (дата звернення: 09.04.2024).

5. Циганок Д. А. Аналіз можливостей алгоритму Kyber CRYSTALS. Радіоелектроніка та молодь у XXI столітті: матеріали 27-го Міжнар. молодіж. форуму. 2023. Т. 6, ч. 1. URL: https://openarchive.nure.ua/handle/document/24830 (дата звернення: 08.04.2024).

6. Горбенко І. Д., Качко О. Г., Єсіна М. В., Пономар В. А. Порівняльна характеристика алгоритмів інкапсуляції ключів Crystals-Kyber та Скеля (ДСТУ 8961-2019). Radiotekhnika. 2022. № 210. URL: http://rt.nure.ua/article/view/268560/264139 (дата звернення: 08.04.2024).

7. J. Bos Crystals-Kyber: A CCA-Secure Module-Lattice-Based KEM. 2018 IEEE European Symposium on Security and Privacy (EuroS&P). 2018. P. 353-367. URL:

https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=8406610 (дата звернення: 07.04.2024).

8. Бодня М. О., Єсіна М. В., Пономар В. А. Основні особливості інфраструктури відкритих ключів. Radiotekhnika. 2023. № 214. URL: http://rt.nure.ua/article/view/297799/290702 (дата звернення: 12.04.2024).

9. R. Avanzi Crystals-Kyber. Algorithm Specifications And Supporting Documentation. 2021. P. 43. URL: https://pq-crystals.org/kyber/data/kyber-specification-round3-20210131.pdf (дата звернення: 07.04.2024).

10. T. Nguyen. An Analysis of Hardware Design of MLWE-Based Public-Key Encryption and Key-Establishment Algorithms.Electronics. 2022. P. 13. URL: https://www.researchgate.net/publication/359210690_An_Analysis_of_Hardware_Design_of_MLWE-Based_Public-Key_Encryption_and_Key-Establishment_Algorithms (дата звернення: 13.04.2024).

11. Crystals-Kyber: The Key to Post-Quantum Encryption. Medium: website. URL: https://medium.com/@hwupathum/crystals-kyber-the-key-to-post-quantum-encryption-3154b305e7bd (дата звернення: 16.04.2024).

12. Kyber. Repository Githhub: website. URL: https://github.com/pq-crystals/kyber (дата звернення: 16.04.2024).

13. Кузнецов О. О., Євсеєв С. П., Король О. Г. Стеганографія: навчальний посібник. Харків, 2011. – 232 с.

14. Конахович Г. Ф., Прогонов Д. О., Пузиренко О. Ю. Комп’ютерна стеганографічна обробка й аналіз мультимедійних даних: підручник. Київ, 2018. – 558 с.

15. Кузнецов О. О., Полуяненко М. О., Кузнецова Т. Ю. Приховування даних у просторовій області нерухомих зображень шляхом модифікації найменш значущого біта: методичні рекомендації до лабораторної роботи з дисципліни «Стеганографія» для студентів спеціальності 125 «Кібербезпека». Харків, 2019. – 48 с.

16. Денисюк В. О., Денисюк А. В., Денисюк Н. В. Програмна реалізація стеганографічного алгоритму захисту інформації. Ефективна економіка. 2018. № 4. URL: http://www.economy.nayka.com.ua/?op=1&z=6223 (дата звернення: 02.05.2024).

17. Хорошко В. О., Яремчук Ю. Є., Карпінець В. В. Комп'ютерна стеганографія: навч. посіб. Вінниця, 2017. 155 с.

18. Кузнецов О. О., Полуяненко М. О., Кузнецова Т. Ю. Приховування даних у просторовій області нерухомих зображень методом блокового вбудовування, методом квантування та методом «хреста»: методичні рекомендації до лабораторної роботи з дисципліни «Стеганографія» для студентів спеціальності 125 «Кібербезпека». Харків, 2019. – 48 с.

19. Кузнецов О. О., Полуяненко М. О., Кузнецова Т. Ю. Приховування даних в просторовій області нерухомих зображень на основі прямого розширення спектра»: методичні рекомендації до лабораторної роботи з дисципліни «Стеганографія» для студентів спеціальності 125 «Кібербезпека». Харків, 2019. – 68 с.

20. Горбенко І. Д., Горбенко Ю. І. Прикладна криптологія: підручник. Форт, 1 та 2 видання. Харків, 2013. 878 с.

21. Кузнецов О. О., Полуяненко М. О., Кузнецова Т. Ю. Приховування даних в аудіоконтейнерах: методичні рекомендації до лабораторних робіт з дисципліни «Стеганографія». Харків, 2019. – 52 с.

22. Блейхут Р. Є. Теорія та практика кодів, що контролюють помилки: монографія. М.: Мир, 1986. – 576 с.

ДОДАТОК А

ПЕРЕЛІК ПУБЛІКАЦІЙ

ISSN 2519-2310 CS&CS, Issue 2(24) 2023

УДК 004.056.5

ДОСЛІДЖЕННЯ МОЖЛИВОСТЕЙ ЗАСТОСУВАННЯ СТЕГАНОГРАФІЧНИХ ТА КРИПТОГРАФІЧНИХ АЛГОРИТМІВ ДЛЯ ПРИХОВУВАННЯ ІНФОРМАЦІЇ

Микита Бодня[1], Марина Єсіна[1,2], Володимир Пономар[1,2]

[1] Харківський національний університет імені В. Н. Каразіна, майдан Свободи, 4, Харків, 61022, Україна
bodnia2020kb12@student.karazin.ua, m.v.yesina@karazin.ua
[2] АТ «ІТ», вулиця Коломенська, 15, Харків, 61166, Україна
Laedaa@gmail.com

Надійшла: Листопад 2023. Прийнята: Грудень 2023.

Анотація: *Організація захисту інформації завжди було актуальною задачею особливо після появи інформаційно-комунікаційних систем. Базисними напрямленнями в області захисту інформації, які прийшли зі стародавніх часів є криптографія та стеганографія. Криптографія реалізує захист інформації шляхом перетворення інформації у нечитабельний вигляд. Стеганографія дозволяє приховати інформацію в різних контейнерах, при цьому факт наявності інформації залишається непомічним для випадкових спостерігачів. У статті розглядаються підходи до криптографії та стеганографії, концепція гібридного застосування криптографічних та стеганографічних методів для забезпечення подвійного рівня захисту даних, загальна архітектура криптографічних та стеганографічних систем. Традиційними криптографічними системами, які застосовуються в сучасних системах захисту інформації є симетричні та асиметричні криптосистеми. Хоча симетричні системи еволюціонували з появою нових математичних перетворень, але вони мають суттєвий недолік. Він полягає в потребі додаткової передачі секретного ключа отримувачу. Така стратегія вимагає використання захищеного каналу зв'язку, оснащеного технічними системами захисту. При цьому існує ризик несанкціонованого доступу, який може спричинити компрометацію секретного ключа. Виходячи з вищенаведених проблем симетричних криптосистем, при розробці механізмів захисту, перевагу віддають асиметричним алгоритмам. Проведено аналіз криптосистеми RSA, яка ґрунтується на асиметричному підході шифрування. Ця система використовується в сучасних протоколах автентифікації та забезпечення конфіденційності в інформаційних системах та Інтернеті. Проведено дослідження швидкодії програмних модулів генерації ключової пари, шифрування та розшифрування для системи RSA, шляхом зміни загальних параметрів алгоритму (модуля перетворень, розміру вихідних даних). Результати часових вимірювань наведені в таблиці, на базі яких побудовані залежності часу від модифікації конкретних параметрів. Досліджено стеганографічний алгоритм модифікації найменш значущого біту (НЗБ), який застосовується для приховування даних в зображеннях. Нині існує широкий спектр стеганоалгоритмів, які розробляються на базі особливостей сенсорних систем людини (системи зору або слуху). Розглядаються властивості зорової системи людини, які використовуються в стеганографії.*

Ключові слова: *криптографія, стеганографія, ключ, інформаційне повідомлення, асиметрична криптосистема, симетрична криптосистема, криптограма, стеганограма.*

1. Вступ

Інформація завжди займала провідне місце в житті людини. Поняття «інформація» [1] можна інтерпретувати як сукупність публічно оголошених або документованих відомостей, які охоплюють явища природи, навколишнього середовища та різноманітні області діяльності соціуму й держави. Вагомість і класифікація інформації визначається її вмістом. Поява інформаційно-комунікаційних систем і глобальних мереж спрощує доступність й обмін інформацією. Стрімкий технологічний прогрес призвів до появи загроз несанкціонованого доступу, порушення конфіденційності, цілісності інформації, фальсифікації даних тощо. Поряд з цим питання забезпечення інформаційної безпеки (ІБ) завжди було актуальним, починаючи зі стародавніх часів і до теперішнього моменту. Основними напрямами, що впроваджують надійні механізми забезпечення ІБ є криптографія і стеганографія [2].

Для розв'язання проблем ІБ широко використовуються відповідні алгоритми криптографії і стеганографії. Сучасні системи ІБ розробляються з реалізацією перспективних криптографічних і стеганографічних методів захисту. Система інформаційної безпеки (СІБ) [1]

УДК 004.056.5 **DOI:10.30837/rt.2023.3.214.02**

М. О. БОДНЯ, М. В. ЄСІНА, канд. техн. наук, В. А. ПОНОМАР, канд. техн. наук

ОСНОВНІ ОСОБЛИВОСТІ ІНФРАСТРУКТУРИ ВІДКРИТИХ КЛЮЧІВ

Вступ

На сьогодні широко застосуються засоби мережевої інфраструктури та інформаційно-комунікаційні системи для організації спілкування та обміну даними між користувачами. Внаслідок цього виникло питання – як забезпечити автентифікацію всіх авторизованих користувачів. Сучасні криптографічні протоколи автентифікації базуються на криптографії з відкритим ключем. Системи захисту інформації та технічні засоби захисту не можуть повністю гарантувати запобігання несанкціонованому доступу до каналу зв'язку, внаслідок цього реалізовуються різноманітні протоколи та системи автентифікації, що ґрунтуються на асиметричній криптографії. Застосування технологій та процедур, що засновані на криптографії з відкритим ключем, широко впроваджуються в системи комерційних організацій та урядових установ, тому що вони забезпечують надійний механізм, який дозволяє підтвердити, що особа є тим, за кого себе видає, та реалізувати конфіденційність, цілісність, неспростовність та автентичність інформації.

Як показує практика, застосування асиметричної криптографії є недостатнім комплексом методів та технологій для забезпечення процедури достовірної автентифікації та обміну інформацією між авторизованими користувачами. Суб'єкти комунікації гіпотетично можуть використовувати паперові документи, які містять персональні дані та відкриті ключі авторизованих користувачів, підписані рукописним підписом та завірені нотаріусом. Але в такому випадку виникає проблема масштабності: проведення нотаріального завірення документів для великої множини суб'єктів спілкування потребує великої кількості аркушів паперу та займає багато часу. Інфраструктура відкритих ключів (ІВК) є надійним інструментом для розв'язання задач, пов'язаних з автентифікацією користувачів та визначенням легітимності, справжності відкритих ключів користувачів у цифровому середовищі.

Структура ІВК складається зі спеціалізованих компонентів, кожен з яких має власний напрям діяльності та фіксований спектр задач. При цьому забезпечуються всі процеси відносно управління цифровими сертифікатами, які включають: видачу, перевипуск, відкликання сертифікатів, управління життєвим циклом та ключами сертифікатів тощо. Такі сертифікати підтверджують факт належності певного відкритого ключа конкретному суб'єкту та наявності у відповідного суб'єкта секретного ключа. Завдяки цифровим сертифікатам всі сторони можуть ідентифікувати один одного та безпечно обмінюватись інформацією через мережу. Фальсифікувати облікові дані цифрового сертифіката, видані центром сертифікації, дуже важко, адже цифровий сертифікат підписується особистим ключем центру сертифікації, який відомий лише йому. Цифровий підпис забезпечує цілісність, автентичність та неспростовність відповідного сертифікату.

ІВК є комплексною системою, яка має раціональну структуру та широкий набір функцій, які спрощують процедуру автентифікації та забезпечують її справжність на підставі цифрових сертифікатів. Суб'єкти комунікації можуть повністю не довіряти один одному, але довіряти третій незалежній стороні, яка регулює механізм встановлення довіри між ними. Цей механізм базується на використанні цифрових сертифікатів і криптографії з відкритим ключем та є важливим елементом для забезпечення безпеки та конфіденційності інформації в Інтернеті та інших цифрових середовищах.

ІВК широко застосовується для проведення безпечних електронних транзакцій, банківських операцій, цифровізації та трансформації уряду, державних установ та організацій задля підвищення рівня якості надання послуг та організації комунікації між суспільством та державними органами. Міжнародна спільнота розгортає та модернізує ІВК у вигляді надійного механізму для забезпечення процесу обміну інформацією та комунікації.

ДОДАТОК Б

ТЕСТУВАННЯ РОЗРОБЛЕНОГО ВЕБ-ЗАСТОСУНКУ

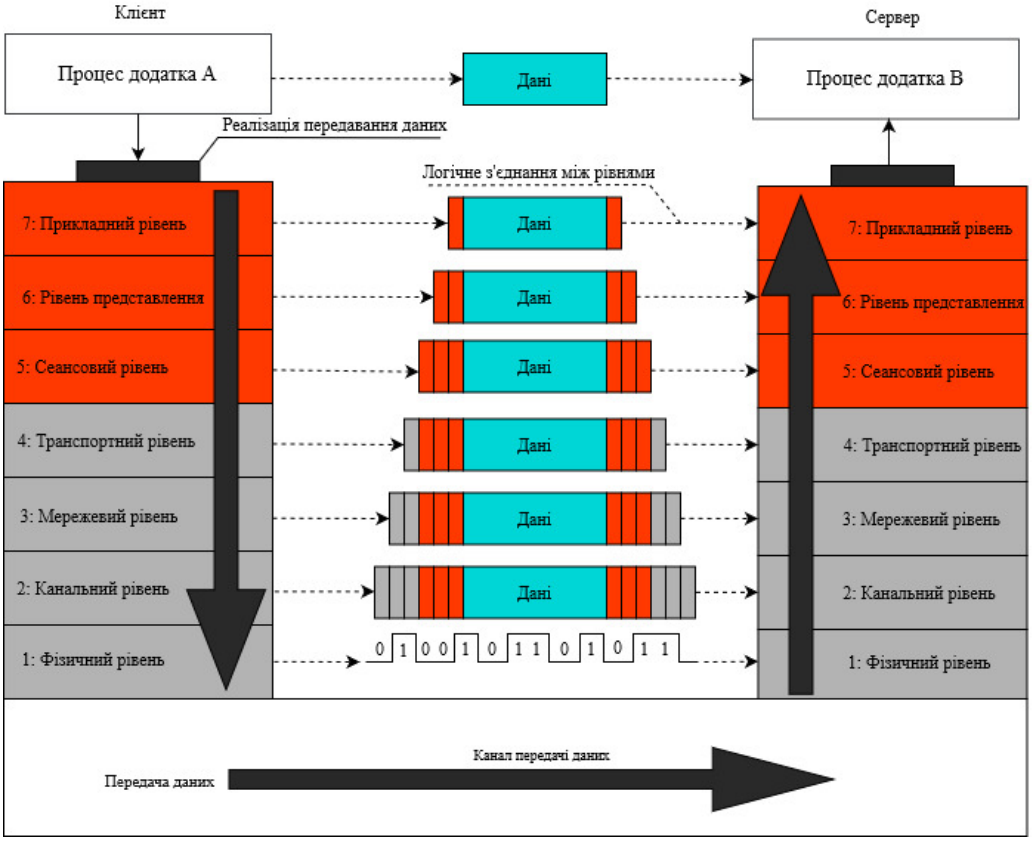


Рисунок Б.1 – Концептуальна модель розробленої системи передачі

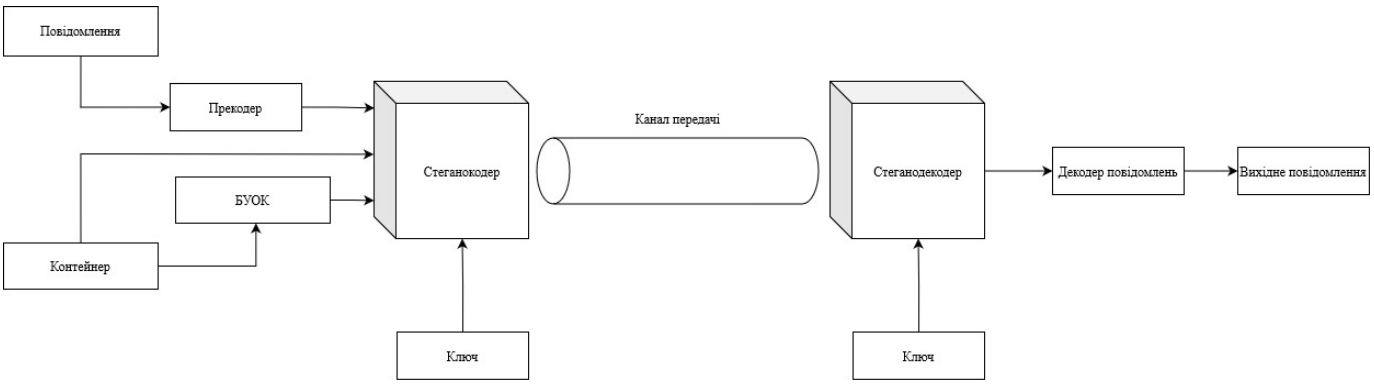


Рисунок Б.2 – Узагальнена структурна схема стеганографічних модулів

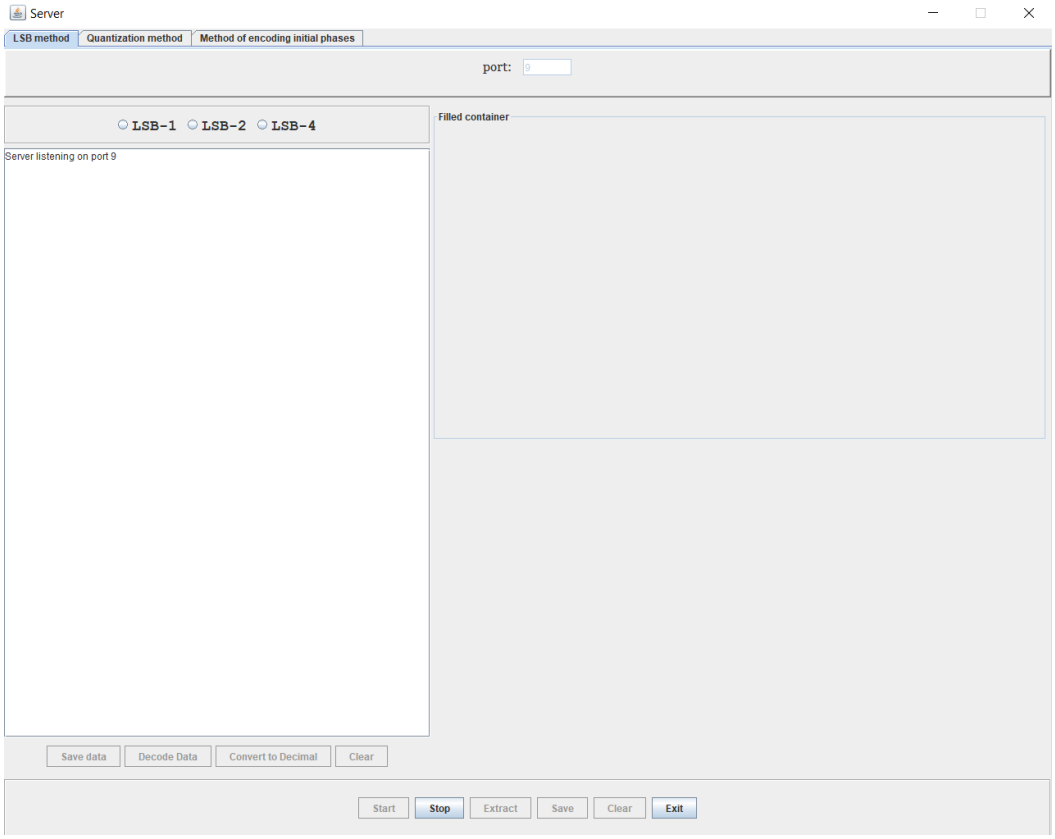


Рисунок Б.3 – Відкриття з’єднання на 9/tcp порті зі сторони сервера

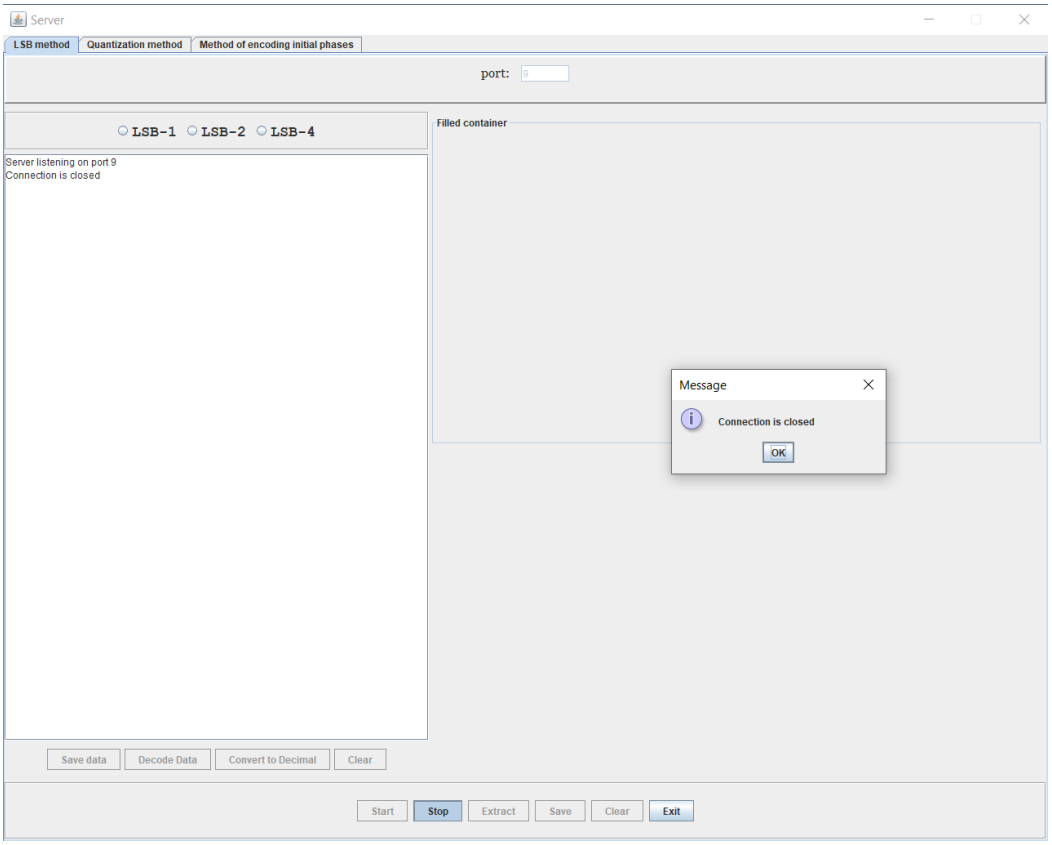


Рисунок Б.4 – Закриття з’єднання

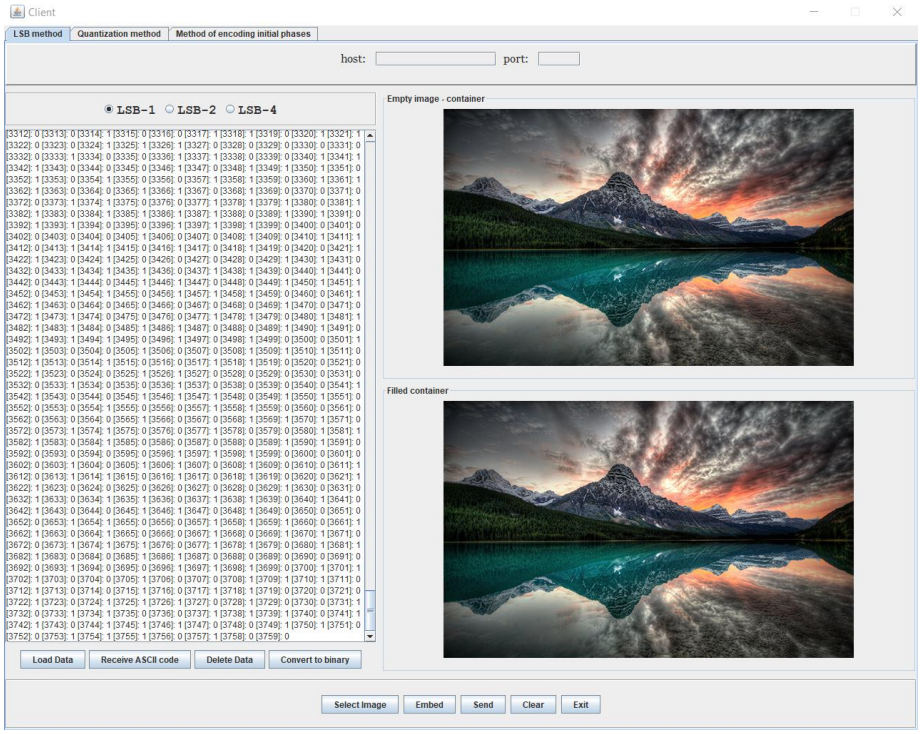


Рисунок Б.5 – Стеганографічне монтування інформації за допомогою методу LSB

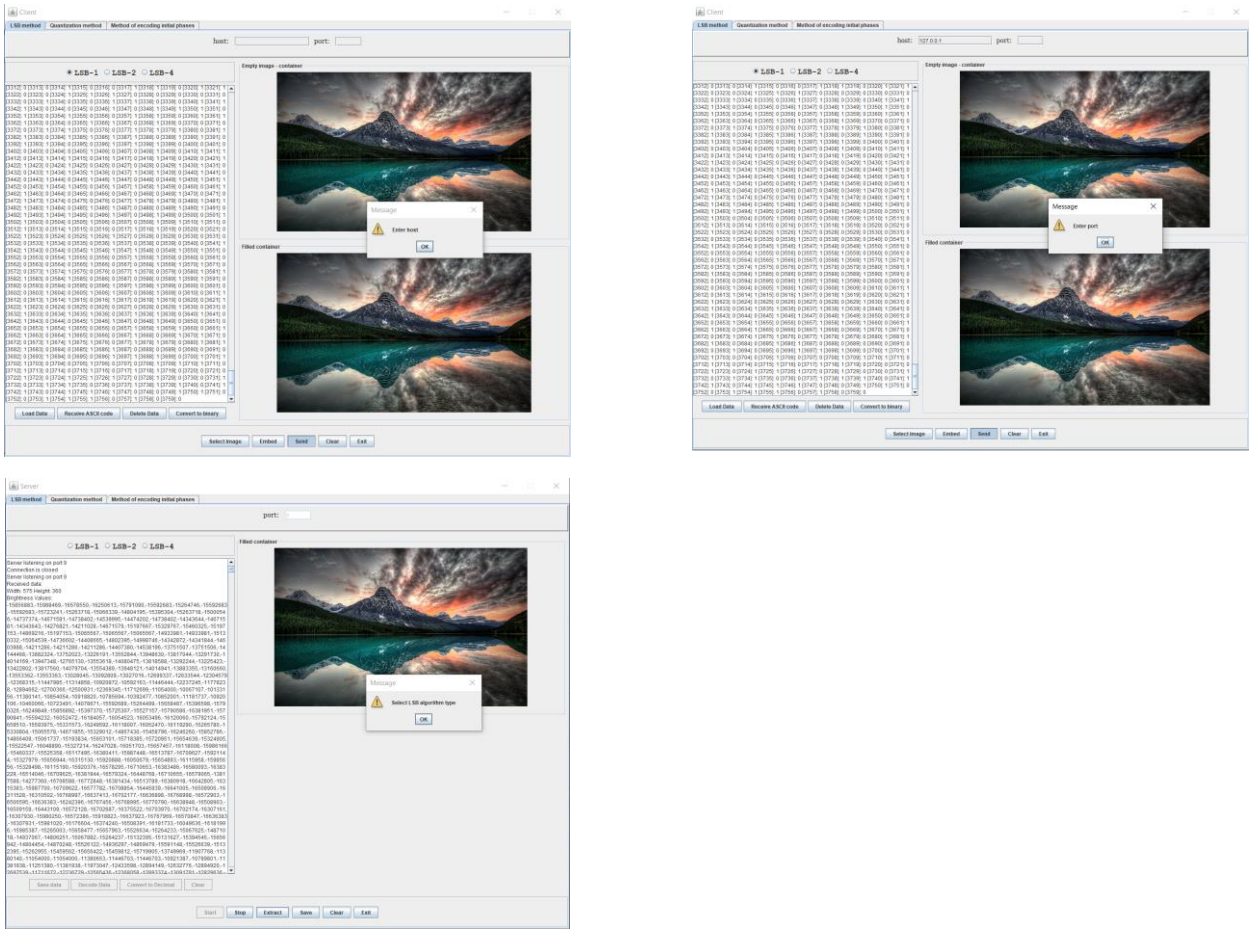


Рисунок Б.6 – Реакція системи на пропущення обов’язкових для заповнення полів

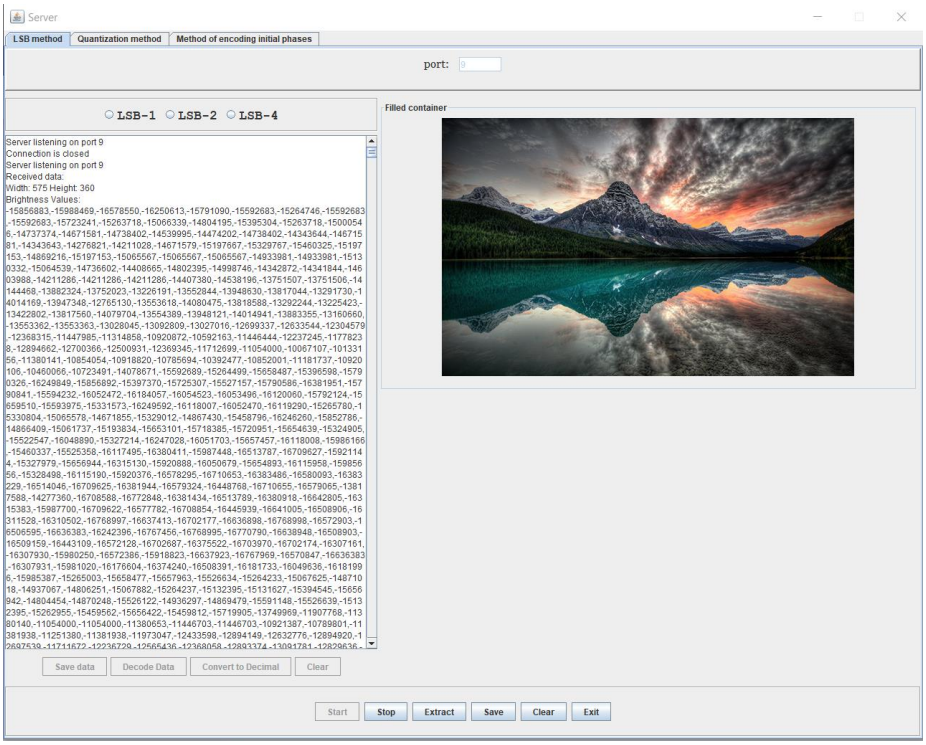


Рисунок Б.7 – Верифікація справності функціонування системи передачі

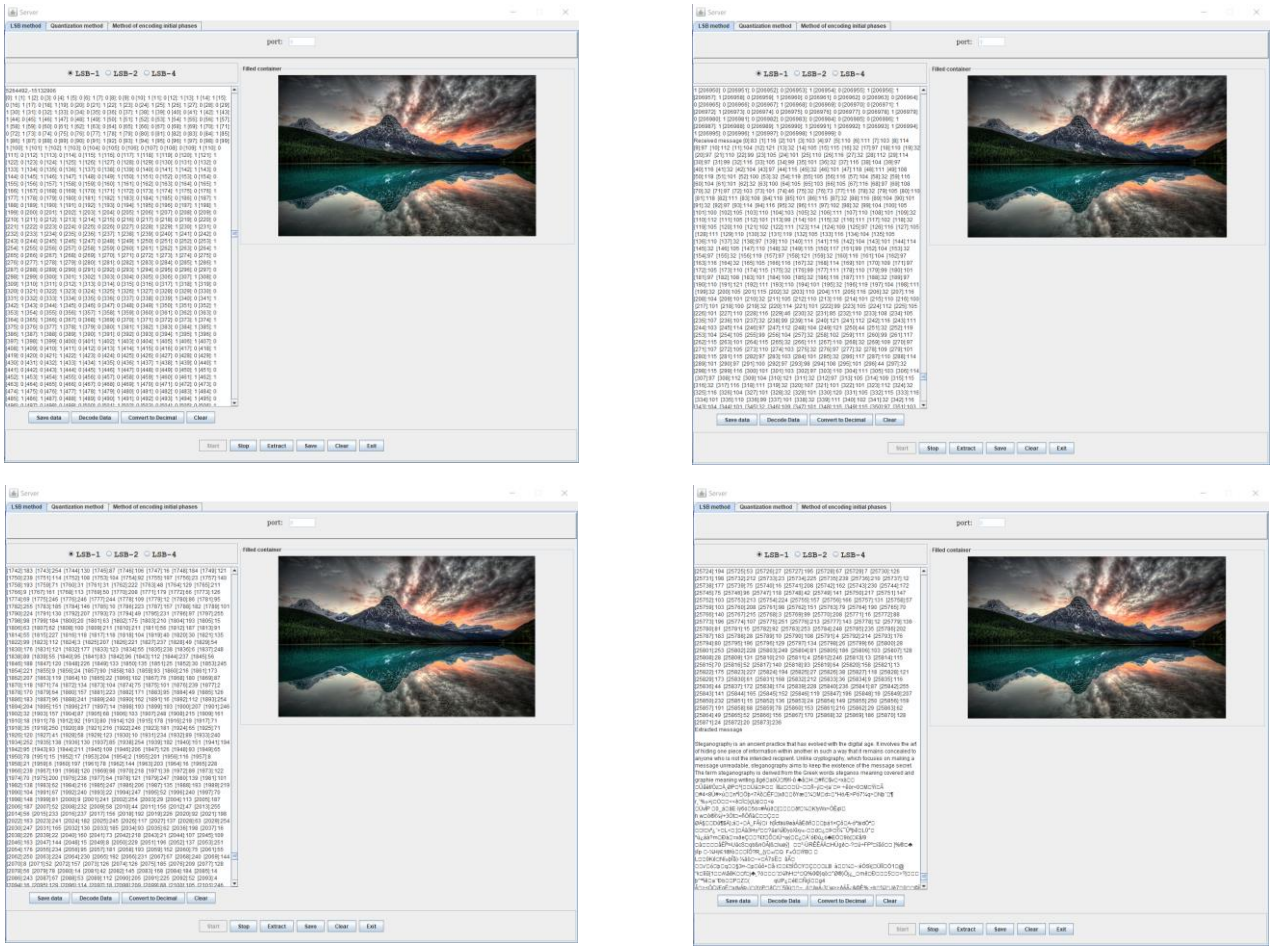


Рисунок Б.8 – Вилучення та декодування інформаційних повідомлень

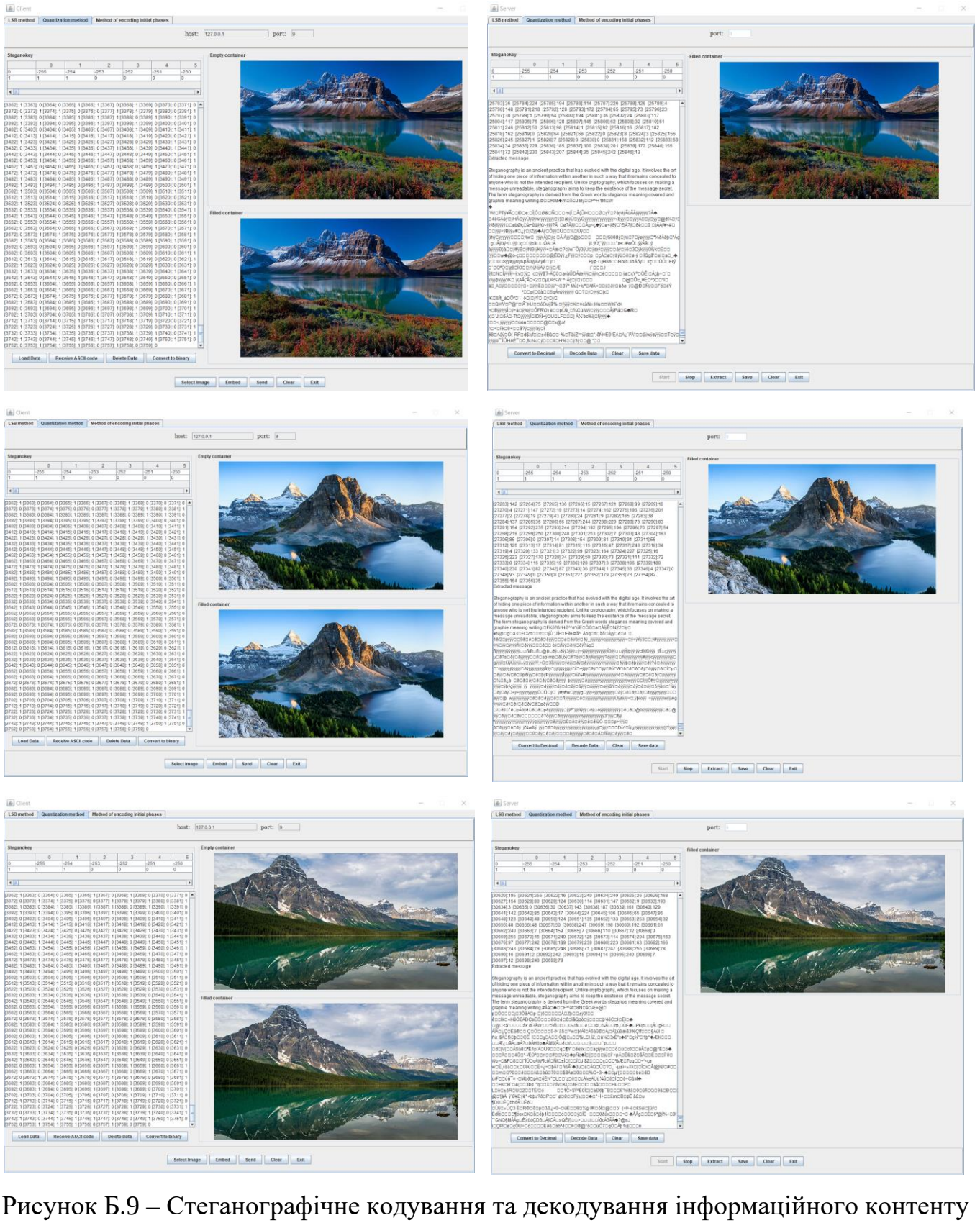


Рисунок Б.9 – Стеганографічне кодування та декодування інформаційного контенту за допомогою алгоритму квантування

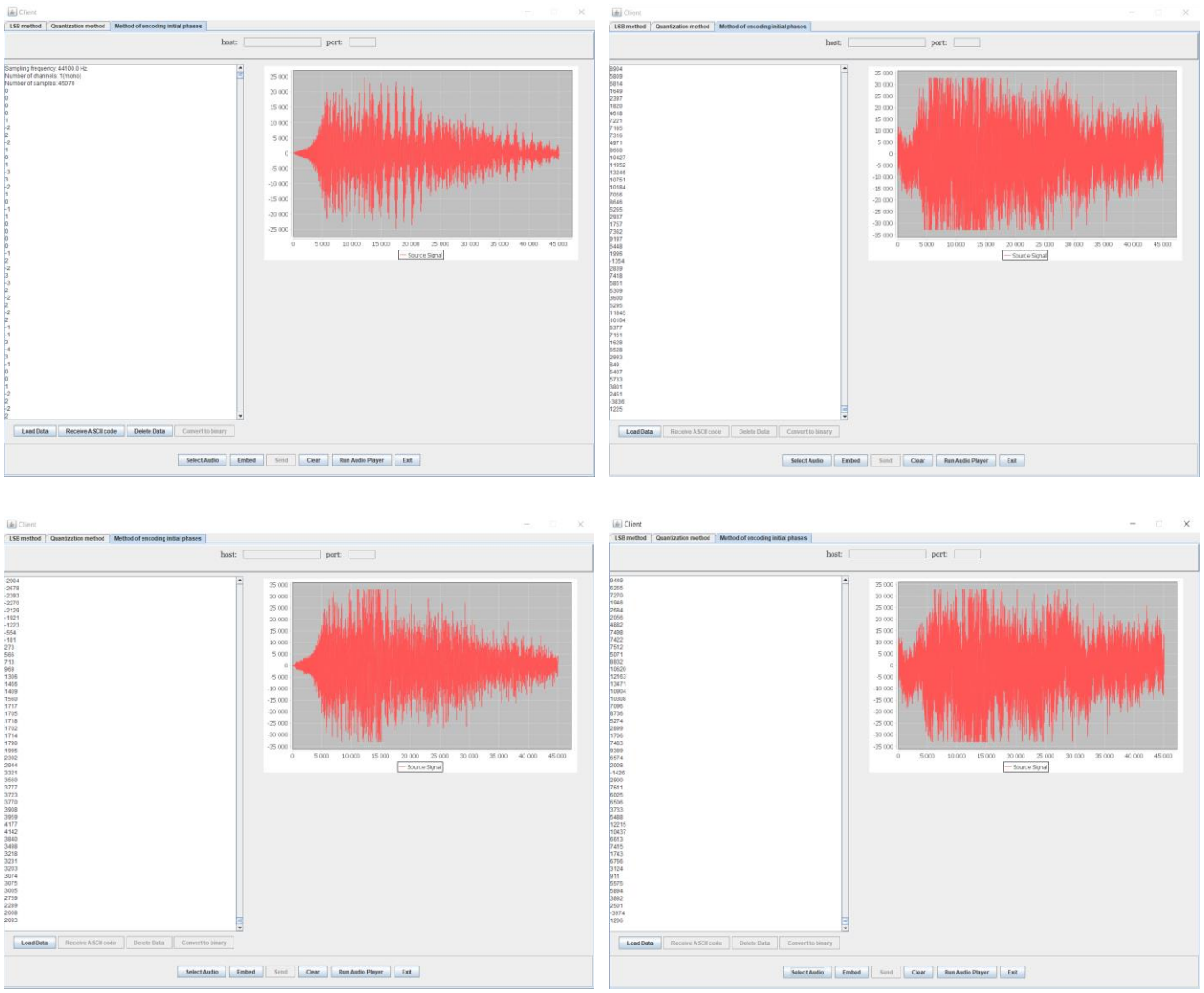


Рисунок Б.10 – Валідація модулю читання аудіопотоків

Рисунок Б.11 – Результат транспортування даних аудіосигналу через мережу

Рисунок Б.12 – Стеганографічне монтування та демонтування методом кодування початкових фаз

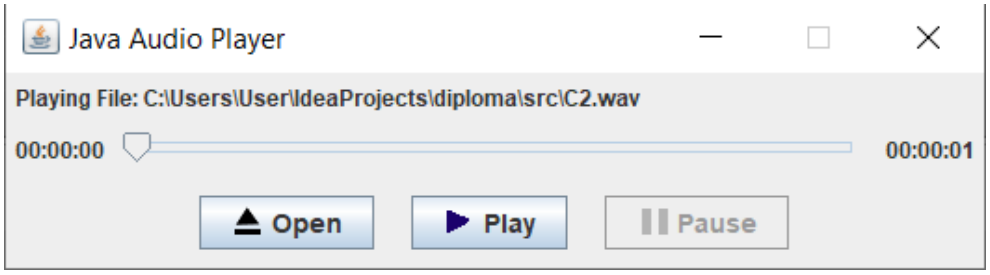


Рисунок Б.13 – Інтерфейс програвача аудіотреків

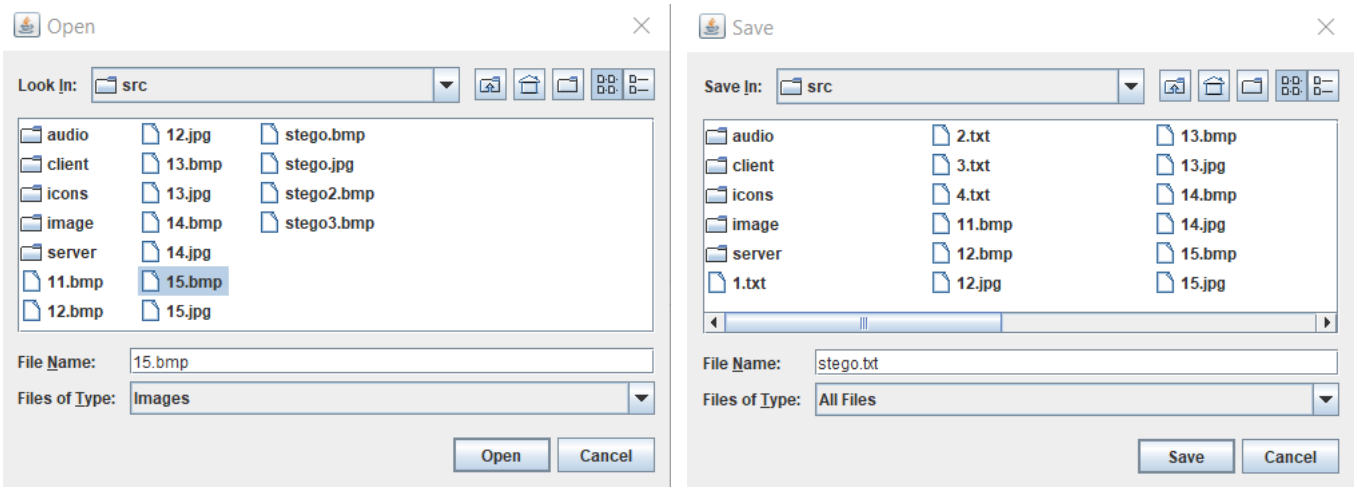


Рисунок Б.14 – Інструментарій веб-застосунку для роботи з файлами

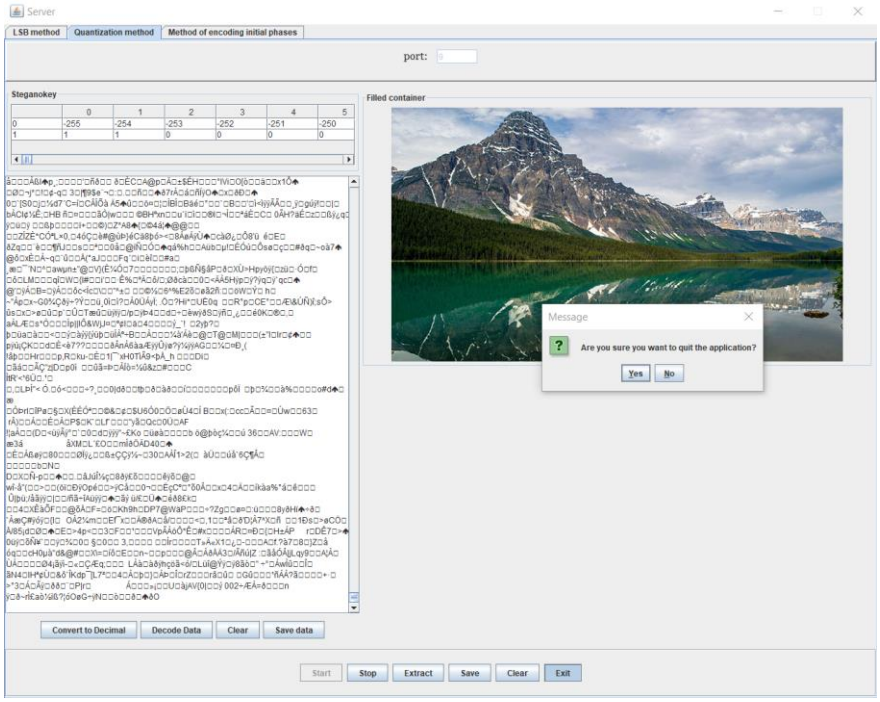


Рисунок Б.15 – Функція завершення роботи веб-застосунку

ДОДАТОК В

ФРАГМЕНТИ ПРОГРАМНОГО КОДУ РОЗРОБЛЕНОГО ВЕБ-ЗАСТОСУНКУ

Лістинг В.1 – Модуль перетворення числа з десяткового до бінарного виду

```
 public static int[] D_B(int x) {
        int V[]=new int[8];
        for (int i=0;i<8;i++){
            V[i]=x % 2;
            x = (int)Math.floor(x/2);
        }
        return V;
    }
```

Лістинг В.2 – Модуль перетворення числа з бінарного до десяткового виду

```
public static int B_D(int[] x) {
        int sum=0;
        for (int i=0;i<8;i++){
            sum += x[i]*Math.pow(2,i);
        }
        return sum;
    }
```

Лістинг В.3 – Функція демонтування даних методом LSB

```
public int[] extract(){
        ArrayList<Integer> M_b1 = new ArrayList<Integer>();
        int V[];
        for (int i=0;i< redChannel.length;i++){
            for (int j=0;j<redChannel[i].length;j++){
                V=D_B(redChannel[i][j]);
                M_b1.add(i* redChannel[i].length+j,V[bit]);
            }
        }
        int M_b11[] = new int[M_b1.size()];
        for (int i=0;i<M_b11.length;i++){
            M_b11[i] = M_b1.get(i);
        }
        return M_b11;
    }
```

Лістинг В.4 – Функція інкапсуляції даних в просторову область нерухомих зображень за допомогою методу квантування

```
public int[][] embed(){
        int S1[][] = new int[R.length][R[0].length];
        int b=0;
        int j1=0;
        for (int i=0;i<R.length;i++){
```

```
            for (int j=0;j<R[0].length;j++){
                S1[i][j] = R[i][j];
            }
        }
        for (int j=0;j<S1[0].length-1;j++) {
            for (int i = 0; i < S1.length; i++) {
                if ((S1.length*j+i) >= M_b.length){
                    break;
                }
                b = S1[i][j+1]-S1[i][j];
                if (M_b[j*S1.length+i] == d[1][b+255]){
                    continue;
                }
                if (M_b[j*S1.length+i] != d[1][b+255]) {
                    j1 = 1;
                    while (M_b[j*S1.length+i] != d[1][b+255+j1] && j1 <
509) {
                        j1 = j1 + 1;
                    }
                    S1[i][j+1] = S1[i][j+1] + d[0][b+255+j1]-b;
                }
            }
        }
        return S1;
    }
```

Лістинг В.5 – Функція декапсуляції даних методом квантування

```
public int[] extract(){
        int[] M_b2 = new int[R.length*R[0].length-R.length];
        int b=0;
        for (int j=0;j<R[0].length-1;j++){
            for (int i=0;i<R.length;i++){
              b = R[i][j+1]-R[i][j];
              M_b2[j*R.length+i] = d[1][b+255];
           }
         }
        return M_b2;
    }
```

Лістинг В.6 – Модуль генерації стегоключа – таблиці квантування

```
public static int[][] generateD(){
        Random random = new Random(SEED);
        d = new int[2][511];
        for (int i=0;i<511;i++){
            d[0][i] = i-255;
            d[1][i] = random.nextInt(2);
        }
        return d;
    }
```

Лістинг В.7 – Модуль перетворення матриць растрових даних в рядок

```
    private static String convertBrightnessMatrixToString(int[][]
brightnessMatrix) {
            StringBuilder data = new StringBuilder();
            for (int i = 0; i < brightnessMatrix.length; i++) {
                for (int j = 0; j < brightnessMatrix[0].length; j++) {
                    data.append(brightnessMatrix[i][j]);
                    if (i < brightnessMatrix.length - 1 || j <
brightnessMatrix[i].length - 1) {
                        data.append(",");
                    }
                }
            }
            return data.toString();
        }
```

Лістинг В.8 – Модуль перетворення масиву амплітуд аудіопотоку в рядок

```
private static String getStrSamples(short[] samples){
        StringBuilder data = new StringBuilder();
        for (int i=0;i<samples.length;i++){
            data.append(samples[i]);
            if (i!=samples.length-1) data.append(",");
        }
        return data.toString();
    }
```

Лістинг В.9 – Вихідний код модуля вибору текстових файлів

```
private static String chooseFile(JTextArea textAreaPointer) {
        String str = null;
        JFileChooser fileChooser = new JFileChooser();
        fileChooser.setFileSelectionMode(JFileChooser.FILES_ONLY);

        FileNameExtensionFilter textFilter = new
FileNameExtensionFilter("Text files", "txt");
        fileChooser.addChoosableFileFilter(textFilter);

        int response = fileChooser.showOpenDialog(null);
        if (response == JFileChooser.APPROVE_OPTION) {
            File selectedFile = fileChooser.getSelectedFile();
            StringBuilder contentBuilder = new StringBuilder();
            try {
                BufferedReader reader = new BufferedReader(new
FileReader(selectedFile));
                String line;
                while ((line = reader.readLine()) != null) {
                    contentBuilder.append(line);
                }
                reader.close();
            } catch (IOException i) {
                i.printStackTrace();
            }
```

```
            str = contentBuilder.toString();
            textAreaPointer.append(str);
            textAreaPointer.append("\n");
        }
        return str;
    }
```

Лістинг В.10 – Модуль вибору цифрового зображення-контейнеру

```
private static BufferedImage selectImageFunc(JLabel iconLabel) {
        BufferedImage image = null;
        JFileChooser fileChooser = new JFileChooser();
        fileChooser.setFileSelectionMode(JFileChooser.FILES_ONLY);
        fileChooser.addChoosableFileFilter(new
FileNameExtensionFilter("Images", "jpg", "png", "gif", "bmp"));
        int response = fileChooser.showOpenDialog(null);
        if (response == JFileChooser.APPROVE_OPTION) {
            File file = fileChooser.getSelectedFile();
            try {
                image = ImageIO.read(file);
            } catch (IOException ioException) {
                ioException.printStackTrace();
            }
            if (image != null) {
                iconLabel.setIcon(new ImageIcon(image));
            }
        } else {
            JOptionPane.showMessageDialog(null, "Could not open the
file", "Message", JOptionPane.INFORMATION_MESSAGE);
        }
        return image;
    }
```

Лістинг В.11 – Модуль для побудови графічного представлення аудіосигналу

```
private static void buildSignal(XYSeriesCollection dataset, short[]
audioSamples, int numSamples, String xLabel){
    XYSeries seriesSignal = new XYSeries(xLabel);
    for (int i = 0; i < numSamples; i++) {
        seriesSignal.add(i, audioSamples[i]);
    }
    dataset.addSeries(seriesSignal);
}
```

ДОДАТОК Г

ДОСЛІДЖЕННЯ СИСТЕМИ ПЕРЕДАЧІ

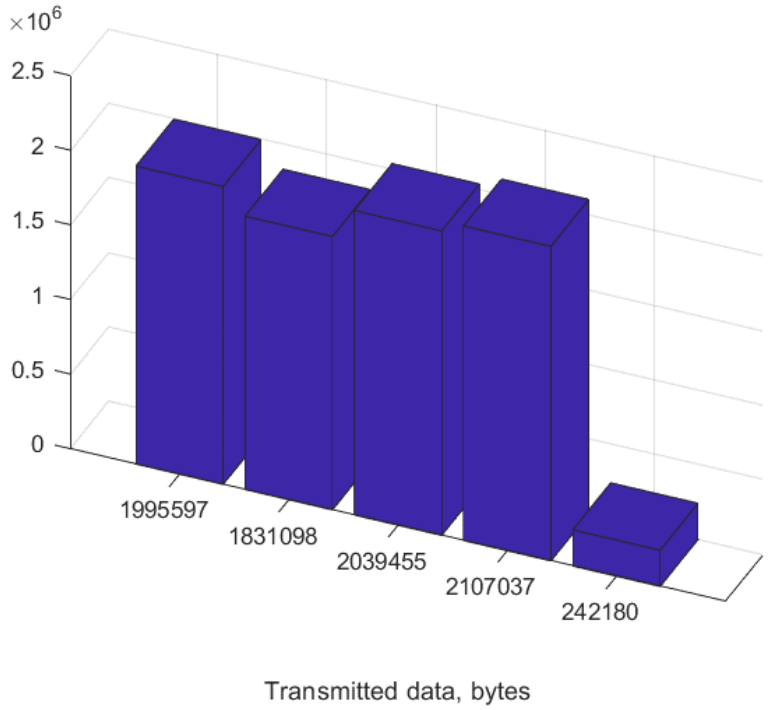


Рисунок Г.1 – Гістограма переданих даних по КЗ

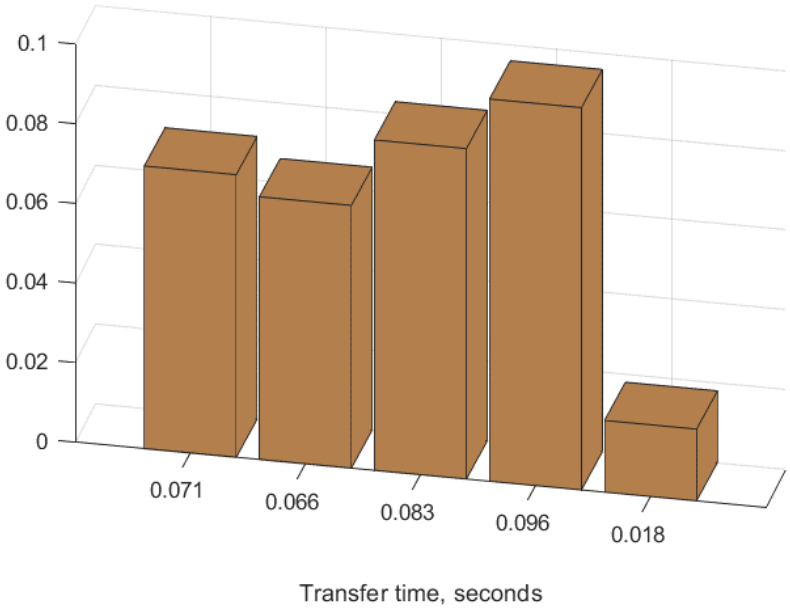


Рисунок Г.2 – Гістограма продуктивності передачі

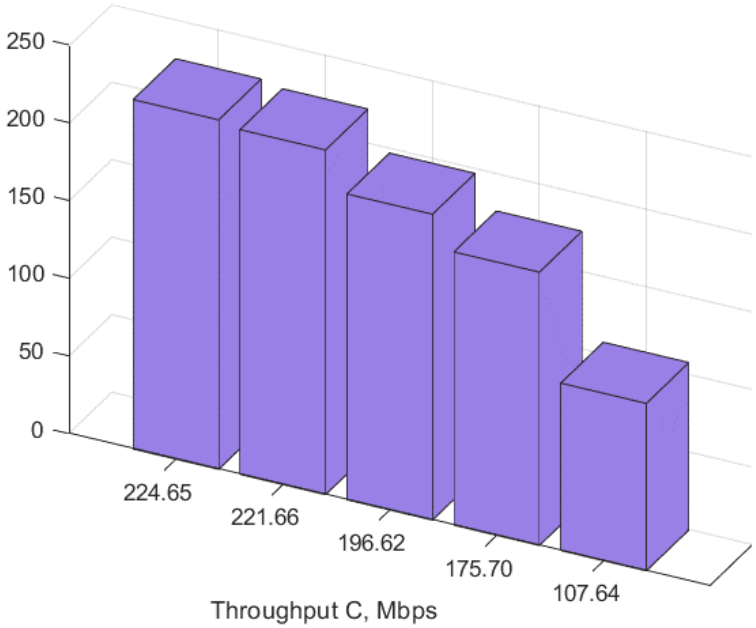


Рисунок Г.3 – Гістограма пропускної здатності