Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ДОПУСТИТИ ДО ЗАХИСТУ:
В.о завідувача кафедри
кібербезпеки та захисту інформації
_____ Іван ПАРХОМЕНКО
“__” червня 2023 р.

ПОЯСНЮВАЛЬНА ЗАПИСКА

кваліфікаційної роботи

галузь знань 12 Інформаційні технології
(шифр і назва галузі знань)
спеціальність 125 «Кібербезпека»
(код і назва спеціальності)
освітній ступень бакалавр
освітня програма Кібербезпека
(назва освітньо-професійної програми)
на тему: Засіб захисту інформації в аудіофайлах

Виконавець: студент IV курсу, групи КБ-42

_____ **Марк ГАВРИЩУК**
(підпис) (ім’я прізвище)

	Ім’я, прізвище	**Підпис**
Керівник роботи	Олександр ТОРОШАНКО	

Нормоконтроль	Андрій БІГДАН	

Київ 2023

Міністерство освіти і науки України
Київський національний університет імені Тараса Шевченка

Факультет інформаційних технологій
Кафедра кібербезпеки та захисту інформації

ЗАТВЕРДЖЕНО:
В.о. завідувача кафедри
кібербезпеки
та захисту інформації
_____ Сергій ТОЛЮПА
«24» жовтня 2022 р.

ЗАВДАННЯ
на виконання кваліфікаційної роботи

спеціальності 125 Кібербезпека
(код і назва спеціальності)

освітньої програми Кібербезпека
(назва освітньої-професійної програми)

Студенту **КБ-42** **Гаврищук Марк Вячеславович**
(група) (прізвище ім'я по-батькові)

Тема кваліфікаційної роботи Засіб захисту інформації в аудіофайлах

1. ПІДСТАВИ ДЛЯ ПРОВЕДЕННЯ РОБОТИ

Тема кваліфікаційної роботи затверджена на засіданні кафедри кібербезпеки та захисту інформації протокол №3 від 20.10.2022 р.

2. ВИХІДНІ ДАНІ ДЛЯ ПРОВЕДЕННЯ РОБІТ

Методика приховування інформації, архітектура засобу захисту інформації

3. ЗМІСТ РОЗРАХУНКОВО-ПОЯСНЮВАЛЬНОЇ ЗАПИСКИ

Необхідно дослідити можливості аудіофайлів. Проаналізувати відомі засоби захисту інформації в аудіофайлах, вмісту аудіофайлу. Порівняння існуючих засобів захисту з використанням стеганографії. Проаналізувати процес накладання водяного знаку на інформацію в аудіофайл та на аудіофайл в цілому. Дослідити наявні вразливості інформації в аудіофайлі та створити новий засіб захисту інформації в аудіофайлі.

4. ВИМОГИ ДО РЕЗУЛЬТАТІВ ВИКОНАННЯ РОБОТИ

Практична цінність Розроблена методика стенографічного захисту інформації в аудіофайлах

5. ДАТА ВИДАЧІ ЗАВДАННЯ

Дата видачі завдання: 24 жовтня 2022 року

Завдання видав _____ (підпис) Олександр ТОРОШАНКО (ім'я, прізвище)

Завдання прийняв до виконання _____ (підпис) Марк ГАВРИЩУК (ім'я, прізвище)

КАЛЕНДАРНИЙ ПЛАН

№ п/п	Найменування етапів робіт	Строки виконання робіт (початок-кінець)	Відмітка про виконання
1	Уточнення постановки задачі	24.10.2022 – 20.01.2023	*виконано*
2	Аналіз літератури	21.01.2023 –20.02.2023	*виконано*
3	Обґрунтування вибору рішення	24.02.2023 – 04.03.2022	*виконано*
4	Дослідження поняття аудіофайлу	05.03.2023 – 21.03.2023	*виконано*
5	Огляд існуючих стеганографічних засобів захисту інформації в аудіофайлах	22.03.2023 – 09.04.2023	*виконано*
6	Дослідження методів накладння водяного знаку на аудіофайл	10.04.2023 – 15.04. 2023	*виконано*
7	Дослідження вразливостей метаданих в аудіофайлах	16.04.2023 – 22.04. 2023	*виконано*
8	Опис розроблюваного засобу захисту	23.04. 2023 – 30.04. 2023	*виконано*
9	Побудова методу, що буде використовуватись в засобі захисту інформації в аудіофайлах	1.05.2023 – 27.05.2023	*виконано*
10	Реалізація консольного додатку для конвертації даних	28.05.2023 – 05.06.2023	*виконано*
11	Оформлення пояснювальної записки	06.06.2023 – 07.06.2023	*виконано*

Завдання видав _____ (підпис) Олександр ТОРОШАНКО (ім'я, прізвище)

Завдання прийняв до виконання _____ (підпис) Марк ГАВРИЩУК (ім'я, прізвище)

Термін подання кваліфікаційної роботи до ЕК 12 червня 2023 року

РЕФЕРАТ

Кваліфікаційна робота складається зі вступу, трьох розділів, висновку, списку джерел, які були використані під час написання роботи, додатків, має 70 сторінок основного матеріалу, 38 рисунків, 7 таблиць та 2 додатки. Список джерел, які були використані містить 22 найменування та займає 2 сторінки.

Метою роботи є створення засобу захисту інформації в аудіофайлах.

Об'єктом дослідження є процес захисту інформації в аудіофайлах.

Предметом дослідження є набір механізмів, що реалізують процес захисту інформації в аудіофайлах.

Методи дослідження:

- аналіз відкритих джерел;
- порівняння засобів захисту інформації в аудіофайлах;
- моделювання нового засобу захисту інформації;

Практичною цінністю є розробка методики стенографічного захисту інформації шляхом зашифровування тексту в звук з подальшим оформленням у відповідний формат аудіофайлу.

Новизна: процес приховування та захисту інформації в аудіофайлі з використанням нового підходу – конвертація повідомлення в музику, накладання шумів та водяного знаку.

Ключові слова: сиквенсор, аудіофайл, ревербератор, аудіосигнал, октава, нотне поле, MIDI-формат, синтезатор, водяний знак, дискретизація.

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ ТА СКОРОЧЕННЬ

MP3	–	MPEG-1 Audio Layer III
WAV	–	Waveform Audio File Format
FLAC	–	Free Lossless Audio Codec
LSB	–	Least Significant Bit
FL	–	Fruity Loops
ІКМ	–	Імпульсно-кодова модуляція
WMA	–	Windows Media Audio
CD	–	Compact Disc
DVD	–	Digital Versatile Disc
MSD	–	Million Song Dataset
FFT	–	Fast Fourier Transform

ЗМІСТ

ВСТУП 8
РОЗДІЛ 1 ОПИС ПОНЯТТЯ АУДІОФАЙЛУ ТА СТЕГАНОГРАФІЇ В АУДІОФАЙЛАХ 12
1.1 Історія виникнення формату MP3 12
1.2 Порівняння існуючих аудіоформатів 14
1.3 Аналіз форматів аудіофайлів 18
1.4 Поняття стеганографії 20
1.5 Приховування даних в аудіофайлах 21
1.6 Опис стеганографії в аудіофайлах 23
1.7 Безпека аудіофайлів 25
1.8 Метод DWT 27
Висновки до першого розділу 28
РОЗДІЛ 2 ЦІЛІСНІСТЬ ТА КОНФІДЕНЦІЙНІСТЬ ІНФОРМАЦІЇ В АУДІОФАЙЛАХ 29
2.1 Важливість захисту інформації в аудіофайлах 29
2.2 Цілісність файлу з вотермаркою 31
2.3 Процедури вбудовування аудіо водяного знаку 33
2.4 Класифікація та проблематика вотермарки 37
2.6 Узгодження метаданих 39
2.7 Вирівнювання аудіо до синтезованого MIDI 40
2.8 Стеганографія в аудіофайлах 42
2.9 Аналіз можливостей застосування стеганографії до аудіофайлів 44
Висновки до другого розділу 45
РОЗДІЛ 3 ЗАСІБ ЗАХИСТУ ІНФОРМАЦІЇ В АУДІОФАЙЛАХ 46
3.1 Загальний огляд ПЗ 46
3.2 Інтерфейс і функції FL Studio 49
3.3 Огляд програмної функції Edison 54
3.4 Опис процесу захисту інформації в аудіофайлах 56
3.5 Опис процесу приховування в аудіофайлах при використанні ПЗ Edison та FL Studio 57

3.6 Процес приховування інформації 59
3.7. Процес декодування повідомлення в аудіофайлі. 64
Висновки до третього розділу 67
ВИСНОВКИ 68
СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ 69
ДОДАТОК А 71
ДОДАТОК Б 72

ВСТУП

Захист інформації став життєво важливим аспектом нашого повсякденного життя. Зберігаючи інформацію у різноманітній формі та передаючи за допомогою різних технологій та пристроїв, є потреба в забезпеченні її цілісності та конфіденційності. Одним із способів досягнення цих потреб є використання шифрування.

Шифрування використовувалося століттями для захисту інформації, але з появою цифрових технологій воно стало ще більш важливим інструментом захисту даних. Інформація в аудіофайлах – це один із типів даних, які потребують захисту, і є кілька способів, як цього можна досягти.

Історію захисту інформації в аудіофайлах можна простежити з перших днів появи цифрового аудіо. У 1980-х роках впровадження цифрового аудіозапису дозволило створювати високоякісні аудіофайли, які можна було легко розповсюджувати та обмінюватися ними. Однак, у міру того, як ці файли ставали все більш популярними, потреба в захисті стала очевидною.

Однією з перших форм захисту інформації в аудіофайлах стало використання цифрових водяних знаків. Ці водяні знаки вбудовувалися в аудіофайл і могли бути використані для ідентифікації власника файлу. Однак цей метод не був надійним і міг бути легко видалений.

Сьогодні існують більш досконалі методи захисту інформації в аудіофайлах, зокрема шифрування. Шифрування передбачає зашифрування даних у файлі таким чином, щоб їх неможливо було прочитати без відповідного ключа для розшифрування. Цей метод є високоефективним для захисту інформації в аудіофайлах від несанкціонованого доступу.

Важливість захисту інформації в аудіофайлах важко переоцінити. Аудіофайли використовуються в різних сферах, включаючи виробництво музики, кіно- і телепродукцію, і навіть в медицині. У кожному з цих випадків безпека аудіофайлів має вирішальне значення для успіху проекту.

Отже, захисту інформації в аудіофайлах є критично важливим аспектом інформаційної безпеки. З постійним розвитком технологій важливо, щоб ми розробляли нові та інноваційні способи захисту наших даних. Шифрування - це лише один з багатьох інструментів, доступних нам у цьому відношенні, і він, безсумнівно, буде відігравати все більш важливу роль у найближчі роки.

Оскільки з плином часу, всі відомі на сьогодні засоби захисту інформації в аудіофайлах вивчаються шахраями і хакерами під мікроскопом, частину з цих засобів і методів вже можна зламати, маючи певні навички і знання роботи певних механізмів, які використовуються у методах, засобах захисту інформації в аудіофайлах.

Але незмінним в аудіофайлах, якщо річ йде саме про музичні композиції чи просто музику, це використання нот. Їх існує всього сім. З одного боку можна сказати, що якщо інформація захищатиметься нотами, то й легше буде перехопити, декодувати повідомлення і тоді можливий несанкціонований доступ до інформації. Таким чином порушуватиметься конфіденційність інформації та цілісність інформації. Але існує прислів’я: «Хочеш заховати щось, поклади на видне місце». Тому й засіб захисту інформації в аудіофайлах розроблений на базі нот і шифрування інформації через ноти є цікавим рішенням щодо захисту конфіденційності інформації аудіофайлів, адже на перший погляд вміст аудіофайлу не буде очевидним.

Ідея використання приховування повідомлення у вигляді музичних нот для захисту інформації в аудіофайлах є досить оригінальною та інноваційною. Така система захисту може мати кілька потенційних плюсів, зокрема:

Високий рівень захисту. Використання музичних нот як символів може зробити захист інформації більш надійним, оскільки кожен знак має свій унікальний звуковий відтінок, що може бути використано як ключ для розшифрування. Такий метод може дозволити забезпечити високий рівень захисту від несанкціонованого доступу до інформації.

Легкість використання. Якщо система захисту буде налаштована правильно, то використання музичних нот як символів може бути досить простим та зручним.

Захищена інформація може бути легко передана та отримана, що дозволить ефективно використовувати цю систему в різних сферах.

Широкі можливості застосування. Система захисту інформації на основі музичних нот може мати широкі можливості застосування в різних галузях, наприклад, в музичній та фільмовій індустрії. Такий захист може допомогти захистити авторські права та запобігти незаконному поширенню музичних творів чи фільмів.

Схована інформація. Оскільки музичні ноти можуть бути використані як символи захисту, то захищена інформація може бути прихована під мелодією. Такий метод може забезпечити додатковий рівень захисту, оскільки захищена інформація може бути складніше виявити незаконним користувачам.

Тому новий засіб захисту інформації в аудіофайлах при доопрацюванні і створенні повноцінної інформаційної системи допоможе захищати інформацію просто пересилаючи аудіофайл отримувачу. Проста і надійність, яка полягає у легкому захисту і шифруванні інформації та дешифруванні і отримання самого повідомлення вкладеного в аудіофайл є актуальним і потрібним в наш час.

Метою роботи є створення засобу захисту інформації в аудіофайлах.

Для досягнення успіху при створенні засобу захисту інформації в аудіофайлах поставлено наступні завдання:

- Дослідження поняття аудіофайлу
- Огляд існуючих стеганографічних засобів захисту інформації в аудіофайлах
- Дослідження методів накладання водяного знаку на аудіофайл
- Дослідження вразливостей метаданих в аудіофайлах
- Опис розроблюваного засобу захисту
- Побудова методу, що буде використовуватись в засобі захисту інформації в аудіофайлах
- Реалізація консольного додатку для конвертації даних

Об’єктом дослідження є процес захисту інформації в аудіофайлах.

Предметом дослідження є набір існуючих механізмів та засобів захисту інформації в аудіофайлах, що реалізують процес захисту зашифрованої інформації в аудіофайлах.

Методи дослідження:

- аналіз відкритих джерел;
- порівняння засобів захисту інформації в аудіофайлах;
- моделювання нового засобу захисту інформації;

Практичною цінністю є розроблена нова методика стеганографічного засобу захисту інформації шляхом зашифровування тексту в звук з подальшим оформленням у відповідний формат аудіофайлу за використання сиквенсору, програмних засобів для детекту нот в музиці і дешифрування зашифрованого тексту в аудіо файлі, який містить музику.

РОЗДІЛ 1
ОПИС ПОНЯТТЯ АУДІОФАЙЛУ ТА СТЕГАНОГРАФІЇ В АУДІОФАЙЛАХ

1.1 Історія виникнення формату MP3

Історія цього формату почалася в 70-х роках двадцятого століття, тоді коли в університеті «Ерланген-Нюрнберга» зібралася група студентів-однодумців під керівництвом професора Дітера Зайцера. Група мала на меті вирішити проблему надійної передачі людського мовлення телефонними лініями. Коли однак з появою оптоволоконних кабелів і цифрової мережі зв'язку ISDN відбулася глобальна інформаційно-комунікаційна революція.

Тому німецька команда звернула свою увагу на ефективне кодування та стиснення аудіосигналів. Карлхайнц Бранденбург, пізніше відомий як «Батько MP3», був першим вченим, який зрозумів, що оптимальне стиснення мови неможливе без урахування особливостей людського слуху.

Подальша історія розвитку MP3 була досить стрімкою – у 1987 році Університет «Ерланген-Нюрнберга» та Інститут «Фраунгофера» створили дослідницьке партнерство, в якому взяли участь американські та канадські технологічні компанії «AT&T Bell Labs» і «Thomson Corporation». У 1988 році було створено перші практичні прототипи нового формату, а рівно через рік, у квітні 1989 року, інститут «Фраунгофера» отримав німецький патент на технологію MP3.

Цікаво, що ця розробка була протестована на конкретній пісні. Це була популярна пісня "Tom's Diner" співачки Сюзанни Веги. Бранденбург слухав цю пісню знову і знову і використовував її для вдосконалення алгоритму стиснення. Після завершення цієї роботи він жартома назвав Сюзанну Вегу "матір'ю MP3".

Офіційною датою появи MP3 вважається 1995 рік, коли було створено перший MP3-файл за допомогою першого у світі кодера. Незабаром після цього винахідники випустили перший практичний MP3-плеєр під назвою WinPlay3. Відтоді мільйони

людей по всьому світу змогли створювати та відтворювати файли цього нового формату на своїх комп'ютерах.

У 1996 році формат був зареєстрований у США [1], а невдовзі після цього в Інтернеті з'явився портал mp3.com. Спочатку він слугував місцем для збору необхідної інформації про нові розширення, деталі плеєрів та інші можливості. Згодом цей ресурс перетворився на найбільший у світі легальний аудіоархів. Водночас, поява MP3 ознаменувала "золотий вік" музичного піратства в Інтернеті, оскільки MP3-файли можна було легко копіювати та поширювати серед користувачів через веб-сайти-агрегатори.

Принцип кодування MP3 простий. При аналізі звукової доріжки видаляються частини аудіопотоку, які ледве розрізняються людським вухом. Чим вищий ступінь стиснення, тим більше деталей видаляється і тим нижча якість звуку.

MP3 стали невід'ємною частиною цифрової епохи завдяки своїм компактним розмірам. Вони можуть відтворюватися на всіх поширених операційних системах і підтримуються всіма без винятку портативними і стаціонарними аудіопристроями. Пересічний меломан, без сумніву, оцінить збільшену місткість дисків і можливість зберігати більше улюблених пісень, навіть якщо він не помітить різниці в якості звуку.

Винахід формату MP3 визнаний найуспішнішим проектом в німецькій цифровій історії. За оцінками експертів, розвиток цієї технології створив у Німеччині понад 10 000 робочих місць, а податки від комерційного використання алгоритму MP3 становлять понад 300 мільйонів євро на рік. Німці витрачають понад 1,5 мільярда євро на рік на MP3-плеєри та інші продукти.

Однією з причин "занепаду" цієї успішної технології є розробка інших, більш досконалих кодеків. Прикладами є формат WMA від Microsoft, Free Lossless Audio Codec (FLAC) та стандарт Apple Lossless, який використовують споживачі продукції Apple.

Але до сих пір на сьогодні цей формат є найвідомішим і вважається стандартом, шаблоном і використовується цей формат частіше за інші.

1.2 Порівняння існуючих аудіоформатів

З такою великою кількістю доступних аудіоформатів, постає питання вибору. Який з них варто використовувати? Який з них краще? В чому різниця?

Люди споживають цифрове аудіо протягом багатьох років та й комп'ютер користувача буде наповнений цифровими музичними файлами у різних форматах. Здебільшого не потрібно про це думати, але існує таке питання чи може медіаплеєр їх читати. Але коли, в процесі занурення в це аудіофільське хобі, ви можете почати задаватися питанням: "Який аудіоформат найкращий?".

Формат аудіофайлів – це формат файлів для зберігання цифрового аудіо на комп'ютерних системах, таких як ПК, мобільний телефон тощо. Аудіоінформація зберігається у файлі у вигляді бітів і може називатися бітовою структурою. Цей бітовий макет може бути нестисненим або стисненим за допомогою кодування без втрат або з втратами.

Не всі формати аудіофайлів однакові. Частота дискретизації та розрядність визначають роздільну здатність, частотний діапазон і динамічний діапазон звуку. Стандартом для аудіо CD якості є частота дискретизації 44,1 кГц і розрядність 16 біт.

Все, що нижче цих значень, не вважається аудіо високої чіткості (HD) або високої роздільної здатності. Багато форматів аудіофайлів використовують параметри, нижчі за ті, що вважаються HD.

Вся музика повинна мати частоту дискретизації щонайменше 44,1 кГц (44100 Гц). Ця частота захоплює частоти до 22050 Гц [2], що трохи вище межі чутності людського слуху (~20 кГц).

Формати аудіофайлів можна розділити на 3 основні групи:

- Нестиснутий аудіоформат
- Стиснутий аудіоформат без втрат
- Формат стисненого аудіо з втратами

В таблиці 1.1 наведена інформація, яка відображає кожен формат аудіофайлу та тип його кодування:

Таблиця 1.1

Порівняння форматів аудіофайла за типами кодування

Формат	Кодування
WAV	Без стиснення
AIFF	Без стиснення
ALAC	Без втрат
FLAC	Без втрат
MP3	З втратами
AAC	З втратами
WMA	З втратами
OGG	З втратами

Нестиснутий аудіоформат.

Якщо ваші аудіофайли нестиснуті, вони на 100% побітно ідентичні тому, в якому вигляді їх створив виробник, тобто цифрові копії. Тобто якщо розглядати зі сторони цілісності інформації в аудіофайлі певного нестиснутого аудіоформату то можна сказати, що вибір нестиснутого аудіоформату це гарне рішення закриваючи питання однієї з трьох властивостей інформації – цілісності.

DVD-диски були дуже популярними, їх використовували для зберігання, оброки та передачі певної інформції, а особливо аудіо-інформації.

Нестиснуті формати створюються за допомогою імпульсно-кодової модуляції, ІКМ. Це той самий формат, що використовується для компакт-дисків та DVD-дисків.

Отже, якщо ваша майстер-студія має такі біти:

1100110011000000

Доріжка у форматі ІКМ міститиме такі самі біти:

1100110011000000

На рисунку 1.1 зображена дискретизація та квантування сигналу для імпульсно-кодової модуляції. На рисунку ми бачимо сінусоїду. Тобто з цього випливає пояснення чому біти будуть одні й ті самі, адже мінусових значень біта не може бути.

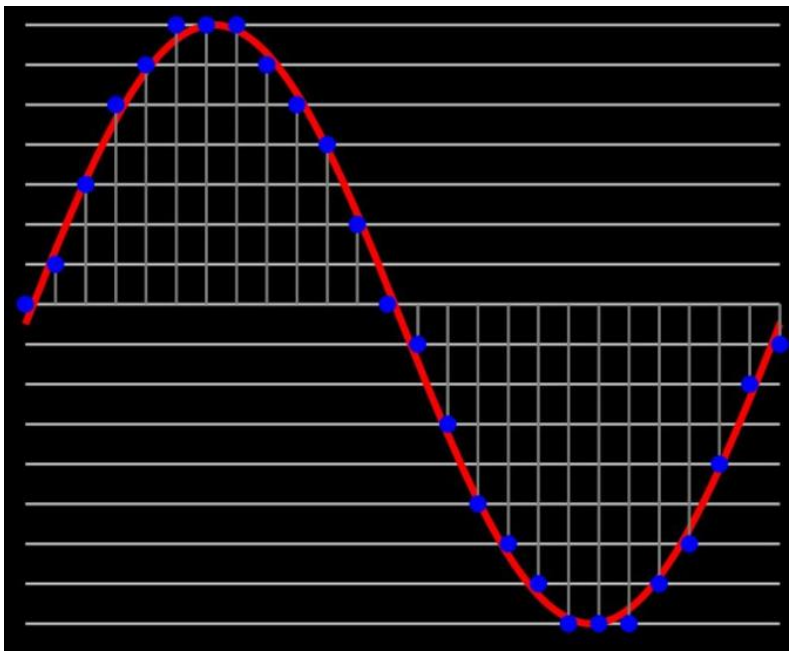

Рисунок 1.1 – Дискретизація та квантування сигналу для ІКМ

Великий розмір файлу. Оскільки ІКМ не стискається, нестиснуті файли займають величезний обсяг дискового простору. Єдиним придатним портативним носієм для зберігання таких файлів є диски CD, DVD або навіть Blu-ray.

Формат стисненого аудіо без втрат.

Кодування без втрат або стиснення без втрат дає змогу отримати найкраще з обох світів. Файли без втрат стискаються, зменшуючи розмір файлу, що полегшує їх зберігання та розповсюдження.

Під час відтворення вони можуть бути декодовані до початкового нестисненого стану без погіршення якості сигналу.

Потрібно уявити, що це наче ZIP-архів. Коли ви запаковуєте документи, вони стискаються і розмір файлу зменшується. Коли ви "розархівуєте" або декодуєте їх, оригінальні документи відтворюються без жодних змін.

Як працює стиснення без втрат?

Стиснення без втрат здається магією. Як ви викидаєте дані, але при цьому точно відтворюєте їх назад. Стиснення з втратами теж втрачає дані, і вони не можуть бути відновлені до початкової копії.

Виявляється, стиснення без втрат видаляє дані в розумний спосіб, який дозволяє відтворити оригінальну копію. Це відбувається шляхом виявлення закономірностей і представлення даних у коротший спосіб.

Існують складні статистичні аналізи для виявлення патернів для стиснення, але ми наведемо простий приклад нижче, щоб продемонструвати, як розумно видаляти дані.

Біти для основної студійної доріжки:

1100011110000000

Кодер без втрат може відкинути всі 0 і замінити їх символом, який займає менше місця, таким чином, зменшуючи розмір файлу:

11___1111_____

Коли декодер отримає вищезгадані біти, він знову замінить символ на 0, щоб відтворити вихідні біти.

Окрім використання для прослуховування музики, формати без втрат корисні для архівування аудіофайлів. У процесі стиснення дані не втрачаються. Хоча ви не отримуєте такого значного зменшення розміру, як при стисненні з втратами, стиснення без втрат все одно дозволяє досягти вражаючого зменшення розміру файлу до 60%.

Формат стиснення аудіо з втратами.

Кодування з втратами або стиснення з втратами використовує психоакустичний аналіз. Алгоритми виявляють вміст, який вважається нечутним через маскування (звуки не чути через інші звуки). Потім він відкидає цю інформацію.

Перевага полягає в тому, що розмір файлу можна значно зменшити, потенційно до 1/10 від початкового розміру. Однак, при цьому неможливо отримати втрачені дані і відновити файл у вихідному форматі без стиснення пізніше.

Недоліком цього є те, що це впливає на якість музики. Показником якості потокового аудіо є бітрейт, який вимірюється в кілобітах на секунду, кбіт/с.

Формула бітрейту = Частота дискретизації х Розрядність х Кількість каналів.

У таблиці 1.2 можна побачити бітрейт різних аудіоформатів і порівняти. Порівнювати слід за принципом «чим більше – тим краще».

Таблиця 1.2

Порівняння бітрейту аудіоформатів

Формат	**Максимальний бітрейт**
MP3	320kbps
AAC	320kbps
OGG	500 кбіт/с
WMA	576 кбіт/с

При бітрейті 320 кбіт/с або вище, залежно від якості обладнання для відтворення, важко відрізнити закодований файл з втратами від нестисненого.

Чим нижчий бітрейт, тим більше даних втрачається, тим гірша якість звуку.

Для декого ця втрата якості не є проблемою, а іноді взагалі непомітна. Для випадкового прослуховування це може бути прийнятним. [4]

1.3 Аналіз форматів аудіофайлів

WAV. Тип стиснення: без стиснення

WAV розшифровується як Waveform Audio. Файли цього формату також називають хвильовими, їх розширення – ".wav". Розроблений компаніями IBM і Microsoft, він був одним з перших типів аудіофайлів, розроблених для ПК.

Хоча він може бути контейнером для стиснених або нестиснених файлів, його зазвичай використовують як контейнер для нестиснених ІКМ-файлів, щоб їх можна було відтворювати в Windows.

Підтримує частоту дискретизації до 192 кГц, розрядність до 32 біт. Це формат високої чіткості, без додаткової обробки чи кодування.

Оскільки він існує з 1991 року, він сумісний з усіма плеєрами, апаратним і програмним забезпеченням, яке може працювати з цифровими файлами.

Єдиним недоліком усіх нестиснутих форматів є те, що розмір файлів може бути великим.

Як наслідок, вони займають більше місця на жорсткому диску комп'ютера. Вони також займають більше часу для завантаження/вивантаження в Інтернеті і використовують велику пропускну здатність.

Переваги:

- Точна копія оригінального запису
- Забезпечує найвищу якість сигналу

Недоліки:

- Великий розмір файлу
- Потребує високої пропускної здатності для передачі через інтернет/ Bluetooth.

FLAC. Тип стиснення: без втрат

FLAC розшифровується як Free Lossless Audio Codec. Він має відкритий вихідний код.

FLAC-файли мають можливість швидше передаватися в потоковому режимі та декодуватися. Це не завжди має значення, окрім випадків, коли ви граєте в ігри або переглядаєте синхронізацію фільмів/діалогів.

У таких ситуаціях вам потрібна якомога менша затримка. Затримка - це часова затримка, що виникає через обробку звуку.

Існує 9 різних рівнів стиснення FLAC, які починаються від 0 до 8. Чим вищий рівень, тим вищий ступінь стиснення, але він супроводжується меншою швидкістю кодування. Швидкість декодування, з іншого боку, більш-менш однакова для всіх рівнів.

Переваги:

- Менший розмір файлів, ніж у нестиснутих (але більший, ніж у файлах з втратами)
- Звук порівнянний з нестиснутими файлами
- Відкритий вихідний код

MP3. Тип стиснення: з втратами

MPEG-1 Audio Layer 3, або MP3 - один з найпопулярніших форматів стиснутих файлів з втратами. Представлений у 1993 році, він швидко став неймовірно популярним.

Компактний розмір файлу дозволив швидко поширювати його в Інтернеті, пропускна здатність якого на той час була набагато нижчою.

Усі MP3-файли спочатку створюються у нестислому форматі [3], наприклад, WAV або AIFF. Вони обробляються і створюються за допомогою кодера. Бітрейт, встановлений у кодері, визначає роздільну здатність і якість музики, яку ви чуєте.

320 кбіт/с - це найвища роздільна здатність, яку може мати MP3-файл. Якщо ви не використовуєте високоякісне обладнання для відтворення, важко відрізнити файл, закодований з такою роздільною здатністю, від аудіо CD-якості.

Основною перевагою MP3 є компактний розмір файлу, сумісність з цифровими медіаплеєрами та хороша якість звуку, якщо кодувати зі швидкістю 320 кбіт/с.

Переваги:

- Компактний розмір файлів
- Універсальна сумісність з апаратним та програмним забезпеченням
- Низька пропускна здатність, необхідна для передачі через інтернет/ Bluetooth
- Достатньо добре для використання на платних платформах потокової музики

Недоліки:

- Низька якість звуку MP3 може бути помітною на високоякісній апаратурі для відтворення
- Низька якість звуку може не в точності передати частоти

1.4 Поняття стеганографії

Стеганографія – це наука про приховування інформації в іншій інформації, зазвичай у цифрових медіафайлах, таких як зображення, аудіо та відео. Стеганографія

відрізняється від криптографії, яка захищає дані шляхом їх шифрування, оскільки вона не тільки шифрує повідомлення, але й робить їх невидимими для сторонніх осіб.

Що таке стеганографія. Стеганографія використовує методи та алгоритми для приховування даних у стегоконтейнерах (прихованих носіях інформації). Цього можна досягти, змінюючи дрібні деталі стегоконтейнера [4], такі як LSB (найменший значущий біт) у бітовому представленні даних, які потрібно приховати.

Як працює стеганографія Стеганографія ґрунтується на тому, що її неможливо виявити. Це означає, що секретні дані має бути надзвичайно важко побачити [5] звичайному спостерігачеві. Використовуючи певні методи, можна розмістити секретну інформацію на стеганоконтейнері так, що зовнішній спостерігач не зможе виявити наявність додаткових даних.

Види стеганографії. Існують різні методи стеганографії:

- текстова стеганографія
- стеганографія зображень
- відеостеганографія
- аудіостеганографія.

Стеганографія в аудіофайлах – це метод приховування інформації в аудіосигналах. Він використовує характеристики аудіофайлу для приховування даних без помітної зміни якості звуку [6]. Оскільки виявлення змін в звуці, стеганографія в аудіофайлах може бути ефективним засобом захисту конфіденційної інформації.

1.5 Приховування даних в аудіофайлах

Як приховати дані в аудіофайлах:

Існує кілька методів приховування даних в аудіофайлах. Один з найпоширеніших методів - це використання методу LSB (найменш значущого біта), де приховані дані вкладаються в менш значущі біти аудіофайлу без впливу на якість звуку. Інші методи включають зміну амплітуди сигналу, використання фазових зміщень або використання спеціально сформованих шумів для приховування даних.

Різні методи приховування даних в аудіофайлах:

У стеганографії в аудіофайлах використовуються різні методи для приховування даних. Крім методу LSB, існують методи, які використовують спектральні характеристики звукового сигналу, такі як частотні перетворення, фазовий спектр та амплітудні модуляції. Деякі методи використовують інформацію про властивості аудіофайлу [7], такі як маска звукової моделі людини, що дозволяє ефективно приховати дані.

Розділ повинен бути добре написаний і добре організований. Він повинений надавати чіткий і стислий огляд теми. Розділ також повинен бути підкріплений літературними даними. Розділ повинен бути написаний у спосіб, доступний для широкої аудиторії.

Переваги використання стеганографії в аудіофайлах:

- Непомітність: однією з головних переваг стеганографії в аудіофайлах є здатність приховати дані без помітних змін в якості аудіо. Передача конфіденційної інформації може відбуватись непомітно для сторонніх спостерігачів.
- Великий обсяг прихованої інформації: аудіофайли можуть містити значну кількість даних, що може бути прихована. Завдяки великому обсягу аудіофайлу, стеганографія в аудіофайлах може передавати більшу кількість інформації порівняно з іншими типами медіа-файлів.
- Відсутність сумісного вражаючого алгоритму: захист інформації, прихованої в аудіофайлах, може бути ефективним, оскільки не існує загальновизнаного алгоритму для виявлення прихованих даних. Це робить стеганографію в аудіофайлах викликом для зловмисників, які намагаються розкрити приховану інформацію.

Недоліки використання стеганографії в аудіофайлах:

- Вплив на якість звуку: процес приховування даних може впливати на якість аудіофайла. Чим більше даних приховується, тим більше може бути помітних змін у якості звуку. Це може спричинити спад у звучанні аудіофайла, зокрема поява шуму або артефактів.

- Підвищення ризику виявлення: незвичайні шаблони або зміни в аудіофайлах можуть привернути увагу аналітиків або зловмисників.

Принцип стеганографії виник і продовжує розвиватися як механізм захисту інформації в усіх типах файлів. Аудіостеганографія використовується для приховування аудіосигналів. Основні вимоги до системи аудіостеганографії полягають в тому, що секретне повідомлення повинно бути непомітним, пропускна здатність повинна бути максимальною [8], здатність приховування повинна бути максимальною, за бажанням секретні дані повинні бути зашифровані, і один або обидва повідомлення і файл прикриття можуть бути аудіосигналами. Деякі дослідники вважають, що часова область даних аудіофайлів підходить для приховування секретних повідомлень.

1.6 Опис стеганографії в аудіофайлах

Так, найбільш поширеним методом був LSB який використовував К. Гопалан, який використовував LSB для вбудовування аудіоповідомлення у файл прикриття. Вигляд LSB зображено у рисунку 1.2.

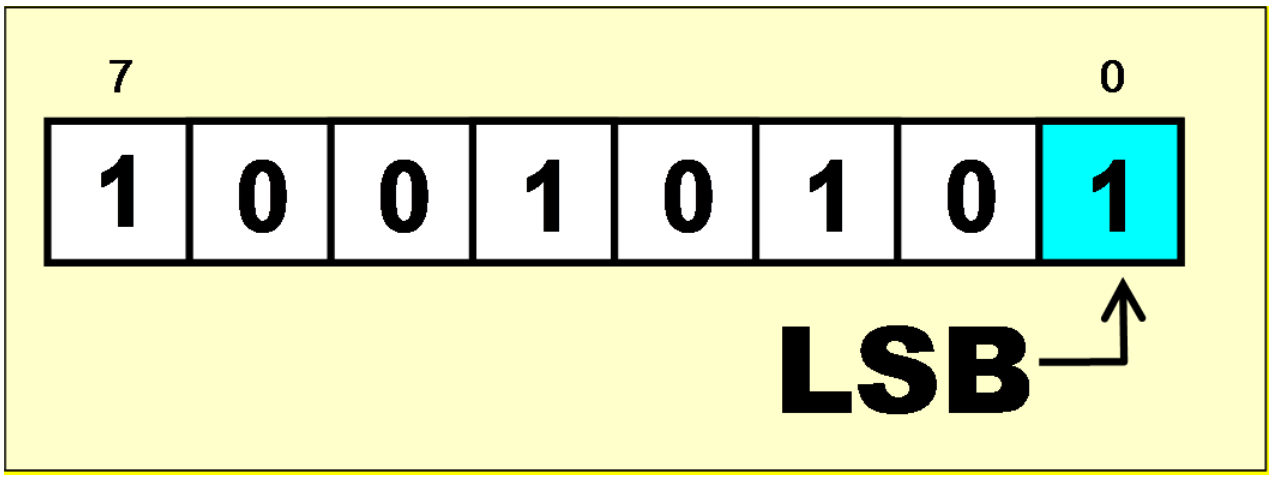

Рисунок 1.2 – Вигляд LSB

Секретне повідомлення стискається і ховається в зразку прикриття шляхом зміни одного біта кожного зразка відповідно до бітів даних за допомогою ключа, який також використовується на приймальній стороні для отримання секретних бітів.

Звуковий сигнал є аналоговим. Тому, щоб використовувати сигнал у цифровому вигляді в комп'ютерах, аналоговий сигнал.

Періодично дискретизується в часі, щоб перетворити його на послідовність відліків за допомогою цифрового перетворювача. Кожен відлік представляє високу пропускну здатність сигналу в дану одиницю часу. У методі LSB відліки аудіофайлу прикриття і піксель секретного зображення перетворюються в двійковий формат, причому LSB прикриття буде замінено на один біт з бітів повідомлення [9], як показано на рисунку 1.3.

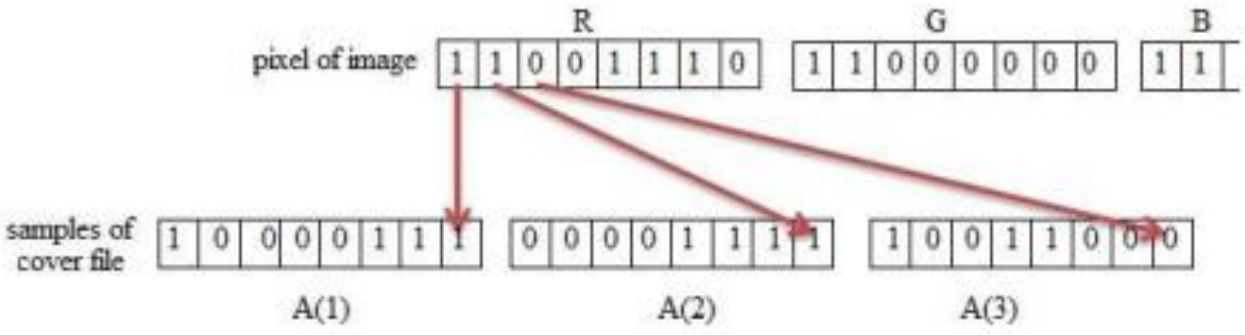

Рисунок 1.3 - Прикриття бітів

Сантос і Бао запропонували підхід аудіо стеганографії на основі DWT для перетворення аудіо в зображення. У їхньому підході оригінальний аудіосигнал перетворюється в зображення і ховається в зображенні, яке потім перетворюється назад в аудіосигнал.

Потім аудіофайл знову перетворюється в аудіосигнал. 2015 року Сінха запропонував метод LSB для приховування зашифрованих текстових повідомлень. По-перше, текстові повідомлення шифруються за допомогою алгоритму шифрування Віженера за допомогою алгоритму шифрування Віженера за допомогою алгоритму шифрування Віженера за допомогою алгоритму шифрування Віженера. По-друге,

текстове повідомлення було вбудовано в обкладинку аудіофайлу за допомогою методу LSB за допомогою методу LSB.

1.7 Безпека аудіофайлів

Безпека стеганографії в аудіофайлах є важливим аспектом при захисті прихованої інформації. Це означає, що приховані дані повинні залишатися недоступними для неповідомлених сторін, і виявлення наявності прихованих даних має бути складним завданням для атакуючих. Деякі фактори, які впливають на безпеку стеганографії в аудіофайлах, включають рівень непомітності, стійкість до різних атак та ефективність методів приховування і виявлення.

Існує кілька способів захисту стего-аудіофайлів від виявлення, які зображені у таблиці 1.3.

Таблиця 1.3

Способи захисту стего-аудіофайлів від виявлення

Криптографічний захист	Застосування криптографічних алгоритмів до прихованих даних може забезпечити їх конфіденційність та цілісність. Перед приховуванням даних, вони можуть бути зашифровані, що зробить їх незрозумілими для осіб без необхідного ключа.
Використання стійких методів приховування	Важливо використовувати стійкі методи приховування, які ускладнюють процес виявлення. Це може включати використання більш складних алгоритмів, розподіл прихованих даних по різних частин аудіофайлу або використання додаткових захисних механізмів.
Маскування прихованих даних	Застосування методів маскування до унеможливлює виявлення прихованих даних і включає використання різних акустичних ефектів, зміну гучності

Для підвищення рівня безпеки аудіофайл був оброблений за допомогою генератора псевдовипадкових чисел Blum Blum Shub.

Генератор псевдовипадкових чисел Blum Blum Shub для транспонування місць відліків. Naseri представили нову стратегію захисту на основі водяних знаків для квантових зображень. Вони мали на меті приховати дані, використовуючи LSB та наймолодший значущий біт (MSB) та старшого значущого біта (MSB). На рисунку 1.4 показано місце розташування старшого та молодшого біта.

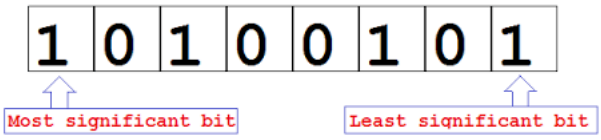

Рисунок 1.4 – Старший та молодший біт

Автори змоделювали і протестували свою систему з розрахунком пікового співвідношення сигнал/шум (PSNR) для забезпечення її безпеки та застосовності порівняно з попередніми дослідженнями, проведеними в той період.

У частотній області Вішванатан розробив модель для приховування текстового повідомлення в аудіофайлах у форматі частотної області (FFT), на рисунку 1.5 зображена схема роботи FFT [10], у якій частоти знаходяться в двох графіках частоти та часу.

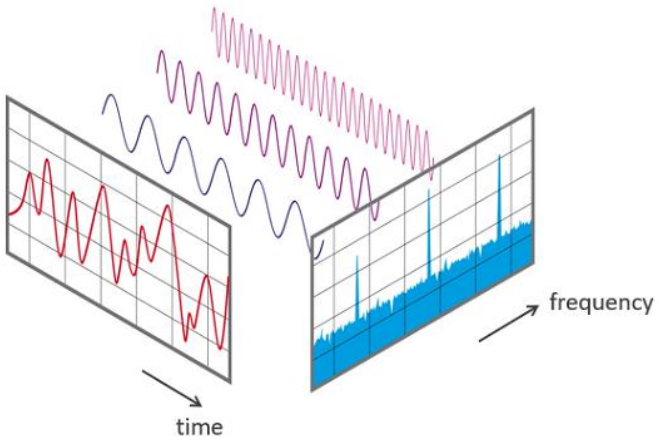

Рисунок 1.5 – Схема роботи FFT

Компонент стеганоаналізу створював стеганоаналізатор, який використовувався для виявлення операцій вбудовування. Ці операції виконувалися за допомогою компонента стеганографії та схеми вилучення ознак для обчислення енергетичної характеристики квантових кадрів аудіосигналу. Ця квантова модель була симулювали і тестували багато разів з різними хвильовими файлами. Абдулраззак вивчали метод, який включав стиснення-шифрування зображення в аудіофайлі. Файл зображення стискався за допомогою методу GMPR що використовує дискретне косинусне перетворення та схему кодування з високочастотною мінімізацією.

Метод стеганографії для приховування стисненого зображення в аудіофайлі стандартного формату "*.wav" використовувався метод стеганографії через його високу ємність як носія інформації. Алгоритм приховування ґрунтувався на методі LSB у змінної та декількох LSB-шарів.

Серед проблем, які існують в літературі, можна виділити проблему безпеки ключа шифрування або точності при точності отримання даних, а також часових витрат і складності обчислень.

1.8 Метод DWT

У цьому методі файл обкладинки і закодоване повідомлення перетворюються у DWT. Перетворення обкладинки виконується у три рівні, як показано на рисунках 5 і 6. Потім вибираються випадкові місця вибираються випадкові місця так само, як і в методі LSB. Цей вибір має на меті розподілити вейвлет-коефіцієнти зображення у файлі обкладинки файлі обкладинки. Потім аудіофайл знову реконструюється за допомогою оберненого перетворення, і отриманий звуковий файл надсилається одержувачу.

На стороні одержувача знову застосовується DWT-перетворення, і коефіцієнти секретного витягуються коефіцієнти секретного зображення [11], маніпулюються зі зворотним перетворенням, і секретне зображення успішно відновлюється. На рисунку

1.6 зображена декомпозиція сигналу X, який проходить через фільтр високих частот та фільтр низьких частот, далі проходить через знижуючу вибірку.

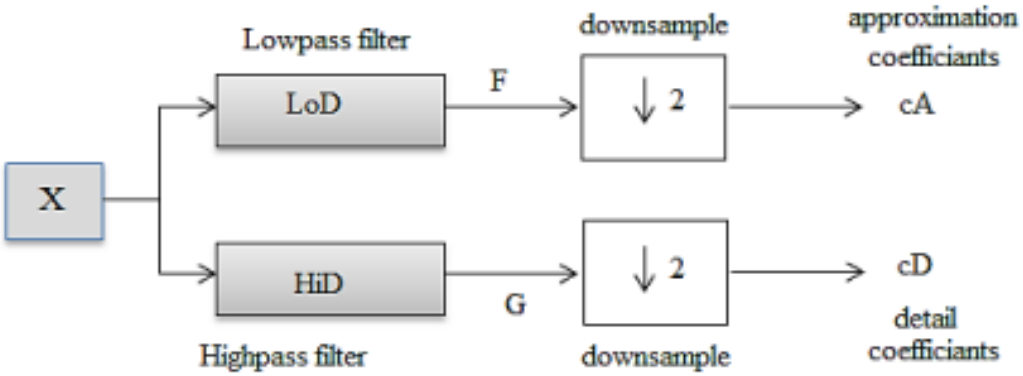

Рисунок 1.6 – Декомпозиція сигналу X

Висновки до першого розділу

Отже, було розглянута історія виникнення аудіоформатів. Також було розглянуто поняття стеганографії та певні методи, які застосовуються для захисту інформації в аудіофайлах. Стеганографія є методом приховування даних всередині носія інформації, такого як аудіофайл, з метою забезпечення конфіденційності та захисту даних.

Аналізовані матеріали та ознайомлення зі схемами методів стеганографії надалі прослужать основою для подальшої розробки засобу захисту інформації в аудіофайлах. Цей засіб буде містити в собі передові технології стеганографії, які дозволять ефективно захищати інформацію в аудіофайлах від несанкціонованого доступу та використання.

РОЗДІЛ 2
ЦІЛІСНІСТЬ ТА КОНФІДЕНЦІЙНІСТЬ ІНФОРМАЦІЇ В АУДІОФАЙЛАХ

2.1 Важливість захисту інформації в аудіофайлах

Важливість захисту цифрового контенту в еру цифрової інформації дуже важлива. Враховуючи легкість, з якою цифровий контент можна копіювати та поширювати, зараз як ніколи важливо вжити заходів для захисту своєї інтелектуальної власності.

Існують способи захисту цифрової інформації:

- Захистити свою роботу авторським правом: авторське право дає виключне право відтворювати, розповсюджувати, виконувати, демонструвати та створювати похідні роботи на основі роботи автора.
- Використання водяних знаків: водяні знаки – це невидимі або майже невидимі знаки, які вбудовуються в цифровий контент. Вони можуть бути використані для ідентифікації власника контенту та запобігання несанкціонованому копіюванню.
- Використання шифрування: шифрування можна використовувати для захисту цифрового контенту від несанкціонованого доступу.
- Використання паролів і налаштувань безпеки: паролі та налаштування безпеки можна використовувати для захисту доступу до цифрового вмісту.
- Вживаючи заходи для захисту свого цифрового вмісту, є можливість допомогти запобігти крадіжці або неправомірному використанню інтелектуальної власності.

Окрім юридичних та фінансових наслідків несанкціонованого копіювання, існують також етичні міркування. Коли щось створюють, вкладається в це час, зусилля і творчий потенціал. Коли хтось інший копіює вашу роботу без дозволу, він, по суті, краде вашу важку працю. Це може розчаровувати і деморалізувати, а також відбити бажання створювати нові роботи в майбутньому.

Захищаючи свій цифровий контент, ви не лише захищаєте свої права, але й надсилаєте повідомлення про те, що ви цінуєте свою роботу і не хочете, щоб нею скористалися.

Ось кілька додаткових причин, чому важливо захищати цифровий контент: щоб запобігти несанкціонованому доступу до конфіденційної інформації.

- Захист приватного життя людей.
- Запобігання поширенню дезінформації.
- Захист прав інтелектуальної власності.

Вживаючи заходів для захисту цифрового контенту, ми можемо допомогти створити більш безпечне і приватне середовище в Інтернеті.

Поняття "водяні знаки" відіграє вирішальну роль у загальному процесі захисту цифрового контенту. Водяні знаки – це видимі або невидимі позначки, які вбудовуються в цифровий контент і слугують формою ідентифікації та стримування від несанкціонованого копіювання або розповсюдження. Вони забезпечують унікальний підпис або ідентифікатор, який вказує на право власності і може допомогти відстежити походження контенту.

Водяні знаки діють як потужний інструмент для творців контенту, художників, фотографів і підприємств для захисту своєї інтелектуальної власності. Включаючи водяні знаки до свого цифрового контенту, вони можуть перешкоджати іншим особам неправомірно використовувати або порушувати їхню роботу. Наявність водяного знаку чітко вказує на те, що контент захищений і не повинен використовуватися без дозволу.

Крім того, водяні знаки можуть слугувати засобом брендингу. Багато компаній використовують водяні знаки для посилення ідентичності свого бренду, гарантуючи, що їхній контент буде впізнаваним і асоціюватиметься з їхньою організацією. Цей аспект брендингу допомагає встановити довіру та авторитет, оскільки споживачі можуть легко ідентифікувати автентичний контент з першоджерела.

Окрім стримування та брендингу, водяні знаки також відіграють важливу роль у підвищенні обізнаності та інформуванні користувачів про важливість дотримання прав інтелектуальної власності. Послідовно застосовуючи водяні знаки до свого

цифрового контенту, творці можуть ініціювати діалог про захист авторських прав, заохочуючи відповідальний обмін і практику атрибуції серед споживачів контенту.

Однак при використанні водяних знаків важливо дотримуватися балансу. Хоча вони забезпечують захист, надмірно нав'язливі водяні знаки можуть заважати загальному користувацькому досвіду і потенційно знижувати естетичну цінність контенту. Творці контенту повинні знайти відповідний баланс між видимістю водяного знаку і доступністю свого контенту, гарантуючи, що водяний знак не затьмарює основну мету або повідомлення цифрового матеріалу.

Отже, водяні знаки є життєво важливим компонентом у ширшому контексті захисту цифрового контенту. Їх використання означає право власності, перешкоджає несанкціонованому копіюванню, допомагає встановити ідентичність бренду і сприяє відповідальному поширенню. Використовуючи методи нанесення водяних знаків разом з іншими захисними заходами, творці контенту і підприємства можуть захистити свої права інтелектуальної власності в постійно мінливому ландшафті обміну інформацією.

2.2 Цілісність файлу з вотермаркою

Вбудовування цифрових водяних знаків.

Метод вбудовування водяних знаків базується на методі прямого послідовного розширення спектра (DSSS). Додаткова інформація, що вбудовується, модулюється псевдошумовою (PN) послідовністю. Сигнал, спектр якого розширюється в частотній області потім генерується в частотній області і додається до вихідного аудіосигналу.

Мета полягає у тому, щоб мати функцію яка працює таким чином, щоб кількість водяного знаку дорівнювала кількості шуму, доданого до аудіосигналу.

Вагова функція повинна бути спроектована таким чином, щоб енергія водяного знаку була максимальною в межах необхідних максимально допустимих спотворень. Сила сигналу вбудованого водяного знаку залежить від особливостей сприйняття мовного сигналу людиною.

Існує схема вбудовування, зображена на рисунку 2.1, яка використовує маскуючий ефект слухової системи людини. Щоб адаптувати дослідження методів нанесення водяних знаків та вбудовування потрібно зробити:

- Обчислити поріг маскування мовного фрейму, що аналізується в даний момент.
- Створити PN-послідовність довжиною 1024.
- Виконати алгоритм FFT для інформації про авторські права, модульованої PN-послідовністю.
- Сформуйте сигнал водяного знаку так, щоб його було неможливо виявити за частотою, використовуючи поріг маскування.
- Зробити його невизначуваним у частотній області.
- Обчислити обернене БПФ згенерованого сигналу водяного знаку.

В кінці потрібно у часовій області додати сигнал з водяним знаком до вихідного мовного сигналу, щоб створити остаточний мовний сигнал з водяним знаком.

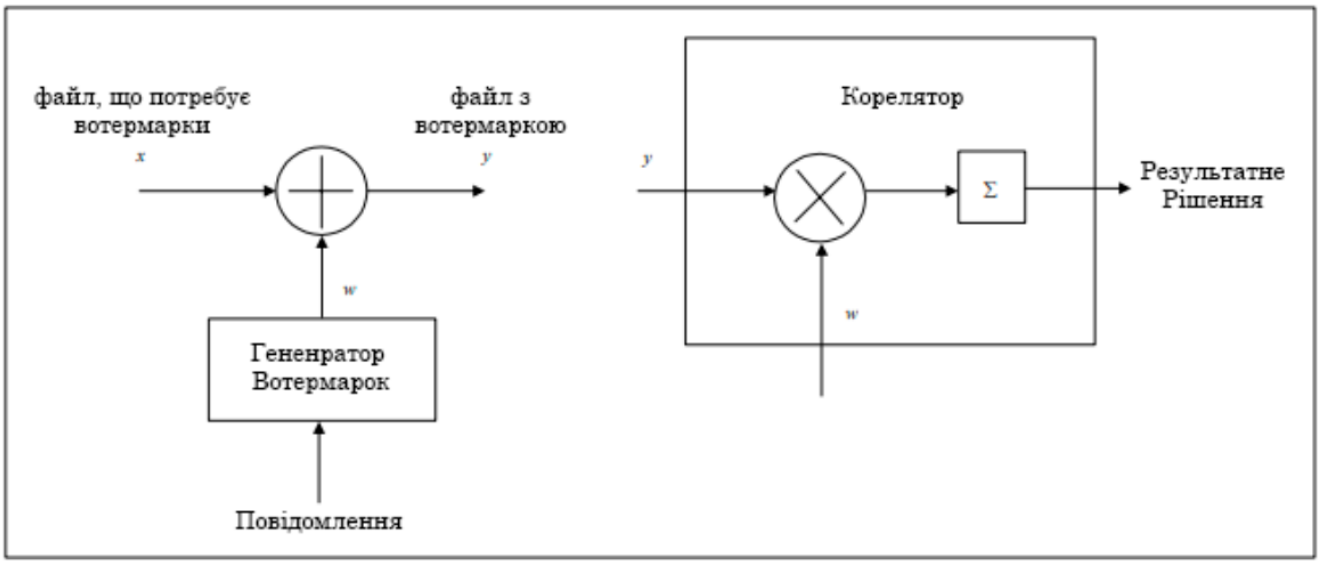

Рисунок 2.1 – Вбудування та виявлення вотермарки

Виявлення водяних знаків.

При розробці системи виявлення водяних знаків враховували бажану продуктивність та надійність системи. Водяний знак повинен бути здатним бути виявленим за допомогою звичайних операцій обробки сигналів, таких як цифро-

аналогове та аналого-цифрове перетворення [12], лінійна та нелінійна фільтрація стиснення та масштабування.

Крім того, у більшості застосувань процеси вилучення водяних знаків не потребують доступу до оригінального сигналу. Насправді, відсутність доступу до оригінального сигналу є важливим у реальних умовах, таких як телерадіомовлення.

Дана процедура виявлення водяних знаків не вимагає доступу до оригінального аудіосигналу для виявлення сигналу водяного знаку. Більшість схем кореляційних детекторів припускають, що канал є білим гауссовим; однак, це не так для реальних аудіосигналів, оскільки звукові зразки мають високу кореляцію.

Для небілого гауссівського каналу можна підвищити ефективність виявлення за допомогою попередньої обробки перед кореляції [12]. Цього можна досягти, застосувавши відбілювання або декореляції перед кореляцією.

2.3 Процедури вбудовування аудіо водяного знаку

По-перше у процесі нанесення аудіо водяних знаків потрібна підготовка аудіосигналу. Тобто потрібно перетворити сигнал в цифровий формат і поділити його на короткий сегмент аудіосигналу 20-40 мс (далі - Кадр).

Після того, як аудіосигнал розділено на кадри, наступним кроком є вилучення ознак з кожного кадру. Ознака - це характеристика, яку можна використати для опису аудіосигналу. До загальних ознак відносяться енергія сигналу, частотний склад сигналу і часові характеристики сигналу.

Вбудовування водяних знаків.

Після того, як ознаки вилучено, наступним кроком є вбудовування водяного знаку в аудіосигнал. Це робиться шляхом зміни характеристик сигналу таким чином, щоб він був невидимим для людського вуха. Існують різні способи вбудовування водяного знаку в аудіосигнал, але є певні загальні методи:

- Вбудовування найменш значущого біта (LSB): цей метод змінює найменш значущий біт ознаки.

- Вбудовування з розширенням спектра: цей метод поширює сигнал водяного знаку на весь аудіосигнал.
- Вбудовування квантування: цей метод квантує сигнал водяного знаку і вбудовує квантований сигнал у мовний сигнал.

Вибір методу вбудовування залежить від застосування. Наприклад, якщо водяний знак використовується для захисту авторських прав, потрібен надійний метод вбудовування. Якщо ж водяний знак використовується для аутентифікації, потрібен безпечний метод вбудовування.

Отже, нанесення водяних знаків на аудіо є складним процесом, але необхідним для багатьох застосувань. Вбудовування водяного знаку в аудіосигнал захищає сигнал від несанкціонованого копіювання і підтверджує, що сигнал походить із законного джерела.

Сигнали з водяними знаками зазвичай набагато коротші за аудіосигнали. Це пов'язано з тим, що сигнал водяного знаку повинен бути вбудований в аудіосигнал таким чином, щоб його не було видно для людського вуха.

Сигнали водяних знаків зазвичай шифруються перед тим, як вбудовуються в аудіосигнал. Це робиться для того, щоб запобігти видаленню водяного знаку несанкціонованими користувачами.

Процедура вбудовування аудіо водяних знаків зазвичай виконується в режимі реального часу. Це необхідно для таких застосувань, як моніторинг мовлення, де водяні знаки потрібно вбудовувати в аудіосигнал, що транслюється.

Нанесення водяних знаків на аудіо, постійно розвивається. Постійно розробляються нові методи вбудовування водяних знаків, а ефективність існуючих методів постійно підвищується. В результаті, аудіо водяні знаки стають все більш важливим інструментом для захисту цифрового аудіо контенту.

На рисунку 8 зображена схема процесу накладання вотермарки на аудіо. В ній описується процедура вбудування в аудіо водяний знак (вотермарк). Процес починається з введення аудіо далі файл обробляється за допомогою алгоритму FFT, проходить психоакустичну модель та накладається шум і далі йде він через алгоритм IFFT [13] і на виході отримуються файл з вотермаркою.

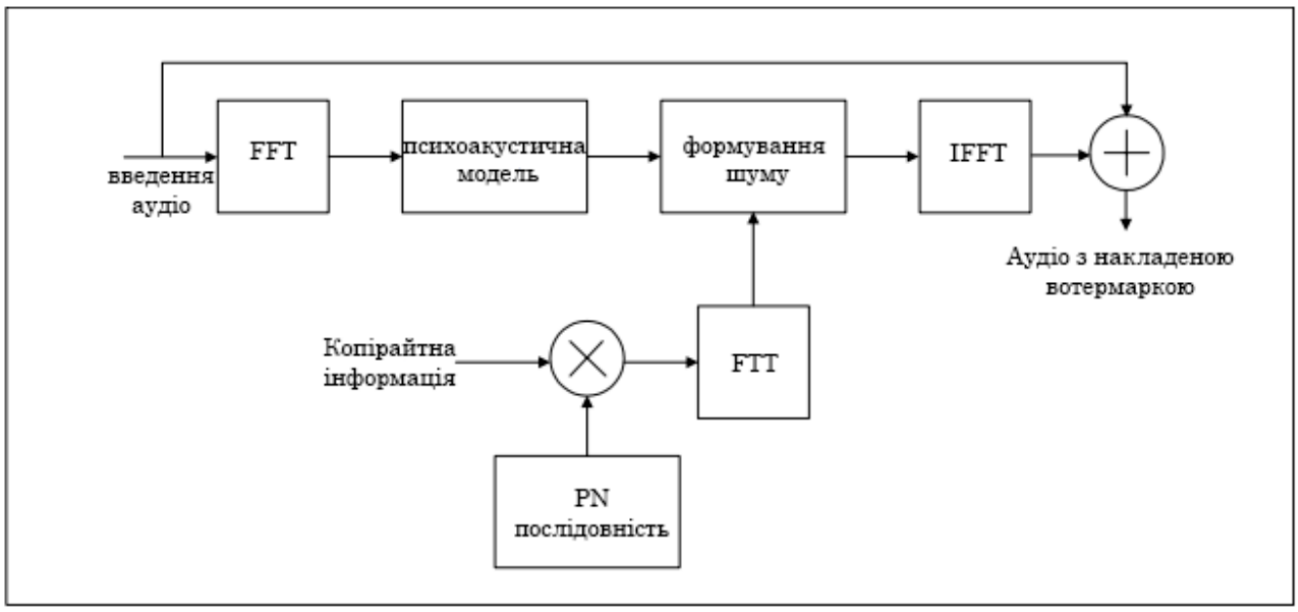

Рисунок 2.2 – Схема процедури вбудовування аудіо водяного знаку

Після успішного використання процедури вбудування в аудіо водяного знаку, проведено додаткову аналізу, що дозволила отримати графічне зображення аудіосигналу з вотермаркою та без неї.

Зазначені результати були відображені на рисунках 2.3-2.5, де можна порівняти візуальні відмінності між обома версіями аудіосигналу.

На рисунку 2.3 представлено зображення аудіосигналу розмови людей без та з водяним знаком, в якому була вбудована вотермарка, що передає важливу інформацію.

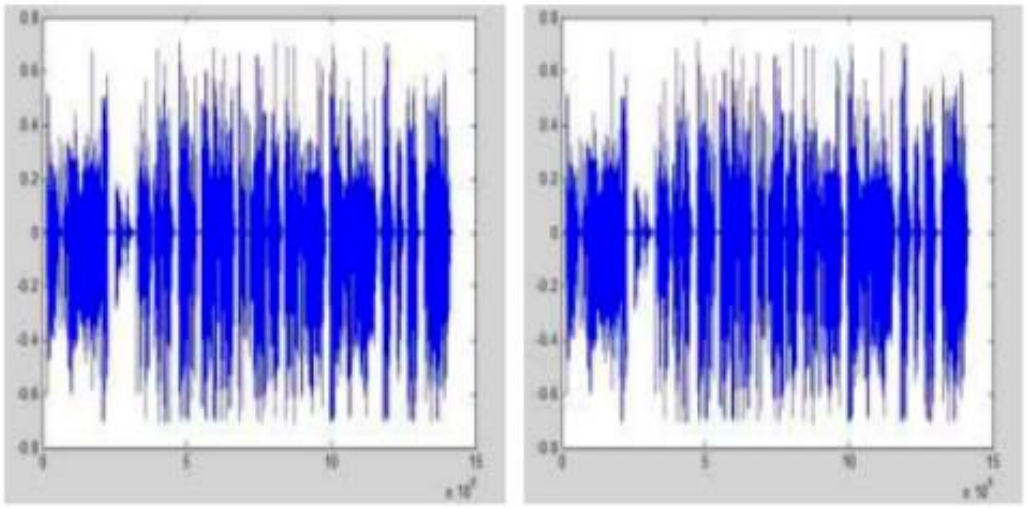

Рисунок 2.3 – Запис розмови людей без та з водяним знаком

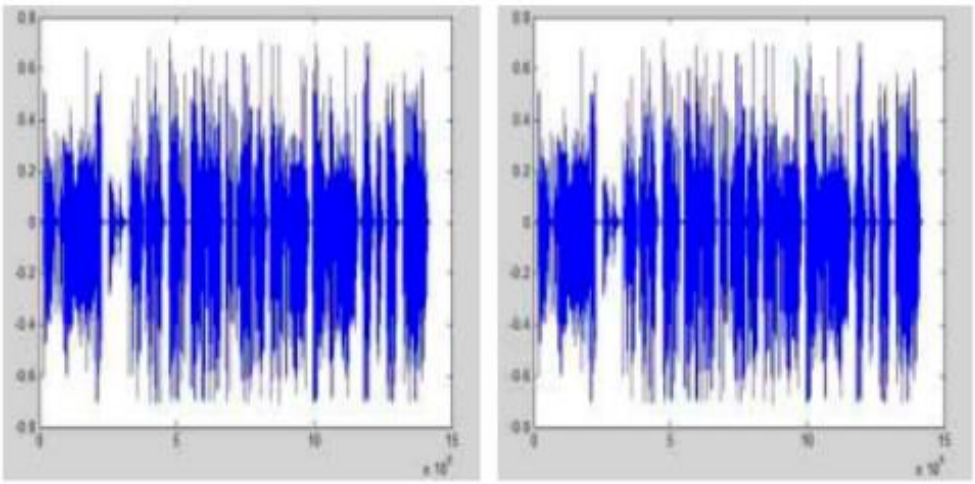

Рисунок 2.4 – Запис голосу з та без водяного знаку

З іншого боку, рисунок 2.4 відображає оригінальний аудіосигнал запису голосу з та без водяного знаку. Без водяного знаку, де жодних змін не було внесено.

Також, рисунок надає можливість прямого порівняння між зображеннями аудіосигналів з і без вотермарки, не демонструючи очевидної різниці.

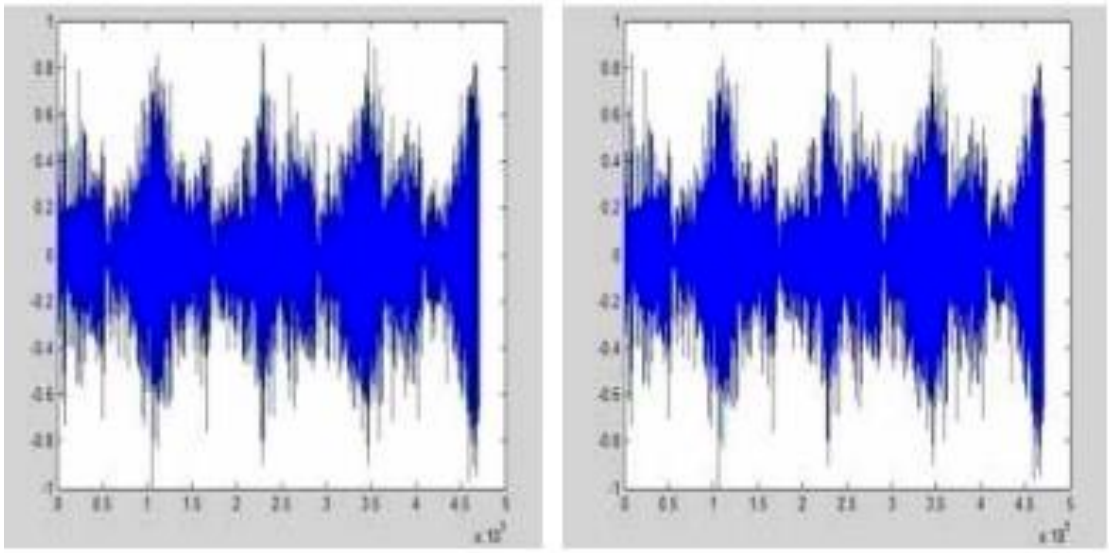

Рисунок 2.5 – Музичний трек з та без водяного знаку

Отже, можна вважати, що процес вбудування вотермарки був успішним. Як видно з рисунків 2.3-2.5 діапазонний вигляд інформації в аудіофайлах не змінився.

2.4 Класифікація та проблематика вотермарки

Проблематика питання вотермарки дуже важливе питання, якщо говорити про цілісність аудіофайлу. В залежності від потреб, типу аудіофайла, наприклад пісня чи запис мікрофону з концерту, є певні категорії вотермарок:

- Захист авторських прав: водяні знаки для аудіо можна використовувати для захисту авторських прав на аудіофайли. Якщо вбудувати унікальний ідентифікатор в аудіофайл, можна відстежити та ідентифікувати джерело файлу в разі його незаконного копіювання.
- Моніторинг мовлення: водяні знаки для аудіо можна використовувати для моніторингу трансляції аудіоконтенту. Вбудовуючи водяний знак з інформацією про власника авторських прав, можна ідентифікувати та відстежувати несанкціоновану трансляцію захищеного авторським правом матеріалу.
- Аутентифікація: аудіо водяні знаки можна використовувати для автентифікації аудіофайлів. Вбудовуючи водяний знак з інформацією про джерело файлу, можна перевірити, що файл є справжнім і не був підроблений.
- Ідентифікація трансляції: водяні знаки для аудіо можна використовувати для ідентифікації джерела аудіотрансляції. Вбудовуючи водяний знак з інформацією про мовника, можна ідентифікувати джерело аудіотрансляції, навіть якщо трансляція була повторно трансльована або змінена.

Цільовою аудиторією для нанесення водяних знаків на аудіо є всі, хто хоче захистити свої авторські права, відстежувати трансляцію свого аудіоконтенту або перевіряти автентичність своїх аудіофайлів.

Сюди входять:

- Музиканти: музиканти можуть використовувати водяні знаки для захисту своїх авторських прав і відстежувати розповсюдження своєї музики.
- Творці контенту: творці контенту, такі як подкастери та радіомовники, можуть використовувати водяні знаки для захисту своїх авторських прав і відстежувати поширення свого контенту.

- Розповсюджувачі аудіо: дистриб'ютори аудіо, такі як потокові сервіси та звукозаписні лейбли, можуть використовувати водяні знаки для перевірки автентичності своїх аудіофайлів і запобігання піратству.

Порівняльна таблиця відносно алгоритму на основі надійності, непомітності та безпеки (далі – ННБ) зображена в таблиці 2.1.

Дана таблиця 2.1 містить в собі дані відносно алгоритмів стеганографічного захисту інформації [14].

Розглядається три різних алгоритму.

Таблиця 2.1

ННБ відносно алгоритму

Алгоритм	Надійність	Непомітність	Безпека
Вбудовування найменшого значущого біта (LSB)	Низький	Високий	Низький
Вбудовування розширеного спектру	Високий	Середній	Середній
Вбудовування квантування	Середній	Низький	Високий

2.5 Метадані в аудіофайлах

Підготовка даних. Для початку потрібно залучити аудіофайли, а саме MIDI-файли з інтернету. Було отримано 23 файли, з яких 11 з них мали унікальні контрольні суми MD5.

Розглянемо ефективного способу ефективного способу зіставлення цього корпусу з базою даних Million Song Dataset (MSD), або, точніше, з коротким прев'ю аудіозаписів, наданим 7-digital аудіозаписами.

Для оцінювання нам потрібна колекція достовірних MIDI-аудіо пар, які правильно підібрані. Тоді наш підхід можна буде оцінити на основі того, наскільки

точно він здатен відновити ці пари, використовуючи лише вміст аудіо та MIDI-файлів.

Щоб застосувати крос-модальну схему хешування, потрібна колекція вирівняних MIDI- та аудіофайлів, щоб отримати відповідні пари векторів ознак з кожної області, які будуть використані для навчання моделі хешування MIDI та аудіо ознак до спільного хешування. Маючи відповідні аудіо та MIDI-файли аудіо та MIDI-файлів, існуючі методи вирівнювання можуть бути для створення цих навчальних даних; однак, ми повинні виключити неправильні збіги та невдалі вирівнювання. Навіть у масштабі цього зменшеного набору навчальних даних, ручна перевірка вирівнювання ручна перевірка вирівнювання є непрактичною, тому ми розробили покращену.

2.6 Узгодження метаданих

Щоб отримати колекцію пар MIDI-аудіо, спочатку виокремити потрібно підмножину MIDI-файлів, для яких назва каталогу відповідала виконавцю пісні, а ім'я файлу назву пісні. Отримані метадані потребують додаткової канонізації; наприклад, "The Beatles", "Beatles, The", "Бітлз" і "Бітлз Джон Пол Рінго Джордж" з'являлися як виконавці. Щоб нормалізувати ці проблеми, застосовується деяку ручну обробку тексту і узгодження імен виконавців і назви пісень. В результаті буде отримано колекцію з MIDI-файлів файлів для унікальних пісень, які будуть називатись називати "чиста підмножина MIDI".

Потрібно використовувати чисту підмножину MIDI у два способи:

- По-перше, для отримання правдивих пар MSD/MIDI-відповідностей.
- А по-друге, для створення навчальних даних для нашої схеми хешування.

Навчальні дані не обов'язково повинні обмежуватися MSD, і використання інших джерел для збільшення розміру навчальної вибірки може покращити продуктивність хешування.

На рисунку 9 зображено величини змінної довжини. Вони э зручним способом запису цілих чисел – від найменших до найбільших, без необхідності відводити під число фіксовану кількість байт.

Біти вихідного числа пакуються в один або більше байт: у кожен б по сім біт (праворуч, біти з 0 по 6-й).

Старший біт у байті є службовим; усі байти в серії, крім останнього, повинні містити в ньому одиницю, останній – нуль.

Декілька прикладів пакування показано на рисунку 2.6.

Число (HEX)	Змінна довжина
00000000	00
00000040	40
0000007F	7F
00000080	81 00
00002000	C0 00
00003FFF	FF 7F
00004000	81 80 00
00100000	C0 80 00
001FFFFF	FF FF 7F
00200000	81 80 80 00
08000000	C0 80 80 00
0FFFFFFF	FF FF FF 7F

Рисунок 2.6 – Приклади пакування даних в MIDI-файлах

2.7 Вирівнювання аудіо до синтезованого MIDI

Нечіткого узгодження метаданих недостатньо для того, щоб гарантувати, що M аудіофайли з однаковим вмістом, наприклад, метадані можуть бути неправильними, нечітка відповідність тексту нечітка відповідність тексту. MIDI може бути поганою транскрипцією (наприклад, відсутні інструменти або секції), та/або MIDI та аудіо дані можуть відповідати різним версіям пісні. Оскільки ми будемо використовувати DTW для вирівнювання аудіовмісту до аудіоресинтезу MIDI-контенту, ми можемо потенційно використати загальну вартість збігу – кількість мінімізовану за

допомогою DTW – як індикатор правильних збігів, оскільки непов'язані пари MIDI та аудіо, ймовірно, призведуть до високу оптимальну вартість збігів.

Калібрування цієї необробленої вартості збігу "довірчої оцінки", як правило, не можна порівняти між різними вирівнюваннями. Програма, вимагає довірчої оцінки DTW, яка може достовірно визначити, коли пара аудіо/MIDI-файлів є дійсною для використання в якості навчальних даних для моделі хешування.

По-перше, синтезуються MIDI-дані за допомогою ПЗ Fluidsynth (рисунок 2.7). Потім оцінюємо розташування часток MIDI, використовуючи інформацію про зміну темпу в MIDI-файлі інформацію про зміну темпу в MIDI-файлі та метод.

Рисунок 2.7 – Інтерфейс ПЗ Fluidsynth

Щоб обійти поширену проблему, коли біт відстежується на половину такту не в фазі, ми подвоюємо значення BPM, поки воно не досягне принаймні 240. Обчислюємо 4 місця розташування бітів для аудіосигналу з обмеженням, що BPM має залишатися близьким до глобальний MIDI-темп. Потім ми обчислюємо лог-амплітудні біт-синхронні перетворення з постійною добротністю (CQT) аудіо та синтезованих MIDI-даних з інтервалом у півтону і діапазоном частот від C3 (65,4 Гц) до C7 (1046,5 Гц).

2.8 Стеганографія в аудіофайлах

Великі обсяги медіа-інформації та слабке регулювання авторських прав дають свободу стеганографії. Достатня кількість інформації в аудіозаписі дає можливість приховати конфіденційну інформацію всередині аудіофайлу. Приховати інформацію легко, але знайти стеганографічне повідомлення може бути проблематично. Тому метою цієї статті є розробка модуля пошуку секретної інформації, який використовує методи стеганографії для виявлення прихованих даних в аудіофайлах.

Наразі стеганографія в аудіофайлах є складною задачею і на ринку не існує програмного забезпечення, яке може виконати комплексний аналіз аудіофайлів та знайти сліди прихованої інформації.

Тому створення такого програмного забезпечення є дуже важливим для вирішення актуальних проблем, пов'язаних з витоком інформації.

Розглянемо тепер порівняння форматів аудіофайлів згідно таблиці 2.2.

Таблиця 2.2

Порівняльна таблиця аудіоформатів відносно зберігання даних

	MP3	WAV	OGG
Обкладинка	Є	Немає	Нема
Метадані	ID3v1 ID3v2 ID3v3	RIFF XMP	RDF XML-похідні XML-метадані MusicBrainz Ogg
Вміст заголовку	Frame sync MPEG version Layer ID Bitrate ID Frequency ID Personal Bit	ChunkId Format Subchunk1Size AudioFormat SampleRate BitsPerSample Data	Capture pattern Version Header type Granule position Bitstream number Page sequence number Checksum

З таблиці 2.2 ми можемо бачити наступні речі – формат MP3 містить багато інформації, до якої може бути організовано несанкціонований доступ, але це не контролюється

Таблиця містить наступні дані про MP3:

- Обкладинка: Є
- Метадані: ID3v1, ID3v2, ID3v3
- Вміст заголовку: Frame sync, MPEG version, Layer ID, CRC, Bitrate ID, Sampling Frequency ID, Personal Bit, Channel, Expansion mode, Original author rights, Mastering Glue.

Про WAV:

- Обкладинка: немає
- Метадані: RIFF, XMP
- Вміст заголовку: RDF, XML-похідні (включаючи RDF, CMML та XMP), XML-метадані MusicBrainz Ogg Skeleton

Про OGG:

- Обкладинка: немає
- Метадані: RDF, XML-похідні (включаючи RDF, CMML та XMP), XML-метадані MusicBrainz Ogg Skeleton
- Вміст заголовку: Capture pattern, Version, Header type, Granule position, Bitstream serial number, Page sequence number Checksum, Page segments, Segment table.

Наведені дані в таблиці 2.2 про аудіоформати MP3, WAV та OGG порівнюють за наявністю обкладинки, типами метаданих та вмістом заголовку. MP3 має обкладинку, а також використовує формати метаданих ID3v1, ID3v2 та ID3v3. У вмісті заголовку MP3 зустрічаються різні параметри, такі як Frame sync, MPEG version, Layer ID, CRC, Bitrate ID, Sampling Frequency ID, Personal Bit, Channel, Expansion mode та Original author rights.

WAV не має обкладинки, але використовує формати метаданих RIFF та XMP. У вмісті заголовку WAV зустрічаються RDF, XML-похідні (включаючи RDF, CMML та XMP) та XML-метадані MusicBrainz Ogg Skeleton.

OGG також не має обкладинки, але використовує формати метаданих RDF, XML-похідні (включаючи RDF, CMML та XMP) та XML-метадані MusicBrainz Ogg Skeleton. У вмісті заголовку OGG зустрічаються параметри [15], такі як Capture pattern, Version, Header type, Granule position, Bitstream serial number, Page sequence number Checksum, Page segments та Segment table.

2.9 Аналіз можливостей застосування стеганографії до аудіофайлів

Аналізуючи можливості застосування стеганографії до аудіофайлів різних форматів за допомогою вільно поширюваного програмного забезпечення, можна виділити наступні місця

Секретне повідомлення перше.

Теги. Теги можуть використовуватися для зберігання нестандартних даних.

Оскільки нестандартні дані можуть зберігатися, в них можуть бути приховані повідомлення.

Наприклад, деякі програми зберігають тут налаштування гучності і нормалізації для кожного окремого файлу. Медіа-плеєри, як правило, не відображають невідомі їм параметри.

Обкладинка. У файлі обкладинки аудіозаписи, яка знаходиться в тезі, після IEND можна додати текст, причому файл так і буде визначатися як зображення. Більш того, абсолютно ніяких заявок не буде зазначено. Зрівнявши HEX представлення вмісту аудіофайлів, можна побачити, що дописаний 16-ий код після IEND-чанки — стеганоповідомлення.

Спектрограма. Вона є доповнюваним об'єктом аудіофайла. Для того, щоб створити стеганоповідомлення в спектрограмі, виконується наступний алгоритм: створюється картинка, на якому написаний потрібний текст і далі за допомогою програми формується аудіозапис.

Розглянувши порівняння аудіофайлів, а також визначивши місця прикриття даних була дослідники створили схему як на рисунку 2.8.

Рисунок 2.8 – Схема засобу пошуку стеганографії

Висновки до другого розділу

Отже, було розглянуто важливість захисту інформації в аудіофайлах та проведено порівняльний аналіз методів стеганографії для захисту аудіоданих. Були розглянуті методи стеганографії для приховання інформації всередині аудіофайлів, забезпечуючи конфіденційність та захист даних.

Після детального аналізу можливостей стеганографії для аудіофайлів була розроблена схема пошуку стеганографічної інформації в аудіофайлах. Також проведено порівняльний аналіз різних аудіоформатів, які використовуються для збереження звукової інформації. Були розглянуті особливості різних форматів і їх використання в контексті захисту даних.

Додатково було досліджено питання метаданих в аудіофайлах різних форматів. Метадані містять додаткову інформацію про аудіофайл. Були розглянуті можливі вразливості метаданих, які можуть використовуватись для ідентифікації, відстеження або порушення конфіденційності даних.

В результаті аналізу і порівняння різних аспектів захисту аудіофайлів, було набуто глибокого розуміння важливості захисту інформації в цьому контексті для створення засобу захисту інформації в аудіофайлах.

РОЗДІЛ 3
ЗАСІБ ЗАХИСТУ ІНФОРМАЦІЇ В АУДІОФАЙЛАХ

3.1 Загальний огляд ПЗ

Що таке сіквенсор?

Сіквенсор може бути використаний для створення засобу захисту інформації в аудіофайлах шляхом вбудовування водяного знаку. Використовуючи сіквенсор у тебе є можливість маніпулювати, змінювати, редагувати звук, звукову доріжку чи інструмент.

За допомогою сіквенсора, можна вбудовувати цей водяний знак в аудіосигнал. Починаючи з оригінального аудіофайла, сіквенсор дає можливість внести невидимі зміни в аудіосигнал, що представляють собою унікальний підпис або інформацію про автора або власника.

Водяний знак може бути реалізований у формі амплітудних змін, фазових змін, зміни частоти або навіть застосування спеціальних аудіофільтрів. Ці зміни зазвичай неприйнятні для людського слуху, але можуть бути виявлені аналітичними програмами або спеціальним обладнанням.

За допомогою сіквенсора ви можете створювати аудіофайли з нуля або редагувати існуючі записи. Ви можете додавати різні інструменти до аудіо, звукові ефекти та міксувати їх разом. Сіквенсор надає широкий набір інструментів для обробки та маніпулювання звуком, таких як зміна темпу, зміна тону, затримка, реверберація, еквалайзер та багато інших.

Сіквенсор є незамінним інструментом, він є програмним засобом, який дозволяє створювати, редагувати і організовувати музичні сигнали в часовій лінії, відтворюючи їх послідовно і узгоджено. Популярними прикладами таких сіквенсорів є FL Studio, Ableton Live, Studio One, Logic Pro та інші. Порівняльна характеристика пов’язана з доступними функціями трьох основних сіквенсорів на ринку в таблиці 3.1.

Таблиця 3.1

Порівняння сіквенсорів відносно доступних функцій

Функція	FL Studio	Ableton Live	Studio One
Музичний інтерфейс	Так	Так	Так
MIDI підтримка	Так	Так	Так
Автоматизація	Так	Так	Так
Вбудовані інструменти	Так	Так	Так
Ефекти	Так	Так	Так
Звукозапис	Так	Так	Так
Редактор паттернів	Так	Так	Так
Візуальне програмування	Так	Ні	Ні
Підтримка VST плагінів	Так	Так	Так
Режим живого виконання	Ні	Так	Так
Можливість колаборації	Ні	Так	Так

Сіквенсори мають потрібні функції для управління MIDI даними та файлами [16], які дозволяють створювати та редагувати аудіо за допомогою MIDI-контролерів, клавіатури або інших засобів введення. Ви можете програмувати різні ноти, акорди. В нашому випадку буде використовуватись клавіатура як контролер інформації в MIDI-файлі.

Маніпулювання акордами та нотами буде використовуватись в засобі захисту інформації в аудіофайлах. Це основний принцип переваги користування саме сіквенсором для роботи зі звуком, ніж звичним старим апаратним ПЗ.

Офіційний сайт FL Studio надає широкий спектр інформації, пов'язаної з цим програмним забезпеченням для створення музики. Основна інформація, доступна на сайті, включає багато інформації, перелік інформації що доступна на основному сайті розробника:

- Опис продукту: детальний опис FL Studio, її можливостей, функцій та інструментів її кабінетів. Також представлені різні версії програми та їхні можливості.
- Завантаження та встановлення: веб-студія FL Studio надає посилання для завантаження останньої версії програмного забезпечення. Тут також показано, як встановити та оновити програмне забезпечення на вашому комп'ютері.
- Документація та посібники користувача: ви знайдете безліч інструментів, документації та ресурсів для вивчення FL Studio і отримання максимальної віддачі від неї. Сюди входять довідники версій, навчальні посібники та форуми спільноти користувачів.
- Демонстрації: доступ до демонстраційних роликів, відеоуроків, ілюстрацій та текстів пісень, що демонструють функціональність FL Studio. Це дає змогу ознайомитися з програмним забезпеченням та його можливостями ще до того, як його буде придбано, і вирішити, чи підходить воно для Ваших потреб чи ні в залежності від задач.
- Інформація про компанію та контактна інформація Цей сайт містить інформацію про компанію Image-Line Software, яка є розробником FL Studio. Ви можете ознайомитися з інформацією, привілеями та іншими важливими відомостями про компанію.
- Оновлення та підтримка. На офіційному веб-сайті FL Studio Ви можете знайти найновіші версії програми, які вже є доступними на офіційному сайті від розробника. Там же ви знайдете інформацію про те, як встановити та оновити програмне забезпечення на вашому комп'ютері використовуючи різні операційні системи.

3.2 Інтерфейс і функції FL Studio

На рисунку 3.1 зображений інтерфейс FL Studio v20.0.5. Виходячи з рисунка ми можемо побачити багато різних налаштувань, але нас цікавитимуть обведені червоним на рисунку функції.

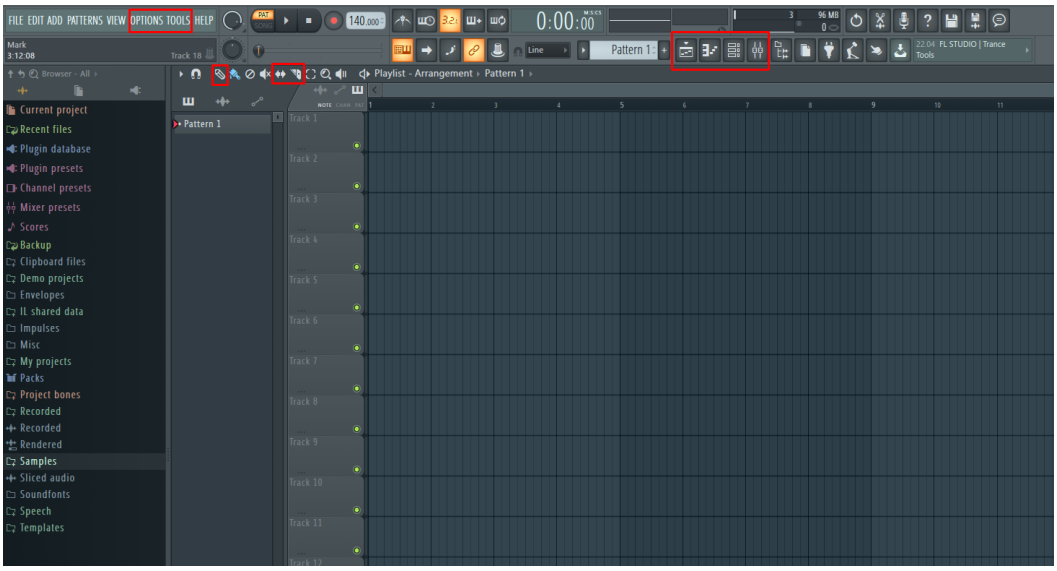

Рисунок 3.1 – Інтерфейс FL Studio 20

В одному зі згаданих вище вікон є функціонал додавання любого звуку, любої довжини. Якщо потрібно завантажити свій аудіофайл – можна скористатись функцією додавання папки з аудіофайлами. Заходимо в налаштування (OPTIONS) і додаємо потрібну папку. В моєму випадку це папка «Samples».

Меню налаштувань зображене на рисунку 3.2.

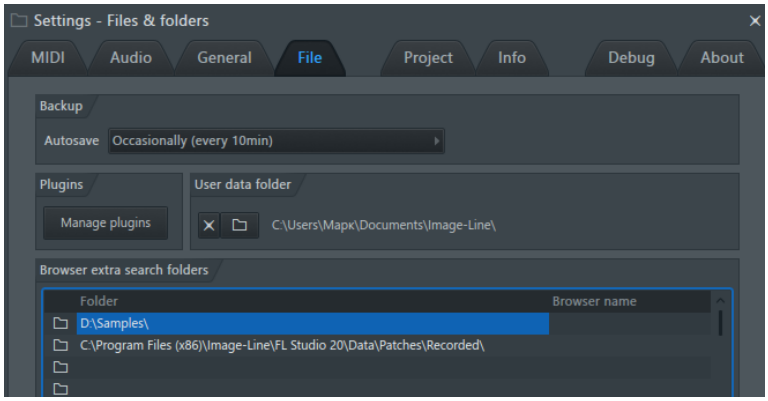

Рисунок 3.2 – Меню налаштувань

Далі, починаючи роботу, потрібно додати аудіофайл. Потрібно натиснути «+» і вибрати потрібний аудіофайл. В нашому випадку це буде аудіофайл з назвою «Сигнал №1». Процес зображений на рисунку 3.3.

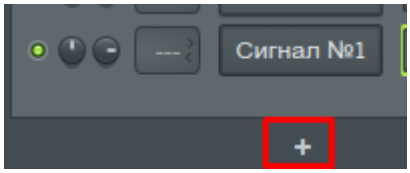

Рисунок 3.3 – Процес додавання аудіофайлу до робочої області

Після додавання аудіофайлу, потрібно оглянути робочу область, а саме інструмент Piano Roll. Дана область зображена на рисунку 3.4. В засобі захисту інформації в аудіофайлах, буде використовуватись потенціал Piano Roll, оскільки він буде використовуватись незвичайним способом.

Дана робоча область використовуватиметься для побудування повідомлення, яке потрібно передати відкритими або закритими каналами зв’язку.

Рисунок 3.4 – Робоча область Piano Roll

На рисунку 3.5 зображений мікшер. Це імітація апаратного мікшера в вигляді частини програмного забезпечення FL Studio. В даній області буде використовуватись стеганографічне додавання шуму і інших звукових сигналів для забезпечення конфіденційності та унеможливлення декодингу повідомлення переданого відкритими чи закритими каналами зв’язку.

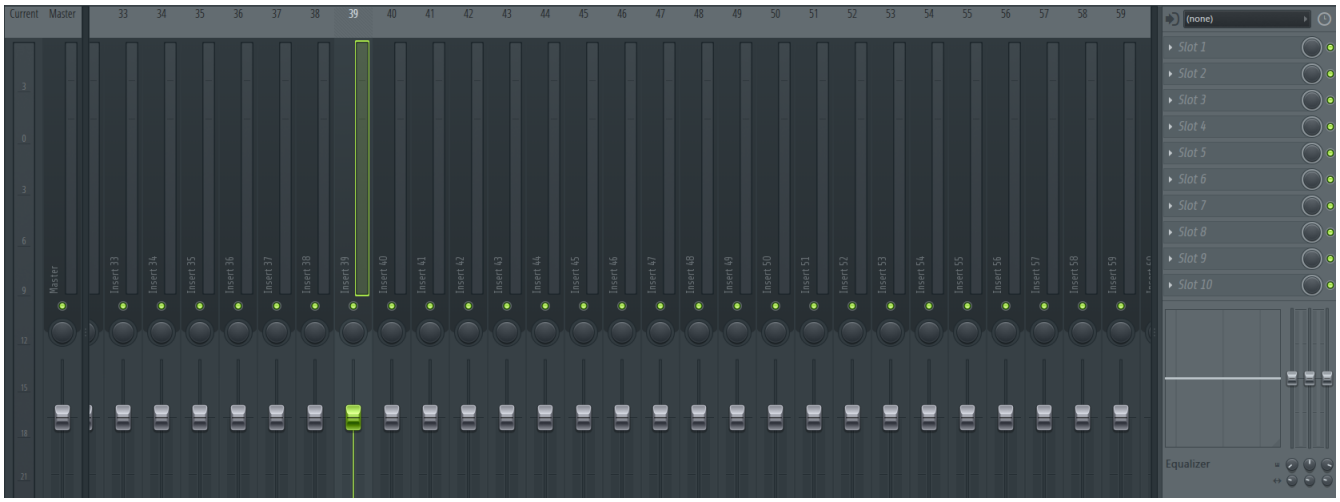

Рисунок 3.5 – Мікшер

На рисунку 3.5, справа, можна бачити вертикальні поля з текстом «Slot 1-10». Саме в ці 10 слотів можна додавати певні програми, які розширюють функціонал самого програмного забезпечення у цілому.

В рисунках 3.6-3.8 зображений функціонал редагування самого аудіофайлу.

Отже, мікшер буде використовуватись не тільки для кореляції гучності аудіосигналу, а й як ще одне робоче поле, яке повинно підвищити рівень конфіденційності в аудіофайлі, прибрати метадані, накласти водяний знак (вотермарку) тощо.

В меню редагування звуку також може бути наявна функція додавання водяного знаку до аудіофайлу. Водяний знак є додатковою інформацією, яка може бути вбудована в аудіофайл для різних цілей, таких як підвищення рівня конфіденційності, відстеження авторства або захист від незаконного використання.

Рисунок 3.6 – Вкладка 1 меню редагування звуку

Ця функція дозволяє вам накласти водяний знак на аудіофайл шляхом вбудовування додаткової інформації в сам файл. Інформація водяного знаку може бути текстовим повідомленням, логотипом, унікальним ідентифікатором або іншими даними, які ви хочете включити.

При додаванні водяного знаку в меню редагування звуку можуть бути доступні додаткові опції, такі як регулювання прозорості знаку, його розміщення в аудіофайлі або налаштування стилю та розміру знаку.

Крім того, меню редагування звуку може містити й інші функції для збереження конфіденційності ваших аудіофайлів, такі як видалення метаданих. Ця функція дозволяє вам видалити всю додаткову інформацію про файл, яка може бути вбудована в нього, таку як назва автора, дата створення або місце запису.

Загалом, наявність функцій додавання водяного знаку і видалення метаданих в меню редагування звуку допомагає вам забезпечити більшу конфіденційність та контроль над вашими аудіофайлами.

Рисунок 3.7 – Вкладка 2 меню редагування звуку

В меню редагування звуку є різноманітні функції та налаштування для зміни і покращення звукових файлів. Існує можливість завантажити файл звуку, обрізати його, змінити гучність, використовувати еквалайзер для налаштування частотного спектру, застосовувати ефекти, а також зменшувати шум.

Це дозволяє вам створювати і редагувати звукові файли з більшою точністю та контролем, дозволяючи досягти бажаного звукового ефекту. Меню редагування звуку забезпечує вам широкий спектр інструментів для творчого виразу та покращення якості звуку.

Переглядаючи рисунок 3.8 можна побачити багато кругових налаштувань. Ці настройки впливають на аудіо. Також в даній вкладці можна змінювати кореневу ноту аудіосигналу.

Оскільки потрібно щоб аудіосигнал відповідав тій ноті, якій він відповідає природньо (для коректного створення повідомлення та декодингу повідомлення), це все можна зробити в даному меню.

Рисунок 3.8 – Вкладка 3 меню редагування звуку

Отже, ми оглянули весь потрібний нам функціонал в програмі FL Studio. Весь оглянутий функціонал використовуватиметься в засобі захисту інформації в аудіофайлах. Але потрібно ще провести порівняльний аналіз кожної з вкладок меню редагування звуку. І провівши аналіз, отримано результат що вкладка, яка зображена на рисунку 3.6 більш функціональна і ми будемо її використовувати під час роботи.

3.3 Огляд програмної функції Edison

Програма Edison має широкий спектр функцій.

FL Studio 20 є музичним програмним забезпеченням, яке включає в себе різноманітні інструменти і плагіни для створення музики. Один з цих плагінів - Edison, є потужним аудіоредактором, який має розширений функціонал для редагування, збереження і обробки аудіо.

Основні функції Edison у FL Studio 20:

- Запис аудіо: Edison дозволяє здійснювати запис звуку з різних джерел, таких як мікрофон, лінійний вхід або інші аудіоінтерфейси. Ви можете записувати в реальному часі або імпортувати вже наявні аудіофайли.

- Редагування аудіо: Edison надає широкі можливості для редагування аудіо. Ви можете вирізати, копіювати, вставляти, переміщувати, змінювати гучність та обрізати аудіофрагменти. Також є можливість використовувати функції звукового згладжування і попередньої обробки шуму.
- Спектральний аналіз: Edison надає можливість відображати спектральний аналіз аудіосигналу. Ви можете побачити спектрограми, графіки гучності, візуалізації частот та інші корисні дані, що допоможуть вам в редагуванні і мастерингу аудіо.
- Відновлення аудіо: Edison має інструменти для відновлення аудіо. Ви можете видаляти шуми, статичний шум, паразитний звук та інші артефакти з аудіосигналу, щоб покращити якість звуку.
- Збереження і експорт: Edison дозволяє зберігати ваші редаговані аудіофайли у різних форматах, таких як WAV, MP3, FLAC та інші. Ви також можете експортувати аудіофайли безпосередньо у FL Studio для подальшої роботи над ними.

Це лише кілька основних функцій Edison у FL Studio 20. Завдяки своєму розширеному функціоналу, він дозволяє здійснювати різноманітні операції з аудіо, що робить його потужним інструментом для створення, редагування.

Вікно перегляду Edison зображено на рисунку 3.9.

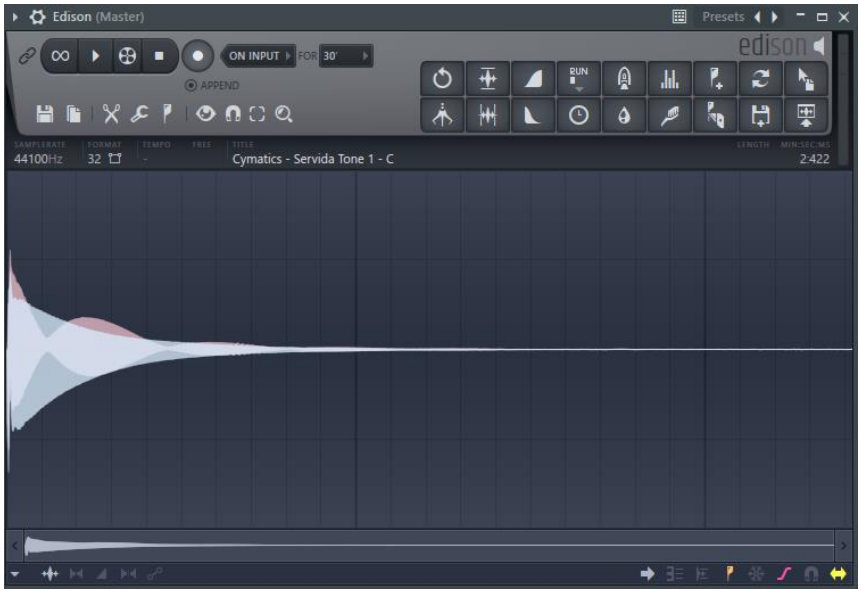

Рисунок 3.9 – Вікно Edison

3.4 Опис процесу захисту інформації в аудіофайлах

Ціль створеного засобу захисту інформації в аудіофайлах полягає у тому, що в аудіофайл приховується інформація. Інформація приховується шляхом конвертації повідомлення в ноти. Після цього ми отримаємо певну мелодію.

Отже, ми вже маємо аудіофайл з прихованим повідомленням. Саме повідомлення можна зашифрувати любим криптографічним засобом шифрування. Але доцільно використовувати криптографічні методи, а саме методи перестановки. Наприклад шифр Цезаря чи шифр Віженера. Ключ до зашифрованого повідомлення можна легко приховати в сам аудіофайл.

Після того як ми вже «розібрались» з повідомленням за допомогою вище згаданого ПЗ можна додати шуми, які зроблять неможливим декодинг повідомлення. Адже ми використовуватимо двофакторну систему – приховування зашифрованого повідомлення.

В цілому в цьому засобі захисту інформації є творчий потенціал і є куди прагнути для покращення і доведення до ідеалу цього засобу захисту інформації.

Ціль створеного засобу захисту інформації в аудіофайлах полягає у тому, що в аудіофайлі приховується інформація шляхом перетворення повідомлення в ноти, які потім вбудовуються в аудіофайл. Таким чином створюється мелодія, яка містить приховане повідомлення.

Саме повідомлення можна зашифрувати за допомогою будь-якого інструменту криптографічного шифрування. Однак бажано використовувати криптографічні методи, які використовують перестановку, такі як шифр Цезаря або шифр Віженера. Це пришвидшить процес декодування повідомлення отримувачем, що збільшить продуктивність даного засобу захисту інформації. Ключ до зашифрованого повідомлення можна легко сховати в самому аудіофайлі.

Після того, як повідомлення зашифровано і вбудовано в аудіофайл, до файлу можна додати шум, щоб ускладнити його декодування. Це створює двофакторну систему, яка ускладнює несанкціонований доступ користувачів до прихованої інформації.

3.5 Опис процесу приховування в аудіофайлах при використанні ПЗ Edison та FL Studio

Всього існує сім нот та ноти з діезом чи бемолем. В англійському алфавіті 26 букв, додаємо крапку та кому – виходить 28 символів. Застосувавши одну з функцій алгебри, ділення (28/7), ми отримаємо цифру 4. Це і буде наш спектр октав, тобто 4 октави.

Далі нам потрібно створити таблицю перетворень. Потрібно визначити, яка нота буде відповідати конкретній букві чи символу. В таблиці 3.2 зображені символи, які використовуватимуться для створення прихованого повідомлення.

Таблиця 3.2

Символи для створення повідомлення

A	H	O	V
B	I	P	W
C	J	Q	X
D	K	R	Y
E	L	S	Z
F	M	T	.
G	N	U	,

Далі, розберемо ноти, які використовуватимуться для приховування повідомлення. Є два можливих методи створення повідомлення.

Перший метод полягає у тому, що існують півтони (чорні клавіши на рисунку 25). Взагалі кажучи, із врахуванням півтонів, ми отримуємо 12 унікальних символів, але, як вже зазначалося, використовувати можна лише 2 октави з "копійками". Однак, якщо ми обмежимося лише цими нотами, то втратимо мелодичність, а результат виглядатиме підозріло, особливо якщо додати до них бемолі (чи дієзи).

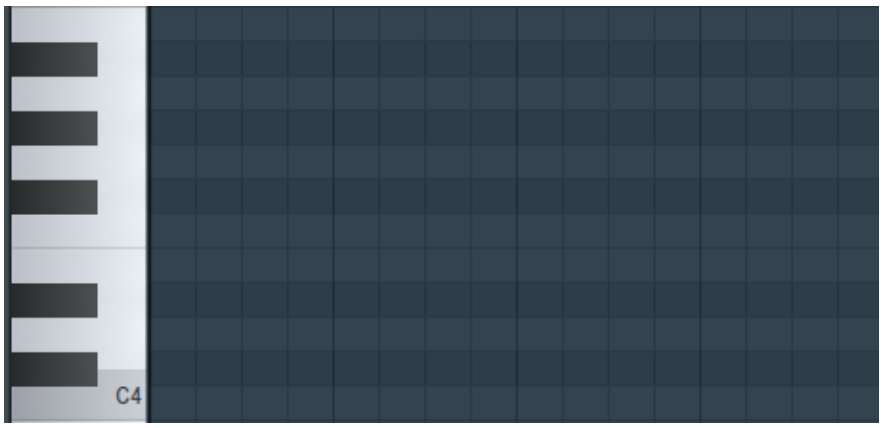

Рисунок 3.10 – Зображення нот октави C4

Однак, обмеження використанням лише двох октав з "копійками" може виявитися обмеженим. Якщо ми обмежимося лише цими нотами, то втратимо гнучкість і широту музичного виразу. Музична лінія стане одноманітною і непохідною, дозволяючи легше отримати несанкціонований доступ до інформації, яка приховується.

І другий метод полягає у тому, щоб уникнути цієї одноманітності, можна розширити музичний арсенал, використовуючи всі можливі «білі» ноти. В тональності Ля-мінор (A min) задіяні всі «білі» ноти. Таким чином, ми зможемо створити більш мелодичні повідомлення і додатково відволічемо від факту, що в аудіофайлі може бути приховане повідомлення.

В FL Studio є вбудована функція побудування тональностей, яка зображена на рисунку 3.11.

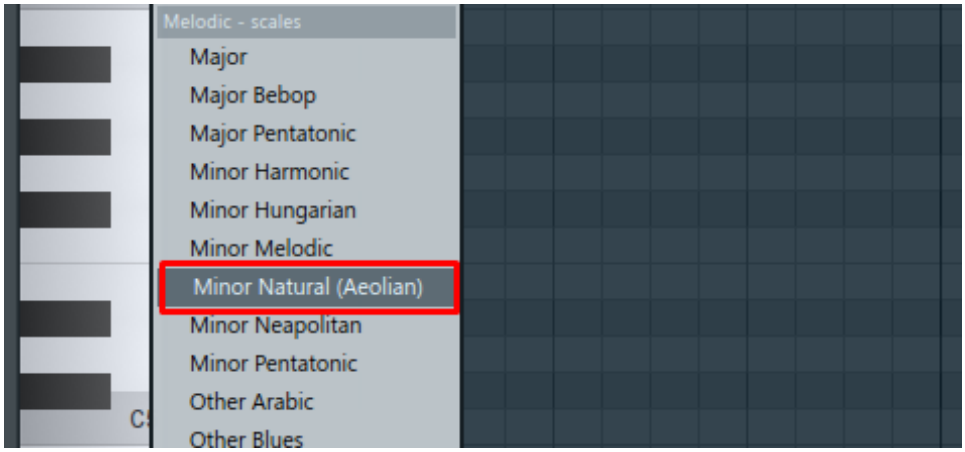

Рисунок 3.11 – Функція побудування тональностей

На рисунку 3.12 зображена тональність Ля-мінор (A min) в різних октавах. З цього можна зрозуміти, що інформація про використання білих нот є правдивою і можна використовувати це для побудування повідомлень в аудіофайлах.

На рисунку 3.12 зображено:

- Синім кольором позначені нотні лінії, які не будуть використовуватись для подальшої роботи;
- Зеленим кольором нотні лінії будуть використовуватись під час роботи для побудування повідомлення в нотній формі.

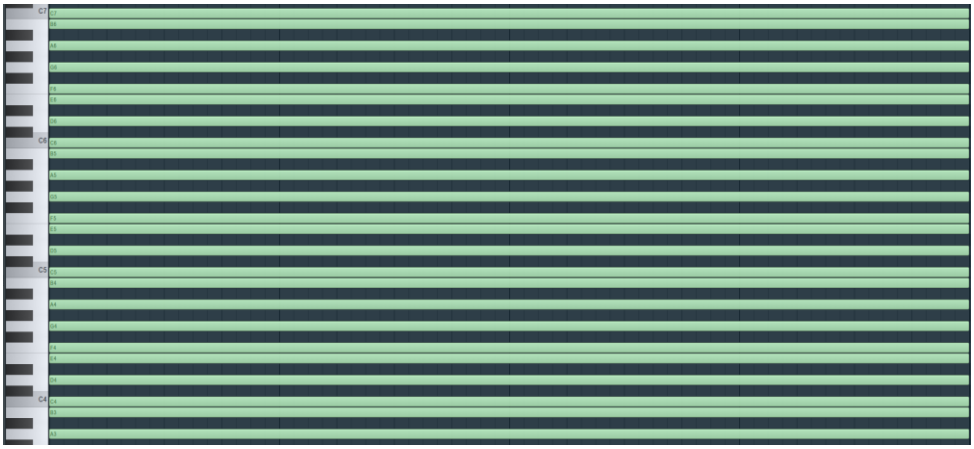

Рисунок 3.12 – Тональність Ля-мінор (A min)

3.6 Процес приховування інформації

На початку потрібно сформувати тестове повідомлення, яке ми будемо приховувати в аудіофайлі. Для цього візьмемо придумаємо довільний текст. Ось приклад повідомлення, яке я використовуватиму за приклад - «Mark what do you think of your home? How can I not love you my Kyiv?».

Коли вже є повідомлення, наступним кроком буде конвертація даних в ноти. Цей процес потрібний для того, щоб коректно вмісти повідомлення в аудіофайл. Для цього можна скористатись таблицею 3.2, і розроблений процес конвертації, який наведений в додатку А.

Далі потрібно використати програму, яка конвертуватиме введене повідомлення в ноти, зберігатиме файл в форматі MIDI. Після збереження програмою

файл, його можна просто перетягнути на робочу область Piano Roll і повідомлення буде вже у вигляді нот (Рисунок 3.13). Програмний код наведений у додатку Б.

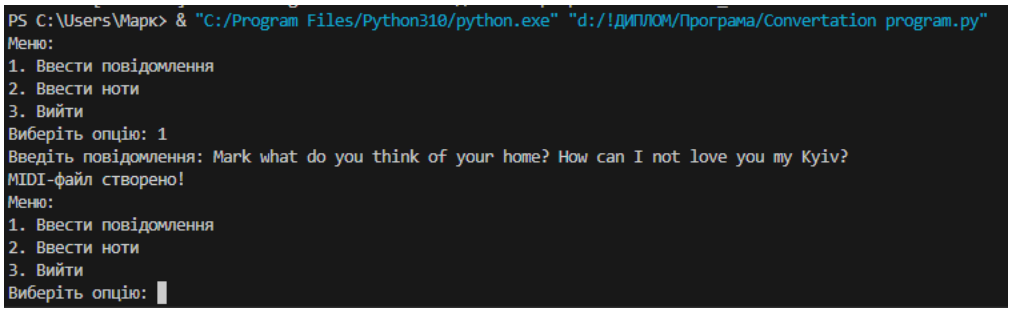

Рисунок 3.13 – Вікно програми-конвертатора

На рисунку 3.14 зображене повідомлення у вигляді нот. Якщо прослухати інформацію з аудіофайлу, який ми створили, то це може нагадати мелодію будильника чи саундтрек з якоїсь дитячої комп'ютерної гри. Тобто відкривши даний аудіофайл, зловмисник не подумає, що вміст аудіофайлу є прихованим повідомленням.

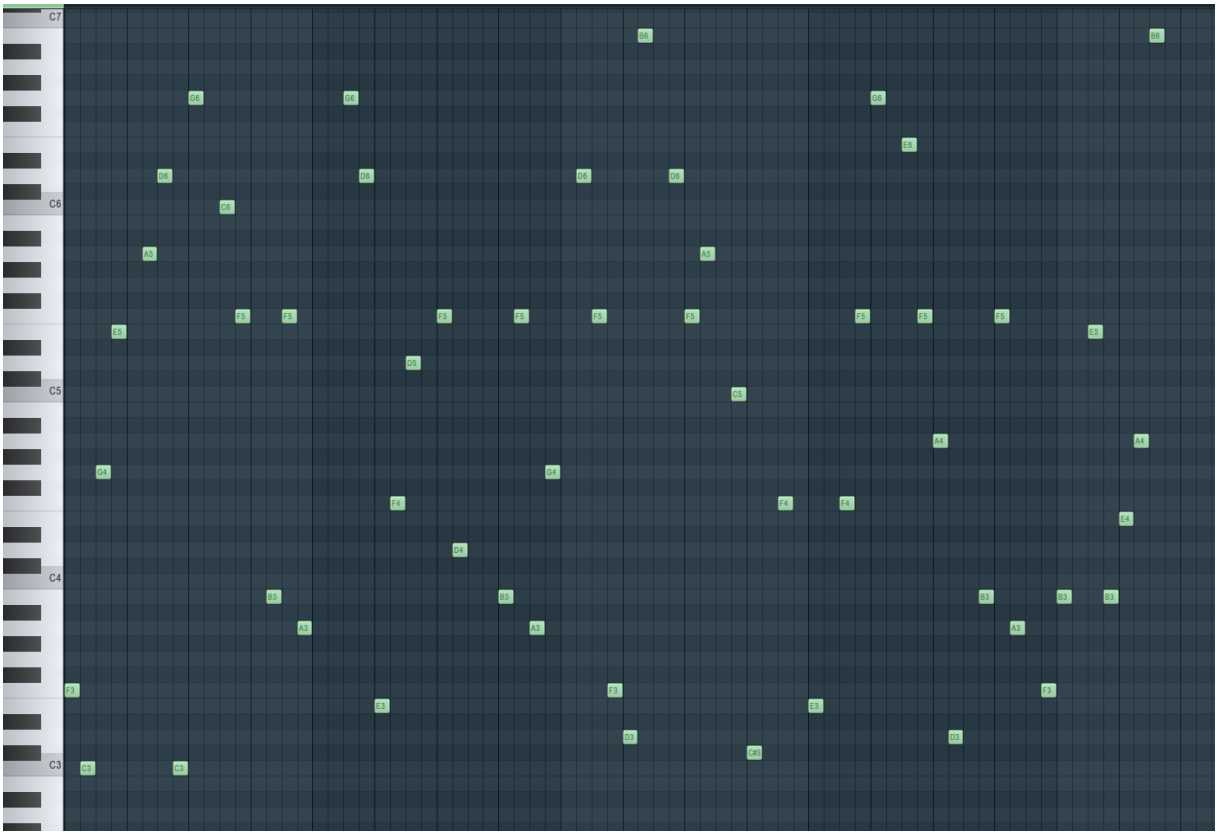

Рисунок 3.14 – Повідомлення зображене у вигляді нот в робочій області

Також існує метод ручного збереження повідомлення (музики) в MIDI-файл. Процес дуже простий, адже це вбудований функціонал любого сиквенсора. І в нашому випадку, процес збережнння зображений на рисунку 3.15.

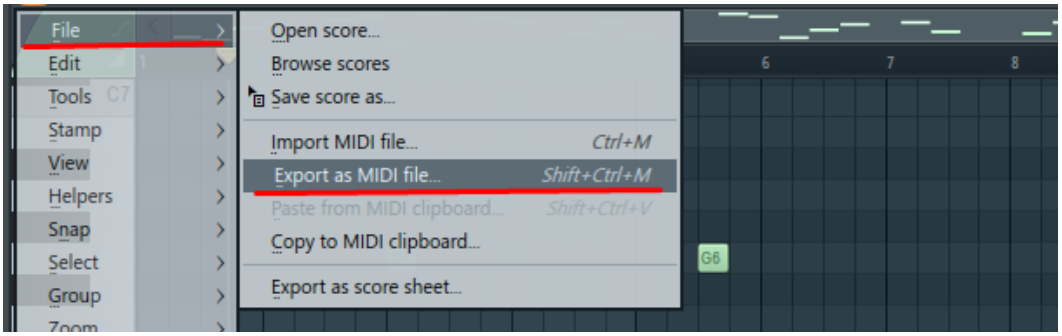

Рисунок 3.15 – Процес збереження повідомлення в MIDI-файл

Сам MIDI-файл не є найкращим вибором для передачі інформації в аудіофайлі. Тому слід використовувати формати, які були згадані в Розділі 1: MP3, WAV, FLAC.

На рисунку 3.16 зображене меню збереження файлу. В ньому можна вибрати формат аудіофайлу, який ти хочеш зберегти та ще додаткові налаштування окремо для кожного формату такі як: бітрейт, компресія, флоат.

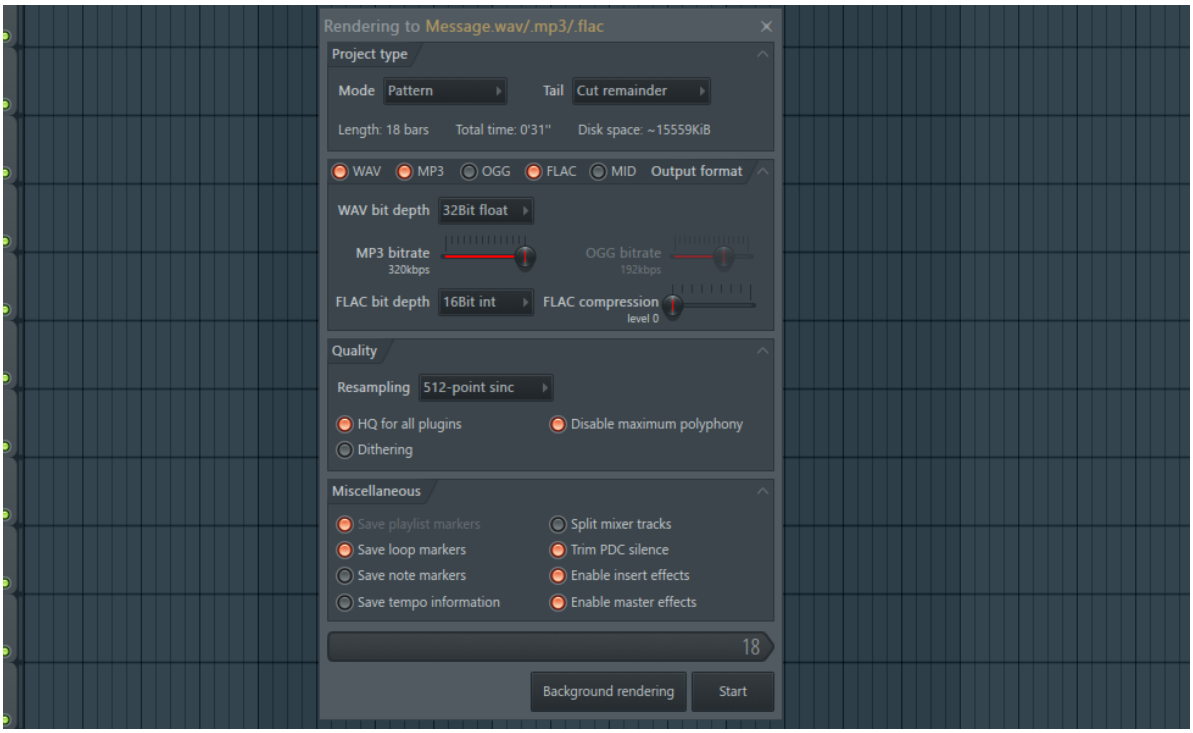

Рисунок 3.16 – Меню збереження файлу

Після введення налаштувань ми зберігаємо файл. В даному меню вибору формату не має значення який формат буде обрано, адже в меню збереження було вибрано і налаштовано, які саме аудіоформати будуть збережені.

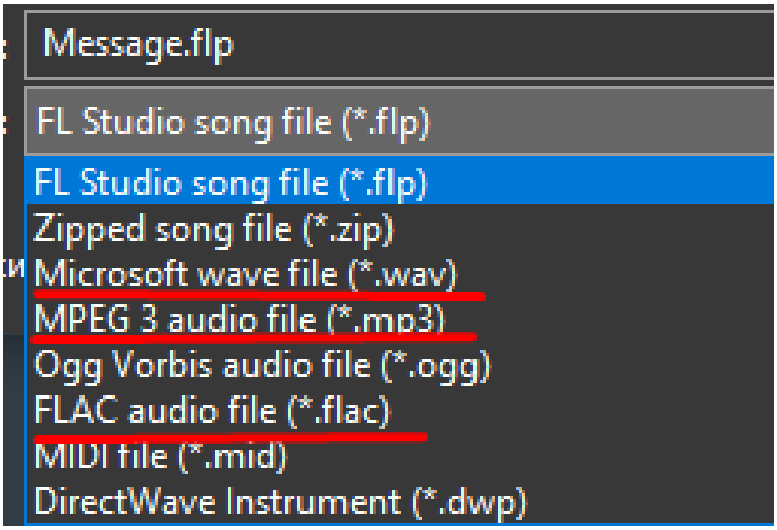

Рисунок 3.17 – Збереження аудіофайлу

На рисунку 3.18 зображено збережені файли, ми можемо побачити їх розмір. І зробити висновки, що файл MP3 формату найлегший та файл формату WAV найтяжчий.

Отже, якщо є ціль використовувати найменший за розміром аудіофайл з інформацією слід використовувати – MP3 або FLAC.

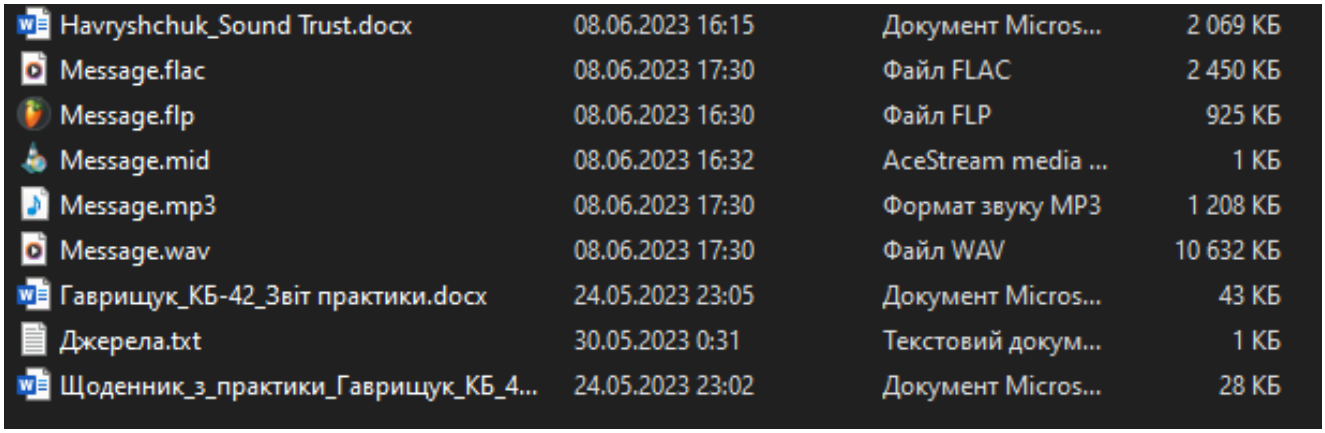

Рисунок 3.18 – Збережені аудіофайли з вбудованим повідомленням

Далі нам потрібно, вивантажити збережені файли в робочу область для накладання вотермарки та додаткових шумів. Продовжуємо роботу без WAV формату.

Оскільки попередньо згаданий формат є найтяжчим з використаних, а нас цікавить найменший розмір. На рисунку 3.19 зображені файли, які відправлені на канал мікшеру 40. Канал мікшеру – це певний аналог порта, тільки при роботі зі звуком. Саме на 40 каналі ми будемо накладати водяний знак та шуми для унеможливлення зчитування інформації.

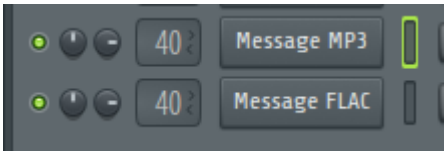

Рисунок 3.19 – Відправлені на 40 канал мікшеру аудіофайли з інформацією

Додаємо водяний знак на інформацію в аудіофайлі використовуючи вбудований функціонал FL Studio. В нашому випадку це моє ім’я та прізвище. Вигляд вотермарки зображений на рисунку 3.20

Рисунок 3.20 – Водяний знак у вигляді імені та прізвища

Далі нам потрібно накинути шуми, а саме реверберацію на файл (рисунок 3.21), це створить об’ємність звуку, програма зімітує ефект великого приміщення і звуки будуть накладатись один на одного. Тим самим нота буде змішуватись з сусідню ноту

і розібрати повідомлення зловмисник не зможе. До того ж при любій модифікації інформації в аудіофайлі чи самого аудіофайлу спадатиме водяний знак, що дасть нам інформацію щодо цілісності інформації в аудіофайлі.

Рисунок 3.21 – Ревербератор імітуючи велике приміщення

Отже тепер, аудіофайл з інформацією можна зберігати і відправляти будь-якими каналами зв’язку.

- Повідомлення приховане в аудіофайл.
- На аудіофайл накладена реверберація (шуми) та водяний знак.
- Цілісність та конфіденційність інформації переданої в аудіофайл забезпечено.

3.7. Процес декодування повідомлення в аудіофайлі

Для декодування нам потрібно використовувати Edison. Для цього нам потрібно відкрити аудіофайл в програмі.

Далі потрібно позбутись від лишніх шумів використовуючи функцію пробору шуму.

По-перше, потрібно вибрати область, яка буде ідентифікатором шуму, як зображено на рисунку 3.22.

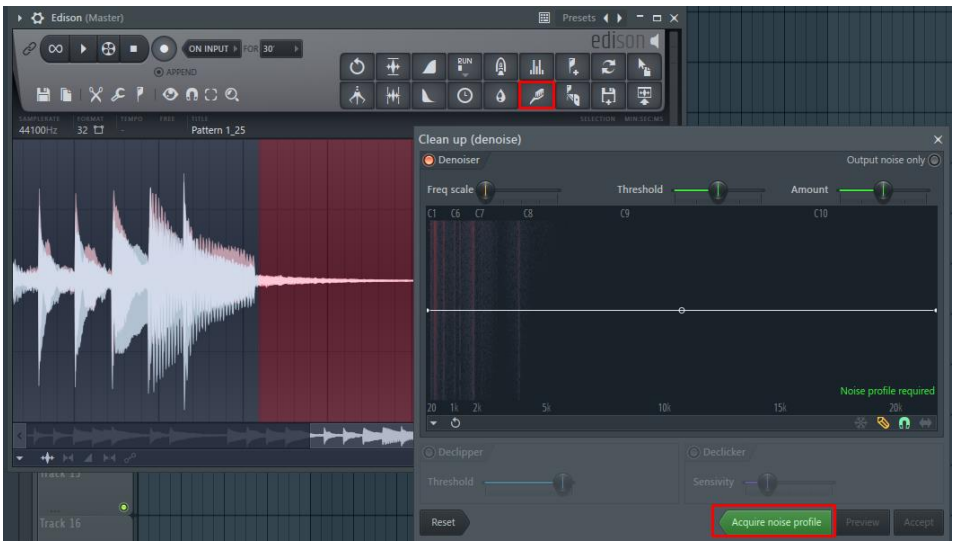

Рисунок 3.21 – Ідентифікація рівня шуму в аудіофайлі

По-друге, потрібно вибрати весь файл та як зображено на рисунку 3.23 натиснути кнопку «accept» і програма самостійно знизить шуми. Також можна бачити червоне забарвлення – це і є частина водяного знаку, який ми попередньо наклали на аудіофайл з інформацією.

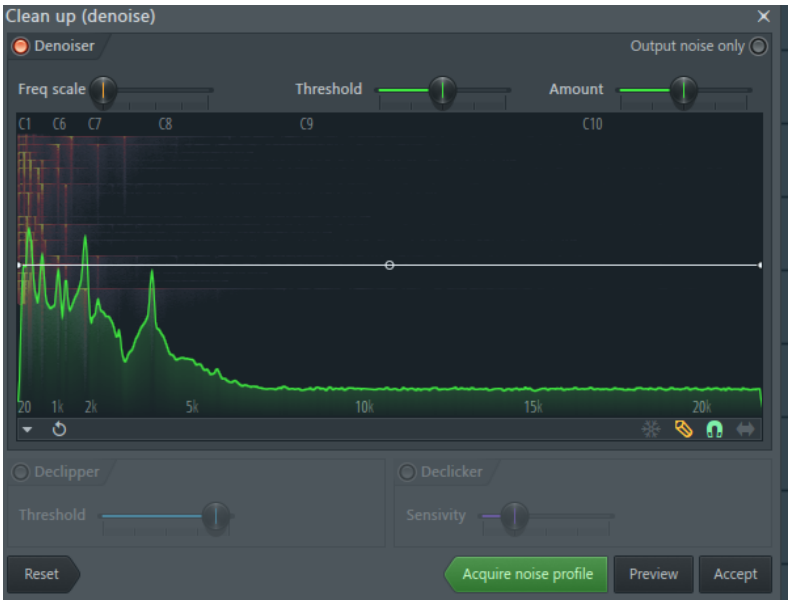

Рисунок 3.22 – Видалення шуму з аудіофайлу

По-третє, потрібно повернутись до головного меню Edison та як зображено на рисунку 3.23 задетектити ноти в аудіофайлі.

Рисунок 3.23 – Детектування нот використаних в аудіофайл

В результаті ми отримаємо ось такий вигляд, як на рисунку 3.24. Ці жовті помітки – ноти, які ми використовували. Тепер нам потрібно ввести ці значення в програму, яка зображена в додатку Б і ми отримаємо повідомлення, яке попередньо приховали в аудіофайл.

Рисунок 3.24 – Ноти-повідомлення в Edison

Висновки до третього розділу

Отже, був розроблений засіб захисту інформації в аудіофайлах форматів Wav, MP3, FLAC (далі – Sound Trust). Мета Sound Trust це приховувати повідомлення в аудіофайлах для подальшої можливої передачі його.

В Sound Trust використовується водяний знак, що забезпечує цілісність аудіофайлу.

Було розроблено програму-конвертатор, яка дає можливість швидко конвертувати повідомлення і додавати його в робочі області за для подальших модифікацій пов’язаних з захистом.

ВИСНОВКИ

Отже, було проведено дослідження поняття аудіофайлу, існуючих аудіоформатів та історії їх створення. В ході дослідження було розглянуто різні аудіоформати, такі як MP3, WAV, FLAC та інші, а також їх характеристики та особливості.

Подальше дослідження зосереджувалося на стеганографічних методах захисту інформації в аудіофайлах. Були розглянуті різні техніки та методи, що дозволяють приховувати інформацію у аудіофайлі, зберігаючи непомітність. Важливою складовою дослідження було вивчення вразливостей метаданих в аудіофайлах, які можуть містити конфіденційну інформацію.

На основі аналізу, досліджень та розуміння проблематики захисту інформації в аудіофайлах було розроблено новий засіб захисту інформації. Цей засіб базується на нових методах приховування інформації та додавання водяного знаку на аудіофайли.

Також була розроблена консольна програма, яка забезпечує конвертацію повідомлення в ноти і зворотну конвертацію з нот в повідомлення. Ця програма використовує розроблені методи захисту інформації в аудіофайлах для забезпечення безпеки даних в аудіофайлах. Вона дозволяє введення повідомлення або нот і перетворює їх у відповідні ноти або повідомлення. Крім того, програма здатна генерувати MIDI-файли на основі введеного повідомлення.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Історія MP3: як створювався популярний аудіоформат [Електронний ресурс] - Режим доступу: https://vikna.if.ua/cikavo/123189/view

2. Конахович Г. Комп'ютерна стеганографічна обробка й аналіз мультимедійних даних / Г. Конахович, Д. Прогонов, О. Пузиренко., 2018. – (Центр навчальної літератури) — 558 с.

3. Шифрування аудіофайлів WAV-формату [Електронний ресурс]: http://ir.lib.vntu.edu.ua/bitstream/handle/123456789/24890/DOKLAD.pdf?sequence=1&isAllowed=y

4. Image Steganography: Concepts and Practice - [Електронний ресурс]. – Режим доступу: http://sharif.edu/~kharrazi/pubs/ims04.pdf

5. Steganography: A Brief History - [Електронний ресурс]. – Режим доступу: https://www.techopedia.com/

6. Colin Raffel. LARGE-SCALE CONTENT-BASED MATCHING OF MIDI AND AUDIO FILES / Colin Raffel. – New York, 2015.

7. Audio steganography with enhanced LSB method for securing encrypted text with bit cycling, 2022. – 185-194 с. – (Bulletin of Electrical Engineering and Informatics). – (Vol. 11; № 1)

8. Alyousuf F. Analysis review on spatial and transform domain technique in digital steganography / F. Alyousuf, R. Din, A. Qasim.. – 122-129 с. – (Bulletin of Electrical Engineering). – (Vol. 9).

9. IEEE International Symposium on Signal Processing and Information Technology / M. Pooyan and A. Delforouzi – 600-603 p – Anchorage, AK, USA, 2007.

10. Md. Iqbal Hasan Sarker, Mohammad Ibrahim Khan, Kaushik Deb and Md. Faisal Faruque «FFT-Based Audio Watermarking Method with a Gray Image for Copyright Protection» - International Journal of Advanced Science and Technology Vol. 47, 2012.

11. DCT and DWT Based Robust Audio Watermarking Scheme for Copyright Protection / Kaushik Deb, Md. Ashikur Rahman, Kazi Zakia Sultana, Md. Iqbal Hasan Sarker Ui-Pil Chong – 3-8 p. – Південна Корея, 2014.

12. Seppanen T. Digital Audio Watermarking Techniques and Technologies / Seppanen T. – Burlington, 2007.

13. Digital Watermarking and Steganography / I.Cox,M.Miller, J.Bloom, J.Fridrich, T.Kalke – Burlington, 2007.

14. Методи цифрової стеганографії: стан та напрями розвитку / Мельник С., Київ, 2014.

15. Steganography: The Art of Hiding Information / K. Schmeh, P. Horster – 118p. – Paperback, 2005.

16. Офіційний сайт FL Studio [Електронний ресурс] - Режим доступу: https://www.image-line.com/

17. Ефективна схема нанесення водяних знаків для захисту медичних даних за допомогою нейронної мережі [Електронний ресурс] - Режим доступу: https://www.scielo.br/j/babt/a/yV9Qyc5K37fPBdmbWG6cQJL/?lang=en

18. Best Audio File Formats: What They Are And Why They Matter [Електронний ресурс] - Режим доступу: https://www.headphonesty.com/2020/04/best-audio-file-formats-explained/#What_Is_an_Audio_File_Format

19. Information Security of the Person, Society and State / Кащук. С. – 2013. – №3. – С. 65–70.

20. Шелест М. Комп’ютерна стеганографія та її можливості // М. Шелест, В. Андреев. // Сучасна спеціальна техніка. – 2011. – №24. – С. 97–104.

21. The Rise of Steganography / A. Siper, R. Farley, C. Lombardo – Pace University, 2005, 7p.

22. Handbook of Information and Communication Security / P. Stavroulakis, M. Stamp – International Journal of Advanced Science and Technology Vol. 17, 2009.

ДОДАТОК А

Таблиця-правило конвертації даних з символів в ноти

C3	A	C5	C
D3	E	D5	G
E3	I	E5	K
F3	M	F5	O
G3	Q	G5	S
A3	U	A5	W
B3	Y	B5	.
C4	B	C6	D
D4	F	D6	H
E4	J	E6	L
F4	N	F6	P
G4	R	G6	T
A4	V	A6	X
B4	Z	B6	?

ДОДАТОК Б

Програмна реалізація конвертації повідомлень

```
import csv
import os
import string
from midiutil import MIDIFile

# Зчитуємо дані з таблиці і створюємо словник для конвертації
conversion_table = {}
with open('conversion_table.csv', 'r') as file:
    reader = csv.reader(file, delimiter='\t')
    for row in reader:
        note = row[0]
        letters = row[1:]
        conversion_table[note] = letters

# Функція для конвертації нот в літери
def convert_to_letters(notes):
    letters = []
    for note in notes:
        if note in conversion_table:
            letters.append(conversion_table[note])
    return letters

# Функція для конвертації літер в ноти
def convert_to_notes(letters):
    notes = []
```

```
    for letter in letters:
        for note, letter_list in conversion_table.items():
            if letter in letter_list:
                notes.append(note)
    return notes

# Функція для створення MIDI-файлу з введеного повідомлення
def create_midi_file(message):
    # Визначаємо властивості MIDI-файлу
    track = 0
    channel = 0
    time = 0
    duration = 1
    tempo = 120

    # Ініціалізуємо MIDI-файл
    midi_file = MIDIFile(1)
    midi_file.addTempo(track, time, tempo)

    # Конвертуємо повідомлення в ноти і додаємо їх до MIDI-файлу
    notes = convert_to_notes(message)
    for note in notes:
        midi_file.addNote(track, channel, note, time, duration, velocity=100)
        time += 1

    # Зберігаємо MIDI-файл
    with open('output.mid', 'wb') as file:
        midi_file.writeFile(file)

# Головна функція
```

```
def main():
    while True:
        # Виводимо меню
        print("Меню:")
        print("1. Ввести повідомлення")
        print("2. Ввести ноти")
        print("3. Вийти")

        choice = input("Виберіть опцію: ")

        if choice == '1':
            message = input("Введіть повідомлення: ").upper()
            letters = list(filter(lambda x: x in string.ascii_uppercase, message))
            converted_notes = convert_to_notes(letters)
            create_midi_file(converted_notes)
            print("MIDI-файл створено!")

        elif choice == '2':
            notes = input("Введіть ноти (розділені пробілом): ").upper().split()
            converted_letters = convert_to_letters(notes)
            print("Конвертовані літери:", ' '.join(converted_letters))

        elif choice == '3':
            break

        else:
            print("Невірний вибір. Спробуйте ще раз.")

if __name__ == '__main__':
    main()
```