Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)
Кафедра кібербезпеки
(повна назва кафедри)

КВАЛІФІКАЦІЙНА РОБОТА

на здобуття освітнього ступеня

бакалавр
(назва освітнього ступеня)

на тему: Методи стеганографічного захисту і стеганоаналізу інформації з використанням аудіофайлів

Виконав: студент IV курсу, групи СБс-42
спеціальності 125 Кібербезпека
(шифр і назва спеціальності)

(підпис) Нога О.В. (прізвище та ініціали)

Керівник (підпис) Золотий Р.З. (прізвище та ініціали)

Нормоконтроль (підпис) Лобур Т.С. (прізвище та ініціали)

Завідувач кафедри (підпис) Загородна Н.В. (прізвище та ініціали)

Рецензент (підпис) Приймак М.В. (прізвище та ініціали)

Тернопіль - 2022

Міністерство освіти і науки України
Тернопільський національний технічний університет імені Івана Пулюя

Факультет комп'ютерно-інформаційних систем і програмної інженерії
(повна назва факультету)

Кафедра кібербезпеки
(повна назва кафедри)

ЗАТВЕРДЖУЮ
Завідувач кафедри
_____ Загородна Н.В.
(підпис) (прізвище та ініціали)
«__» _____ 2021 р.

ЗАВДАННЯ
НА КВАЛІФІКАЦІЙНУ РОБОТУ

на здобуття освітнього ступеня Бакалавр
(назва освітнього ступеня)

за спеціальністю 125 Кібербезпека
(шифр і назва спеціальності)

Студенту Нозі Олександру Васильовичу
(прізвище, ім'я, по батькові)

1. Тема роботи Методи стеганографічного захисту і стеганоаналізу інформації з використанням аудіофайлів

Керівник роботи Золотий Роман Захарійович., к.т.н., доц. каф. КТ
(прізвище, ім'я, по батькові, науковий ступінь, вчене звання)

Затверджені наказом ректора від «23» 03 2022 року № 4/7-178

2. Термін подання студентом завершеної роботи 23.06.2022р.

3. Вихідні дані до роботи наукові літературні джерела

4. Зміст роботи (перелік питань, які потрібно розробити)
1. Аналіз предметної області. 1.1. Аналітичний огляд. 1.2. Класифікація методів стеганографії. 1.3. Особливості аудіо файлів для застосування стеганографії. 1.4. Вибір формату аудіо файлу. 1.5. Опис стандарту WAVE. 2. Теоретична частина. 2.1. Методи стеганографії для аудіо файлів. 2.2. Огляд наявного ПЗ для стеганографії в аудіо файлах. 2.3. Існуючі методи та ПЗ для стеганоаналізу в аудіо файлах. 2.4. Опис розробленого підходу стеганоаналізу. 3. Практична частина. 3.1. Розробка додатку для стеганографії та стеганоаналізу для аудіо файлів. 3.2. Тестування БД файлів щодо вкладень за допомогою частотного аналізу. 3.3. Метод СтА аудіо файлів на основі алгоритмів стиснення. 4. Безпека життєдіяльності, основи хорони праці

5. Перелік графічного матеріалу (з точним зазначенням обов'язкових креслень, слайдів)
1. Титулка. 2. Актуальність. 3. Мета, задачі дослідження. 4. Порівняння існуючих аналогів. 5. Загальна схема роботи програми. 6. Клієнтська частина програми. 7 Огляд методів виявлення аномалій. 8,9. Дослідження методів детектування аномалій на реальних даних. 10. Метрики, використані для оцінки. 11. LOF (Local Outlier Factor) 12. Моніторинг БД та переривання запитів. 13. Результати проведеного дослідження

6. Консультанти розділів роботи

Розділ	Прізвище, ініціали та посада консультанта	Підпис, дата	
		завдання видав	завдання прийняв
Безпека життєдіяльності, основи хорони праці	Гурик О.Я., доцент кафедри МТ		

7. Дата видачі завдання 2022 р.

КАЛЕНДАРНИЙ ПЛАН

№ з/п	Назва етапів роботи	Термін виконання етапів роботи	Примітка
1.	Ознайомлення з завданням до кваліфікаційної роботи	23.03 – 26.03	*Виконано*
2.	Підбір джерел про стеганографію та стеганоаналіз		
	на аудіо файлах -	27.03 – 09.04	*Виконано*
3.	Опрацювання джерел про стеганографію та		
	стеганоаналіз на аудіо файлах -	10.04 – 16.04	*Виконано*
4.	Виконання дослідження щодо стеганографії		
	та стеганоаналіу на аудіо файлах	17.04 – 23.04	*Виконано*
5	Розроблення програмного коду	24.04 – 29.04	
6.	Оформлення розділу «Аналіз предметної області»	30.04 – 07.05	*Виконано*
7.	Оформлення розділу «Теоретична частина»	08.05 – 15.05	*Виконано*
8.	Оформлення розділу «Практична частина»	16.05 – 21.05	*Виконано*
9.	Виконання завдання до підрозділу «Безпека		
	життєдіяльності, основи хорони праці»	14.05 – 21.05	*Виконано*
10.	Оформлення кваліфікаційної роботи	22.05 – 05.06	*Виконано*
11.	Нормоконтроль	06.06 – 12.06	*Виконано*
12.	Перевірка на плагіат	10.06 – 15.06	*Виконано*
13.	Попередній захист кваліфікаційної роботи	16.06 – 19.06	*Виконано*
14.	Захист кваліфікаційної роботи	24.06	

Студент _____ Нога О.В.
(підпис) (прізвище та ініціали)

Керівник роботи _____ Золотий Р.З.
(підпис) (прізвище та ініціали)

АНОТАЦІЯ

Методи стеганографічного захисту і стеганоаналізу інформації з використанням аудіофайлів // Нога Олександр Васильович // Тернопільський національний технічний університет імені Івана Пулюя, факультет комп'ютерно-інформаційних систем та програмної інженерії, кафедра кібербезпеки, група СБс–42 // Тернопіль, 2022 // С. – 53, рис. – 26, табл. – 2, слайдів – 13, бібліогр. – 39.

Ключові слова: СТЕГАНОГРАФІЯ, СТЕГАНОАНАЛІЗ, АУДІО ФАЙЛИ, WAVE, КОНТЕЙНЕР, LSB, МЕТОДИ СТИСНЕННЯ, СТАТИСТИЧНІ ТЕСТИ

Кваліфікаційна робота присвячена застосуванню методів стеганографії та стеганоаналізу для аудіо файлів.

Розглядається метод стеганоаналізу, заснований на стисканні файлів, а також запропонований новий метод, що базується на частотному аналізі. Запропоновано власну методику для визначення стеговкладень в звукових файлах з використанням пакету NIST. Розроблена методика дозволяє виявляти вкладення, зроблені за допомогою алгоритму LSB, у деякі види аудіо файлів формату WAVE.

Для стеганографії на аудіофайлах та для розглянутих методів стеганоаналізу розроблено оригінальне програмне забезпечення для стеганографічного приховування даних в аудіо файлах формату WAVE, реалізоване у вигляді пакета Wave_Hide_Stego. В програмі реалізовано алгоритм вилучення прихованих даних. Для шифрування даних перед приховуванням застосовується алгоритм XOR.

ANNOTATION

Methods of steganographic security and steganographic analysis of information using audio files // Noha Oleksandr // Ternopil Ivan Pul'uj National Technical University, Faculty of Computer Information Systems and Software Engineering, Department of Cyber Security // Ternopil, 2022 // P. - 53, Fig. - 26, Table - 2, Slides - 13, References - 39.

Keywords: STEGANOGRAPHY, STEGANANALYSIS, AUDIO FILES, WAVE, CONTAINER, LSB, COMPRESSION METHODS, STATISTICAL TESTS

Thesis deals with the use of steganography and stegoanalysis methods for audio files.

The method of stegoanalysis based on file compression is considered, and a new method based on frequency analysis is proposed. We offer our own method for determining attachments in audio files using the NIST package. The developed technique allows to detect attachments made using the LSB algorithm in some types of audio files in WAVE format.

For steganography on audio files and for the considered methods of stegoanalysis the original software for steganographic hiding of data in audio files of the WAVE format developed in the form of the Wave_Hide_Stego package is developed. The program implements an algorithm for extracting hidden data. The XOR algorithm is used to encrypt data before hiding.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ СКОРОЧЕНЬ І ТЕРМІНІВ

LSB (Least Significant Bit) – МЗР (молодший значущий розряд).

PCM (Pulse Code Modulation) – імпульсно-кодова модуляція.

WAVE (Waveform Audio File Format) – формат аудіофайлу.

АЦП – аналого-цифровий перетворювач.

БД – база даних.

Бітність – кількість біт у семплі.

Відсоток заповнення контейнера – частка контейнерного обсягу (для заданого методу вбудовування), зайнята вбудованим повідомленням.

Заповнений контейнер – контейнер, що містить вбудоване повідомлення.

Контейнер – вихідний файл, призначений для приховування даних (повідомлень).

Об'єм контейнера – максимально можлива частина файлу, придатна для вбудовування повідомлення, залежить від методу вбудовування.

ПК – персональний комп’ютер.

ПЗ – програмне забезпечення.

Повідомлення – будь-які дані, призначені для передачі.

Порожній контейнер – вихідний файл без вбудованого повідомлення.

Семпл – сукупність амплітуди і короткого проміжку часу.

СтА (Стеганоаналіз) –наука про виявлення факту передачі прихованої інформації в аналізованому повідомленні (є розділом стеганографії).

СтГр (Стеганографія) – тайнопис, при якому повідомлення, закодоване таким чином, що не виглядає як повідомлення — на відміну від криптографії/

Стегоповідомлення – інформація, що вбудовується у контейнер.

ФК – файл-контейнер.

ФКл – файл-ключ.

ФП – файл-повідомлення.

ЧД – частота дискретизації.

ЗМІСТ

ВСТУП

Основним завданням СтГр не є повна заміна, а властиво доповнення до криптографії. Методи СтГр застосовуються для поміщення секретної інформації в нейтральні повідомлення для унеможливлювання навіть підозри щодо існування якогось вбудованого власне прихованого послання. До СтГр належить множина секретних способів надсилання інформації, наприклад невидимі чорнила, мікрофотознімки, умовне розміщення символів, таємні канали і засоби зв'язку на плаваючих частотах і т. д.

СтГр стала доступнішою для багатьох користувачів та може бути застосована поза закономими цілями, в т.ч. для передачі інформації, яка містить державну таємницю. Тому актуальною є розробка ефективних методів виявлення прихованих даних у мультимедійних файлах, котрі передаються з використанням комп’ютерних мереж.

СтА напряму пов’язаний із передачею прихованої інформації [1]. Робляться спроби розв'язання задач СтА, зокрема, для графічних файлів [2]. Однак, загальних підходів для виявлення прихованих вкладень для всіх типів файлів поки що не створено.

Метою роботи є дослідження методів СтГр і СтА та створення ПЗ для приховування даних у аудіо файлах на їх основі.

В процесі виконання роботи потрібно вирішити наступні завдання:

– розглянути особливості приховування та виявлення прихованих вкладень в аудіо файлах;

– виконати огляд існуючого ПЗ для приховування інформації у файлах WAVE;

– запропонувати методи виявлення прихованих даних в аудіо файлах;

– провести порівняльний аналіз реалізованого алгоритму СтА та відомого методу, котрий базується на алгоритмі стиснення;

– створити програмний додаток для приховування даних у WAVE файлах.

1 АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Аналітичний огляд

Використання методів СтГр значно зменшує ймовірність визначення власне факту як такої передачі повідомлення. За умови, що воно ще й зашифровано, тоді таке повідомлення володіє ще одним, додатковим, рівнем захисту [3].

СтГр бере свій початок у Стародавній Греції. Перші згадки про реалізацію прихованого зв'язку було знайдено у літописі Геродота, написаної V столітті до нашої ери. У ньому було описано метод непомітної передачі послань, що застосовувався у війні між Персією та Грецією. Сам метод полягав у наступному. У давнину для листа використовувалися дощечки, покриті воском. Для прихованої передачі повідомлення з дощечки зіскоблювався віск, текст писався відразу на дереві, потім віск наносився знову [4].

СтГр не є новим термін. Його існування можна прослідкувати десь з 1500 року приблизно.

Відносно нещодавно з'явився новий напрямок СтГр - комп'ютерна (або цифрова) СтГр, яка спрямована на вбудовування повідомлень у файлів різних типів, як то текстові, графічні, аудіо, відео та ін. У зв'язку зі зростанням ролі глобальних мереж цифрова СтГр набуває великої значущості. Аналіз Internet-джерел дозволяє зробити висновок [5], що зараз цифрова СтГр використовується для наступного:

– прихована передача повідомлень для різних цілей;
– захист приватних даних від НСД;
– боротьба із системами моніторингу та керування ресурсами мережі;
– камуфлювання ПЗ;
– збереження авторського права на визначені види самої інтелектуальної власності.

В даний час розробляються нові методи комп'ютерної СтГр, що базуються на особливостях представлення інформації в цифровому вигляді [6].

Частина цих методів використовує модифікацію палітри, неточність пристроїв оцифровки, надмірність аудіо та відео файлів та ін. підходи [7].

Незважаючи на бурхливий розвиток стеганографічних методів, у вільному доступі є недостатньо ПЗ для СтГр в аудіо файлах. Проблема пов'язана з тим, що методи вкладення інформації в аудіо-файли різних бітностей дещо різні. В даний час немає універсальних програмних рішень для роботи з аудіо файлами різних бітностей. Цифрова СтГр може використовуватись і для кримінальних цілей. Наприклад, у посібнику [8] є глава, присвячена СтГр.

1.2 Класифікація методів СтГр

При опрацюванні літературних джерел зустрічаються дещо відмінні варіанти такої класифікації, зокрема в [9] автори пропонують поділити СтГр на технологічну та інформаційну (рис. 1.1).

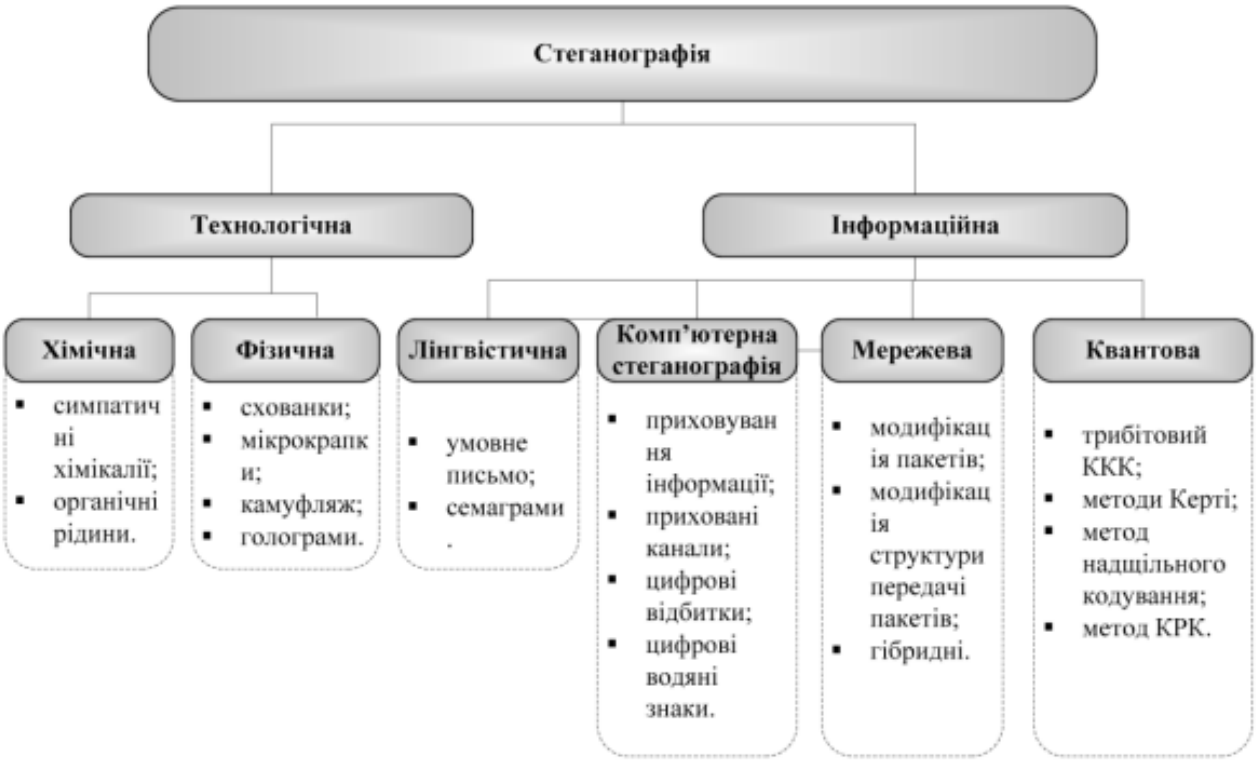

Рисунок 1.1 – Класифікація СтГр

Досить часто використовується поділ СтГр на чотири складові [10]:

– класичний;

– цифровий;

– мовний (лінгвістичний);

– квантовий.

При аналізі літературних джерел, зокрема [11], можна чітко виокремити ще один варіант СтГр – СтГр мережі, тут мережні протоколи моделі OSI застосовуються як приховані носії інформації.

Затримки між пакетами в якомусь цифровому потоці завжди будуть дещо різними. Якщо окремі пакети штучно "пригальмовувати", тоді з’явиться можливість передати успішно якесь приховане повідомлення (рис. 1.2).

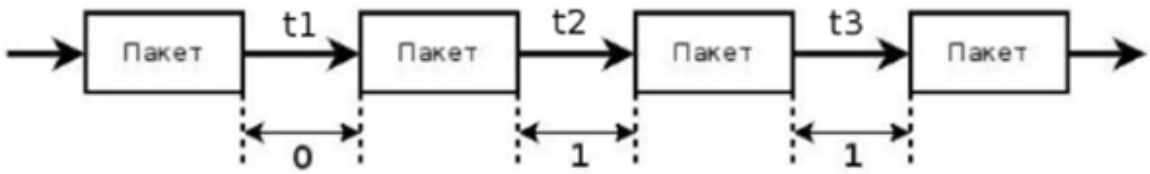

Рисунок 1.2 - Принцип мережевої тимчасової СтГр

("0" - коротка затримка, "1" - довга затримка)

1.3 Особливості аудіо файлів для застосування СтГр

Файли з оцифрованим звуком [12] – це аудіофайли, в котрих аналоговий вихідний сигнал формується з набору перервних величин амплітуд власне звукового сигналу, які вибрані через однакові часові проміжки. Цей варіант запису - PCM [13, 14]. Є таким, що широко застосовується для збереження та передачі цифрового звуку в нестиснутій формі. Достатньо багато програмних додатків мають здатність зчитувати та формувати такі файли. PCM застосовується в звукозаписах у спеціалізованому форматі WAV [6].

Варто згадати і про файли, в котрі вміщені ноти. Це аудіо- файли з набором команд для вказівки відтворюваному обладнанню, яку власне ноту чи

який власне інструмент і як довго відтворювати в визначений проміжок часу (наприклад, файли формату MIDI).

Властиво нестиснутий файл WAVE PCM має послідовні ланцюжки величин дискретизації аналогового звукового сигналу. Якість цього сигналу, котрий збережений в такому форматі, визначається властиво ЧД та рівнем квантування. ЧД показує, яку кількість разів за одну секунду аналоговий сигнал піддається оцифровуванню. Чим ця кількість більша, тим краща звукова якість. Файли WAV можуть бути, наприклад, з ЧД 8000, 11025, 22050, 44100 Гц і т .д.

Поміщення даних в аудіофайли потрібне для захисту авторських прав, інтелектуальної власності, захисту від копіювання та несанкціонованого зберігання, використання, та передачі інформації. Варто зауважити, процес приховування даних в аудіо- сигналі має кілька проблемних моментів, в т.ч. через більш широкий діапазон системи слуху людини у порівнянні із будь-якими іншими відчуттями [15]. Людини здатна вловлювати зміни потужності на $1/10^{12}$ частку і зміни частоти на одну $1/10^{3}$. Проте існують окремі варіанти для вміщення інформації навіть в цій властиво делікатній галузі. Зокрема, вбудовування інформації як шуму [16] є таким варіантом поміщення даних в аудіосигнал.

Зараз досить велике число робочих програмних засобів використовують лише простий LSB для реалізації звукових даних у вигляді контейнерів.

1.4 Вибір формату аудіо файлу

Власне WAVE взято, оскільки він абсолютно влаштовує для реалізації алгоритму LSB через надмірність. В області даних звукових файлів WAVE-формату містяться нестиснені і ніяк не піддані змії дані, котрі взяті безпосередньо з АЦП. Саме тому стеганографічні алгоритми на файлах даного типу реалізовувати дещо простіше і зрозуміліше [17].

Оскільки WAVE- файли є досить великими за своїм об’ємом даних, вони не використовуються для обміну в Інтернеті і для зберігання музики на портативних пристроях (плеєри, мобільні телефони). WAVE- файли

використовуються там, де необхідно зберегти первісний вид файлу високої якості і там, де немає обмеження розміру вільного простору на диску. Наприклад, вони використовуються на студіях звукозапису в програмах, котрі спеціалізуються на редагуванні аудіо, заощаджуючи таким чином час на стисканні, а також розпаковуванні даних.

Надалі під аудіо файлом розумітиметься лише файл формату WAVE.

1.5 Опис стандарту WAVE

Створений інженерами Microsoft та Intel у серпні 1991 року. Він розроблявся як стандартний формат зберігання звукових даних в операційній системі Windows 3.1 [18].

1.5.1 Структура файлу WAVE

Він складається із заголовка, в якому описано формат і всі характеристики аудіо файлу, і безпосередньо області звукових даних (рис. 1.3) [19].

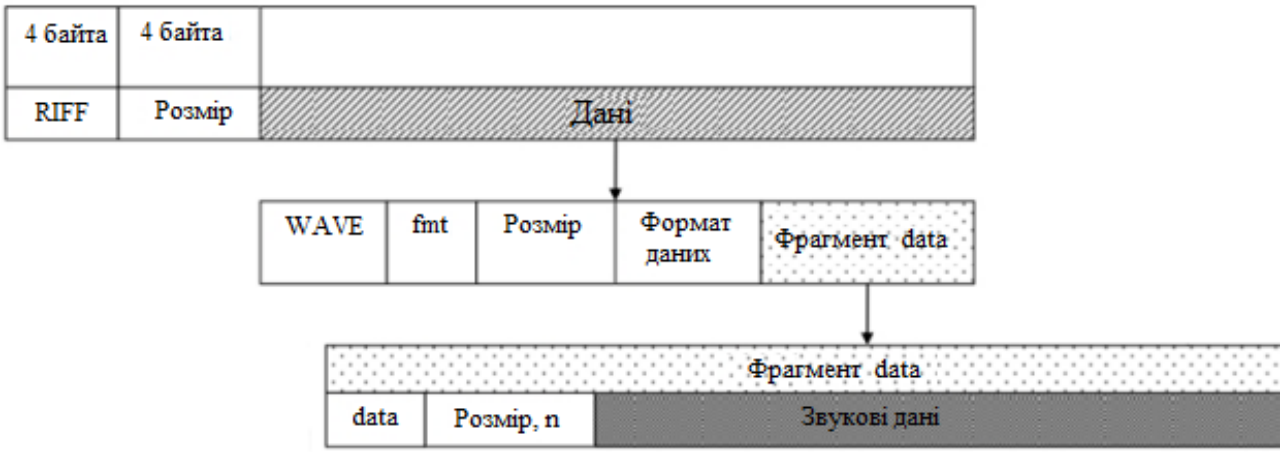

Рисунок 1.3 – Структура WAVE- файлу

1.5.2 Заголовок WAVE- файлу

В таблиці 1.1 наведено заголовок та опис полів [20] (тут PCM - це імпульсно-кодова модуляція [21]).

Таблиця 1.1 - Заголовок та опис полів WAVE- файлу

Розташування	Поле	Опис
0..3 (4 байти)	chunkId	Містить символи “RIFF” в ASCII кодуванні (0x52494646 у big-endian представленні). Є початком RIFF -ланцюжка.
4..7 (4 байти)	chunkSize	Це розмір ланцюжка, що залишився, починаючи з цієї позиції. Інакше кажучи, це розмір файлу - 8, тобто виключені поля chunkId і chunkSize
8..11 (4 байти)	format	Містить символи “WAVE” (0x57415645 в big-endian представленні)
12..15 (4 байти)	subchunk1Id	Містить символи “fmt “ (0x666d7420 у big-endian представленні)
16..19 (4 байти)	subchunk1Size	16 для формату PCM . Це розмір підланцюжка, що залишився, починаючи з цієї позиції.
20..21 (2 байти)	audioFormat	Аудіо формат. Для PCM = 1 (тобто Лінійне квантування). Значення, що відрізняються від 1, позначають певний формат
23 (2 байти)	numChannels	Кількість каналів. Моно = 1, стерео = 2 і т.д.
27 (4 байти)	sampleRate	Частота дискретизації. 8000 Гц, 44 100 Гц і т.д.
31 (4 байти)	byteRate	Кількість байт, надісланих за секунду відтворення
33 (2 байти)	blockAlign	Кількість байт для одного семпла, включаючи всі канали.
34..35 (2 байти)	bitsPerSample	Кількість біт у семпле. 8 біт, 16 біт тощо.
36..39 (4 байти)	subchunk2Id	Містить символи “data” (0x64617461 у big-endian представленні)
40..43 (4 байти)	subchunk2Size	Кількість байт у сфері даних.
44..	data	Безпосередньо WAVE -дані.

1.5.3 Приклад читання заголовка з конкретного файлу WAVE

Для прикладу описаний файл WhiteNoise.wav, що являє собою білий шум (рис. 1.4).

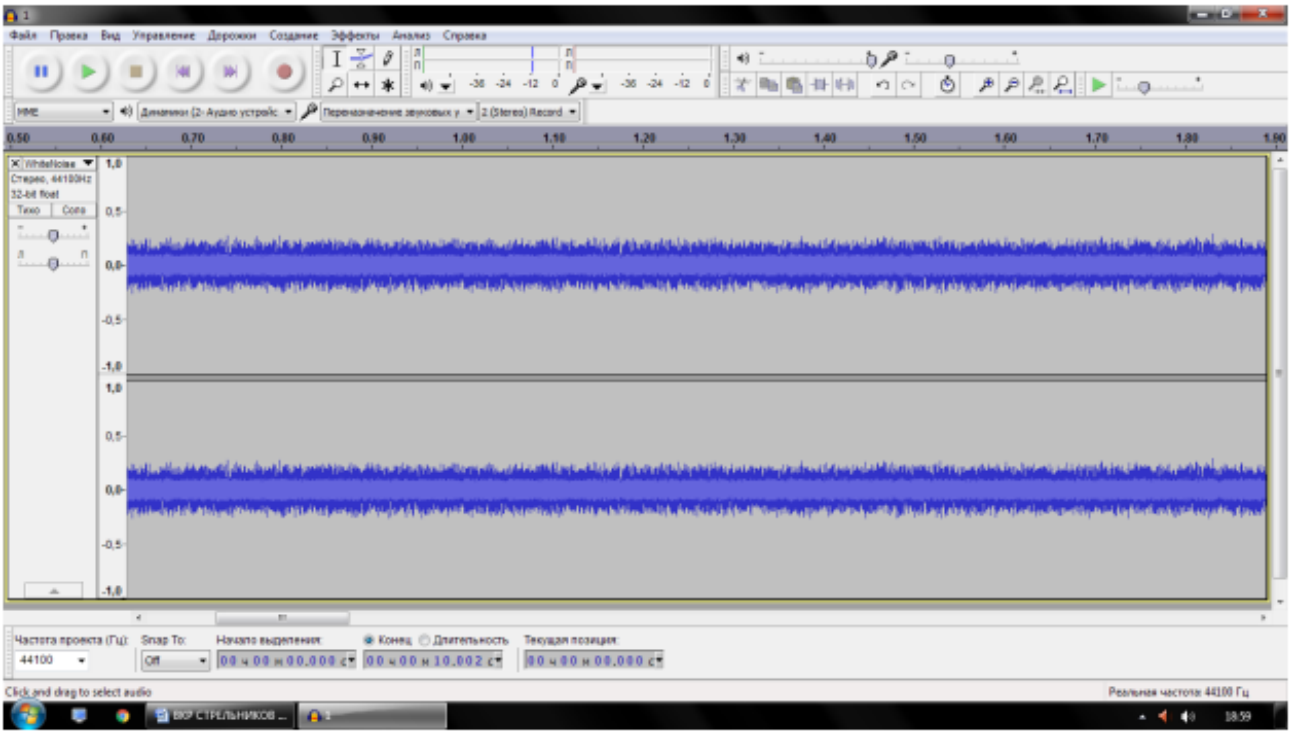

Рисунок 1.4 – Осцилограма файлу "WhiteNoise.wav"

Таблиця 1.2 – Позначення полів файлу з білим шумом

Розташування	Опис
0..3	символи 'R' 'I' 'F' 'F'
4..7	2538452, розмір файлу
8..15	символи 'W' 'A' 'V' 'E' 'f 'm' 't' ".
16..19	вміщено число 16, що вказує на те, що це PCM.
20..21	число 3, PCM зі стисненням
22..23	число 2 (кількість каналів)
24..27	число 44100 – ЧД
28..31	число 352800 – кількість байт, переданих за секунду
32..33	число 8 - кількість байт для одного семпла, включаючи всі канали
34..35	число 32 - кількість біт у семплі
36..39	символи 'd' 'a' 't' 'a'
40..43	число 2538384 - кількість байт у області даних

1.5.4 Визначення області вбудовування повідомлень

Для завдання СтГр та СтА фрагмент data представляє аналоговий сигнал. Набір значень фрагмента data і є основним джерелом уваги організації вкладень

(зокрема, методом LSB [18]).

За визначенням звук формується із коливань, котрі під час оцифровуванн набувають ступінчастого вигляду (див. рис. 1.5). Такий вигляд отримується завдяки тому, що ПК здатний видавати впродовж якого-небудь малого часового діапазону звук визначеної амплітуди (гучності) і, властиво, цей ніби короткий момент далеко не є таким нескінченно коротким. Довжину цього проміжку власне і визначає сама ЧД. Як варіант, якщо є файл із ЧД 44,1 КГц, тоді це означатиме, що такий короткий часовий інтервал рівний 1/44100 с (оскільки Гц = 1/с). В даний час звукові карти можуть підтримувати ЧД в діапазоні до 192 КГц.

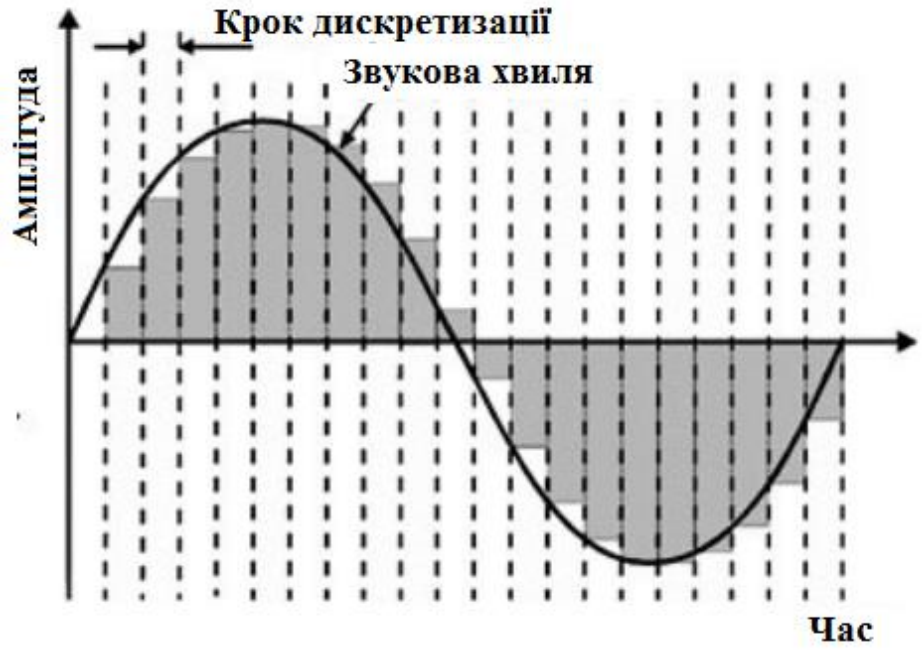

Рисунок 1.5 – Дискретизація звуку

Амплітуда - це гучність звуку у визначений момент часу. Вона визначається числовою величиною, котра займає від 8 до 32 біт пам’яті (хоча в теорії можна і більше). Оскільки 1 байт = 8 біт, то котрась із амплітуд може займати в пам'яті 1, 2, 3, 4 байти відповідно в якийсь короткий проміжок часу. Як наслідок:

– 1 байт – 0..255;
– 2 байти - 0..65 535;
– 3 байти - 0..16 777 216;
– 4 байти - 0..4 294 967 296.

Для варіанту «моно» величини амплітуди є розміщеними послідовно. Для «стерео», спочатку може бути величина амплітуди для лівого каналу, а вже потім для правого, після того знову для лівого і так далі по колу.

ЧД (в Гц) є частотою взяття відліків недискретного сигналу в часі за його дискретизації (зокрема, АЦП) [22].

Отже, основні характеристики WAVE- файлів – це бітність і ЧД.

2 ТЕОРЕТИЧНА ЧАСТИНА

2.1 Методи СтГр для аудіо файлів

2.1.1 Широкосмугове кодування

У сигнал додається модульований повідомленням шум з амплітудою трохи вищою, ніж межа маскування. Перевагою даної схеми є ефективність роботи і висока пропускна здатність, недоліком - чутні спотворення, що вносяться в сигнал.

При приховуванні одного біта послідовності коефіцієнтів вихідна послідовність символів x_i' обчислюється за формулою (2.1):

$$x_i' = \begin{cases} x_i + \omega_i |x_i| \alpha_i & s_i = 1 \\ x_i - \omega_i |x_i| \alpha_i & s_i = 1 \end{cases}, \quad (2.1)$$

де $\omega_i \in -1, +1$ – випадкова двійкова послідовність, α_i – поріг чутності i-ї підсмуги, x_i - символ вихідної послідовності, s_i - біт, що приховується.

Для обчислення порогу чутності може бути використана психоакустична модель, що міститься у форматі кодування MP3 або будь-яка інша. Таким чином, метод дозволяє управляти психоакустичним характером спотворень, що вносяться в сигнал.

Для отримання прихованого біта з послідовності коефіцієнтів використовується функція кореляції прийнятих коефіцієнтів і вихідної випадкової послідовності. Необхідно відмітити, що через ненадійність вилучення цей метод вимагає застосування кодів корекції для помилок. Це призводить до зменшення як швидкодії, і пропускної спроможності методу.

2.1.2 Фазове кодування

У цьому методі використовується той факт, що людське вухо сприймає не значення фази, а лише їх різницю.

Сигнал розбивається на ділянки, значення фази на першій ділянці

використовуються для кодування повідомлення, що приховується, значення фаз інших ділянок береться таким, щоб різниця фаз між ділянками залишилася незмінною.

Для кодування значень фаз, на множині фаз виділяється набір рівномірно розподілених значень, що відповідають бітам 0 і 1. Значення фази замінюється найближчим значенням, що відповідає необхідному біту. Різниця значень у наборі залежить від частоти смуги, та варіюється від π/12 на чутливих смугах до π/4 на високочастотних смугах.

Для кодування одного біта повідомлення, що приховується, використовується певна послідовність змін фаз, різна для кодування 0 і для кодування 1. Для вилучення прихованого повідомлення використовується функція виявлення (2.2):

$$q = \sum r_i(v_i - \varphi_i)^2 - r_i(u_i - \varphi_i)^2 \qquad (2.2)$$

де r_i, φ_i - амплітуда та фаза i- го отриманого сигналу $u = \{\alpha_0, \beta_1, \alpha_2, \beta_3\}$,

$u = \{\alpha_0, \beta_1, \alpha_2, \beta_3\}$ - очікувана послідовність фаз при кодуванні біта 1,

$v = \{\alpha_0, \beta_1, \alpha_2, \beta_3\}$ - очікувана послідовність фаз при кодуванні біта 0,

α_i , β_i- найближчі до φ_i значення фаз, відповідні 1 та 0.

Якщо $q > 0$, біт прихованого повідомлення береться як 1, в потивному випадку 0.

Метод забезпечує високу ефективність кодування за параметром відношення власне сигнал-шум, хоча його пропускна здатність невелика, і становить від 8 до 32 біти за секунду.

2.1.3 Кодування відлунням

Застосовує проміжки різної довжини між сигналами відлуння, щоб закодовувати послідовності величин. За накладення ряду обмежень все таки зберігається умова непомітності для сприйняття людиною. Відлуння володіє трьома характеристиками: початковою амплітудою, затримкою, а також

ступенем згасання. Якщо досягнуто наперед визначений поріг між сигналом і луною відбувається їх змішування. В такій точці людина не може почути для обох таких сигналів відмінність. Існування такої точки дуже складно визначити, оскільки вона перебуває в залежності від якості властиво вихідного запису та власне самого слухача. Як правило, застосовується затримка близько 10^{-3} секунди, що цілком достатня для сприйняття для записів та слухачів (принаймні більшості з них). Використовуються дві різні затримки при кодуванні 0 та 1. Вони обидві мають бути нижчими, аніж поріг чутливості слухацького вуха до одержуваної луни.

Наведені вище методи були взяті з роботи [23]. Ці методи володіють певними недоліками: вони складні для реалізації та розуміння, вносять явні спотворення в аудіо файл та мають надзвичайно погану пропускну здатність. Тому при розробці ПЗ для СтГр нині вони використовуються досить рідко.

2.1.4 Метод LSB

Використовує надмірність звукових файлів. Як відомо, молодші розряди цифрових відліків містять дуже мало корисної інформації. Їхнє заповнення додатковою інформацією практично не впливає на якість сприйняття, що і забезпечує можливість приховування.

Ця група методів володіє рядом відмінностей [24].

Спочатку розглянемо негативні особливості. Зі зміною інформації спотворенню піддаються власне статистичні атрибути цифрових потоків. Зважаючи на це для зменшення компрометуючих ознак необхідно застосувати їх корекцію.

До переваг можна віднести:

– можливість прихованої передачі великих даних;

– можливість захисту авторського права, прихованого зображення, товарної марки, реєстраційних номерів тощо.

Приклад роботи методу LSB наведено на рис. 2.1.

Вихідний файл:

(00100111 11101001 11001000)

(00100111 11001000 11101001)

(11001000 00100111 11101001)

Повідомлення: 01000001

Результат роботи методу:

(00100110 11101001 11001000)

(00100110 11001000 11101000)

(11001000 00100111 11101001)

Рисунок 2.1 – Приклад застосування методу LSB

Підкреслено лише змінені біти вихідного файлу.

В силу своєї простоти та прозорості реалізації метод LSB в даний час широко застосовується у СтГр. Ймовірно, можна запропонувати і інші ефективні методи.

2.2 Огляд наявного ПЗ для СтГр в аудіо файлах

2.2.1 Бажані вимоги до реалізації стеганографічних методів для аудіо файлів

Перед проведенням аналізу спеціалізованого ПЗ варто навести необхідні вимоги:

- збереження цілісності контейнера;
- невідмінність на слух файлів із повідомленням та без нього;
- файл не повинен викликати підозри;
- можливість роботи з файлами будь-якої бітності;
- можливість визначати об'єм контейнера;

– багатотомність контейнера.

Як загальне зауваження:

– все перераховане нижче ПЗ для приховування інформації використовує метод LSB;

– ПЗ, котре використовує інші методи поширене не так широко.

2.2.2 DeepSound [25]

ПЗ, що вільно розповсюджується, з дружнім інтерфейсом, заявленою підтримкою форматів WAVE, APE і FLAC і вбудованим конвертером.

При спробі приховати картинку в аудіо-файлі, що складається з тиші, ФК був наповнений шумами, як видно на рис. 2.2.

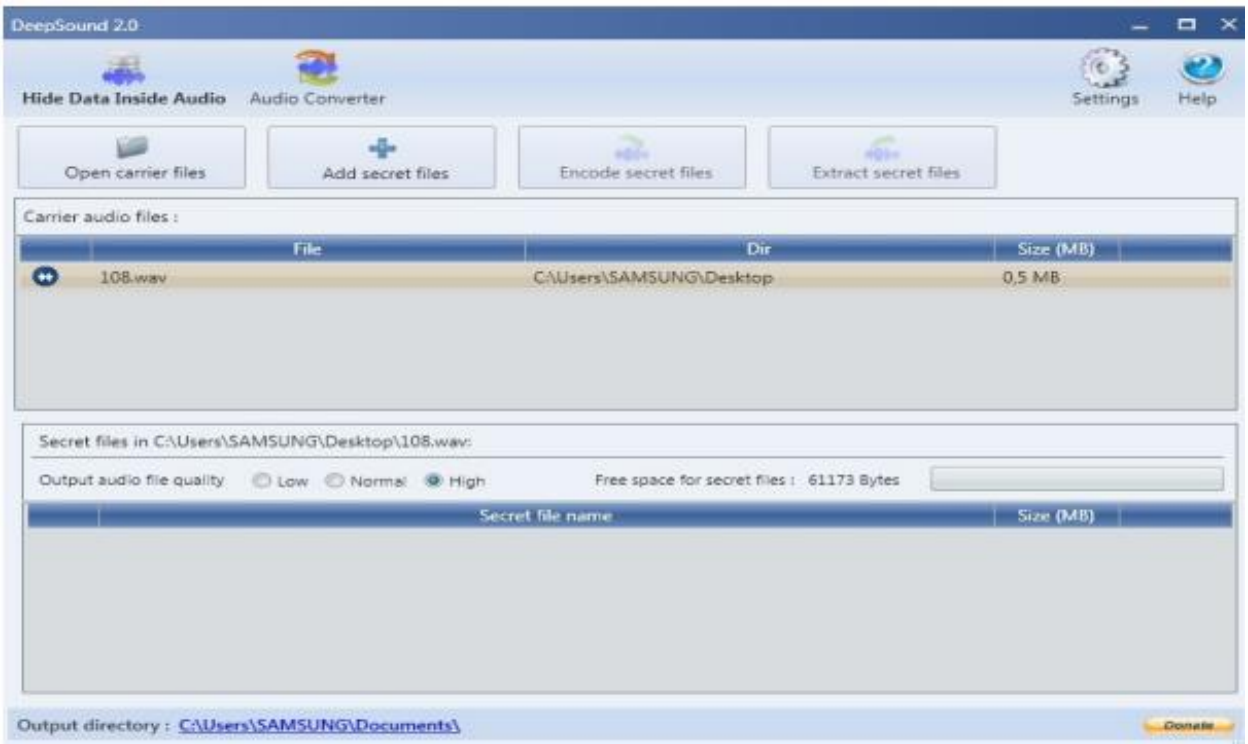

Рисунок 2.2 – Інтерфейс DeepSound

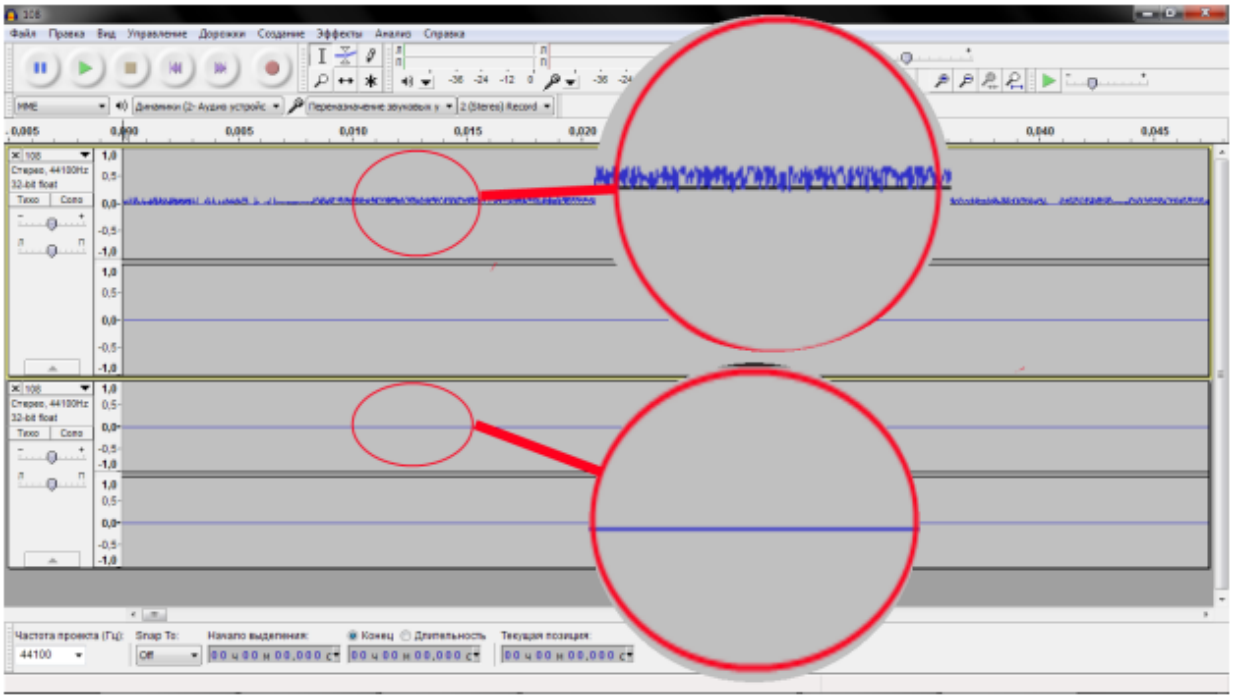

Рисунок 2.3 - Осцилограма файлу з вкладенням за допомогою програми DeepSound (зверху) та файлу тиші

Звідси можна зробити висновок, що програма працює некоректно хоча б з якимсь відсотком файлів, а саме вносить помітні людському вуху зміни в аудіо файл, отже вона не є універсальним стеганографічним рішенням для аудіо файлів.

2.2.3 Xiao Steganography [26]

Вільне розповсюджуване ПЗ для приховування даних у файлах типу BMP і WAV. Є можливість вибору розсіюючої функції (SHA, MD5, MD4, MD2) та алгоритму шифрування (RC2, RC4, DES). При спробі приховати інформацію були чутні спотворення (рис. 2.4). Отже, це ПЗ знову ж таки не є універсальним.

Рисунок 2.4 - Xiao Steganography

2.2.4 SilentEye [27]

ПЗ з вільною ліцензією для приховування даних у файлах різних типів, у тому числі і WAVE (рис. 2.5). Відзначилося дуже незручним діалогом відкриття файлів, а також відмовою у роботі з аудіо файлами формату WAVE.

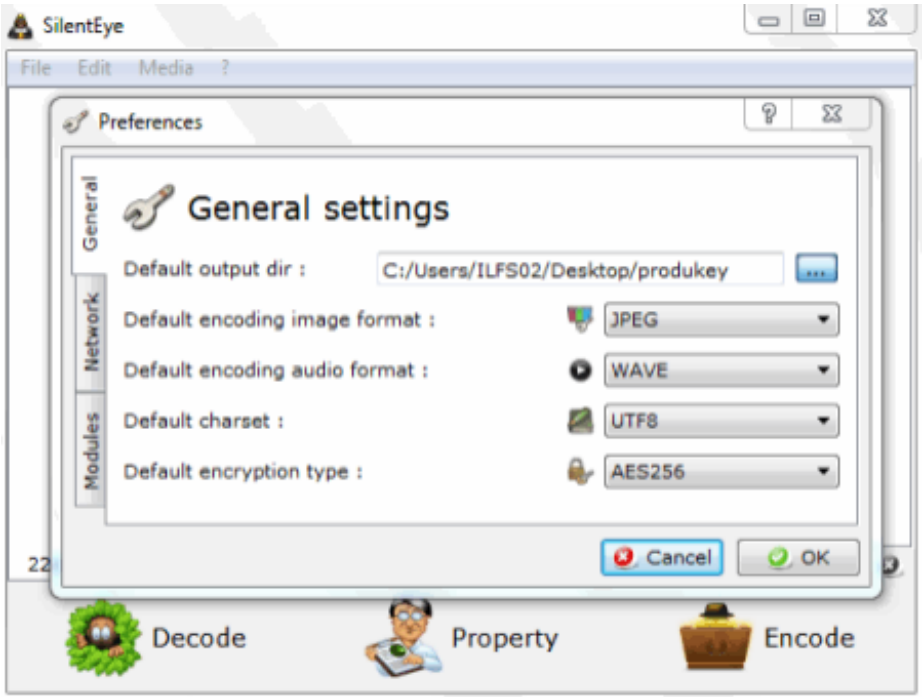

Рисунок 2.5 – SilentEye

2.2.5 StegoStick [28]

Вільне ПЗ для приховування даних у файлах різних типів, у тому числі і WAVE.. Не відображається максимально можливий розмір контейнера, у файлі зі стегоповідомленням виникли шуми (як на рис. 2.6).

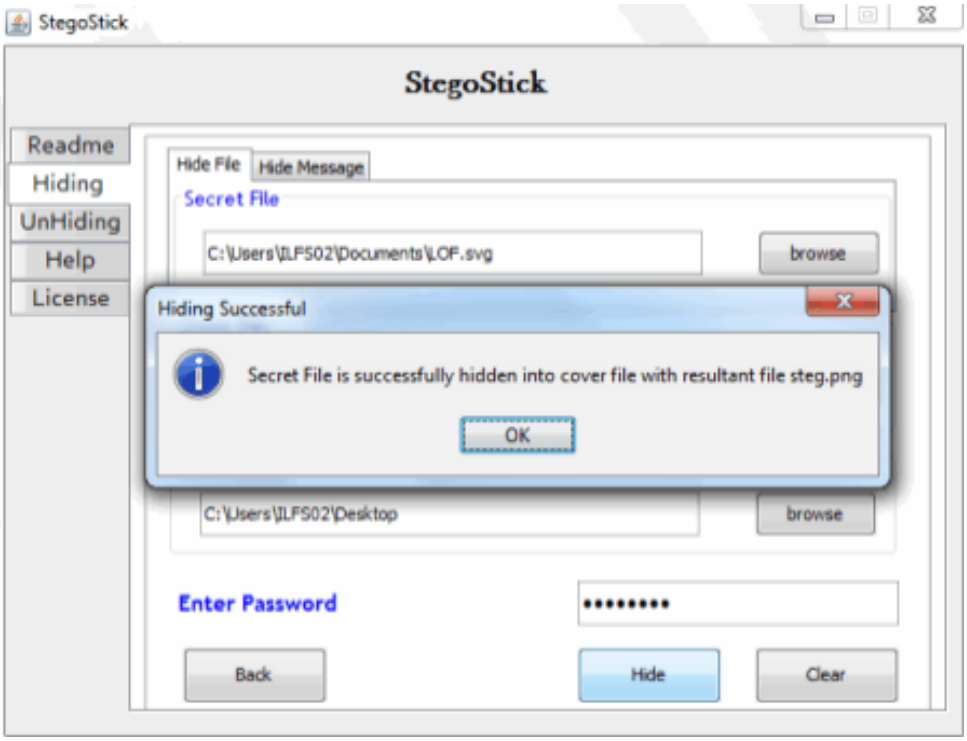

Рисунок 2.6 – StegoStick

Таким чином, проведене тестування згаданих вище пакетів показало, що жоден з них не відповідає викладеним в п.2.2.1 вимогам до стеганографічних пакетів. Програми або вносять чутні спотворення аудіо файли, або відмовляються з ними працювати.

2.3 Існуючі методи та ПЗ для СтА в аудіо файлах

Для виявлення вкладень в аудіо файли дослідниками в різний час було запропоновано наступні методи.

2.3.1 Пошук у фазовій області аудіоданих

Алгоритм пошуку вставок у фазову область полягає у наступному [29].

При обробці кожного кадру даних аудіо сигнал виймався та декодувався. Залежно від параметрів стиснення кадр може містити 578 або 1156 відліків на канал. Одержані дані розбиваються на фрагменти по 512 відліків. Для отриманих фрагментів обчислюється 512 точкове швидке перетворення Фур'є. Далі за знаком уявної частини комплексного спектра визначаються квадранти фаз гармонік і проводиться частотний аналіз цих значень. Потім відшукуються переважаючі значення. Для цього обчислюються рейтинги пар квадрантів 1-2 та 3-4 за формулою (2.3):

$$R_{i,j} = \frac{N_i + N_j}{N} \tag{2.3}$$

де $R_{i,j}$ - рейтинг пари квадрантів i,j; N_i — частота, з якою зустрічається і-й квадрант ; N - кількість компонентів у векторі фаз.

2.3.2 СтА аудіо файлів на основі методів стиснення

Ідея запропонованого методу у тому, що файл, котрий містить однорідні дані, має власну статистичну структуру [18]. Якщо використовувати його як контейнер для секретного повідомлення, то після впровадження повідомлення в контейнер порушиться статистична структура контейнера та підвищиться його ентропія. Таким чином, при використанні алгоритмів стиснення вихідний («порожній») контейнер стискається, як правило, краще, ніж заповнений. Значить, якщо ступінь стиснення передбачуваного контейнера більше деякого порогового значення, то з великою часткою впевненості можна сказати, що контейнер порожній, інакше з великою часткою впевненості можна судити про наявність повідомлення в контейнері.

Метод аналізу полягає у порівнянні коефіцієнтів стиснення вихідного контейнера та його повністю заповненої копії. Повністю заповнена копія виходить за допомогою псевдовипадкового зміни молодших біт вихідного («порожнього») контейнера. До обох файлів застосовується метод стиснення даних та аналізуються їх коефіцієнти стиснення. Якщо ці коефіцієнти близькі за

значенням, то ймовірно, що вихідний файл містив приховане повідомлення. І, навпаки, за великої різниці в коефіцієнтах стиснення виноситься результат про відсутність прихованої інформації у файлі. При практичній реалізації цього методу контейнер та його заповнена копія розглядається не повністю, а діляться на кілька рівних частин. Це дозволяє збільшити точність методу, крім того, вводиться додатковий параметр, що регулює співвідношення помилок I та II роду. Зауважимо, що метод аналізу застосовується не до всього файлу, а саме фрагменту файлу, що містить звукові дані.

З формального боку розроблений алгоритм має такий вигляд. Нехай $X = \{x_1,...,x_n\}$ - послідовність байт звукових даних. $|X| = N$ – довжина послідовності. Послідовність X розіб'ємо на рівні відрізки. Позначимо як Ф(X) - алгоритм стиснення, котрий застосовується до X. Введемо величину $f(X,n)=|Ф(X_n)|/|X_n|$ - коефіцієнт стиснення відрізка n послідовності X універсальним кодом Ф.

Позначимо через φ(X) псевдовипадкову зміну молодших біт послідовності X (алгоритм LSB). Тоді Y = φ (X) - заповнений контейнер X. Введемо величину: Δ (X, n) = | f (X, n) - f (Y, n) |.

Для визначення факту включення секретного повідомлення береться граничне значення для величини Δ і проводиться оцінка кількості відрізків, на яких значення величини не перевищує поріг. Якщо таких відрізків більше, ніж половина їх загальної кількості, то вважається, що вихідна послідовність X мала в собі приховані дані; інакше - X вважається «порожньою».

2.3.3 Існуюче ПЗ для СтА

Шляхом дослідження інтернет джерел вдалося виявити, що у вільному доступі найбільша кількість пакетів для СтА передбачає роботу, здебільшого, з графічними файлами, зокрема, з файлами формату JPEG [30], [31]. Що стосується аудіо файлів, то для них ПЗ для СтА виявити виявилося важко (або у вільному доступі воно взагалі відсутнє), що підкреслює актуальність роботи в даному напрямку.

2.4 Опис розробленого підходу СтА

В основу пропонованого підходу покладена гіпотеза про невипадковість МЗР аудіофайлів [32]. Хоча, є гіпотеза, що МЗР аудіофайлів є випадковими.

Реально, це не так. Хоча людське вухо не помітить змін звукового файлу при зміні останніх бітів, статистичні атрибути звукового файлу будуть помінені.

Перед приховуванням дані, зазвичай, піддаються архівації (для зменшення обсягу) або зашифровуються (з метою надання додаткової стійкості повідомленню при попаданні в чужі руки). Це робить розряди даних дуже наближеними до випадкових. І тут послідовне вмонтовування інформації такого роду поміняє МЗР звукового файла випадковими бітами. У цьому полягає основна суть підходу - уловлювання різниці між розподілом молодших розрядів у порожньому і заповненому контейнерах.

Для перевірки цього підходу було розроблено певну методику, що передбачає використання інструменту статистичного тестування NIST, розробленим в США [33].

Тести програмного комплексу NIST були розроблені для статистичної перевірки бінарних послідовностей на випадковість [34]. Ці тести засновані на статистичних властивостях, які мають лише випадкові послідовності.

Для відпрацювання методики була створена тестова БД з 108 різнотипних аудіо файлів, припускаючи використання їх як контейнер. У БД було включено шуми (білий, чорний, коричневий, рожевий, зелений тощо), рівні синусоїди, записи мови, інструментів (гітара, саксофон, барабани тощо), і навіть повноцінні музичні композиції. БД складається з файлів різних бітностей (8, 16, 24, 32 біти) та ЧД. Розмір файлів від 38 КБ до 41 522 КБ.

Щоб відстежувати поведінку випадковості молодших біт, необхідно впорядкувати файли за якоюсь ознакою. Як таку ознаку було обрано відносну кількість нульових байт у файлі. Відносна кількість нульових байт визначається як відношення нульових байт файлу до загальної кількості байт. Відповідно до цієї ознаки було впорядковано всі 108 тестових файлів.

Для перевірки гіпотези у вихідну БД (крім порожніх контейнерів) були додані частково заповнені, а саме у всі файли з БД були здійснені стеговкладення (за допомогою розробленого ПЗ Hide_Wave_Stego) на 10%, 50% та 100% від максимальної можливості розглянутого стегоконтейнера.

Як тест був використаний частотний побітовий тест пакету NIST [34], який оцінює співвідношення між нулями та одиницями в двійковій послідовності. Алгоритм цього тесту полягає в наступному [33]: файл сприймається як бітова послідовність, одиниця приймається за +1, нуль за 0. Обчислюється сума послідовності. Потім обчислюється статистика за формулою (2.4):

$$S_{obs} = \frac{|S|}{\sqrt{n}}, \tag{2.4}$$

де | S | - сума послідовності, n - кількість елементів у послідовності.

Розраховується P –значення за формулою (2.5) через додаткову функцію помилок:

$$P_{value} = erfc(\frac{s_{obs}}{\sqrt{2}}), \tag{2.5}$$

Додаткова функція помилок розраховується за співвідношенням (2.6):

$$erfc(x) = \frac{2}{\sqrt{\pi}} \int_{x}^{\infty} e^{-t^2}\, dt \tag{2.6}$$

І якщо результат більше ніж 0,01, то послідовність визнається випадковою з рівнем довіри 99%.

3 ПРАКТИЧНА ЧАСТИНА

3.1 Розробка додатку для СтГр та СтА для аудіо файлів

3.1.1 Призначення та структура ПЗ

Для СтГр на аудіофайлах та для розглянутих методів СтА розроблено оригінальне ПЗ, котре реалізоване у вигляді пакету Hide_Wave_Stego.

Додаток виконує стеговкладення в аудіо файли формату WAVE будь-якої ЧД та бітності з властивістю багатотомності, а також у додатку реалізовано власну методику виявлення таких вкладень та метод виявлення повідомлень в аудіо файлах на основі методів стиснення.

ПЗ було розроблено у середовищі Embarcadero Delphi XE5.

3.1.2 Програмна структура пакету та процедури

Структура пакету на ведена на рис. 3.1.

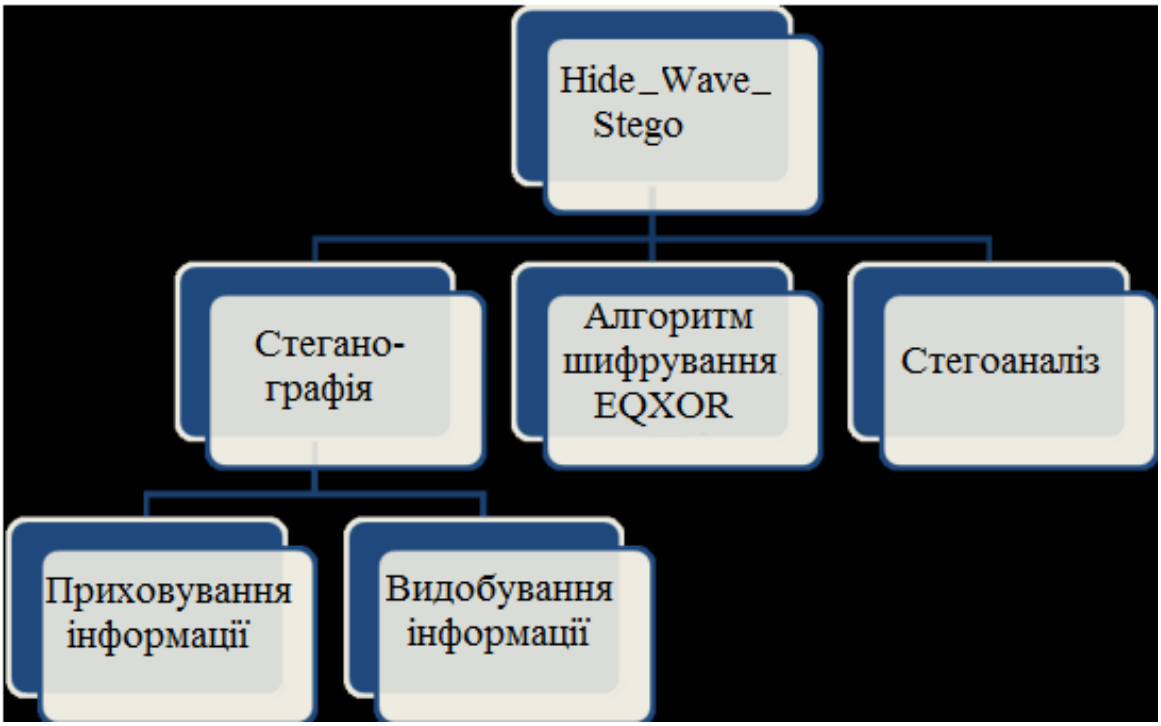

Рисунок 3.1 - Структура розробленого ПЗ

У додатку розроблено такі процедури та функції:

1. FromBytesTo Cardinal – переводить дані з типу Byte у тип Cardinal. Використовується у WaveUnHide.

2. FromCardinalToBytes – переводить дані з типу Cardinal у тип Byte. Використовується у HideWave.

3. ByteToBin і BinToByte - переводять дані з типу Byte до типу Bin і назад (bin - масив [0..7] елементів із множини 0..1). Використовуються у HideWave, WaveUnhide.

4. first – реалізує EQXOR- шифрування. Використовується у HideWave.

5. last – реалізує EQXOR- дешифрування. Використовується у WaveUnHide.

6. HideWave – реалізує LSB алгоритм стеганографічного приховування в WAVE- файлі.

7. WaveUnHide – реалізує видалення інформації, котра була прихованна за допомогою алгоритму LSB.

8. Button2Click - запитує імена файлів для вилучення прихованої інформації та викликає WaveUnHide.

9. Button3Click – запитує ім'я ФК для подальшого використання, відображає його розмір.

10. Button1Click – запитує ім'я файлу, який потрібно сховати, викликає HideWave.

11. BitBtn1Click – запитує ім'я ФКл.

12. Button4Click – реалізує метод СтА, заснований на частотному аналізі.

13. Button5Click – виводить результати статистичних тестів NIST.

14. Button6Click – реалізує метод СтА, заснований на алгоритмах стиснення.

3.1.3 Опис стеганографічного блоку програми

У програмі як стеганографічний алгоритм був реалізований алгоритм LSB. Як мовилося раніше суть даного алгоритму полягає у заміні найменших

значущих бітів аудіо файлу бітами повідомлення. На вхід до програми подається ФК, ФП, а також ФКл.

Файл-повідомлення шифрується за допомогою алгоритму EQXOR та ФКл. Суть алгоритму EQXOR полягає в наступному: береться дві послідовності біт (у цьому випадку перша послідовність це ФП, друга - ФКл). Наприклад, 01001001 та 11010100. Якщо і-й біт першої послідовності аналогічний біту другої, то у вихідній послідовності на і-тому місці пишеться одиниця, інакше - нуль. Таким чином, для наших послідовностей з прикладу результат буде таким: 01100010. За своєю суттю, це алгоритм XOR, тільки логічною операцією замість «виключає АБО» виступає операція еквівалентності [35].

Потім отримуємо розмір зашифрованого файлу та місткість контейнера. Місткість контейнера обчислюється за формулою (3.1):

$$Size = \frac{F}{BPS} - 132, \tag{3.1}$$

де F – розмір ФК, BPS – кількість байт на семпл (є в заголовку аудіо файлу формату WAVE, зчитується безпосередньо з файлу).

Потім залежно від кількості байт на семпл послідовно проводиться заміна двох молодших бітів ФК (якщо BPS = 1, то заміні піддається кожен байт даних ФК, якщо BPS = 2, кожен третій, тощо.). Спочатку записується розмір в байтах файлу, що приховується. Заміна проводиться до тих пір, поки всі біти ФП не будуть записані або до ФК. Якщо кінець ФК досягнуто, виводиться повідомлення про те, що контейнер закінчено, і потрібно вибрати наступний контейнер.

У програмі це реалізовано в такий спосіб (рис. 3.2).

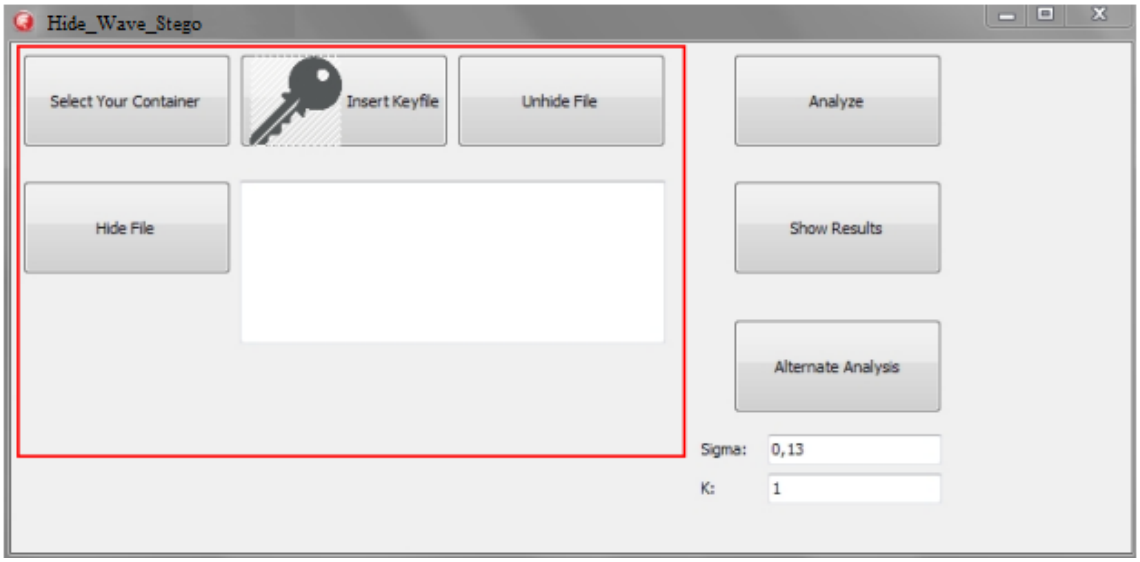

Рисунок 3.2 – Головне вікно програми. Червоним прямокутник виділена стеганографічна частина

У цій частині програми є ряд процедур та функцій: first, last, HideWave, BitBtnlClick, WaveUnHide.

Перелік дій при приховуванні файлів наступний:

1. Натисканням на кнопку "Insert Keyfile" викликається BitBtnlClick. У процедурі вибирається ФКл за допомогою стандартного інструменту Delphi OpenDialog. Глобальної змінної Ideal надається ім'я ФКл.

2. Натисканням на кнопку "Select Your Container" вибирається ФК. У стандартному текстовому редакторі Memol виводиться розмір контейнера (кількість кілобайт максимально можливої інформації для приховування).

3. Натисканням на кнопку «Hide File» вибирається файл, що приховується. Потім викликається процедура first, у якій реалізується алгоритм XOR для вибраного та ФКл. Результатом процедури є зашифрований файл, який згодом і буде прихований. Потім у процедурі зчитується інформація про бітність аудіо файлу. І, нарешті, відбувається саме приховування доти, доки не досягнуто кінець ФК, або зашифрованого ФП.

Приховування відбувається так: в аудіо файлі, послідовно, залежно від його бітності, два останніх значущі біти замінюються бітами зашифрованого ФП. Якщо досягнуто кінець ФП, то у контейнер дописуються байти вихідного файла. Якщо ж досягнуто кінець ФК, то користувачеві необхідно вибрати

додатковий контейнер (рис. 3.3).

Рисунок 3.3 - Вимога програми надати новий контейнер для приховування інформації

На виході виходить аудіо файл із вкладенням, ідентичний початковому файлу за розміром, а також за звучанням. Результати роботи програми наведено на рисунках 3.4 та 3.5.

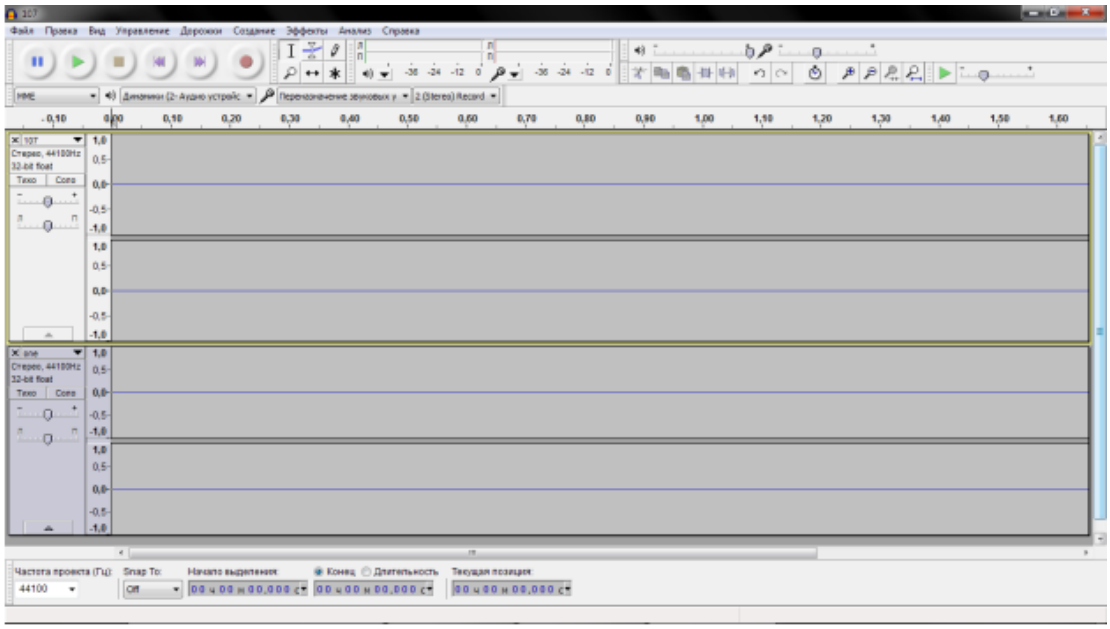

Рисунок 3.4 - Осцилограма «порожнього» аудіо файлу (зверху, 107.wav) та аудіо файлу з вкладенням з використанням ПЗ Hide_Wave_Stego (знизу, one.wav)

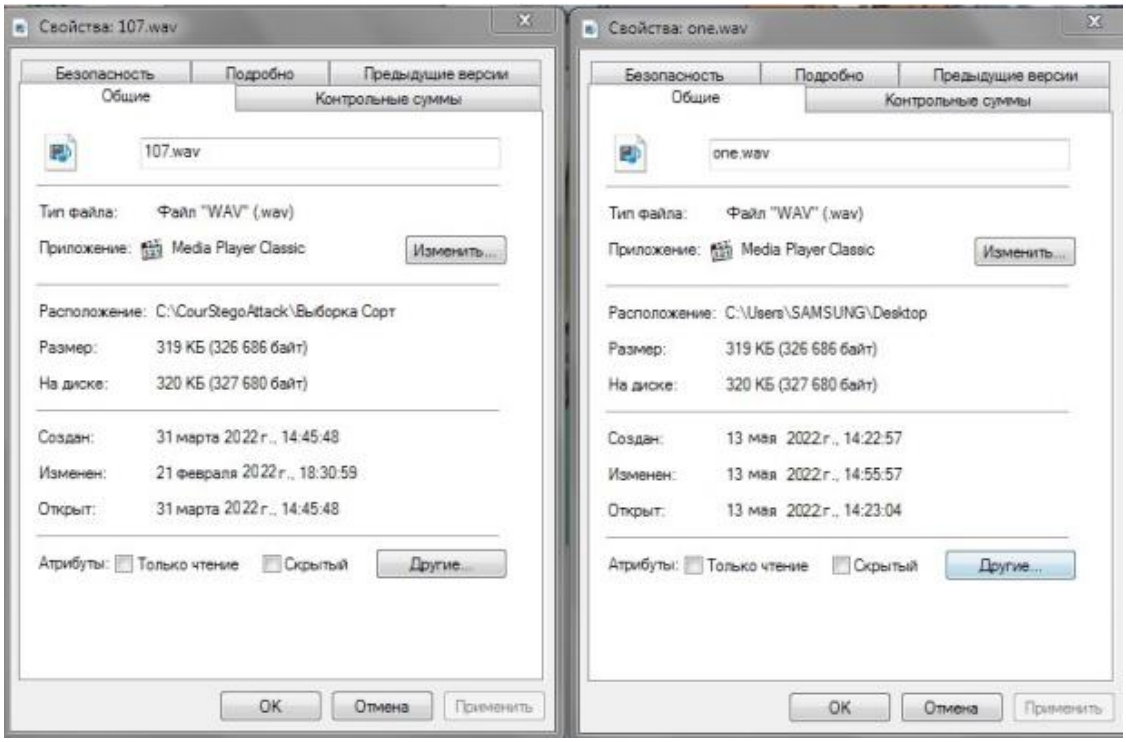

Рисунок 3.5 – Відповідність розмірів файлу з вкладенням та без

3.1.4 Процедура виймання повідомлення з файлу

На вхід подається аудіо ФК та ФКл. З ФК зчитується його бітність, і навіть розмір прихованого у ньому ФП. Послідовне зчитування молодших бітів іде відповідно до розміру ФП. Якщо кінець ФК досягнуто, користувач подає на вхід програмі ще один ФК.

У програмі це реалізовано в такий спосіб.

Порядок дій під час отримання стегоповідомлення наступний:

1. Натисканням на кнопку "Insert Keyfile" викликається BitBtn1Click. У процедурі вибирається ФКл за допомогою стандартного інструменту Delphi OpenDialog. Глобальної змінної Ideal надається ім'я ФКл.

2. Натисканням на кнопку "Unhide File" викликається WaveUnHide. Користувачеві потрібно вибрати файл зі стегоповідомленням. Спочатку з файлу зчитується його бітність. Потім зчитується розмір прихованого файлу. Запускається цикл від нуля до розміру прихованого файлу, в якому згідно з бітністю проводиться зчитування двох останніх біт та запис їх у вихідний файл. Якщо досягнуто кінець файлу, то користувачев повинен запропонувати додатковий контейнер програмі (рис. 3.6).

На виході отримується ФП формат якого не є наперед із визначеним. Передбачається, що одержувачу формат файлу повідомлення є відомим.

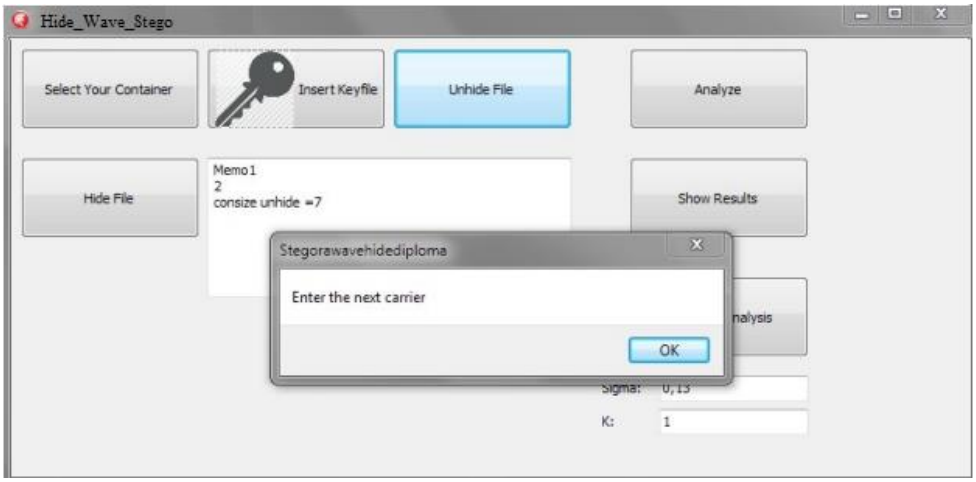

Рисунок 3.6 - Вимога програми надати наступний контейнер для вилучення інформації

3.2 Тестування БД файлів щодо вкладень за допомогою частотного аналізу

З допомогою розглянутого в п. 2.4 тесту послідовності найменших значущих біт всіх 108 файлів з бази було перевірено на випадковість.

Результати представлені рис. 3.7 – 3.10.

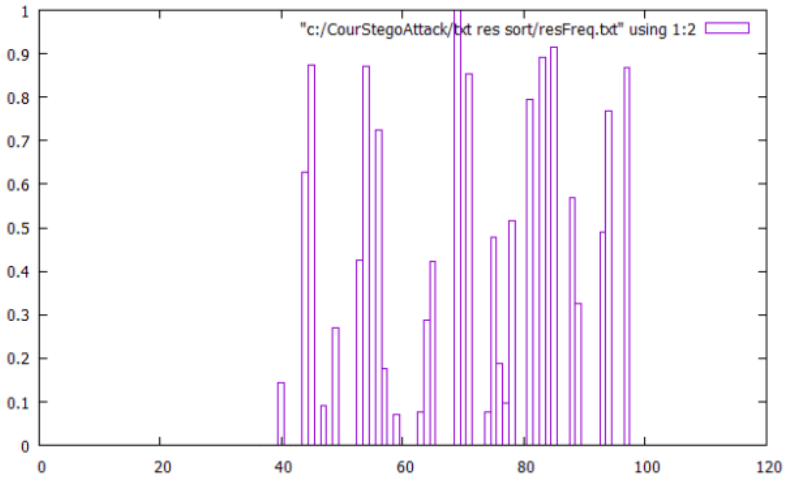

Рисунок 3.7 – Перевірка на випадковість послідовності LSB 108 «порожніх» аудіо файлів.

Як видно з рис. 3.7, послідовності LSB у файлів з відносною кількістю нульових байт менше 0,08 (значення для 40 файлів) невипадкові.

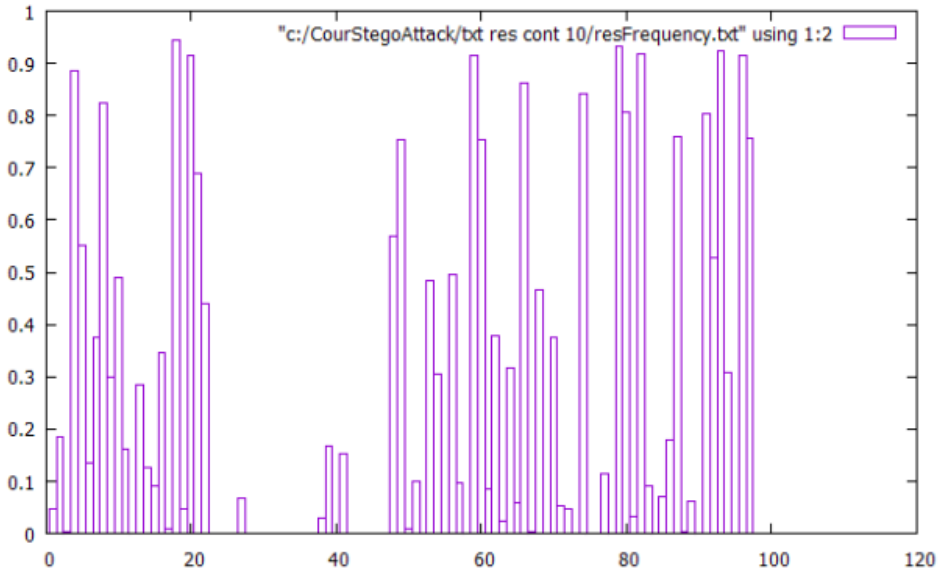

Рисунок 3.8 - Перевірка на випадковість послідовності LSB 108 аудіо файлів, заповнених на 10% максимальної можливості

З рис. 3.8 видно, що при 10 % стеговкладенні послідовності LSB у файлів із відносним кількістю нульових байт менше 0,0348 (значення для 22-го файла) випадкові.

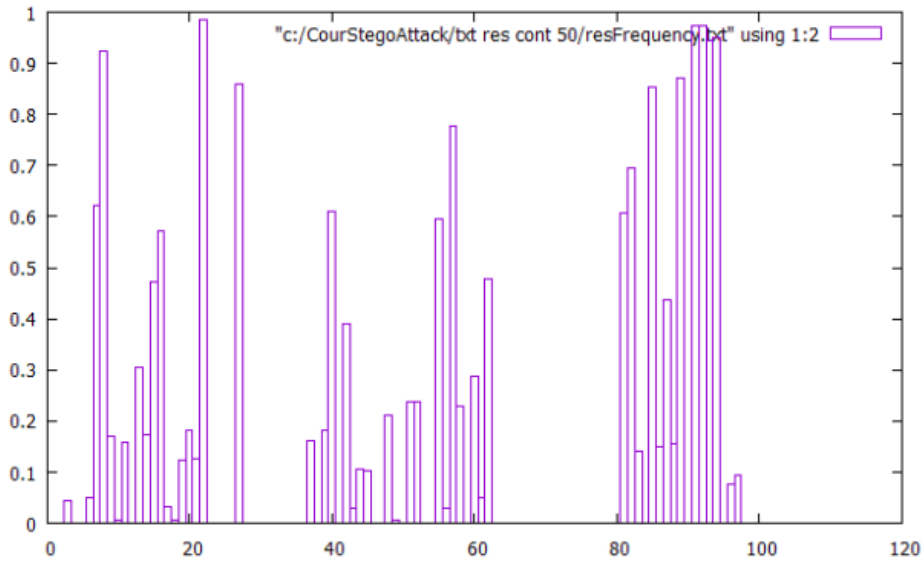

Рисунок 3.9 - Перевірка на випадковість послідовності LSB 108 аудіо файлів, заповнених на 50% максимальної можливості

З рис. 3.9 видно, що при 50 % стеговкладенні послідовності LSB у файлів з відносним кількістю нульових байт менше 0,0348 (значення для 22-го файла) здебільшого випадкові

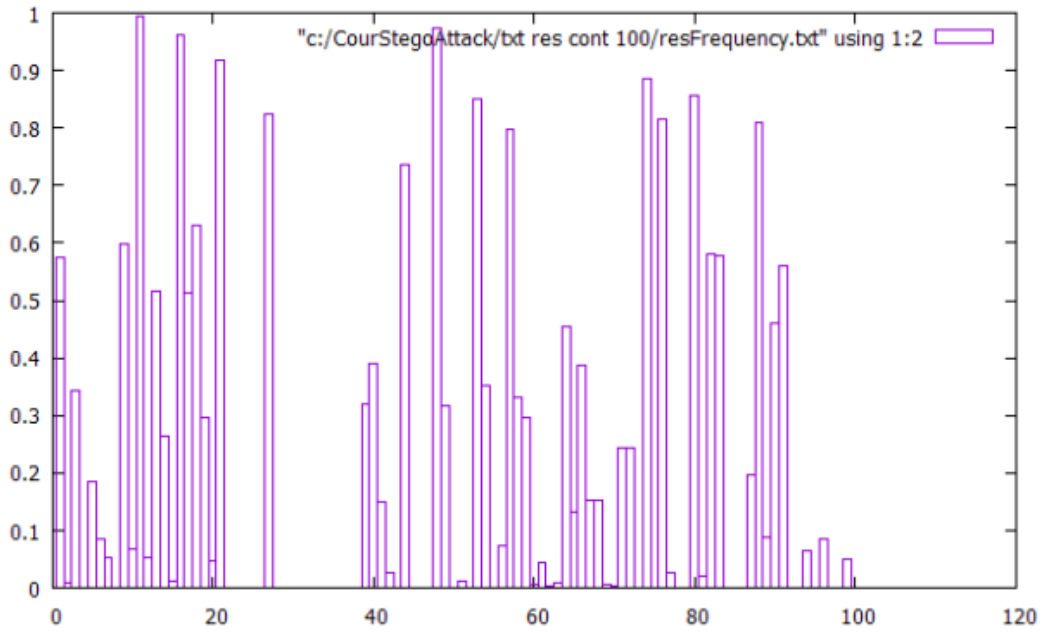

Рисунок 3.10 - Перевірка на випадковість послідовності LSB 108 аудіо файлів, заповнених на 100% максимальної можливості

З рис. 3.10 видно, що при 100 % стеговкладенні послідовності LSB у файлів з відносною кількістю нульових байт менше 0,0348 (значення для 22-го файлу) випадкові.

Практичне використання методики визначення вкладень в аудіо файл:

– береться файл, що перевіряється;

– визначається відносне число нульових байт;

– якщо це число менше, ніж 0,0038 (підібрано емпірично), файл перевіряється частотним тестом NIST.

Якщо тест показав, що послідовність молодших бітів файлу є випадковою (тобто результат частотного тесту більше, ніж 0,01), це з великою часткою впевненості свідчить, що у файлі є стеговкладення.

У програмі ця методика була реалізована в такий спосіб:

– при натисканні на кнопку "Analyze" користувач має перш за все

вказати файл для аналізу, потім файл, в який збережеться послідовність LSB вибраного файлу. У стандартному текстовому редакторі Memol з'явиться відносна кількість нульових байт. Відкриється пакет статистичних тестів NIST (рис. 3.11). У ньому необхідно вибрати файл з послідовністю молодших біт аналізованого файлу, формат читання: текстовий , кількість послідовностей: 1, довжину послідовності виставити відповідно до розміру файлу та натиснути на кнопку «Тест»;

– при натисканні на кнопку "Show Results" відкривається текстовий документ з результатами тестів з їх назвами.

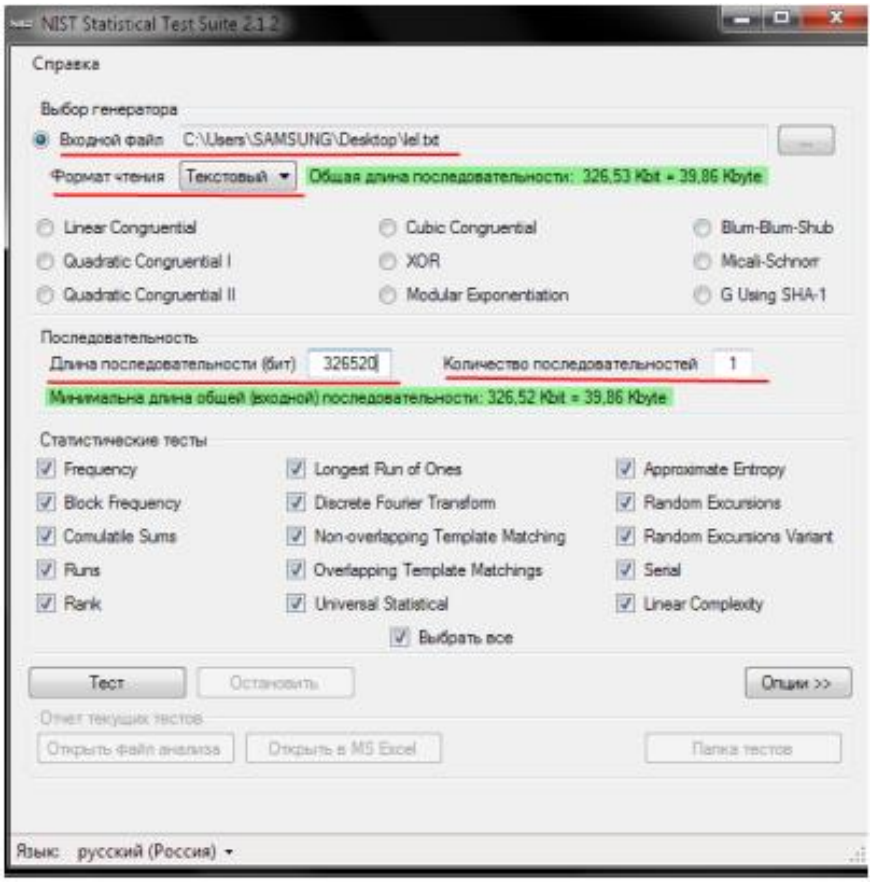

Рисунок 3.11 – Вікно пакету NIST

Із проведеного тестування встановлено, що запропонована методика працює дуже ефективно (визначаються навіть 10% вкладення), але лише на аудіофайлах з малою відносною кількістю нульових байт.

3.3 Метод СтА аудіо файлів на основі алгоритмів стиснення

Для перевірки порівняльної ефективності розробленої в кваліфікаційній роботі методики СтА в роботі був реалізований відомий метод СтА, заснований на алгоритмі стиснення, описаному в [18].

3.3.1 Зауваження та недоліки СтА на основі алгоритму стиснення

У роботі [18]:

– не вказано кількість молодших біт, що змінюються;

– відсутня жодна інформація про вид аудіо файлів, що піддавалися дослідженню;

– цілком ймовірно, що були використані аудіо файли подібного вигляду, отже, цей алгоритм не можна вважати універсальним;

– були використані непопулярні у наш час 8- та 16-бітні WAVE файли. Сьогодні більш актуальні аудіо файли бітністю від 24 і вище.

3.3.2 Реалізація методу СтА на основі алгоритму стиснення

При реалізації зважаючи на зроблені вище зауваження, прийнято:

– види досліджуваних файлів максимально довільні (представлені в БД);

– кількість молодших біт, що замінюються на випадкові - 2 (як це було зроблено в стеганографічній частині розробленого додатка);

– бітність файлів від 8 до 32.

У розробленому ПЗ цей метод реалізовано так.

В інтерфейсі розробленого пакету Hide_Wave_Stego при натисканні на кнопку «Alternate Analysis» та введенні коефіцієнтів sigma та K користувачеві потрібно вибрати файл для аналізу. Спочатку файл розбивається на K рівних частин. Потім проводиться стиснення кожної з частин за допомогою методів стандартної бібліотеки zlib і обчислюється відношення розміру стисненого куска файла до нестисненого. Після того, як у початковому файлі два молодші біти змінюються на випадкові і процедура повторюється, обчислюється

відношення розмірів стисненого куска до стисненого. Для кожного куска розраховується параметр Δ. Якщо кількість кусків, параметр Δ для яких менше, ніж введений коефіцієнт sigma, більше половини, то виноситься рішення про наявність у файлі, що розглядається стеговкалдень. В іншому випадку виноситься рішення про їх відсутність.

3.3.3 Тестування бази файлів щодо вкладень за допомогою методу, заснованому на алгоритмі стиснення

Тестова БД з 108 аудіо файлів була перевірена на стеговкладення за допомогою цього методу. На графіках нижче показані значення параметра Δ. Порівнювалися значення цього параметра для «порожніх» та заповнених (за допомогою розробленого ПЗ Hide_Wave_Stego) на 100% файлів. Результати подано на графіках нижче (рис. 3.12 – 3.15). По осі обсцис на графіках знаходяться номери файлів, що тестуються, по осі ординат - отримане значення параметра Δ.

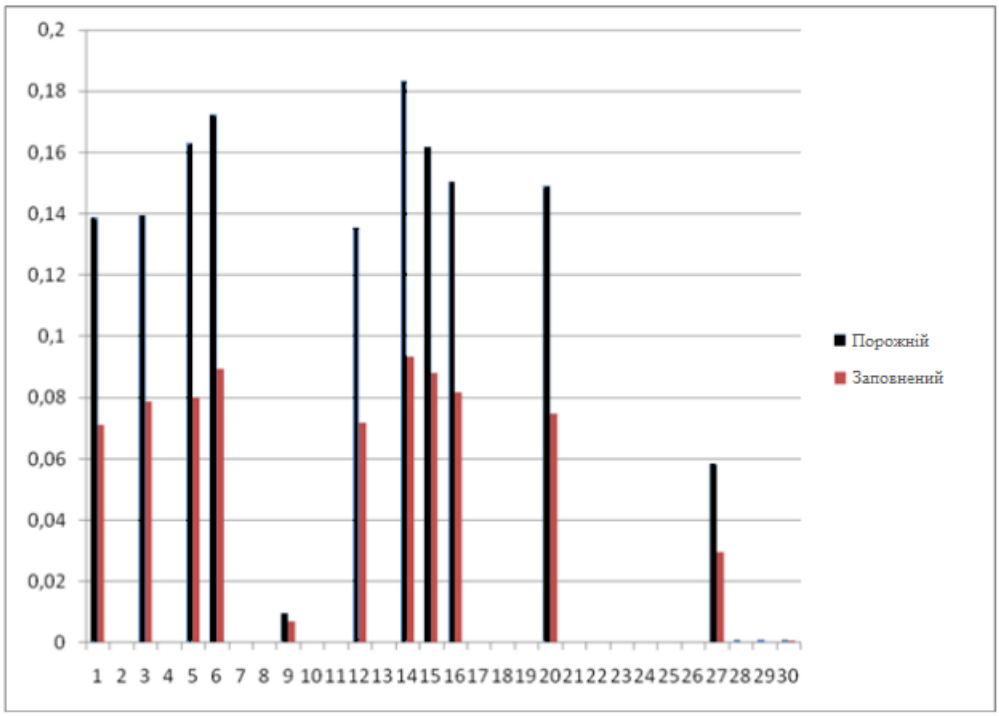

Рисунок 3.12 - Тестування файлів з 1 до 30

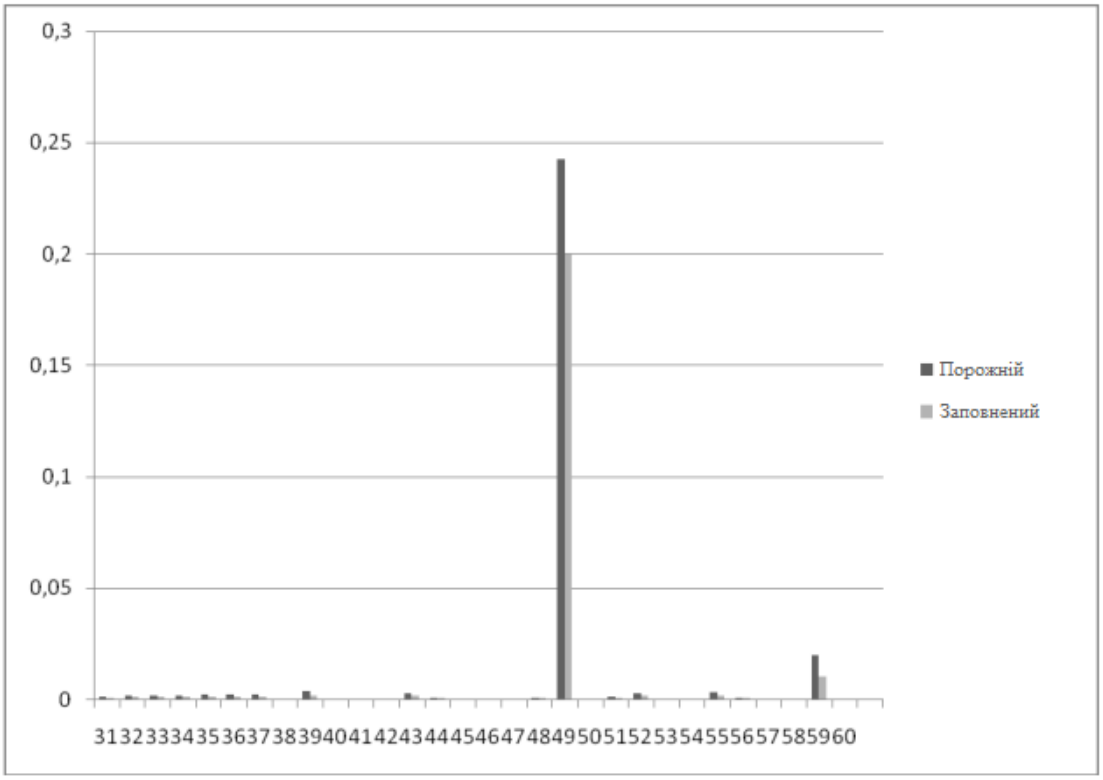

Рисунок 3.13 - Тестування файлів з 31 до 60

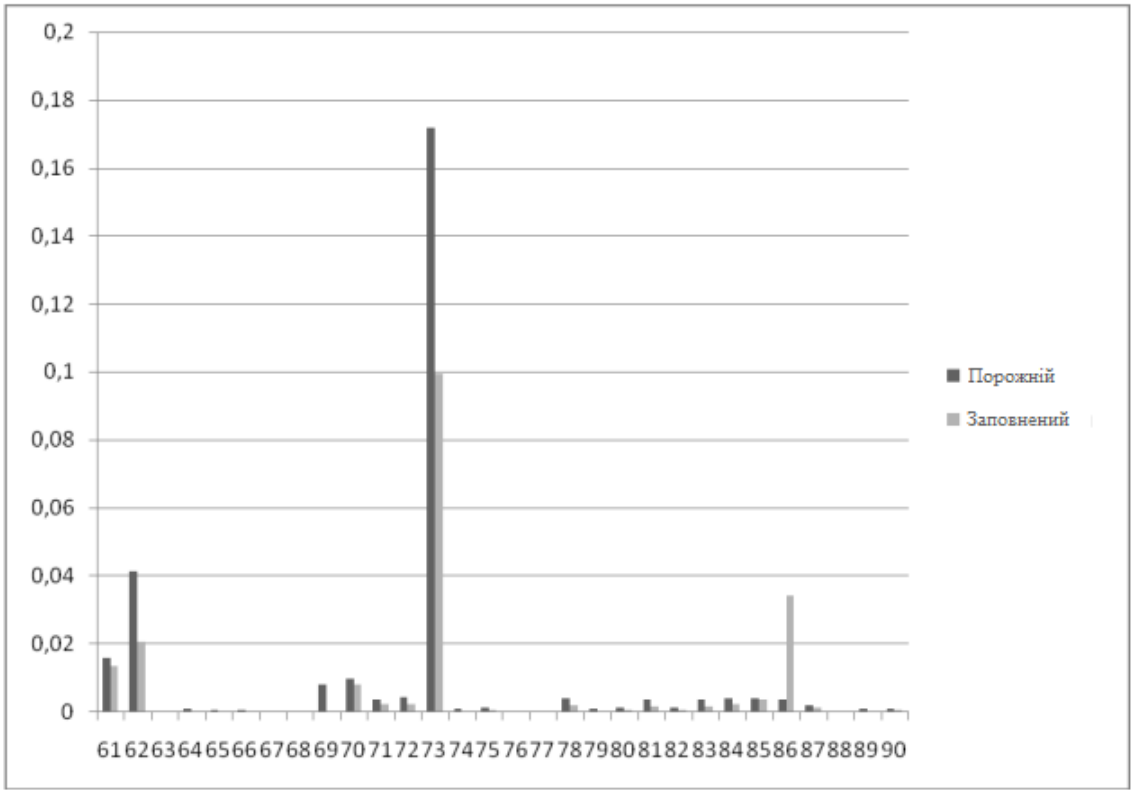

Рисунок 3.14 - Тестування файлів з 61 до 90

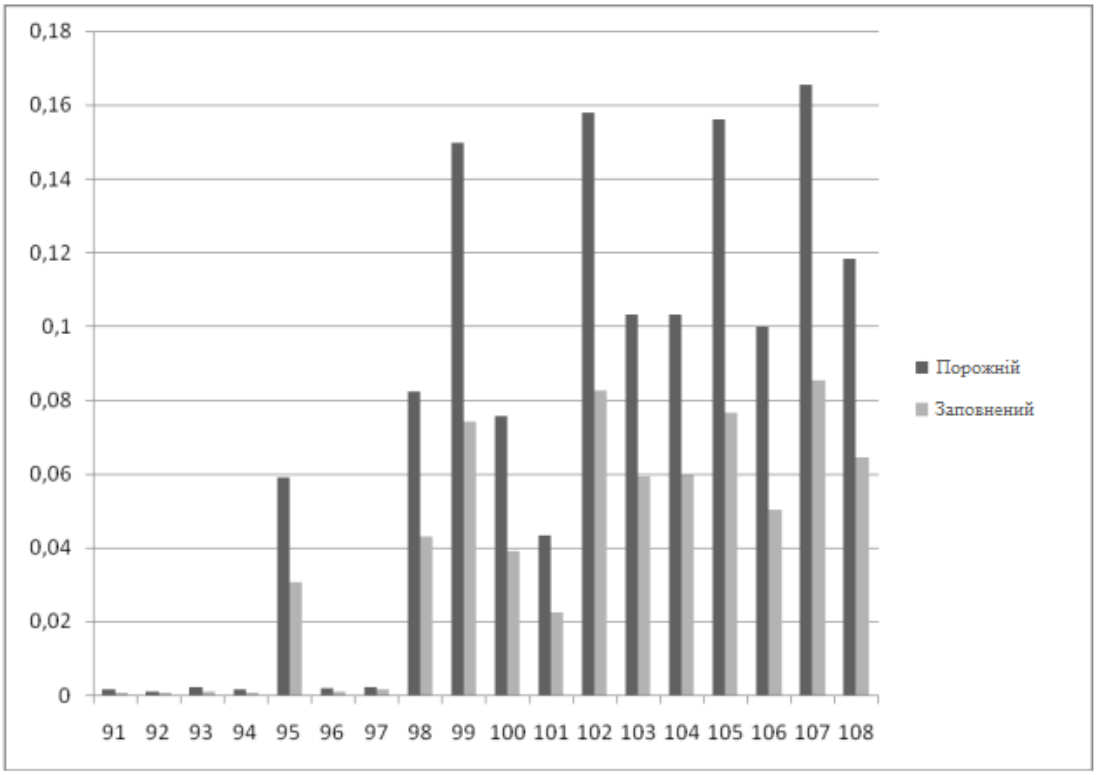

Рисунок 3.15 - Тестування файлів з 91 до 108

Як видно з результатів тестування розробленого ПЗ (див. рис. 3.12 – 3.15), заповнений контейнер стискається в середньому вдвічі гірше. Отже, метод СтА, заснований на алгоритмах стиснення, можна використовувати для визначення наявності вкладень в аудіо файлах.

Але оскільки параметр Δ різний для різних видів файлів, для ефективного застосування даного методу необхідно класифікувати файли та для кожного класу встановити своє граничне значення Δ.

3.4 Порівняльний висновок за методами СтА

Результати тестування БД з 108 різних аудіо файлів формату WAVE показали, що метод, заснований на алгоритмах стиснення [18], вимагає додаткової доробки в плані класифікації аудіо файлів. З великою часткою впевненості можна говорити про можливість застосування даного методу для визначення стеговкладень в аудіо файлах, що належать до одного класу.

Результати тестування цієї БД, але застосовуючи метод, заснований на

частотному аналізі, показали наступне. Метод працює дуже ефективно для файлів, відносна кількість нульових байт для яких менша, ніж 0,038. Файли, що належать до даного класу, відрізняються великою інформаційною навантаженістю. До цього класу відносяться шуми, записи музичних інструментів із великою кількістю шумів (велика кількість шуму виникає через використання неякісних АЦП).

Тому для ефективного виявлення стеговкладень необхідне доопрацювання обох методів і, можливо, їх комплексне використання.

4 БЕЗПЕКА ЖИТТЄДІЯЛЬНОСТІ, ОСНОВИ ОХОРОНИ ПРАЦІ

4.1 Вимоги ергономіки до організації робочого місця оператора ПК

Робоче місце — це зона простору, що оснащена необхідним устаткуванням, де відбувається трудова діяльність одного працівника чи групи працівників [40].

Раціональне планування робочого місця має забезпечувати: найкраще розміщення знарядь і предметів праці, не допускати загального дискомфорту, зменшувати втомлюваність працівника, підвищувати його продуктивність праці. Площа робочого місця має бути такою, щоб працівник не робив зайвих рухів і не відчував незручності під час виконання роботи. Важливо мати також можливість змінити робочу позу, тобто положення корпуса, рук, ніг. Проте доцільно виключати або мінімізувати всі фізіологічно неприродні і незручні положення тіла. Проведені дослідження показують, що при раціональній організації робочих місць продуктивність праці зростає на 15–25% [41].

Організація робочого місця користувача ПК має відповідати ергономічним вимогам ДСТУ 8604:2015 «Дизайн і ергономіка. Робоче місце для виконання робіт у положенні сидячи. Загальні ергономічні вимоги», ДСан ПіН 3.3.2.007-98, характеру та особливостям трудової діяльності.

Площа одного робочого місця користувача ПК повинна складати не менше 6 м2, а об’єм – не менше 20 м3. Конструкція робочого місця користувача ПК повинна відповідати сучасним вимогам ергономіки, характеру виконуваної роботи і забезпечити оптимальне розміщення на робочій поверхні документів та обладнання ПК (монітора, системного блоку, клавіатури, мишки та інших периферійних пристроїв. Монітор на робочому місці встановлюється так, щоб верхній край екрана знаходився на рівні очей.

Розташування монітора ПК має забезпечувати:

– безпечність роботи в цілому;

– зручність та ефективність зорової роботи з екраном в вертикальній площині під кутом 300 від лінії зору, площина екрана при цьому має бути

перпендикулярною нормальній лінії зору користувача.

Клавіатура розміщується на поверхні столу або висувній полиці на відстані 100-300мм від краю, ближчого до користувача. Кут нахилу клавіатури має бути в межах 5-150. Поверхня клавіатури повинна бути матовою з коефіцієнтом відбиття 0,4. клавіші клавіатури мають бути зручними в роботі і м'якими при натисканні (хід всіх клавіш має бути одинаковим з мінімальним опором натискання 0,25Н та максимальним – не більше 1,5Н) [42].

При розміщенні робочих місць з ПК слід дотримуватися вимог, зазначених в ДНАОП 0.00-1.31-99:

– робочі місця розміщуються на відстані не менше 1м від стін з світловими прорізями; – відстань між бічними поверхнями моніторів ПК має бути не менше 1,2м; – відстань між тильною поверхнею монітора одного ПК та екраном монітора іншого ПК має бути не меншою 2,5м.

Вимоги двох останніх пунктів враховуються також при розміщенні робочих місць з ПК в суміжних приміщеннях з урахуванням конструктивних особливостей стін та перегородок.

Загальні принципи організації робочого місця:

– на робочому місці не повинно бути нічого зайвого. Усі необхідні для роботи предмети мають бути поряд із працівником, але не заважати йому;

– ті предмети, якими користуються частіше, розташовуються ближче, ніж ті предмети, якими користуються рідше;

– предмети, які беруть лівою рукою, повинні бути зліва, а ті предмети, які беруть правою рукою – справа;

– якщо використовують обидві руки, то місце розташування пристосувань вибирається з урахуванням зручності захоплювання його двома руками;

– робоче місце не повинно бути захаращене;

– організація робочого місця повинна забезпечувати необхідну оглядовість.

Статичні напруження працівника в процесі праці пов'язані з підтриманням у нерухомому стані предметів і знарядь праці, а також підтриманням робочої пози.

Робоча поза – це основне положення працівника у просторі: зручна робоча поза має забезпечувати стійкість положення корпуса, ніг, рук, голови працівника під час роботи, мінімальні затрати енергії та максимальну результативність праці. Неправильна сидяча поза може викликати застій крові в ногах, а якщо виконується великий обсяг роботи для пальців рук – запалення суглобів.

Організація робочого місця користувача комп'ютера повинна забезпечувати відповідність усіх елементів робочого місця та їх взаємного розташування ергономічним вимогам (рисунок 4.1).

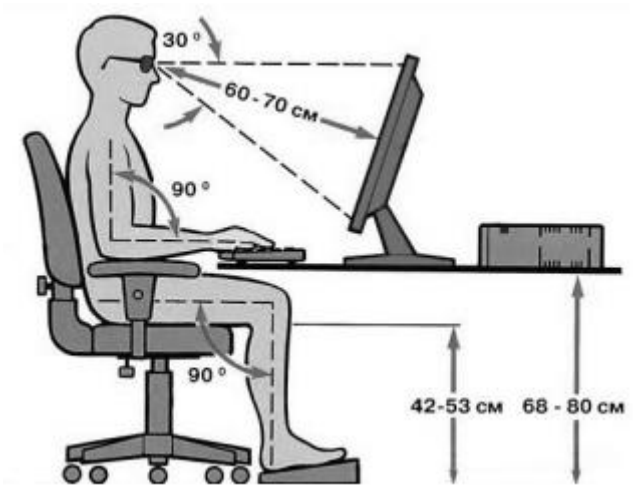

Рисунок 4.1 – Робоче місце і робоча поза користувача ПК

Найпоширенішими у процесі праці є пози сидячи і стоячи. Проектуючи робоче місце, потрібно враховувати, що при виконанні роботи з фізичним навантаженням бажана поза стоячи, а при малих зусиллях – сидячи.

Робоча поза стоячи втомлює людину більше, ніж сидяча. Вона вимагає на 10% більше енергії, спричиняє підвищення артеріального і венозного тиску крові, розширення вен на ногах, пошкодження ступень, викривлення хребта [42].

4.2 Заходи захисту від випромінювань оптичного діапазону

До випромінювання оптичного діапазону відносяться інфрачервоні й

ультрафіолетові хвилі, видиме світло, лазерне випромінювання.

По фізичній природі інфрачервоні промені мають хвильові (довжина хвилі 0,78-540 мкм) і квантові властивості. Генератором випромінювання є будь-яке тіло, температура якого вище абсолютного нуля. За законом Стефана-Больцмана інтегральна густина випромінювання, Вт/м^2, абсолютно чорного тіла пропорційна четвертому ступеню його абсолютної температури [40].

З підвищенням температури тіла змінюється спектральний склад його випромінювання. Чим вища температура тіла, тим коротша довжина хвилі, максимального випромінювання.

Інфрачервона енергія, яка потрапляє на тіло людини, діє передусім на незахищені його частини (лице, руки, шию, груди), причому конвективне тепло впливає на зовнішній шкіряний покрив, тоді як інфрачервоне випромінювання може проникнути на деяку глибину в тканину. При довготривалому перебуванні людини в зоні інфрачервоного випромінювання, як і при систематичній високій температурі наступає різке порушення теплового балансу в організмі. Для вимірювання густини потоку випромінювання на робочому місці застосовують актинометр – прилад, який дозволяє вимірювати густину потоку інфрачервоного випромінювання у діапазоні від 0 до 14кВт/м^2 . Основні види захисту від інфрачервоного випромінювання–захист часом, захист віддалю, усунення джерела тепловиділення, теплоізоляція, охолодження гарячої поверхні, забезпечення тепловіддачі тіла людини та індивідуальні засоби захисту. Потужність випромінювання можна знизити за рахунок конструкторських і технологічних рішень (зміною нагрівання виробів у нагрівальних пічках індукційним нагріванням та ін.) і за рахунок покриття поверхні, яка нагрівається, тепло ізолювальним матеріалом. Для захисту очей застосовують світлофільтри зі спеціального жовто-зеленого або синього скла.

Ультрафіолетове випромінювання змінює склад виробничої атмосфери. Утворюється озон, оксиди азоту і пероксид водню. Короткохвильове випромінювання іонізує повітря, утворює в атмосфері ядра конденсації, які зменшують освітленість робочих місць і призводять до утворення туманів.

Основні засоби захисту. Першочергові заходи – це конструкторські і

технологічні рішення, які виключають генерацію або понижують інтенсивність випромінювання. Спеціальні засоби захисту (екранування джерел випромінювання, фарбування стін у світлі кольори) попереджують розповсюдження і знижують інтенсивність цих випромінювань у виробничих приміщеннях. Очі захищають окулярами або щитками зі склом – світлофільтром. Для захисту шкіри використовують мазі з речовинами – світлофільтрами для цих променів (салол, саліцилово-метиловий ефір та ін.), а також спецодяг з бавовняних тканин і грубововняного сукна. Руки захищають рукавицями [41].

Діапазон довжин хвиль які випромінюють оптичні квантові генератори (ОКГ) – лазери, охоплює видимий спектр і розповсюджується в інфрачервоній і ультрафіолетовій областях.

Найбільш чутливими до дії випромінювання ОКГ є очі. Випромінювання викликають опіки і пошкодження сітківки ока, це може призвести до сліпоти. Небезпечне не тільки пряме випромінювання, але й відбите від стін, обладнання.

Існують спеціальні норми, до яких ввійшли організаційні та інженерно-технічні заходи, які можуть забезпечити зменшення густини потоків енергії (потужності) на робочих місцях до величин, значно менших від допустимих. ОКГ розміщують в окремих або відгороджених приміщеннях. Саме приміщення і обладнання не повинні мати дзеркальної поверхні. Стіни, стелі, обладнання й інші предмети фарбують матовою фарбою з малою сорбційною здатністю. Приміщення повинно мати високу освітленість, а також припливно-витяжну вентиляцію. При розміщенні в одному приміщенні декількох ОКГ їх огороджують ширмами, шторами або екранами, що не пропускають випромінювання. Надійним захистом від випадкового попадання випромінювання на людину є світловод, який екранує промінь на усьому шляху його дії (від ОКГ до мішені) [41].

ВИСНОВКИ

У кваліфікаційній роботі розглянуто проблему СтГр та СтА на аудіо файлах та отримано такі результати.

– зроблено огляд існуючого ПЗ для приховування інформації у файлах WAVE;

– розглянуто проблему СтА в аудіо файлах формату WAVE;

– запропоновано власну методику для визначення наявності стеганографічних вкладень в аудіо файлах на основі статистичних тестів послідовностей молодших біт. Розроблена методика дає змогу виявляти вкладення, котрі виконані при допомозі алгоритму LSB, в деякі види аудіо файлів формату WAVE. У майбутньому планується вдосконалити цю методику та розширити її на всі види аудіо файлів.

– розроблено додаток Hide_Wave_Stego для стеганографічного приховування даних в аудіо файлах формату WAVE за допомогою методу LSB. Програма підтримує WAVE файли будь-якої бітності, ЧД, визначає об'єм контейнера та реалізує багатотомність. Також реалізовано алгоритм вилучення прихованих даних. Для Шифрування даних перед приховуванням застосовується алгоритм XOR. Програма відповідає всім вимогам до стеганографічного ПЗ і може використовуватися для приховування даних в аудіо файлах формату WAVE;

– для порівняльного аналізу у роботі реалізований алгоритм СтА в аудіо файлах з урахуванням відомого методу, заснованого на алгоритмі стиснення, описаного у статті [8];

– проведено порівняльний аналіз цих двох підходів до виявлення вкладень. Виявлено їх відносні переваги та недоліки.

Подальший розвиток цієї роботи полягає в удосконаленні розробленого ПЗ для СтГр в аудіо файлах формату WAVE, розробці універсального методу визначення вкладень в аудіо файли. Також планується розглянути проблему на відео файлах.

ПЕРЕЛІК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Быков, С. Ф., Мотуз, О. В. Основы стегоанализа // Защита информации. – 2000. – №3. – С. 38-41.

2. Стегоанализ графических данных в различных форматах [Електронний ресурс]. – Режим доступа: http://www.sbras.ru/ws/YM2007/12817/paper.html (дата звернення 18.04.2022).

3. Advanced Statistical Steganalysis. R. Böhme, Springer, 2010.

4. Гребенников В.В., История Криптологии & Секретной Связи [Електронний Ресурс]. – Режим доступа: http://cryptohistory.ru/book/chast-6-istoriyasteganografii/61-vstuplenie/ (дата звернення 20.02.2022).

5. Стасюк О. І. Сучасні стеганографічні методи захисту інформації // Захист інформації. — 2011. — Т. 13. — № 1 (50). - С. 56–63.

6. Садов В.С. Компьютерная стеганография – Минск : РИВ, 2014. –172 с.

7. Стеганография в XXI веке. Цели. Практическое применение. Актуальность [Електронний ресурс]. – Режим доступу: https://habrahabr.ru/post/253045/ (дата звернення 20.03.2022).

8. Web Archive [Електронний ресурс]. – Режим доступа – http://web.archive.org/web/20101013075447/www.jamestown.org/programs/gta/single/?tx_ttnews%5Btt_news%5D=1057&tx_ttnews%5BbackPid%5D=182&no_cache=1 (дата звернення 20.04.2022).

9. Юдін О. К. Аналіз стеганографічних методів приховування інформаційних потоків у контейнери різних форматів // Радиоэлектроника и информатика. — Х. : НХНУРЕ, 2015. — № 3. — С. 24-31.

10. Тарасов Д. О. Класифікація та аналіз безкоштовних програмних засобів стеганографії // Інформаційні системи та мережі. Вісник НУ «Львівська політехніка». — 2010. — № 673. — С. 365-374.

11. Стеганографический метод Куттера-ДжорданаБоссена [Електронний ресурс]. - 2018. - Режим доступудо ресурсу:https://habr.com/ru/post/115287/.

12. Саломан А. Криптографія з відкритим ключем. - К.: «Наука», 2013. - 342 с.

13. Конахович Г. Ф. Сучасні методи квантової стеганографії // Захист інформації. — 2011. — Т. 13. — № 2 (51)

14. С.М. Горобець. Основи комп’ютерної графіки. - К.: Центр навч. літератури, 2016. – 232 с.

15. Дацюк Р.К. Метод приховування великого об'єму даних в файлах формату JPEG // «Інтелектуальний потенціал – 2020» , Хмельницький: ПВНЗ УЕП, 2020. – Частина 2. С. 37-42

16. Wu C.-P. Robust audio watermarking for copyright protection / C.-P. Wu, P.- C. Su, K.C.-C. Jay // Proceedings of SPIE, Advanced Signal Processing Algorithms, Architectures, and Implementations IX. - 1999.- Vol. 3807. - P. 387-397

17. Полный список аудио форматов [Електронний ресурс]: Audio Coding. – Режим доступа: http://audiocoding.ru/assets/meta/2021-05-22-wav-file-structure/wav_formats.txt (дата звернення 20.02.2022).

18. С. Ю. Очимов, Стегоанализ аудиофайлов, базирующийся на алгоритмах сжатия // Вестник СибГУТИ. – 2010. – №1 . – С. 33-37.

19. Wave file format – формат звукового файла WAVE [Електронний ресурс]. – Режим доступа: http://microsin.net/programming/pc/wav-format.html (дата звернення 20.04.2022).

20. Структура WAVE файла [Електронний ресурс]: Audio Coding. – Режим доступа – http://audiocoding.ru/article/2021/05/22/wav-file-structure.html (дата звернення 20.02.2022).

21. Імпульсно-кодова модуляція [Електронний ресурс]: – Режим доступа: https://gos2014.at.ua/index/impulsno_kodova_moduljacija_ikm_diferencialna_impulsno_kodova_moduljacija_dikm/0-35 (дата звернення 20.02.2022).

22. Гурский, Д. И. ActionScript 2: программирование во Flash MX / Д. И. Гурский. – Санкт-Петербург : Питер, 2004. – 860 с.

23. Е.Л. Зорин, Н.В. Чичиварин. Стеганография в САПР : учебное пособие. – Москва : МГТУ им. Н.Э. Баумана, 2013. – 90 с.

24. Забелин, М. А. Стегоанализ аудиоданных на основе методов сжатия // Вестник СибГУТИ. – 2010. – №1 . – С. 41-46.

25. DeepSound. [Електронний ресурс]. – Режим доступа:

http://jpinsoft.net/DeepSound/Overview.aspx (дата звернення 20.02.2022).

26. Xiao Steganography. [Електронний ресурс]. – Режим доступа: http://download.cnet.com/Xiao-Steganography/3000-2092_4-10541494.html (дата звернення 20.02.2022).

27. SilentEye. [Електронний ресурс]. – Режим доступа: http://silenteye.v1kings.io/index.html?i1s1 (дата звернення 20.02.2022).

28. StegoStick beta. [Електронний ресурс]. – Режим доступа: https://sourceforge.net/projects/stegostick/ (дата звернення 20.02.2022).

29. Кокорин П. П. О методах стегоанализа в аудиофайлах // Труды СПИИРАН. – 2007. – №4. – С. 239-246.

30. Ben-4D. [Електронний ресурс]. – Режим доступа: https://sourceforge.net/projects/ben4dstegdetect/ (дата звернення 20.02.2022).

31. Digital Invisible Ink Toolkit. [Електронний ресурс]. – Режим доступа: https://sourceforge.net/projects/diit/?source=typ_redirect (дата звернення 20.02.2022).

32. Raggo M.T. Data Hiding: Exposing Concealed Data in Multimedia, Operating Systems, Mobile Devices and Network Protocols. M.T. Raggo. – Массачусетс : Syngress, 2012. – 270 с.

33. Статистическая проверка случайности двоичных последовательностей методами NIST [Електронний ресурс]. –Режим доступа: https://habrahabr.ru/ company/securitycode/blog/237695/ (дата звернення 20.02.2022).

34. Пакет статистичних тестів NIST [Електронний ресурс]: – Режим доступа: http://csrc.nist.gov/groups/ST/toolkit/rng/documentation_software.html

35. Алгоритм XOR [Електронний ресурс]. – Режим доступа: https://en.wikipedia.org/wiki/XOR_swap_algorithm (дата звернення 20.03.2022).

36. Стеблюк М.І. Цивільна оборона: Підручник. – Знання, 2006. – 487 с.

37. Толок А.О. Крюковська О.А. Безпека життєдіяльності: Навч. посібник. – 2011. – 215 с.

38. Агєєв Є .Я. Основи охорони праці: Навчально-методичний посібник для самостійної роботи по вивченню дисципліни – Львів: «Новий Світ – 2000», 2009. – 404 с.

39. Основи охорони праці: Підручник.; 3-те видання, доповнене та перероблене / За ред. К. Н Ткачука. – К.: Основа, 2011. – 480 с.

40. Зеркалов Д.В. Безпека життєдіяльності та основи охорони праці. Навчальний посібник. К.: «Основа». 2016. – 267 с.

41. Яремко З. М. Безпека життєдіяльності: Навч. посіб. — Львів., 2005. – 301 с.

42. Желібо Є. П. Заверуха Н.М., Зацарний В.В. Безпека життєдіяльності. Навчальний посібник. – К.; Каравела, 2004. -328 с.