Центральноукраїнський національний технічний університет
Механіко-технологічний факультет
Кафедра кібербезпеки та програмного забезпечення

”Допущено до захисту”
Завідувач кафедри кібербезпеки
та програмного забезпечення
д.т.н., професор
_____ Олексій СМІРНОВ
“____” _____ 2025 р.

ВИПУСКНА КВАЛІФІКАЦІЙНА РОБОТА
за другим (магістерським) рівнем вищої освіти
на тему
“Дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM”

Виконав здобувач вищої освіти
ІІ курсу, групи КН-24М
ОПП «Комп’ютерні науки»
спеціальності 122 «Комп’ютерні науки»
_____ Скрипка М.А.
«____» _____ 2025 р.

Керівник проекту
кандидат технічних наук
_____ Смірнова Т.В.
«____» _____ 2025 р.
Рецензент _____
_____

м. Кропивницький

АНОТАЦІЯ

Скрипка М.А. Дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM. 122 Комп'ютерні науки. Центральноукраїнський національний технічний університет. Кропивницький. 2025.

В даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуального керування цифровими правами на основі DRM.

Метою розробки є дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

Об'єктом дослідження є процес інтелектуального керування цифровими правами на основі DRM.

Предметом дослідження є методи інтелектуального керування цифровими правами на основі DRM.

Методи дослідження базуються на методах захисту інтелектуальної власності, методах математичної статистики, методах розробки програмного забезпечення.

Результат роботи – програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

В процесі роботи над програмною моделлю виконано аналіз існуючих апаратних та програмних засобів. В повній мірі описані всі компоненти розробленого програмного забезпечення.

Розроблено зручний інтерфейс користувача. Наведені інструкції по роботі з програмними засобами.

Програма може використовуватися на ПЕОМ з ОС Windows 10/11.

Програму розроблено в середовищі Python.

Ключові слова: комп'ютерні науки, інтелектуальне керування цифровими правами

ABSTRACT

Skrypka M.A. Research and software implementation of the intellectual digital rights management system based on DRM. 122 Computer Science. Central Ukrainian National Technical University. Kropyvnytskyi. 2025.

In this final qualification work for the second (master's) level of higher education, software has been developed, which is intended for the intellectual digital rights management system based on DRM.

The purpose of the development is the research and software implementation of the intellectual digital rights management system based on DRM.

The object of the research is the process of intellectual digital rights management based on DRM.

The subject of the research is the methods of intellectual digital rights management based on DRM.

The research methods are based on methods of intellectual property protection, methods of mathematical statistics, methods of software development.

The result of the work is the software implementation of the intellectual digital rights management system based on DRM.

In the process of working on the software model, an analysis of existing hardware and software tools was performed. All components of the developed software are fully described.

A user-friendly user interface has been developed. Instructions for working with the software are provided.

The program can be used on a PC with Windows 10/11.

The program is developed in the Python environment.

Keywords: computer science, intelligent digital rights management

ЗМІСТ

ПЕРЕЛІК УМОВНИХ ПОЗНАЧЕНЬ, СИМВОЛІВ, ОДИНИЦЬ І ТЕРМІНІВ

ВДТ	–	відео-дисплейні термінали
ЕОМ	–	електронно-обчислювальна машина
ЕПТ	–	електронно-променева трубка
ЕЦП	–	електронний цифровий підпис
ЗІ	–	захист інформації
ПЗ	–	програмне забезпечення
ПК	–	персональний комп’ютер
СБ	–	служба безпеки
ТЗ	–	технічне завдання
ЦВЗ	–	цифрові водяні знаки
DES	–	стандарт шифрування США
DSA	–	Digital Signature Algorithm
ECDSA	–	Elliptic Curve Digital Signature Algorithm
EGSA	–	El Gamal Signature Algorithm
IDEA	–	International Date Encryption Algorithm – алгоритм шифрування
IP	–	Internet Protocol
LSB	–	Least Significant Bits – метод стеганографії
PGP	–	Pretty Good Privacy – міжнародний криптографічний стандарт
RSA	–	алгоритм асиметричного шифрування
SHA-1	–	Secure Hash Algorithm 1 – алгоритм криптографічного хешування
TDES	–	Triple DES –модифікація DES з трьома незалежними підключами
JPEG	–	Joint Photographic Experts Group – растровий формат зображення

ВСТУП

Актуальність теми. Управління цифровими правами (DRM) стосується алгоритмів та процесів, створених для забезпечення дотримання авторських прав під час споживання цифрового контенту. Без DRM кінцевий користувач може легко скопіювати ваш контент. Цей процес зазвичай називають піратством. Таким чином, це необхідно в архітектурі онлайн-розповсюдження відео, але споживач не бачить цього.

Найбільш відоме призначення DRM – захист добутків від копіювання й інших дій, що забороняються авторами або іншими правовласниками на підставі авторських або суміжних прав після продажу кінцевому користувачеві. Термін «технічні засоби захисту авторських прав» використовується в законодавстві України, що забороняє обхід таких засобів. Закон про Авторське Право в Цифрову Епоху був прийнятий у США в 1998 році з метою ввести кримінальну відповідальність за обхід засобів DRM. [1]

Мета й завдання дослідження. Метою роботи є дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

Для досягнення поставленої мети визначена програма дослідження, що складається з наступних завдань:

– Огляд існуючих систем інтелектуального керування цифровими правами на основі DRM.

– Дослідження системи інтелектуального керування цифровими правами на основі DRM.

– Програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

Об'єктом дослідження є процес інтелектуального керування цифровими правами на основі DRM.

Предметом дослідження є методи інтелектуального керування цифровими правами на основі DRM.

Методи дослідження базуються на методах захисту інтелектуальної власності, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод інтелектуального керування цифровими правами на основі DRM.

– Розроблено вітчизняний продукт інтелектуального керування цифровими правами на основі DRM, який має більш широкі можливості, на відміну від існуючих аналогів.

Практична цінність отриманих результатів полягає в тому, що розроблені алгоритми дозволяють успішно вирішувати задачі інтелектуального керування цифровими правами на основі DRM.

Достовірність наукових результатів підтверджена теоретичними викладеннями, даними комп'ютерного моделювання, коректними дослідженнями параметрів на функціонуючій обчислювальній мережі, а також відповідністю отриманих результатів окремим результатам, наведеним у науковій літературі.

Робота апробована на LVII Науково-технічній конференції здобувачів вищої освіти LV науково-технічної конференції «Наука в ЦНТУ: основні досягнення та перспективи розвитку» (2025 р.), основні положення випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти надруковані у статті збірника праць молодих науковців ЦНТУ, випуск №15.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

1 ПРИЗНАЧЕННЯ ТА ОБЛАСТЬ ВИКОРИСТАННЯ

1.1 Призначення системи

«DRM» – абревіатура від англійського вираження «digital rights management», слова якого переводяться як «керування цифровими правами». Деякі супротивники DRM призивають розшифровувати другу букву як «restrictions» – «обмеження». Потреба в потокових можливостях серед медіаіндустрії та дистриб'юторів відеоконтенту загалом є найвищою за весь час.

Споживачі та розробники змагаються у пошуку та розповсюдженні найкращого контенту, який є в їхньому розпорядженні. На жаль, цей високий попит на відеоконтент часто підривається відсутністю безпеки оригінальних цифрових активів. В результаті, творці та розповсюджувачі опиняються в ситуації, коли їм потрібно захищати себе та свої матеріали, захищені авторським правом, від неавторизованих користувачів; вводяться технології DRM.

1.2 Область застосування

Наразі управління цифровими правами може бути реалізоване як програмне та/або апаратне рішення; і в більшості випадків воно реалізується як комбінація обох. Незалежно від типу апаратного чи програмного забезпечення DRM, усі постачальники, які прагнуть захистити свій цифровий контент, побачать, як їхні файли проходять цикл шифрування та дешифрування.

Таким чином, виходячи з вищеперерахованого, дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM, є актуальною задачею, яка потребує вирішення у даній випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти.

2 ПЕРЕГЛЯД АНАЛОГІЧНИХ ІСНУЮЧИХ СИСТЕМ

2.1 Огляд існуючих систем, технологій, архітектур, програмних рішень за профілем теми випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти

Anubis

Анубіс – супергерой древніх єгиптян. Людина-шакал, на честь якого через шість тисяч років назвали безкоштовну програму. Утиліта вийшла кросплатформної, але потребуючої установки JRE, а також (у випадку з Windows 10-11) віртуальної машини DOS – NTVDM.

Основне вікно програми виглядає максимально аскетично. Натискаємо Encrypt і у вкладці, що відкривається, указуємо необхідні дії: який файл помістити усередину якого й де зберегти результат. Гарантовано працює тільки приховання текстових файлів всередині картинок формату BMP. Трохи таких уже є в Windows 10 – це іконки користувачів. Було б цікаво сховати в user.bmp список паролів або ще яку-небудь конфіденційну інформацію. Давно помічено, що кращі схованки стоять на видному місці.

Додатково можна захистити отриманий файл пин-кодом – тоді він буде потрібно для зворотного перетворення. Утиліта некоректно обробляє рядок із вказівкою місця результуючого файлу. Він може бути збережений на рівень вище заданого або взагалі у вихідному каталозі.

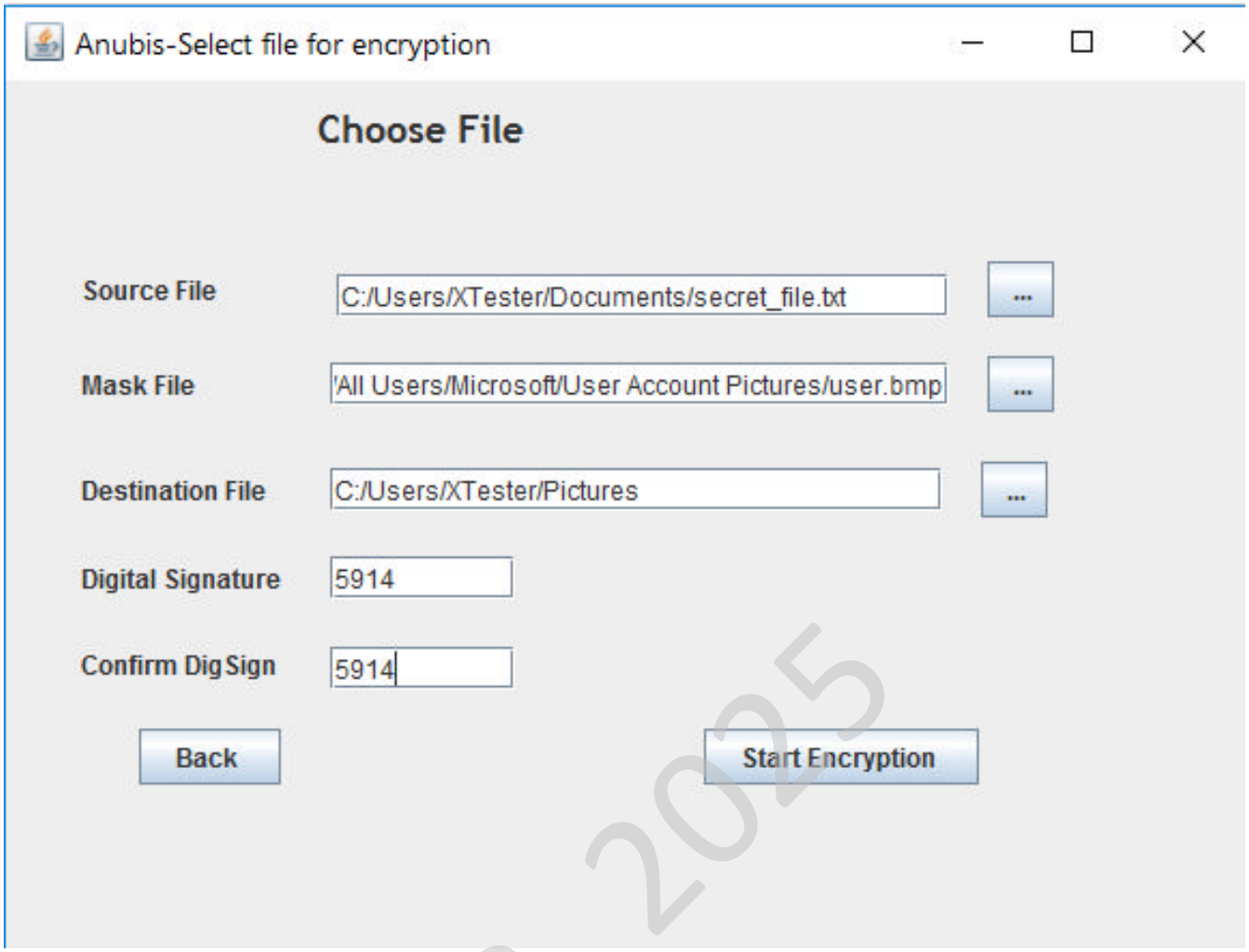

Рисунок 2.1 – Ховаємо TXT в BMP

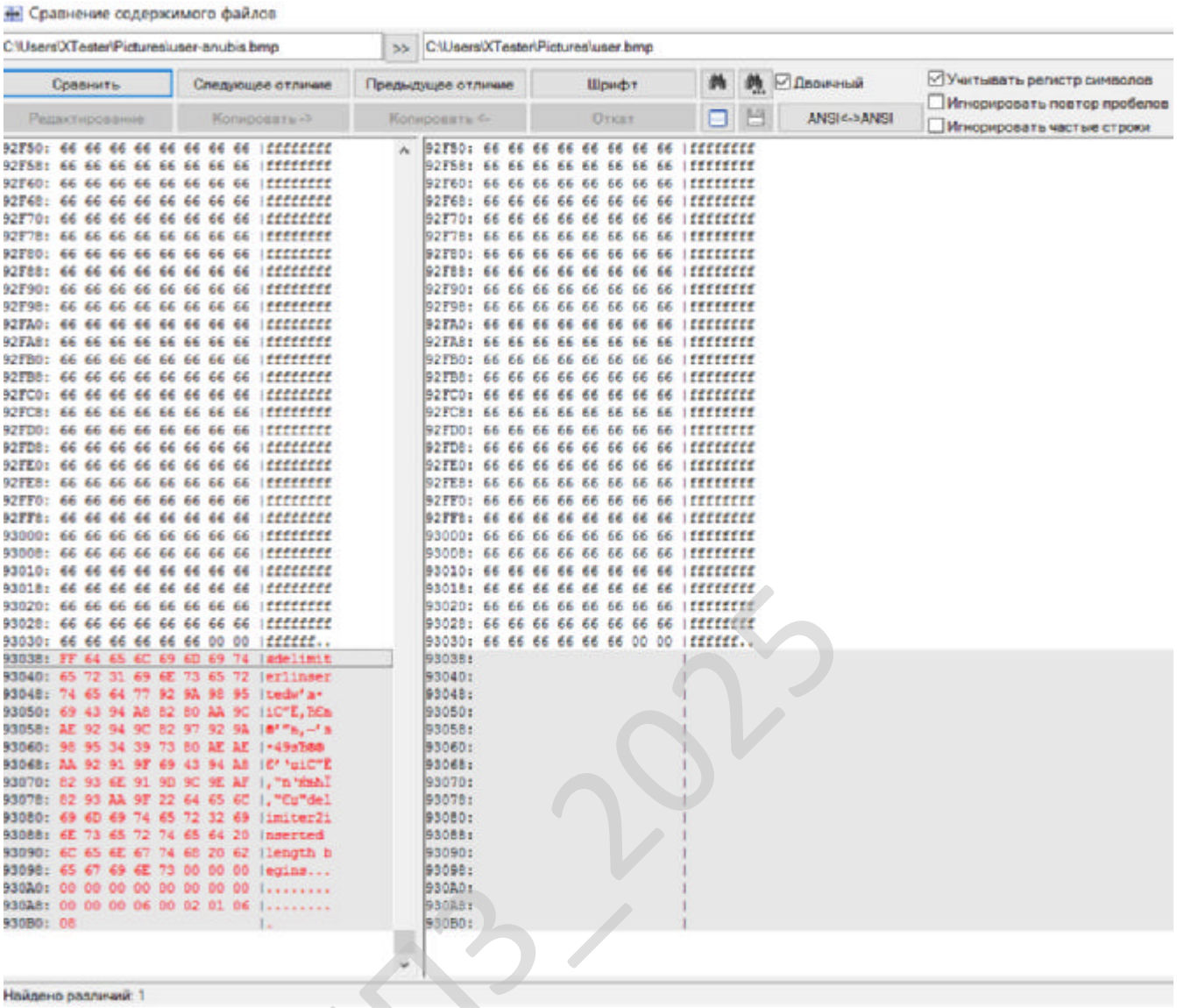

Рисунок 2.2 – Порівняння порожнього й наповненого контейнера

Як видно з побайтного порівняння вихідного файлу BMP з утримуючий схований текст, програма працює дуже примітивно. Вона просто дописує дані в кінець файлу. Дані зашифровані, але постачені характерними покажчиками: limiter1, limiter2,inserted length begins. Простим пошуком файлів, що містять такі рядки, легко знайти всі стегоконтейнери. Таку утиліту можна використовувати як ілюстрація найпростішого методу стеганографії, але для серйозних завдань вона зовсім не підходить.

DeEgger Embedder

DeEgger Embedder – ще одна маленька програма для стеганографії. У ній реалізований уже більший набір функцій, але його використання вимагає установки .NET Framework 3.5. Крім рідко використовуваних сьогодні картинок BMP, програма підтримує як контейнери PNG, JPG, відеофайли AVI і музичні MP3. Утиліта веде докладний лог своїх дій, що відображається прямо в головному вікні.

Кнопка запуску алгоритму називається Combine, а не Encrypt, що більш точно відбиває процес впровадження файлів. Витягають приховувані файли (стегоповідомлення) з мультимедійних контейнерів натисканням єдиної кнопки Extract. Ніякого захисту пін-кодом тут немає.

Зате програма може обробляти кілька файлів відразу. Можна помістити кілька повідомлень в один контейнер або одне в різні контейнери.

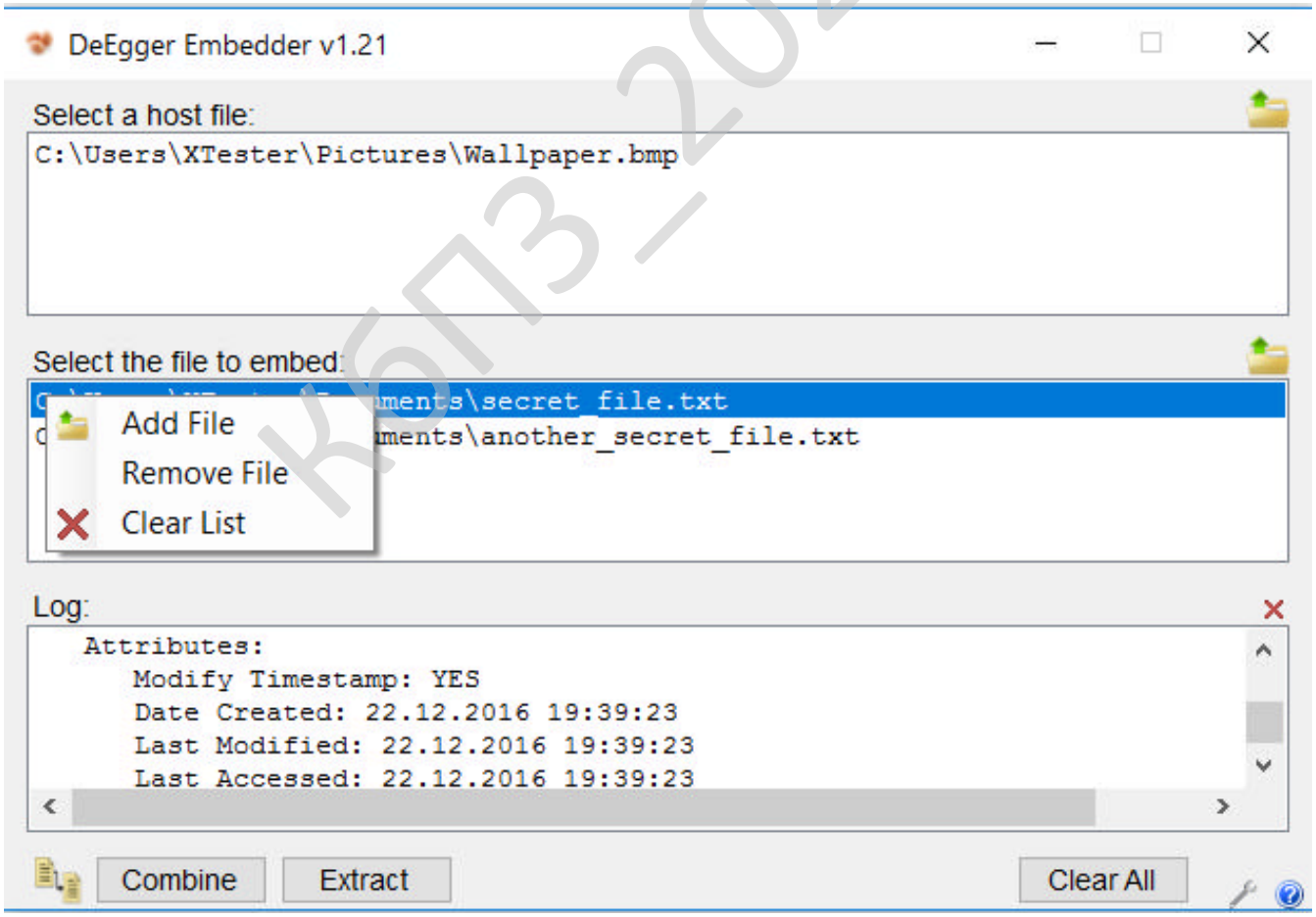

Рисунок 2.3 – Записуємо кілька файлів в один контейнер

Після обробки в DeEgger утиліти для порівняння зображень уважають ідентичними вихідний і кінцевий файли BMP. Реально ж це порожній і заповнений контейнер у термінології стеганографії.

Зробимо побайтне порівняння. Знайома картина? Так само як і Anubis, утиліта DeEgger Embedder дописала стегоповідомлення в кінець файлу-контейнера. У картинці user.bmp багато однотонних областей, тому такий апендикс виглядає особливо помітно.

На перший погляд, тут немає явних покажчиків, по яких можна зробити пошук файлів, що містять певний рядок. Однак придивимося уважніше. Для цього зробимо ще один контейнер з іншим повідомленням і зрівняємо вже два заповнених контейнери між собою.

От однакова ділянка в шістнадцятковому виді: 24 23 26 29 2A 40 26 28 23 5E 2A 00 D1 8B 87 8B FF.

Як бачимо, незважаючи на підтримку більшого числа форматів, DeEgger недалеко пішов від Anubis. Приховувані файли так само записуються в кінець файлу-контейнера й мають характерний вигляд, по якому їх легко виявити.

Irdeto

Irdeto – світовий лідер галузі безпеки цифрових платформ, що обслуговує компанії у сфері відеорозваг, відеоігор, підключеного транспорту та підключених галузей Інтернету речей. Вони надають клієнтам можливість захищати свої доходи, впроваджувати інновації з новими пропозиціями та ефективно боротися з кіберзлочинністю. Маючи 50-річний досвід у сфері безпеки, Irdeto наразі захищає понад 5 мільярдів пристроїв та програм для відомих брендів по всьому світу. Їхня заявлена місія – створити безпечне майбутнє, надаючи людям можливість користуватися зв’язком без шкоди для безпеки та довіри.

NAGRA

NAGRA, підрозділ цифрового телебачення Kudelski Group (SIX:KUD.S), спеціалізується на пропонуванні комплексних рішень безпеки та багатоекранного користувацького досвіду для монетизації цифрових медіа.

Їхня експертиза полягає в оснащенні провідних постачальників контенту та операторів цифрового телебачення по всьому світу безпечними, відкритими та бездоганно інтегрованими платформами та додатками для мовлення, широкосмугового зв'язку та мобільних платформ. H4: Verimatrix

Verimatrix

Verimatrix є світовим постачальником надійних рішень у сфері безпеки та аналітики, що спеціалізується на захисті пристроїв, послуг та програм на широкому спектрі ринків.

Незліченні постачальники послуг та новатори галузі довіряють Verimatrix захистити важливі системи, від яких люди покладаються щодня.

Verimatrix пропонує зручні програмні рішення, хмарні сервіси та передові кремнієві IP-адреси, забезпечуючи надійні заходи безпеки та надаючи підприємствам цінну аналітику та аналітичні дані.

PallyCon

PallyCon, що працює на базі INKA ENTWORKS, є лідером галузі, що пропонує перше хмарне SaaS-рішення для комплексної безпеки контенту.

Їхнє комплексне рішення охоплює широкий спектр функцій, включаючи Multi DRM, судово-медичне додавання водяних знаків, видимі водяні знаки, захист від захоплення екрана, послуги боротьби з піратством та безпеку додатків, всі вони бездоганно інтегровані в єдиний робочий процес.

Маючи понад 20 років досвіду в галузі безпеки контенту, PallyCon надає клієнтам можливість захистити свої доходи за допомогою масштабованого, глобально доступного, надійного та економічно ефективного рішення.

Intertrust ExpressPlay

Intertrust ExpressPlay надає комплекс послуг захисту та боротьби з піратством, призначених для власників прав та розповсюджувачів як живого контенту, так і контенту на вимогу.

Intertrust ExpressPlay

Їхній хмарний пакет ExpressPlay Media Security Suite пропонує такі рішення, як служба ExpressPlay з кількома DRM-захистом, рішення для безпеки мовлення ExpressPlay XCA, а також служби захисту від піратства та водяних знаків ExpressPlay.

Вони відомі своєю масштабованістю та користуються довірою основних потокових OTT-платформ по всьому світу. Крім того, ExpressPlay DRM Offline забезпечує безпечну потокову передачу преміум-контенту через офлайн-платформу з кількома DRM.

EZDRM

EZDRM – експерт у сфері управління цифровими правами як послугою (DRMaaS), що надає комплексні рішення для захисту та монетизації відеоконтенту. Вони існують з 2001 року.

EZDRM

Вони використовують розміщену та керовану пропозицію з кількома DRM, розроблену для спрощення підтримки послуг доставки відео в реальному часі, на вимогу, для завантаження та офлайн-відео. Вони дуже гнучкі, коли йдеться про врахування різних бізнес-моделей.

Їхня універсальна DRM-технологія поєднує в собі Widevine від Google та PlayReady від Microsoft з використанням Common Encryption (CENC) поверх DASH, а також потокову передачу Apple FairPlay від EZDRM.

BuyDRM

BuyDRM – відомий постачальник послуг безпеки контенту, що обслуговує такі галузі, як розваги, освіта, підприємництво та готельний бізнес.

Працюючи під управлінням OVHcloud, платформа безпеки контенту KeyOS від BuyDRM широко використовується відомими брендами в медіа та технологічному секторах.

Вони мають великий досвід у впровадженні рішень безпеки комерційного контенту та медіатехнологій, а також мають хороший досвід роботи з такими

великими брендами, як ABC (Австралійська мовна корпорація), AMPAS (The Academy), Blizzard Entertainment, Cinedigm, Crackle, Crunchyroll, Daily Rounds, Deluxe Digital, EPIX, FuboTV, POPS Worldwide, Rakuten Viki, Redbox, SBS Belgium, Sinclair Digital та Zee5.

Axinom

Axinom – відомий постачальник цифрових рішень, що обслуговує провідні бренди в медіа та розважальній індустрії.

Аксіном

Їхній портфель OTT охоплює управління контентом (CMS), DRM та попередньо створені довідкові програми (додатки) для контенту на вимогу, подій у прямому ефірі та лінійного контенту в прямому ефірі.

Axinom може запропонувати комплексне рішення, яке охоплює весь робочий процес, від отримання відео до його доставки на різні пристрої, такі як HTML5, iOS, Android, Windows 10, Xbox, телеприставки та Smart TV.

Axinom зосереджена на створенні OTT-відеорішень наступного покоління, які забезпечують швидкий вихід на ринок.

Friend MTS

Friend MTS – надійний постачальник рішень для безпеки контенту для медіа- та розважальних компаній.

Їхні передові послуги охоплюють комплексне вимірювання, моніторинг, виявлення та запобігання піратству контенту. Пропонуючи цілісний підхід до боротьби з онлайн-піратством, Friend MTS надає бізнесу чітке розуміння постійно мінливого ландшафту піратства.

Вони проактивно випереджають складні технології та поведінку онлайн-піратства, забезпечуючи зростання доходів та процвітання творчості в безпечному середовищі.

2.2 Обґрунтування вибору засобів для побудови системи та мови програмування

Python – це об’єктно-орієнтована мова програмування високого рівня загального призначення з відкритим кодом. Це визначення може бути важким для новачків, тому розглянемо кожну характеристику окремо, щоб зрозуміти, що вона означає:

– Відкритий вихідний код: це безкоштовно та доступно для подальших покращень, таких як додавання корисних функцій або виправлення помилок.

– Об’єктно-орієнтована: заснована не на функціях, але в об’єктах з певними атрибутами й методами.

– Високий рівень: зручний для людини, а не для комп’ютера.

– Загальне призначення: можна використовувати для створення будь-яких програм.

Ця мова використовується в будь-якому програмному забезпеченні, про яке ви тільки можете подумати. Ви можете використовувати його для створення веб-сайтів, штучного інтелекту, серверів, програмного забезпечення для бізнесу та багато іншого. Також застосовується в науці про дані, аналізі даних, машинному навчанні, інженерії даних, веб-розробці, розробці програмного забезпечення та інших галузях.

Переваги та недоліки Python

Переваги:

– Її легко читати, вчити та писати. Це мова програмування високого рівня з англійським синтаксисом. Це полегшує читання та розуміння коду. Її дійсно легко зрозуміти і вивчити, тому багато людей рекомендують Python новачкам. Вам потрібно менше рядків коду для виконання того ж завдання в порівнянні з іншими основними мовами, такими як C/C++ та Java.

– Підвищує продуктивність. Це дуже продуктивна мова. Завдяки її простоті розробники можуть зосередитися на розв’язанні проблеми. Їм не

потрібно витрачати багато часу на розуміння синтаксису або поведінку мови програмування. Ви пишете менше коду та виконуєте більше завдань.

– Інтерпретована мова. Python мова, що інтерпретується, а це означає, що вона безпосередньо виконує код по рядку. Якщо сталася помилка, вона зупиняє подальше виконання та повідомляє про її виникнення. Вона показує лише одну помилку, навіть якщо у програмі їх кілька. Це спрощує налагодження.

– Динамічно типізована. Python не визначає тип змінної, доки ми не запустимо код. Вона автоматично надає тип даних, коли відбувається процес виконання. Фахівець може не турбуватися про оголошення змінних та типи даних.

– Безкоштовна та з відкритим вихідним кодом. Ця мова постачається під схваленою OSI ліцензією з відкритим вихідним кодом. Це робить його безкоштовним для використання та розповсюдження. Ви можете завантажити вихідний код, змінити його та навіть розповсюджувати свою версію. Це корисно для організацій, які хочуть використати свою версію для розробки.

– Підтримка великих бібліотек. Стандартна бібліотека Python є величезною, ви можете знайти майже всі функції, необхідні для вашого завдання. Таким чином ви не залежите від зовнішніх бібліотек.

– Портативність .У багатьох мовах, таких як C/C++, потрібно змінити свій код, щоб запустити програму на різних платформах. З Python все інакше. Ви тільки пишете один раз і запускаєте її будь-де.

Недоліки:

– Низька швидкість. Вище ми обговорювали, що це інтерпретована мова з динамічною типізацією. Порядкове виконання коду часто призводить до повільного виконання. Динамічна природа Python також є причиною її низької швидкості, оскільки їй доводиться виконувати додаткову роботу при виконанні коду. Тому вона не підходить для цілей, де швидкість важливий аспект проєкту.

– Неефективна для пам’яті. Ця мова програмування використовує великий обсяг пам’яті, це може бути недоліком при створенні програм, коли віддають перевагу оптимізації пам’яті.

– Слабка у мобільних обчисленнях. Python зазвичай використовується у серверному програмуванні. Ми не бачимо – її на стороні клієнта або в мобільних програмах з таких причин: вона не заощаджує пам’ять і має повільну обчислювальну потужність у порівнянні з іншими мовами.

– Доступ до бази даних. Програмувати на цій мові легко, але коли ми взаємодіємо з базою даних, її не вистачає. Рівень доступу до бази даних у Python примітивний та недостатньо розвинений у порівнянні з іншими популярними технологіями.

– Помилки виконання. Це мова з динамічною типізацією, тому тип даних змінної може змінюватись у будь-який час. Змінна, що містить ціле число, у майбутньому може містити рядок, що може призвести до помилок виконання.

Застосування Python:

– Для аналізу даних. Дані стали цінним активом у будь-якій сучасній галузі, і більшість компаній зацікавлені у збиранні, обробці та аналізі релевантних даних, щоб витягти з них цінну інформацію для бізнесу. І тут Python виходить за межі будь-якої конкуренції. Python особливо цінна тим, що крім великої стандартної бібліотеки надає величезний набір додаткових модулів, розроблених спеціально для аналітичних цілей. Найвідоміші бібліотеки Python для аналізу даних – це pandas і NumPy . Ці інструменти дозволяють робити з вашими даними майже все, наприклад, очищати і аналізувати їх, вивчати статистику або візуалізувати приховані тенденції у ваших даних.

– Для візуалізації даних. Візуалізація даних – це окрема частина аналізу даних, яка допомагає нам подавати інформацію, необроблену чи очищену, у більш змістовній формі. Тут Python знову входить у гру, пропонуючи широкий спектр інструментів візуалізації даних. Найпопулярніші з них – matplotlib і

заснований на ній seaborn. Використовуючи їх, ми можемо створювати буквально всі види візуалізації: від найпростіших до складніших.

– Для машинного навчання. Машинне навчання (ML) є основою більшості завдань науки даних. Він є областю штучного інтелекту, пов'язаною з використанням алгоритмів, що дозволяють машинам вивчати закономірності та тенденції на основі історичних даних, щоб робити прогнози на основі невідомих даних. – Використовуючи методи ML, ми можемо створювати моделі, які можуть точно передбачити швидкість відтоку клієнтів компанії, оцінити ризик виникнення у людини певного захворювання, визначити оптимальне розташування автомобілів таксі й т.д. За допомогою Python ми можемо побудувати модель ML, використовуючи лише три рядки коду.

– Для розробки програмного забезпечення. Крім свого багатостороннього застосування в галузях науки про дані, Python використовується на кожному етапі розробки програмного забезпечення, включаючи контроль складання, автоматичну безперервну компіляцію, прототипування, відстеження помилок, тестування та обслуговування програмного забезпечення. За допомогою цієї мови можемо створювати аудіо- або відеопрограми на основі методів штучного інтелекту, машинного навчання, API (інтерфейсів прикладного програмування), GUI (графічних інтерфейсів) або будь-якого іншого типу програмного забезпечення.

– Для веброзробки. У той час як для створення візуальної частини вебсайту ми переважно будемо використовувати такі мови, як HTML, CSS та JavaScript, для його невидимої частини ми часто вибираємо Python. Серед масштабних вебсайтів та програм, створених за допомогою цієї мови, варто згадати Google, Facebook, Instagram, YouTube, Dropbox та Reddit.

– Для автоматизації задач/скриптингу. Це відмінний інструмент для написання програм для автоматизації різних завдань, що повторюються. Цей процес називається скриптингом. Зокрема, можна робити скрипти для роботи з файлами та папками. Наприклад, можна створювати, перейменовувати,

перетворювати, розділяти, об’єднувати або видаляти файли, перевіряти їх на наявність помилок. Ви також можете використовувати автоматизацію Python для пошуку та завантаження інформації з Інтернету, заповнення та надсилання онлайн-форм та надсилання регулярних повідомлень або електронних листів.

Яким фахівцям потрібно володіти Python:

– Фахівець з даних.

– Аналітик даних.

– Інженер даних.

– Інженер з машинного навчання.

– Журналіст даних.

– Архітектор даних.

– Повний стек веб-розробника.

– Backend-розробник.

– DevOps-інженер.

– Інженер-програміст.

Можемо зробити висновок, що Python ще довго буде популярною мовою, хоч і має низку недоліків. Цю мову використовують для створення вебсайтів, штучного інтелекту, серверів, програмного забезпечення для бізнесу, аналізу даних, машинного навчання, інженерії даних та для багатьох інших областей. Це перспективна і затребувана навичка, яка необхідна у всіх галузях.

2.3 Розгорнута постановка завдання

Згідно з технічним завданням на випускну кваліфікаційну роботу за другим (магістерським) рівнем вищої освіти, реалізації підлягає програмне забезпечення, яке призначено для системи інтелектуального керування цифровими правами на основі DRM.

В процесі розробки випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти необхідно виконати наступний обсяг роботи:

а) провести аналіз існуючих систем-аналогів для виявлення їх позитивних і негативних якостей. Результати аналізу врахувати в подальших розробках;

б) вибрати та обґрунтувати методику побудови системи контролю роботи технологічного обладнання на виробництві в автоматизованому режимі. Розробити функціональну та структурну схеми системи;

в) розробити програмне забезпечення системи, що дозволить реалізувати поставлену технічним завданням задачу. Побудувати блок-схеми алгоритмів програми та підпрограми;

г) організувати інтерфейс користувача з метою формування та виводу на екран ЕОМ повідомлень про некоректні дії користувача та нестандартні ситуації в роботі технологічного обладнання;

д) розробити рекомендації по організаційних та методичних заходах, які забезпечать впровадження системи в промислову експлуатацію та її подальшу успішну експлуатацію;

е) провести розрахунки по визначенню економічної ефективності розробленої системи;

ж) розробити заходи по охороні праці при впровадженні та експлуатації системи, а також розробити заходи з цивільного захисту;

з) сформувати висновки про виконаний обсяг робіт та одержані результати.

3 ОПИС І ОБҐРУНТУВАННЯ ПРОЕКТНИХ РІШЕНЬ

3.1 Опис функціонування системи

Сучасні технології DRM

Звук, музичні добутки

Музика в Інтернеті

Багато інтернет-магазини США, що продають музику онлайн, використовують DRM. Один з найбільших – Apple iTunes Store – використовував систему DRM FairPlay аж до 2009 року. Система використовує звичайні аудіофайли формату MP4. Кожний файл містить звуковий потік у форматі AAC, зашифрований за допомогою AES з використанням основного ключа(англ. master key), а також сам основний ключ, зашифрований за допомогою ключа користувача (англ. user key). Ключі користувача генеруються випадково для кожного сеансу, їхні копії зберігаються на серверах Apple і в захищеному репозиторії iTunes (клієнтської програми, використовуваної для доступу до iTunes Store). Той самий аккаунт iTunes Store можна використовувати не більше ніж на п'яти комп’ютерах. iTunes дозволяє копіювати аудіофайл на необмежену кількість плеєрів iPod (при цьому ключі користувача також копіюються у внутрішній репозиторій плеєра), однак на одному iPod можна використовувати музику, отриману не більш ніж з п'яти різних аккаунтів [13]. Apple не видавала ліцензії на власний DRM стороннім компаніям, у результаті чого тільки пристрою від Apple, а також їх медіа-програвач QuickTime могли відтворювати музику з iTunes. iTunes також дозволяє записувати аудіофайли на компакт-диски. Той самий плей-аркуш можна записати не більше семи разів, однак кожний окремий файл можна записувати необмежене число раз [14]. Отримані аудіо-CD не містять DRM, тому нескладно одержати аудіофайли без захисту, зробивши рип компакт-диску, однак при цьому якість звуку може зменшитися при

перекодуванні. Програма Requiem дозволяє витягати ключі користувачів зі сховища iTunes, однак Apple регулярно випускає відновлення, міняючи способи зберігання ключів.

Однак 6 лютого 2007 р. глава Apple Стив Джобс опублікував відкритий лист «Думки про музику» (англ. Thoughts on Music), у якому призвав звукозаписні компанії продавати музику без DRM [15]. З початку 2009 року музика в iTunes Store за згодою з більшістю видавців поступово стала повністю доступна без DRM.

Крім стандартних підходів DRM, деякі магазини пропонують DRM-схему підписки. Наприклад, сервіс Sony Music Unlimited або онлайн музичний магазин Napster. Користувачі можуть завантажувати і прослуховувати необмежену кількість музики доти, поки діє підписка. Однак із закінченням підписки всі файли перестають відтворюватися.

У зв'язку з тим, що схеми DRM у різних виробників відрізняються між собою, іноді стає неможливим програвати музику від різних виробників на одному пристрої (пристрій може просто не підтримуватися DRM-схемою). Рішенням подібних проблем займаються, наприклад, в Англії. Так, в 2006 році Ендрю Гауерс склав список пропозицій по поліпшенню політики захисту авторських прав (англ. Gowers Review of Intellectual Property), що містить 54 пункту. Цей список перебуває у відкритому доступі, і ознайомитися з ним може будь-який бажаючий. Серед всіх інших виправлень пункти з 8 по 12 містять пропозиції по створенню деяких виключень для сумлінного використання авторських прав, наприклад, бібліотеками (розглядається можливість переходити від однієї схеми DRM до іншої). Згодом планувалося ввести подібні виключення й для звичайних користувачів. Взагалі проблема з різними DRM у програвачах стояла досить гостро, наприклад, Apple відмовилися від DRM-захисту в музиці повністю, завдяки чому музика з iTunes програється спокійно на будь-якому пристрої, що підтримує формат AAC. Деякі магазини, наприклад, німецький Musicload, також оголосили про відмову від DRM, тому що з'ясувалося, що 3 з 4

дзвінків у їхню службу підтримки надходило від незадоволених DRM-користувачів. [17]

Відеозображення, фільми, телебачення

Запобігання перехоплення відео- і аудіопотока

Інтерфейси DVI (необов'язково) і HDMI підтримують технологію HDCP (High-bandwidth Digital Content Protection, укр. захист широкополосного цифрового вмісту), що використовує шифрування при передачі сигналу між відеопрогравачем і монітором/телевізором для запобігання перехоплення відеопотоку, а також дозволяє здійснювати вивід тільки на сертифіковані пристрої. Однак виявилося, що ця технологія має низку криптостійкістю й може бути зламана [1].

Компанія Microsoft включила у свою операційну систему Windows Vista технологію Protected Media Path (укр. захищений канал даних), що дозволяє шифрувати інформацію, передану відеокарті або монітору, а також забороняти відтворення, якщо запущені програми без цифрового підпису. [1]

Телевізійні програми

Для захисту телепрограм, переданих по телебаченню високої чіткості, передбачається наявність прапора передачі (англ. Broadcast flag), що дозволяє визначити, чи дозволений запис. Ця концепція була розроблена компанією Fox Broadcasting в 2001 році й була підтримана MPAA і Федеральним Агентством по зв'язку (ФАС) США. Однак у травні 2005 року Апеляційний Суд США ухвалив, що ФАС не має достатню владу для накладення подібних обмежень на телеіндустрію в США.

Куди більшого успіху ця система домоглася, коли була прийнята Проектом Цифрового Відео Віщання – консорціумом, що включає більше 250 вещателів, виробників, операторів мережі, розроблювачів програмного забезпечення й керуючих органів більше 35 країн. Цей консорціум намагався розробити нові цифрові стандарти для DRM у телемовленні. Одним з найбільш перспективних стандартів є варіант із поліпшеним прапором передачі,

розроблений для європейського телебачення DVB-CPCM (DVB Content Protection and Copy Management, укр. захист умісту й керування копіюванням). Цей стандарт був наданий на розгляд європейським урядам в 2007 році. Всі нормативні частини на даний момент уже схвалені для публікації Керівною Радою DVB і будуть опублікований ETSI як офіційний європейський стандарт ETSI TS 102 825-X (X – номер підрозділу). На сьогоднішній день ще ніхто не взяв на себе забезпечення Сумісності й Надійності (англ. Compliance and Robustness) для даного стандарту (однак розробки в даному напрямку ведуться багатьма компаніями), що не дозволяє сьогодні впровадити цю систему повсюдно. [1]

У США постачальниками кабельного телебачення використовується стандарт CableCard, що обмежує доступ користувача тільки тими послугами, на які він підписаний.

Текст, документи, електронні книги

Керування цифровими правами на підприємстві – це застосування технологій DRM для керування доступом до корпоративних документів (файли Microsoft Word, PDF, AutoCAD, електронні листи, сторінки внутрішньої мережі інтранет). Ці технології, більше відомі як Керування Інформаційними Правами (англ. Information Rights Management), в основному використовуються для запобігання несанкціонованого використання документів, що є інтелектуальною власністю підприємства (наприклад, з метою промислового шпигунства або випадкового витоку інформації). Звичайно ця система убудована в програмне забезпечення системи керування вмістом, однак деякі корпорації (наприклад, Samsung Electronics) розробляють свої власні системи DRM.

Електронні книги, призначені для читання на ПК, мобільних пристроях або спеціальних «читалок», звичайно використовують DRM з метою обмежити копіювання, печатка або викладання книг у загальний доступ. Звичайно такі книги обмежені кількістю пристроїв, на яких їх можна прочитати, а деякі видавці взагалі забороняють будь-яке копіювання або печатку. Деякі компанії й оглядачі вважають, що наявність DRM створює безліч проблем для видання книг. [3]

Існує чотири основних DRM-схеми для електронних книг, по однієї від Amazon, Adobe, Apple і Martin Trust Management Organization (MTMO):

– DRM від Amazon є адаптацією споконвічного кодування Mobipocket, і використовується в електронних книгах від Amazon (підтримуються, наприклад, читалкой Amazon Kindle) форматів Mobipocket, KF8 і Topaz.

– DRM-схема Adept від корпорації Adobe застосовується до ePub і PDF, причому читати книги можуть різні читалки від сторонніх розроблювачів, а не тільки програмне забезпечення від Adobe. Формат Adobe PDF підтримує різні методи захисту вмісту: **Повне** криптостійке **шифрування документа**, що вимагає уведення пароля для будь-яких операцій з документом, включаючи відкриття й перегляд; **захист документа**, що визначає, чи можливо копіювання, добування тексту, печатка або зміна документа. Хоча стандарт ISO вимагає, щоб всі програми перегляду PDF додержувалися встановлених обмежень, наприклад, Okular має опцію, що дозволяє ігнорувати обмеження в переглядаються файлах, що, [1]; **Adobe DRM** – технологія захисту, використовувана в Adobe Reader версії 6.0 і вище. Використовується в різних книжкових інтернет-магазинах, підтримує прив'язку можливості перегляду до комп'ютера користувача або іншому пристрою (наприклад, КПК або електронній книзі), дозволяє обмежена кількість разів копіювати документ із одного пристрою на інше (авторизоване в Adobe Content Server), дозволяє заборонити добування тексту й печатка документа, а також обмежити строк, протягом якого можливий доступ до документа [5].

– DRM схема FairPlay від Apple Inc. Вона застосовується до формату ePub, причому прочитати такі файли можуть тільки пристрою Apple за допомогою додатка iBook.

– DRM схема Marlin була створена й підтримується у відкритій галузевій групі Marlin Developer Community (MDC), заснованої компаніями Intertrust, Panasonicm Philips, Samsung і Sony. Ця схема ліцензована MTMO.

Комп'ютерні ігри

DRM у комп'ютерних іграх використовується для різних цілей, але в цілому всі схеми спрямовані на захист від копіювання й поширення піратських копій ігор. Найчастіше при запуску таких ігор необхідно вставити диск із грою в оптичний привод, при цьому перевіряються низькорівневі особливості ліцензійних CD і DVD-дисків, які неможливо відтворити при копіюванні в домашніх умовах. Також подібні системи DRM часто встановлюють у систему драйвер для захисту від емуляторів дисководів (таких как DAEMON Tools і Alcohol 120%), а іноді вимагають реєстрації через Інтернет.

Ігрові приставки, такі, як Xbox 360, Xbox One, PlayStation 3 і Playstation 4, також містять систему перевірки диска на ліцензійність.

Активація для обмеження кількості установок

Приблизно із середини 2008 року випуск Mass Effect запустив цілу хвилю продуктів, що використовують DRM-схему SecuROM, що вимагає онлайн-автентифікації на серверах видавця. У цьому ж році використання подібного захисту в грі Spore від Electronic Arts привело до того, що більшість користувачів віддало перевагу використанню піратської версії гри. Однак незалежні дослідники з TweakGuides прийшли до виводу, що подібне використання DRM не впливає на кількість піратських копій гри, відзначивши, що інші ігри (начебто Call of Duty 4: Modern Warfare, Assassin's Creed, Crysis), що використовують схему SafeDisc, що не прибігає до онлайн-автентифікації, також поширювалися в порівнянні з Spore кількостях серед піратів. До того ж гри, що використовують онлайн-автентифікацію так само, як і Spore, начебто BioShock, Crysis і той же Mass Effect, у списках самих завантажуємих ігор на різних торрент-трекерах не значаться. [7]

Постійна онлайн-автентифікація

Багато видавців, серед яких, наприклад, Electronic Arts, Ubisoft, Valve і Atari, використовували онлайн DRM-схеми аж до початку 2009 року. Наприкінці 2008 року компанія Ubisoft провела експеримент, випустивши серію ігор Prince of

Persia без DRM-захисту на веб-сайті GOG.com, з метою перевірити, «наскільки люди праві» у відношенні того, що DRM тільки збільшує піратство й провокує людей використовувати не ліцензійні копії. Хоч сама компанія так і не оголосила результати експерименту, незалежні експерти з Tweakguides помітили, що всього лише із двох торрентів на Mininova гру скачало більше 23 тисяч людей протягом 24 годин після релізу. [8]

Ubisoft офіційно оголосили про повернення онлайн-автентифікації 9 лютого 2010 року. Вони представили свою нову онлайн ігрову платформу Uplay, що почали використовувати в таких іграх, як Silent Hunter 5, The Settlers 7 і Assassin's Creed II. Silent Hunter 5 зламали протягом 24 годин з моменту релізу. Однак, користувачі піратської версії могли грати тільки в початкові рівні гри. Система Uplay працює таким чином, що на користувальницький ПК гра встановлюється не повністю, а дозавантажує уміст із ігрових серверів Ubisoft у міру проходження гри. Ледве більш, ніж через місяць після релізу на ПК, у перший тиждень квітня, було випущено ПЗ, за допомогою якого можна було обійти DRM-захист в Assassin's Creed II. ПЗ виявляло собою емулятор сервера Ubisoft для гри. Трохи пізніше, у цьому ж місяці, була випущена версія, що забирала необхідність у з'єднанні із серверами повністю. [9]

На початку березня 2010 року сервера Ubisoft піддалися масштабної DoS-атаці, що привело до закриття доступу до ігор для ~5 % гравців. Як компенсація за принесені незручності, компанія надала постраждалим користувачам по безкоштовній завантажуємій грі. З березня 2010 року сервера Ubisoft більше не падали.

Прикладу Ubisoft пішли й інші розроблювачі, такі, як Blizzard Entertainment. Вони також перейшли на варіант захисту, коли більша частина ігрової логіки перебуває «на стороні», або обробляється серверами творця гри. Blizzard використовує подібний підхід у своїй грі Diablo III. Electronic Arts використовували такий підхід у своєму перезавантаженні серіалу SimCity. Треба сказати, що подібний підхід негативно вплинув на обидві компанії, тому що вони

просто не змогли впоратися з кількістю гравців на серверах, що привело до численних скарг і зростаючого невдоволення користувачів. Electronic Arts намагається забрати необхідність постійного підключення до серверів, але поки це не представляється можливим, тому що вся гра була створена з обліком цього. [9]

Втручання в ПЗ

Деякі студії як захист використовують не зовсім стандартні підходи. Bohemia Interactive використовує DRM-схему (починаючи з 2001 року, з виходом Operation Flashpoint: Cold War Crisis), що при запуску нелегальної копії гри просто заважає грати. Гра починає створювати ситуації, у яких у гравців знижується точність зброї, або, наприклад, самі гравці перетворюються в птахів. Компанія Croteam у своїй грі Serious Sam 3: BFE використовувала схожий підхід, нацьковуючи на гравців, що використовують нелегальні копії гри, монстра, якого неможливо було вбити. [1]

Критика DRM

Деякі критики DRM уважають, що DRM використовуються не щоб захистити виключні права й обмежити масове незаконне копіювання («піратство»), а щоб змусити законослухняних клієнтів платити більше за звичні дії начебто «сумлінного використання» або «вільного використання» добутків. Наприклад, звичайну електронну книгу можна читати й на настільному комп'ютері, і на мобільному пристрої, слухати за допомогою синтезатора мови, копіювати в буфер обміну цитати (нікого при цьому не повідомляючи), а DRM дозволяє змусити користувача купувати окремі версії для кожного способу використання.

Неавтономні DRM

Контролююча особа (а іноді й інші обличчя) може збирати інформацію про поводження покупця: його режимі дня, способах використання добутку й т.п. (порівн. phoning homeruen). [5]

Потенційна шкода для навколишнього середовища

В апаратному забезпеченні DRM іноді перешкоджає використанню пристрою для цілей або з матеріалами, не схваленими виробником пристрою. Це прискорює його перетворення в електронні відходи, унеможливлюючи або незаконним «продовження життя» пристрою (ремонт; модернізацію; використання з метою, відмінної від заявленої виробником). [6] [7] Порівн.: пропрієтарне abandonware, добутку з недоступними правовласниками, плановане старіння. Приклади:

– Регіональне кодування оптичних дисків (стосується як програвачів, так і самих дисків)

– Видалення функції «OtherOS» із прошивання ігрової консолі Sony PlayStation 3.

– Тивоизация, певні способи впровадження функції «Secure Boot» UEFI і т.п.

Користувачі також можуть бути змушені здобувати нові пристрої заради сумісності з DRM, у тому числі через необхідність використання для DRM програмного забезпечення з більше високими системними вимогами. [8]

Продукція без DRM

Багато видавців, реагуючи на численну критику DRM, випускають свою продукцію зі спеціальною позначкою « DRM-Free», що на українську мову можна перевести як «Вільно від DRM», або «Без DRM». Безліч великих компаній підтримують цю політику:

– Apple Inc. продають музику без DRM через свій iTunes Store із квітня 2007 року, відзначаючи всю музику знаком « DRM-Free» із січня 2009 року. Музика все-таки містить цифрові водяні знаки для ідентифікації покупця. Інша продукція, продавана через iTunes (наприклад, електронні книги, фільми, додатки й т.д.) продовжує підтримувати DRM. [50]

– Tor Books, великий видавець книг у жанрах наукова фантастика й фентезі, продає книги без DRM з липня 2012 року. Через рік вони заявили, що продовжать дотримуватися політики « DRM-Free», тому що відсутність DRM-

захисту ніяк не шкодить їхньому бізнесу. Більше дрібні видавці почали позбуватися від DRM ще раніше. [51]

– GOG.com, цифровий постачальник відеоігор для ПК, також дотримується строгої політики у відношенні DRM. Весь їхній каталог ігор продається без DRM, у той час як більшість цифрових магазинів продовжують використовувати DRM. [52]

– DotEmu – ще один цифровий магазин класичних відеоігор, що пропонує у своєму каталозі ще й власні порти класичних ігор на мобільні пристрої. Всі « DRM-Free». [53]

– The Humble Indie Bundle – серія продуктів, створена Humble Bundle Inc., містить набори ігор, музики й електронних книг без DRM. Крім того, компанія дотримується цікавої цінової політики – користувач платить стільки, скільки вважає потрібним.

– Краудфандинг – новий плин в області створення й просування проектів. Суть даного плину полягає в тім, що гроші на створення проекту збираються в користувачів, без безпосередніх видавців. Наприклад, на сайті kickstarter.com зібрані засоби можуть досягати декількох мільйонів. [54]

W3C все-таки схвалив стандарт DRM для HTML5

6 липня 2017 року консорціум World Wide Web Consortium (W3C) привселюдно оголосив про намір прийнять Encrypted Media Extensions (EME) – стандарт DRM, що надає API для контролю відтворення контенту в браузері через елементи HTML5 <video> і <audio>. Тобто безпосередньо в браузері з'являться убудовані засоби DRM, так що правовласники зможуть забороняти/обмежувати відтворення фільмів і музики на комп'ютерах користувачів. Таким чином, навіть у вільних браузерах open source буде працювати зашифрований пропрієтарний код – «чорний ящик», що загрожує безпеки й приватності користувачів, а також позбавляє їхнього контролю над власними комп'ютерами.

Фонд електронних рубежів, що 9 липня разом з Фондом вільного ПЗ, організаціями Creative Commons і Document Foundation проводить Міжнародний

день проти DRM виступив категорично проти такого рішення: воно прийнято «без яких-небудь гарантій доступності, досліджень безпеки або наявності конкуренції, незважаючи на безпрецедентні внутрішні суперечки між співробітниками й членами W3C по цьому питанню».

DRM у браузері реалізований так, що JavaScript API забезпечує тільки інтерфейс для взаємодії, а модуль дешифрування контента Content Decryption Module (CDM) контролює одержання ліцензії й обмін криптографічними ключами й реалізує пропрієтарні методи керування ліцензіями. Схематично це зображено на ілюстрації.

Уперше представлена в 2012 році, технологія EME дотепер викликає запеклі суперечки в співтоваристві. Великому табору супротивників стандартизації цієї технології протистоїть потужне корпоративне лобі правовласників, які ведуть у Мережі комерційну діяльність, продаючи цифровий контент і програмне забезпечення.

Проти прийняття EME виступають не тільки вищезгадані Фонд вільного ПЗ й Фонд електронних рубежів. Під відкритим листом підписалися сотні фахівців з безпеки, свої заперечення висловили академічні дослідники, навіть правозахисна група Just Net Coalition і директор по комунікаціях і інформації ЮНЕСКО.

Пропрієтарне шифрування EME просувають Netflix, Google, Microsoft і Apple, а також правовласники з Motion Picture Association of America (MPAA). Всі вони є членами W3C, тобто перераховують фінансові внески на зміст Консорціуму.

Зараз в опонентів EME залишилася остання можливість протистояти прийняттю стандарту – подати апеляцію в Консультативний комітет W3C (Advisory Committee of the World Wide Web Consortium). Цей комітет очолює Тім Бернерс-Чи, що вже схвалив EME. Проте, якщо 5% з 475 членів комітету протягом двох тижнів підпишуться під апеляцією, то буде ініційоване загальне голосування. Воно й установить остаточне рішення, чи включати EME у стандарт.

Організації, що входять в W3E, повинні взяти на себе відповідальність за цифрові права користувачів інтернету й опротестувати згубне рішення Тіма Бернерса-Чи. Головними пріоритетами при прийнятті стандартів повинні бути воля користувачів, приватність, безпека, сумісність і доступність, а не допомога Голливуду й стримінговим компаніям, як зробити більше ефективним їх DRM, спрямований проти користувачів.

Якщо стандарт EME буде прийнятий, то можна чекати поступове збільшення пропрієтарного зашифрованого відео в Мережі, оскільки стримінговим сервісам буде простіше впровадити захист DRM. На думку опонентів, це викличе додаткові проблеми для користувачів і загрожує волі інформації в інтернеті: таке відео важко переводити, коментувати, архівувати, здійснювати інші законні дії. Вони називають EME справжніми «цифровими наручниками» для видеоконтента в Мережі. До того ж, пропрієтарний DRM у браузері – це справжній руткіт для стеження за користувачами з боку правовласників, як показала історія Adobe.

Відповідно до американського закону DMCA Section 1201 (у багатьох інших країнах теж є такі закони) обхід захисту DRM навіть із метою пошуку небезпечних уразливостей у безпеці, закладок і руткітів, карає адміністративним і карним покаранням. Обійти захист DRM нескладно: наприклад, захист Widevine у контенті Google розкрили ще в 2010 році, але це не має значення. Закон дає правовласникові винятково прибуткове право переслідувати користувачів і конкурентів, які намагаються обійти захист або використовувати контент способом, не передбаченим правовласником.

Як альтернатива Фонд електронних рубежів пропонує угода, у рамках якого правовласники-члени W3C обіцяють переслідувати порушників DRM тільки за порушення копирайту, але не за обхід захисту DRM, якщо це пов'язане із законними діями (дослідження безпеки, спрощення доступу до контенту для інвалідів і т.д.).

3.2 Розробка структурної схеми

Цикл шифрування

Щоб розпочати цикл «безпеки», зв’язок між програмним забезпеченням для кодування, що запитує, та сервером ліцензій шифрується.

Кожен сегмент шифрується відповідно до специфікації MPEG Common Encryption (CENC) для ISO-BMFF.

Що таке ISO-BMFF?

ISO-BMFF – це стандартизований формат файлів, який служить контейнером для аудіо- та відеоконтенту. Відомою реалізацією ISO-BMFF (і часто використовується як його синонім) є формат файлів MP4 або фрагментований MP4 (fMP4). У робочому процесі DRM мультимедійний контент шифрується, а контейнер ISO-BMFF покращується за допомогою метаданих та алгоритмів шифрування, специфічних для DRM.

Системи DRM використовують ISO-BMFF для зберігання та транспортування зашифрованих медіаданих і забезпечують пов'язання з ліцензією DRM. Коли користувачі намагаються отримати доступ до захищених медіаданих, система DRM перевіряє, чи має користувач на це право, залежно від пов'язаної ліцензії.

Коротше кажучи, це забезпечує безпечне зберігання, доставку та контроль цифрових медіафайлів у рамках DRM.

Сегменти можуть бути повністю зашифровані або частково зашифровані, коли шифруються лише деякі кадри або навіть лише частини кадрів.

Стандарт MPEG-CENC визначає, як шифрується сегмент, і відображає, який ключ дешифрування потрібно використовувати для якого сегмента (або його частин), пов'язуючи з ним ідентифікатор ключа. MPEG-CENC використовується для потоків DASH та HLS, якщо сегменти мають формат контейнера fMP4.

Стандартне шифрування контенту виконується за допомогою алгоритму Advanced Encryption Standard (AES) з використанням 128-бітних ключів. Залежно

від використовуваної системи DRM, вона використовується або в режимі лічильника (CTR), або в режимі ланцюжка блоків шифрування (CBC).

Ці два режими відрізняються тим, як шифрується корисне навантаження.

Важливо зазначити, що шифруються лише необроблені аудіо- та відеодані в сегменті, а метадані, додані в контейнер, – ні.

Існує три основні постачальники DRM: Google Widevine, Apple FairPlay та Microsoft Playready.

Їхнє застосування може значно відрізнятися залежно від багатьох унікальних факторів – необхідність вибору постачальника, який відповідає потребам дистриб'ютора контенту щодо доставки та відтворення (залежно від того, які пристрої підтримуються), може значно ускладнити процес впровадження DRM.

Для підвищення безпеки та зменшення ризику зворотного проектування систем DRM зазвичай немає чітких повідомлень журналу.

Фактично, частини процесу розглядаються як чорна скринька, і в результаті налагодження може бути ще складнішим на пристроях (наприклад, SmartTV або телеприставках) зі старими версіями програмного забезпечення DRM.

У браузері або операційній системі контент потім буде розшифровано модулем розшифрування контенту (CDM), який розшифровує кожен зашифрований аудіо- та відеосегмент.

Цикл дешифрування

Коли веб-плеєр ідентифікує контент, захищений DRM, він викликає процеси та інтерфейси, визначені розширеннями зашифрованого медіа (EME), які використовуються в браузерах для ініціювання процесу запиту ліцензії.

EME використовується для взаємодії з модулем розшифрування контенту (CDM), який реалізований у браузері та може покладатися або не покладатися на функції операційної системи, такі як HDCP.

Під час відтворення контенту, захищеного DRM, запити на ліцензію генеруються CDM та передаються програвачу через EME.

Всю роботу з розшифрування виконує CDM. Найголовніше, що розшифрований контент залишається в CDM – він не є і не повинен бути доступним для програмного забезпечення для відтворення, оскільки інакше можна було б створити розшифровані копії контенту.

Для відтворення захищеного контенту, після виявлення того, що контент захищений, програвач або програмне забезпечення для відтворення надсилає запит на ліцензію на сервер ліцензування.

Якщо ліцензія кешується локально, цей запит можна пропустити, і замість нього можна використовувати кешовану ліцензію.

Запит на ліцензію, що надсилається програвачем програмного забезпечення для відтворення, завжди містить метадані, які однозначно ідентифікують відтворюваний контент, а формат цих метаданих залежить від використовуваного DRM-рішення.

Ці метадані DRM можуть міститися або в маніфесті (наприклад, MPEG-DASH або вбудовані в HLS), або в конфігурації програвача, або в окремих сегментах.

Хоча це не є обов'язковою вимогою, запит зазвичай містить додаткові дані із пристрою, що запитує, такі як ідентифікатор, який можна використовувати для його унікальної ідентифікації.

Якщо надана вся обов'язкова інформація, сервер може надати ліцензію програвачу або програмному забезпеченню для відтворення з ключами розшифрування, необхідними для безпечного відтворення запитуваного контенту на клієнті.

Повернена ліцензійна угода може містити інформацію про необхідний рівень безпеки розшифрування контенту, наприклад: розшифрування контенту за допомогою програмного забезпечення значно менш безпечне, ніж розшифрування за допомогою апаратного забезпечення.

З точки зору гравця, отримання ліцензії за допомогою EME починається зі створення клієнтом відтворення так званого ключового сеансу. Використовуючи

цей ключовий сеанс та метадані DRM, взяті з сегментів, маніфесту або інших джерел, гравець запускає процес запиту ліцензії за допомогою EME.

Потім CDM генерує підписане ключове повідомлення, яке програвач або програмне забезпечення для повернення коштів надсилає на сервер ліцензій.

Сервер ліцензій повертає запитувану ліцензію, а також приймає рішення про те, чи надано клієнту права на відтворення запитуваного контенту; якщо ні, відтворення зупиняється та відображається помилка.

Або ж сервер ліцензій може визначити, що, наприклад, програвач може відтворювати лише SD-версії контенту.

Якщо запит ліцензії був успішним, клієнт оновлює ключовий сеанс повернутою ліцензією.

Потім розшифрування контенту повністю обробляється CDM.

За деяких обставин ліцензія кешується на певний час і може бути використана для відтворення захищеного контенту офлайн (наприклад, Netflix).

Робочий процес дуже схожий для невеб-платформ, таких як нативні додатки для Android, iOS або tvOS. Кожна платформа має власний набір API, подібних до EME on Web, для взаємодії з базовою інтегрованою CDM.

Ліцензія та розшифровані дані не повинні бути доступні клієнтам, окрім користувача ліцензованого контенту.

Таким чином, закриті ключі та розшифровані дані зберігаються в безпечному середовищі в браузері, операційній системі або навіть на обладнанні (якщо підтримується), як-от у надійних середовищах виконання.

Використання різних форматів контейнерів, таких як fMP4 та MPEG-2 TS, ускладнювало розповсюдження однакового контенту на всіх платформах.

Однак швидке впровадження **CMAF** та стандартизація CENC серед виробників обладнання та розробників програмного забезпечення зменшують складність впровадження для галузі.

Хоча CMAF та CENC все ще дозволяють використання AES CTR та AES CBC, постачальники DRM поступово переходять до використання AES CBC.

Запобігання копіюванню контенту, захищеного авторським правом, у інших правовласників

Припустімо, ви розміщуєте онлайн-платформу відео на вимогу, яку можна використовувати для перегляду всіляких голлівудських фільмів. Власник прав на контент, який ви розповсюджуєте, не хоче, щоб ваші користувачі могли просто створювати копії цього контенту.

Таким чином, постачальник платформи може бути зобов'язаний за контрактом використовувати певну форму захисту контенту для дотримання прав власника прав на контент.

Це часто трапляється з мовниками, які не лише розміщують власний контент, але й, наприклад, транслюють прямі трансляції телепередач чи інші фільми чи серіали. Системи DRM можуть використовуватися для захисту контенту від незаконного копіювання користувачами цього сервісу.

Вибір найкращих DRM-сервісів

Існує кілька варіантів контролю доступу до вашого цифрового контенту, обмежуючи його лише авторизованими користувачами. Постачальники DRM пропонують рішення та послуги творцям контенту, видавцям та дистриб'юторам.

Вони спеціалізуються на розробці та впровадженні технологій, інструментів і систем, що забезпечують захист, розповсюдження та управління вашим цифровим контентом. Вони також забезпечують дотримання умов ліцензування та законів про авторське право.

Такі рішення, як шифрування, контроль доступу, управління ліцензіями, захист контенту та моніторинг, можуть бути надані хорошим партнером з DRM.

Набір послуг розробляємої системи інтелектуального керування цифровими правами на основі DRM виглядає так:

– Інтеграція системи DRM: Постачальники DRM інтегрують свої технології в існуючі платформи розповсюдження контенту, веб-сайти або потокові платформи, забезпечуючи безперебійну функціональність DRM та захист цифрового контенту.

– Шифрування контенту: Рішення для шифрування захищають цифровий контент від неавторизованих користувачів та онлайн-піратства. Гарний партнер застосує надійні алгоритми шифрування для захисту вашого контенту під час зберігання, передачі та відтворення.

– Керування ліцензіями: Системи керування ліцензіями займаються створенням, видачею та керуванням ліцензіями DRM. Ці системи гарантують, що користувачі мають необхідні дозволи та права для доступу до вашого захищеного контенту.

– Забезпечення дотримання прав: ці механізми забезпечують дотримання прав використання, визначених ліцензіями DRM. Це може включати обмеження кількості пристроїв, на яких можна отримати доступ до вашого контенту, забезпечення обмеженого за часом доступу або контроль можливості копіювання чи обміну контентом.

– Аналітика та моніторинг: Постачальники DRM пропонують інструменти аналітики та моніторингу для відстеження використання контенту, виявлення потенційних порушень та збору інформації про поведінку користувачів.

Зрозуміло, що управління цифровими правами – це складна тема, до якої не існує універсального підходу. Але це невід'ємна частина відеороботу для кожного, хто хоче захистити або монетизувати свій цифровий відеоконтент. Це сфера постійного розвитку, оскільки ті, хто має намір займатися піратством, шукають нові способи обійти захист вашого контенту заради власної вигоди.

У даній роботі система керування цифровими правами на основі DRM реалізується за рахунок використання методів стеганографії.

Структурна схема розробленого програмного забезпечення представлена на рисунку 3.1.

В якості даних, що вбудовуються може використовуватися будь-яка інформація: текст, повідомлення, невелике зображення тощо, які дозволяють підтверджувати та захищати авторські права.

Роль контейнеру буде відігравати будь-яке кольорове цифрове зображення, яке потребує захисту у рамках забезпечення авторських прав, що задовольняє стандартним вимогам до контейнерів для стегоповідомлень.

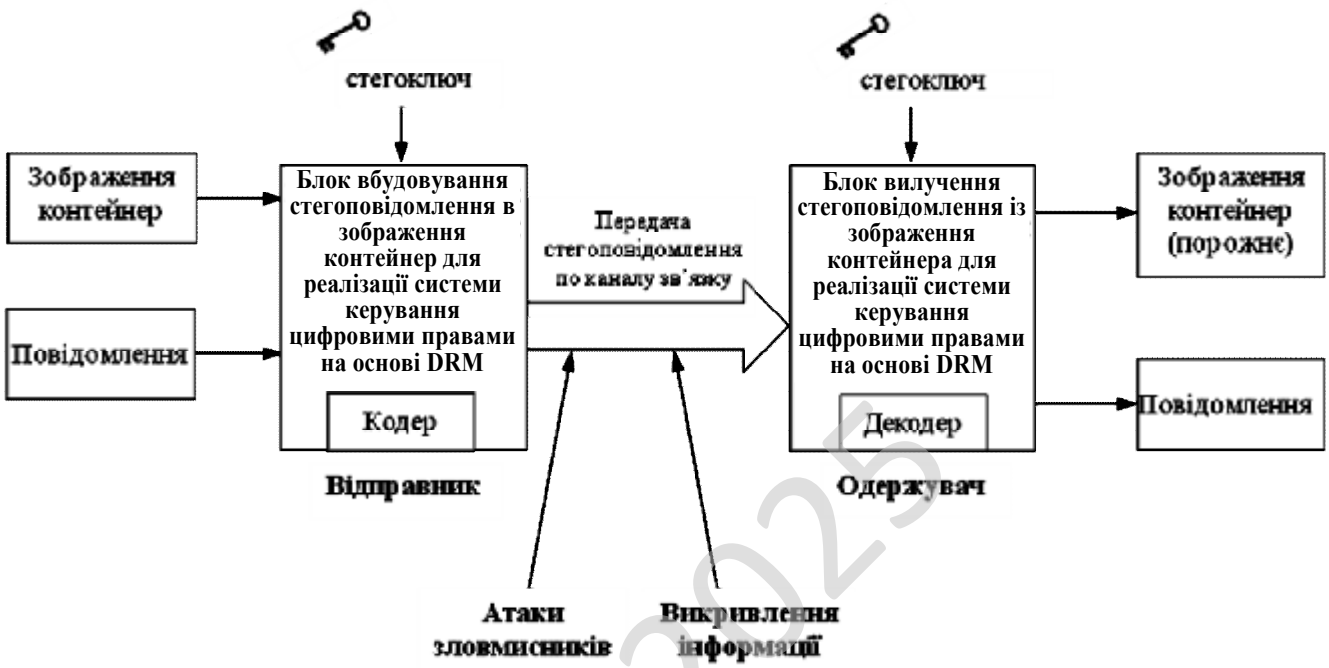

Рисунок 3.1 – Структурна схема системи

Стегоключ – секретний ключ, необхідний для приховування інформації. Залежно від кількості рівнів захисту (наприклад, вбудовування заздалегідь зашифрованого повідомлення) в стегосистемі може бути один або декілька стегоключів.

Вбудовування повідомлення в зображення контейнер відбувається за допомогою стегокодера, який крім приховування інформації здійснює також і перешкодостійке кодування.

Після цього зображення з прихованим повідомленням передається по каналу зв’язку, де може зазнавати атак зловмисників, а також викривлень інформації в наслідок перешкод у каналі зв’язку або застосувань алгоритмів стиснення з втратами.

Вилучення повідомлення із зображення контейнера здійснюється за допомогою стегодетектора. Стегодекодер перевіряє наявність прихованого повідомлення і в разі його існування, вилучає інформацію.

3.3 Розробка функціональної схеми

Більш докладну взаємодію між структурними блоками системи представлено на функціональній схемі роботи системи (рисунок 3.2).

Система включає дві процедури:

– процедуру вбудовування конфіденційної інформації системи керування цифровими правами на основі DRM в зображення контейнер;

– процедуру вилучення конфіденційної інформації системи керування цифровими правами на основі DRM із зображення контейнера.

Рисунок 3.2 – Функціональна схема системи

Робота системи відбувається наступним чином.

Спершу опишемо процедуру вбудовування конфіденційної інформації системи керування цифровими правами на основі DRM в зображення контейнер.

Для цього береться повідомлення, над яким відбувається операція кодування у вибраному зображені, яке є контейнером.

Контейнер – це те зображення, куди буде приховано записано повідомлення.

Принцип кодування наступний:

1. Обчислюється контрольна сума пароля

2. Обчислюється контрольний добуток пароля

3. Всі закодовані дані представляються як масив байтів

4. Від кожного байта даних віднімається байт контрольної суми пароля

5. З результатом попереднього обчислення робиться XOR з байтом контрольного добутку пароля

6. До результату попереднього обчислення додається код відповідного символу з рядка пароля.

7. Як тільки рядок пароля закінчується знову переходимо на його початок.

Для реалізації цього методу відбувається генерація стегоключа, за допомогою якого повідомлення буде зашифроване.

Щоб стегоключ неможливо було взнати зловмиснику, він зашифровується алгоритмом DES, та передається отримувачу повідомлення.

Для більш високої надійності передачі даних, повідомлення кодується за допомогою перешкодостійкого кодека Хеммінга.

Після цього відбувається передача закодованого зображення з прихованим повідомленням, по каналам зв’язку.

На приймальній стороні отримують зображення й реалізують процедуру вилучення конфіденційної інформації системи керування цифровими правами на основі DRM із зображення контейнера.

Це відбувається наступним чином.

Спершу отримане повідомлення перевіряють на наявність помилок, за рахунок застосування кодеку Хеммінга.

Якщо є помилки, то вони виправляються, за рахунок властивостей цього кодеку з виявлення та виправлення помилок.

Після цього відбувається дешифрування ключа. За допомогою отриманого ключа відбувається створення файлу з повідомлення та очищення контейнера.

Після цього на стороні отримувача є інформація, яка була прихована у зображені, та саме зображення, яке може служити контейнером для наступної інформації, яку треба приховано передати.

Розглянувши усі блоки функціональної схеми перейдемо до розгляду діаграми взаємодії процесів, які відбуваються у системі.

3.4 Розробка діаграми процесів

Діаграма процесів розробленої системи зображена на рисунку 3.3. При детальному її розгляді можна побачити як саме проходе взаємодія у розробленій системі. Використовується модель проектування, графічне представлення «потоків» даних в інформаційній системі.

Діаграма взаємодії процесів використовується для візуалізації процесів обробки даних (структурне проектування). Для розробника вважається звичним спочатку креслити діаграму взаємодії процесів даних рівня контексту, завдяки чому буде показано взаємодію системи. Ця діаграма в подальшому підлягає уточненню шляхом деталізації процесів та потоків даних з метою показати систему що розробляється.

Діаграми потоків даних містять чотири типи елементів:

– Процеси які являють собою трансформацію даних в рамках описуваної системи.

– Сховища даних (репозиторії).

– Зовнішні по відношенню до системи сутності.

– Потоки даних між елементами трьох попередніх типів.

Рисунок 3.3 – Діаграма взаємодії процесів

Таким чином, розглянувши опис системи, структурну, функціональну схеми системи, та діаграму взаємодії процесів перейдемо до опису блок-схем основної програми, та підпрограм, які використовуються, для реалізації системи.

4 РЕАЛІЗАЦІЯ ПРОЕКТУ. РОЗРАХУНКИ І ЕКСПЕРИМЕНТАЛЬНІ ДАНІ, ЩО ПІДТВЕРДЖУЮТЬ ПРАВИЛЬНІСТЬ ПРОЕКТНИХ РІШЕНЬ

4.1 Блок-схеми та опис алгоритмів функціонування системи

Під час роботи над магістерською роботою було створено блок-схеми. Перед їх розглядом необхідно провести роз’яснення який саме тип блок-схем використовується.

Блок-схема це представлення задачі для її аналізу або розв'язування за допомогою спеціальних символів (геометричних образів), які позначають такі елементи, як операції, потік, дані тощо. Блок вхідних та вихідних даних прийнято позначати паралелограмом, блок обчислень (обробки) даних – прямокутником, блок прийняття рішень – ромбом, еліпсом – початок та кінець алгоритму.

У інформаційних технологіях функціональна схема складається з функціональних блоків, які являють собою конструктивно відособлені частини (елементи або пристрої) автоматичних систем, які виконують певні функції. Функціональні блоки на схемі позначають прямокутниками, всередині яких надписують їх найменування відповідно до функцій, що виконуються. Зв'язки між функціональними блоками (внутрішні впливи) позначаються лініями зі стрілками, які вказують напрям впливів.

Функціональні схеми можуть виконуватися в укрупненому і розгорненому вигляді. У першому випадку на схемі зображають найважливіші блоки системи і зв'язки між ними.

У другому варіанті схема відображається більш детально, що полегшує її читання та ілюструє принцип роботи.

Основні елементи схем алгоритму це термінатор, процес, рішення, зумовлений процес (підпрограма), дані та з'єднувач.

Термінатор це елемент відображає вхід із зовнішнього середовища або вихід з неї (найчастіше застосування – початок і кінець програми). Всередині фігури записується відповідна дія.

Процес це виконання однієї або кількох операцій, обробка даних будь-якого виду (зміна значення даних, форми подання, розташування). Всередині фігури записують безпосередньо самі операції.

Рішення це показує рішення або функцію перемикального типу з одним входом і двома або більше альтернативними виходами, з яких тільки один може бути обраний після обчислення умов, визначених всередині цього елементу. Вхід в елемент позначається лінією, що входить зазвичай у верхню вершину елементу. Якщо виходів два чи три то зазвичай кожен вихід позначається лінією, що виходить з решти вершин (бічних і нижній). Якщо виходів більше трьох, то їх слід показувати однією лінією, що виходить з вершини (частіше нижній) елемента, яка потім розгалужується. Відповідні результати обчислень можуть записуватися поруч з лініями, що відображають ці шляхи.

Зумовлений процес (підпрограма) це символ відображає виконання процесу, що складається з однієї або кількох операцій, що визначені в іншому місці програми (у підпрограмі, модулі). Всередині символу записується назва процесу і передані в нього дані.

Дані це перетворення у форму, придатну для обробки (введення) або відображення результатів обробки (виведення). Цей символ не визначає носія даних (для вказівки типу носія даних використовуються специфічні символи).

З'єднувач це символ відображає вихід в частину схеми і вхід з іншої частини цієї схеми. Використовується для обриву лінії та продовження її в іншому місці (приклад: поділ блок-схеми, що не поміщається на листі). Відповідні сполучні символи повинні мати одне (при тому унікальне) позначення.

Блок-схеми є першоджерелами стратегії розвитку ПЗ. Тому від точності і детальної блок-схеми залежить результат всієї програми. При виборі початкової точки відліку при побудові схем було враховано, що виходячи з вибору мови

програмування і інших технічних засобів, програма буде об’єктно-орієнтована що вимагає оптимізації програми високого рівня, також те, що при розробці програми слід надати особливу увагу модулю системи керування цифровими правами і основному модулю.

На рисунку 4.1 зображена основна блок-схема програми, на рисунку 4.2 зображено роботу підпрограми. З яких видно що робота основної програми складається з початкових етапів ініціалізації ПЗ, перевірки наявності ресурсів системи, блоку початку основного циклу з чеканням запиту від користувача в якому відбувається виклик підпрограми та останньої стадії – перевірка поточного стану з завершенням роботи розробленого ПЗ. При роботі підпрограми виконується основний функціонал системи з циклічними послідовностями, перевіркою поточного стану та поверненням в основну програму прапорів стану виконання.

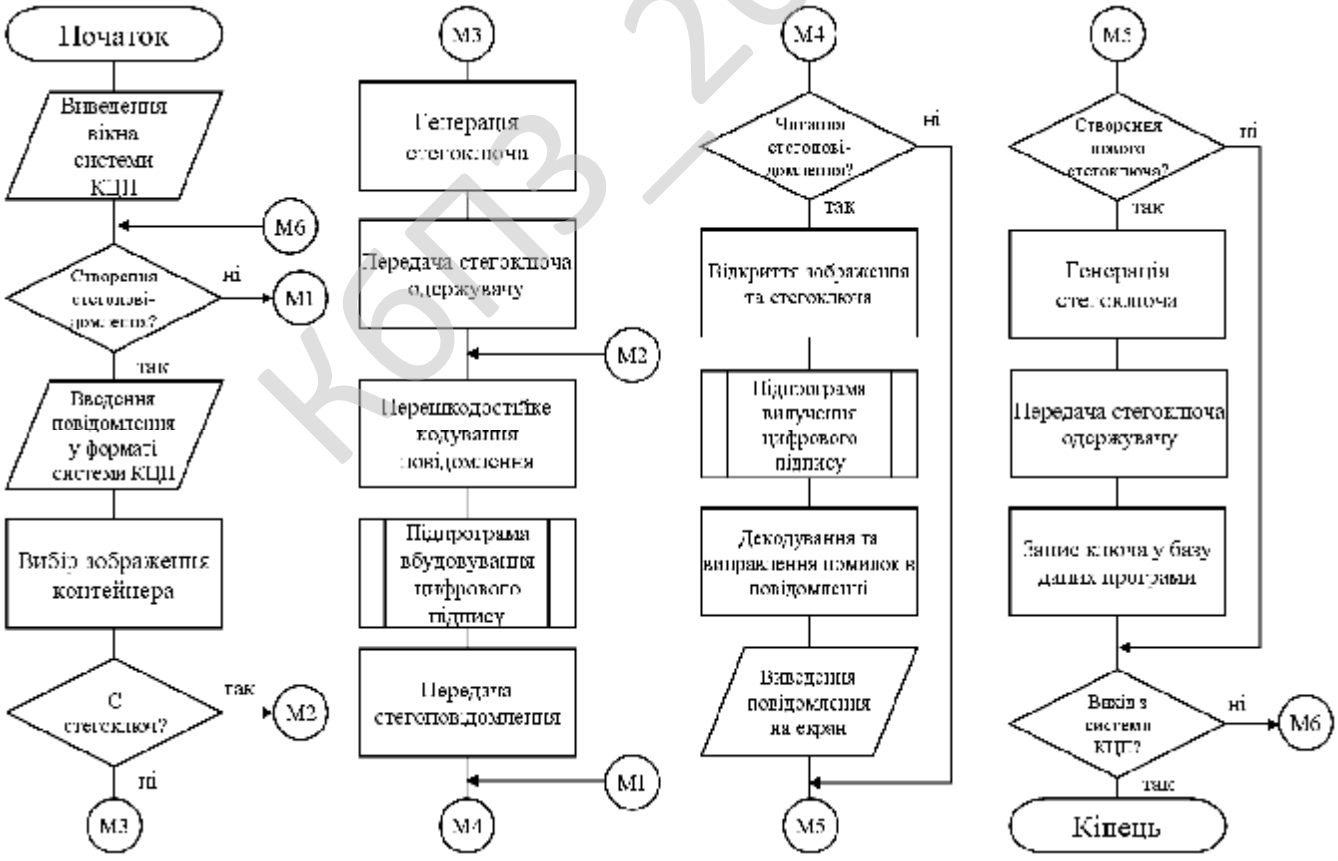

Рисунок 4.1 – Блок-схема основної програми

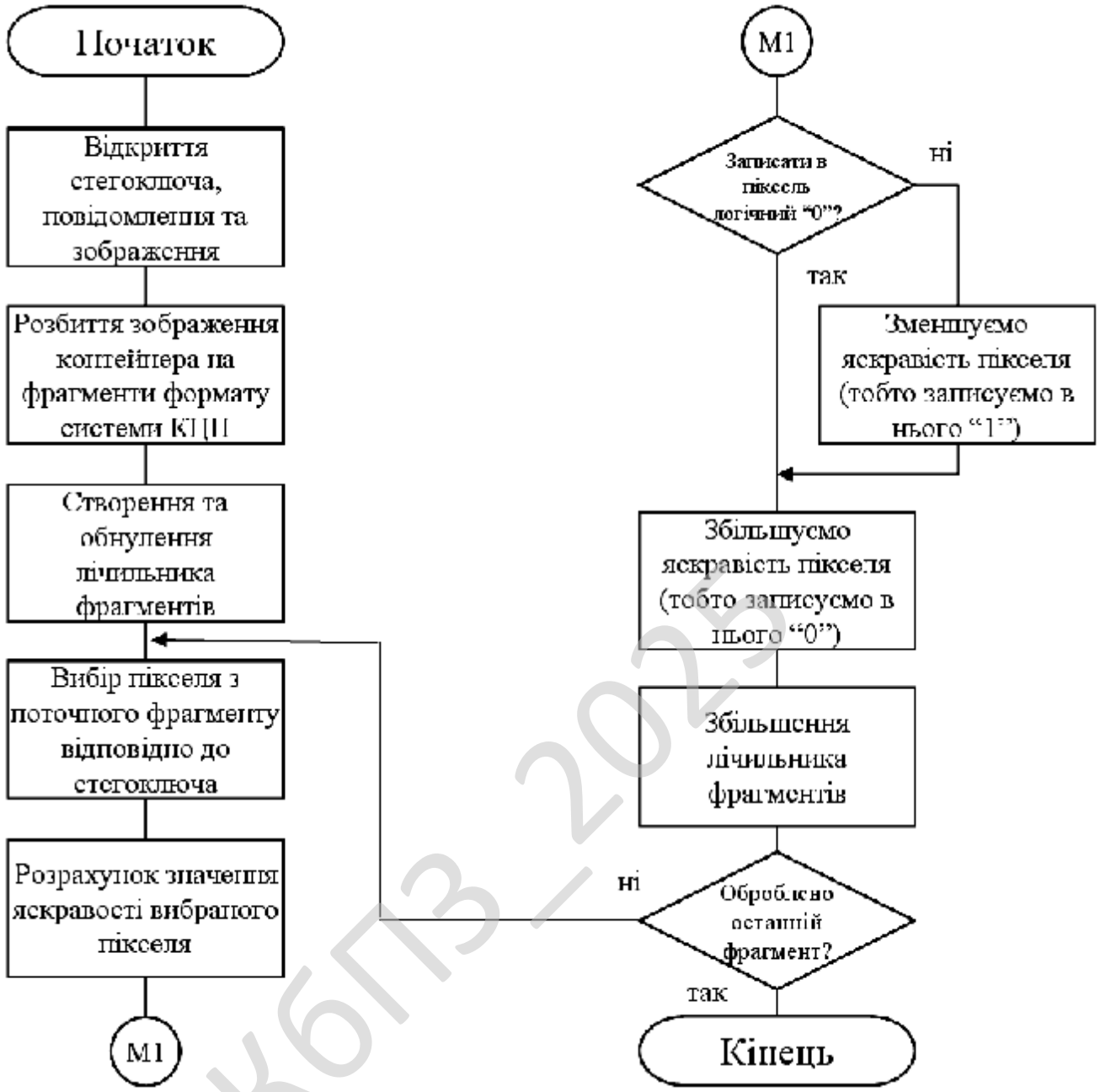

Рисунок 4.2 – Блок-схема роботи підпрограми

Було використано підходи з використанням UML, це уніфікована мова моделювання, використовується у парадигмі об'єктно-орієнтованого програмування. Є невід'ємною частиною уніфікованого процесу розробки програмного забезпечення. UML є мовою широкого профілю, це відкритий стандарт, що використовує графічні позначення для створення абстрактної моделі системи, називаної UML-моделлю. UML був створений для визначення,

візуалізації, проектування й документування в основному програмних систем. UML не є мовою програмування, але в засобах виконання UML-моделей як інтерпретованого коду можлива кодогенерація.

UML може бути застосовано на всіх етапах життєвого циклу аналізу бізнес-систем і розробки прикладних програм. Різні види діаграм які підтримуються UML, і найбагатший набір можливостей представлення певних аспектів системи робить UML універсальним засобом опису як програмних, так і ділових систем.

Діаграми дають можливість представити систему (як ділову, так і програмну) у такому вигляді, щоб її можна було легко перевести в програмний код. Основною причиною використання мови UML є спілкування розробників між собою.

Крім того, UML спеціально створювалася для оптимізації процесу розробки програмних систем, що дозволяє збільшити ефективність їх реалізації у кілька разів і помітно поліпшити якість кінцевого продукту.

UML прекрасно зарекомендувала себе в багатьох успішних програмних проектах. Засоби автоматичної генерації кодів дозволяють перетворювати моделі мовою UML у вихідний код об’єктно-орієнтованих мов програмування, що ще більш прискорює процес розробки. Практично усі CASE-засоби (програми автоматизації процесу аналізу і проектування) мають підтримку UML. Моделі розроблені в UML, дозволяють значно спростити процес кодування і направити зусилля програмістів безпосередньо на реалізацію системи.

Діаграми підвищують супроводжуваність проекту і полегшують розробку документації.

UML необхідний:

– Керівникам проектів, які керують розподілом завдань і контролем за проектом.

– Проектувальникам інформаційних систем які розробляють технічні завдання для програмістів.

– Бізнес-аналітикам, які досліджують реальну систему і здійснюють інжиніринг і реінжиніринг бізнесу компанії.

– Програмістам які реалізовують модулі інформаційної системи.

При модифікації системи об'єктний підхід дозволяє легко включати в систему нові об'єкти і виключати застарілі без істотної зміни її життєздатності. Використання побудованої моделі при модифікаціях системи дає можливість усунути небажані наслідки змін, оскільки вони не ламають структури системи, а тільки змінюють поведінку об'єктів.

Також при розробці магістерської роботи було використано наступні підходи UML: діаграма діяльності (діаграми поведінки типу); діаграма прецедентів (діаграми поведінки типу); Діаграма класів; Діаграма компонент; Діаграма об'єктів; Діаграма розгортання.

Діаграма діяльності. Це візуальне представлення графу діяльностей. Граф діяльностей є різновидом графу станів скінченного автомату, вершинами якого є певні дії, а переходи відбуваються по завершенню дій. Дія є фундаментальною одиницею визначення поведінки в специфікації. Дія отримує множину вхідних сигналів, та перетворює їх на множину вихідних сигналів.

Одна із цих множин, або обидві водночас, можуть бути порожніми. Виконання дії відповідає виконанню окремої дії. Подібно до цього, виконання діяльності є виконанням окремої діяльності, буквально, включно із виконанням тих дій, що містяться в діяльності. Кожна дія в діяльності може виконуватись один, два, або більше разів під час одного виконання діяльності. Щонайменше, дії мають отримувати дані, перетворювати їх та тестувати, деякі дії можуть вимагати певної послідовності.

Специфікація діяльності (на вищих рівнях сумісності) може дозволяти виконання декількох (логічних) потоків, та існування механізмів синхронізації для гарантування виконання дій у правильному порядку.

Діаграма прецедентів це діаграма, на якій зображено відношення між акторами та прецедентами в системі. Також, перекладається як діаграма варіантів використання.

Діаграма прецедентів є графом, що складається з множини акторів, прецедентів (варіантів використання) обмежених границею системи (прямокутник), асоціацій між акторами та прецедентами, відношень серед прецедентів, та відношень узагальнення між акторами. Діаграми прецедентів відображають елементи моделі варіантів використання.

Суть даної діаграми полягає в наступному: проектована система представляється у вигляді безлічі сутностей чи акторів, що взаємодіють із системою за допомогою так званих варіантів використання. Варіант використання (use case) використовують для описання послуг, які система надає актору. Іншими словами, кожен варіант використання визначає деякий набір дій, який виконує система при діалозі з актором.

При цьому нічого не говориться про те, яким чином буде реалізована взаємодія акторів із системою.

У мові UML є кілька стандартних видів відношень між акторами і варіантами використання:

– асоціації (association relationship);

– включення (include relationship);

– розширення (extend relationship);

– узагальнення (generalization relationship).

При цьому загальні властивості варіантів використання можуть бути представлені трьома різними способами, а саме – за допомогою відношень включення, розширення і узагальнення.

Відношення асоціації – одне з фундаментальних понять у мові UML і в тій чи іншій мірі використовується при побудові всіх графічних моделей систем у формі канонічних діаграм.

Включення (include) у мові UML – це різновид відношення залежності між базовим варіантом використання і його спеціальним випадком. При цьому відношенням залежності (dependency) є таке відношення між двома елементами моделі, при якому зміна одного елемента (незалежного) приводить до зміни іншого елемента (залежного).

Відношення розширення (extend) визначає взаємозв'язок базового варіанта використання з іншим варіантом використання, функціональна поведінка якого задіюється базовим не завжди, а тільки при виконанні додаткових умов.

Діаграма класів це статичне представлення структури моделі. Відображає статичні (декларативні) елементи, такі як: класи, типи даних, їх зміст та відношення.

Діаграма класів, також, може містити позначення для пакетів та може містити позначення для вкладених пакетів. Також, діаграма класів може містити позначення деяких елементів поведінки, однак їх динаміка розкривається в інших типах діаграм.

Діаграма класів (class diagram) служить для представлення статичної структури моделі системи в термінології класів об'єктно-орієнтованого програмування. На цій діаграмі показують класи, інтерфейси, об'єкти й кооперації, а також їхні відносини.

В UML існують наступні типи зв'язків які використовуються у діаграмі класів: Асоціації; Агрегація; Композиція.

Асоціації це якщо між двома класами визначена асоціація, то можна переміщатися від об'єктів одного класу до об'єктів іншого. Цілком припустимі випадки, коли обидва кінці асоціації відносяться до одного і того ж класу. Це означає, що з об'єктом деякого класу дозволено зв'язати інші об'єкти з того ж класу. Асоціація, що зв'язує два класи, називається бінарної. Можна, хоча це рідко буває необхідним, створювати асоціації, що зв'язують відразу кілька класів. Графічно асоціація зображується у вигляді лінії, що з'єднує клас сам з собою або з іншими класами.

Асоціації може бути присвоєно ім'я, яке описує природу відносини. Зазвичай ім'я асоціації не вказується, якщо тільки ви не хочете явно задати для неї рольові імена або у вашій моделі настільки багато асоціацій, що виникає необхідність посилатися на них і відрізняти один від одного. Ім'я буде особливо корисним, якщо між одними і тими ж класами існує кілька різних асоціацій.

Клас, що бере участь в асоціації, грає в ній деяку роль. По суті, це "обличчя", яким клас, що знаходиться на одній стороні асоціації, звернений до класу з іншого її боку. Можна явно позначити роль, яку клас грає в асоціації.

Часто при моделюванні буває важливо вказати, скільки об'єктів може бути пов'язано допомогою одного примірника асоціації. Це число називається кратністю (Multiplicity) ролі асоціації та записується або як вираз, значенням якого є діапазон значень, або в явному вигляді.

Вказуючи кратність на одному кінці асоціації, ви тим самим говорите, що на цьому кінці саме стільки об'єктів повинно відповідати кожному об'єкту на протилежному кінці. Кратність можна задати рівною одиниці (1), можна вказати діапазон: "нуль або одиниця" (0..1), "багато" (0 .. *), "одиниця або більше" (1 .. *). Дозволяється також вказувати певне число (наприклад, 3). За допомогою списку можна задати і більш складні кратності, наприклад 0. . 1, 3..4, 6 .. *, що означає "будь-яке число об'єктів, крім 2 і 5".

Агрегація це проста асоціація між двома класами відображає структурний відношення між рівноправними сутностями, коли обидва класу знаходяться на одному концептуальному рівні і ні один не є більш важливим, ніж інший. Але іноді доводиться моделювати відношення типу «частина/ціле», в якому один з класів має більш високий ранг (ціле) і складається з декількох менших за рангом (частин).

Ставлення такого типу називають агрегацією; воно зараховане до відносин типу «має» (з урахуванням того, що об'єкт-ціле має кілька об'єктів-частин). Агрегація є окремим випадком асоціації і зображується у вигляді простої асоціації

з незафарбованим ромбом з боку «цілого». Графічно агрегація представляється порожнім ромбом на блоці класу, і лінією, яка від цього ромба до міститься класу.

Композиція це більш суворий варіант агрегації. Відома також як агрегація за значенням.

Композиція має жорстку залежність часу існування екземплярів класу контейнера та примірників містяться класів. Якщо контейнер буде знищений, то весь його вміст буде також знищено. Графічно представляється як і агрегація, але з зафарбовані ромбиком.

Діаграма компонент в UML це діаграма, на якій відображаються компоненти, залежності та зв'язки між ними.

Діаграма компонент відображає залежності між компонентами програмного забезпечення, включаючи компоненти вихідних кодів, бінарні компоненти, та компоненти, що можуть виконуватись.

Модуль програмного забезпечення може бути представлено в якості компоненти. Деякі компоненти існують під час компіляції, деякі – під час компонування, а деякі під час роботи програми.

Діаграма компонент відображає лише структурні характеристики, для відображення окремих екземплярів компонент слід використовувати діаграму розгортування.

Компоненти об'єднуються разом використовуючи структурні зв'язки (assembly connector) щоб об'єднати інтерфейси двох компонент. Це ілюструє зв'язок типу «клієнт-сервер».

Структурна взаємодія – «зв'язок двох компонент, який передбачає, що один з них надає послуги, потрібні іншому компоненту».

При використанні діаграми компонент щоб показати внутрішню структуру компонента, клієнтські та серверні інтерфейси можуть утворювати пряме з'єднання з внутрішніми. Таке з'єднання називається з'єднанням делегації.

Діаграма об'єктів в UML це діаграма, що відображає об'єкти та їх зв'язки в певний момент часу. Діаграма об'єктів може розглядатись як окремий випадок

діаграми класів, на якій можуть бути представлені як класи, так і екземпляри (об'єкти) класів. Схожою за змістом є діаграма взаємодії (collaboration diagram).

Діаграми об'єктів не мають власної нотації. Оскільки діаграми класів можуть відображати об'єкти, то діаграма класів, на якій відображено лише об'єкти, та не відображено класи, може вважатись діаграмою об'єктів.

Діаграма об'єктів відображає об'єкти та зв'язки в певний момент роботи програми. Об'єкти можуть містити інформацію про власні значення а не про описання. Для відображення загальних шаблонів об'єктів та зв'язків, що можуть багаторазово створюватись під час роботи програми, слід використовувати діаграму взаємодії, яка може відображати характеристики об'єктів та зв'язків. Екземпляр діаграми взаємодії створює діаграму об'єктів.

Діаграма об'єктів не відображає еволюцію системи під час роботи. Натомість, слід використовувати діаграми взаємодії з повідомленнями, або діаграми послідовності.

Діаграма розгортання (deployment diagram) це діаграма в UML, на якій відображаються обчислювальні вузли під час роботи програми, компоненти, та об'єкти, що виконуються на цих вузлах. Компоненти відповідають представленню робочих екземплярів одиниць коду. Компоненти, що не мають представлення під час роботи програми на таких діаграмах не відображаються; натомість, їх можна відобразити на діаграмах компонент. Діаграма розгортання відображає робочі екземпляри компонент, а діаграма компонент, натомість, відображає зв'язки між типами компонент.

4.2 Захист розробленого програмного забезпечення

Захист розробленого програмного забезпечення буде відбуватися за допомогою алгоритму FEAL – блоковий шифр, запропонований Акіхіро Симідзу і Седзі Міягуті.

У ньому використовуються 64-бітовий блок і 64-бітовий ключ. Його ідея полягає і в тому, щоб створити алгоритм, подібний DES, але з більш сильною функцією етапу. Використовуючи менше етапів, цей алгоритм міг би працювати швидше. На жаль, дійсність виявилася далекою від цілей проекту.

Як вхід процесу шифрування використовується 64-бітовий блок відкритого тексту. Спочатку блок даних підлягає операції XOR з 64 бітами ключа. Потім блок даних розщеплюється на ліву і праву половини. Об'єднання лівої і правої половин за допомогою XOR утворює нову праву половину. Ліва половина і нова права половина проходять через N етапів (спочатку 4). На кожному етапі половина об'єднується за допомогою функції F[1] з 16 бітами ключа і за допомогою XOR – з лівою половиною, створюючи нову праву половину. Вихідна права половина (на початок етапу) стає новою лівою половиною. Після N етапів (ліва і права половини не переставляти після N-го етапу) ліва половина знову об'єднується з допомогою XOR з правою половиною, утворюючи нову праву половину, потім ліва і права об'єднуються разом в 64-бітове ціле. Блок даних об'єднується за допомогою XOR з іншими 64 бітами ключа і алгоритм завершується.

5 МЕТОДИКА ВПРОВАДЖЕННЯ СИСТЕМИ В ПРОМИСЛОВУ ЕКСПЛУАТАЦІЮ

Розглянемо розроблене ПЗ керування цифровими правами на основі DRM яке зображено на рисунку 5.1. З рисунку можна побачити що інтерфейс головного вікна розподілено на наступні функціональні розділи:

– Вибрати зображення.

– Перевірити на наявність стегоконтейнера.

– Створити стегоключ.

– Шифрувати стегоключ.

– Дешифрувати стегоключ.

– Вбудувати стегоконтейнер.

– Вилучити стегоконтейнер.

– Зберегти стегоповідомлення.

– Навігаційне меню: Робота з файлом; Стегокодер; Стегодекодер; Ключі; Параметри; Довідка.

Рисунок 5.1 – Головне вікно ПЗ

Розроблена програма має дуже простий і інтуїтивно зрозумілий інтерфейс з користувачем. Кожен, хто в достатньому обсязі володіє операційним середовищем Windows без особливих складностей освоїть і цю програму, оскільки її інтерфейс інтуїтивно зрозумілий.

Якщо програма не видала ніяких помилок, і працює, то можна використовувати, інакше слід слідувати інструкціям, які пропонує програма.

На рисунку 5.2 зображено авторські дані розробленого програмного забезпечення.

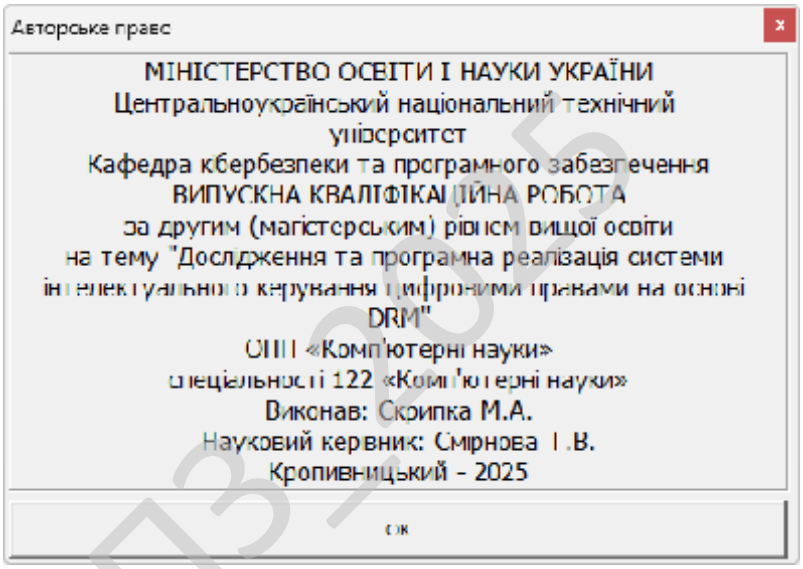

Рисунок 5.2 – Авторське право

Розглянемо процес впровадження програмного забезпечення, це процес налаштування програмного забезпечення під певні умови використання, а також навчання користувачів роботі з програмним продуктом. Впровадження програмного забезпечення це усі дії, що роблять розроблену програмну систему готовою до використання. Даний процес є частинною життєвого циклу програмного забезпечення.

Загалом процес розгортання складається з кількох взаємопов'язаних дій із можливими переходами між ними. Ця активність може відбуватися як з боку

виробника так і з боку споживача. Оскільки кожна програмна система є унікальною, то усі процеси та процедури під час розгортання важко передбачити. Тому, "розгортання" можна трактувати як загальний процес відповідно до певних вимог та характеристик. Розгортання може здійснюватись програмістом і в процесі розробки програмного забезпечення.

До діяльностей пов'язаних із розгортанням програмного забезпечення відносять:

– Випуск.

– Встановлення та активація.

– Деактивація.

– Адаптація.

– Обновлення.

– Вмонтування.

– Відстежування версій.

– Видалення.

– Вилучення з обігу.

При впровадженні програмного забезпечення потрібно урахувати наступні дії:

– Виділення критичних, з точки зору загального результату, процедур в діяльності організації. Коли набір таких процедур визначений, необхідно в першу чергу використовувати ІТ рішення для автоматизації операцій усередині саме цих процедур. Таким чином, розроблене ІТ рішення автоматично стає життєво важливим і затребуваним для організації, а також буде забезпечена публічність процесу впровадження.

– Розширення нормативної бази організації шляхом включення до неї регламентів, що описують порядок виконання процедур автоматизованих процесів. В іншому випадку є небезпека виникнення неузгодженості між автоматизованими процедурами та іншими процесами організації.

– Виконання робіт з загальної стандартизації існуючої діяльності організації, коли виділяються кращі практики виконання процедур і включаються в IT рішення за принципом найбільшої корисності для більшості учасників. Відсоток таких процедур щодо загального обсягу автоматизації може бути невеликий, але це надає процесу побудови рішення вагу в організації за рахунок збільшення його необхідності.

Під час роботи над програмою було проведено тестування програмного забезпечення, тобто технічне дослідження, призначене для виявлення інформації про якість продукту відносно контексту, в якому воно має використовуватись.

Тестування включає як процес пошуку помилок або інших дефектів, так і випробування програмних складових з метою їх оцінки.

Проводилась оцінка:

– відповідності поставленим вимогам;

– правильна відповідь для усіх можливих вхідних даних;

– виконання функцій за прийнятний час;

– практичність;

– сумісність з ОС та стороннім ПЗ.

Оскільки число можливих тестів для програмних компонент практично нескінченне, тому стратегія тестування полягала в тому, щоб провести всі можливі тести з урахуванням наявного часу та ресурсів.

Як результат ПЗ тестувалось стандартним виконанням програми з метою виявлення помилок або інших дефектів.

Проводилось тестування форматом чорної скриньки.

Основне місце програми тестів «чорної скриньки» – інтерфейс ПЗ. Відомі: функції програми. Досліджується: робота кожної функції на всій області визначення.

Ці тести демонструють:

– Як виконуються функції програми.

– Як приймаються вихідні дані.

– Як виробляються результати.

– Як зберігається цілісність зовнішньої інформації.

При тестуванні «чорної скриньки» розглядаються системні характеристики програм, ігнорується їхня внутрішня логічна структура. Вичерпне тестування, як правило, неможливе.

Наприклад, якщо в програмі 10 вхідних величин і кожна приймає по 10 значень, то кількість тестових варіантів становитиме 10^10. Тестування «чорної скриньки» не реагує на багато особливостей програмних помилок.

Тестування «чорної скриньки» (функціональне тестування) дозволяє отримати комбінації вхідних даних, які забезпечують повну перевірку всіх функціональних вимог до програми.

Програмний виріб тут розглядається як «чорна скринька», чию поведінку можна визначити тільки дослідженням його входів та відповідних виходів. При такому підході бажано мати:

– Набір, утворений такими вхідними даними, які призводять до аномалій у поведінці програми (назвемо його ITc).

– Набір, утворений такими вхідними даними, які демонструють дефекти програми (назвемо його OT).

Будь-який спосіб тестування «чорної скриньки» повинен:

– Виявити такі вхідні дані, які з високою ймовірністю належать набору ITc;

– Сформулювати такі очікувані результати, які з високою імовірністю є елементами набору OT.

Принцип «чорної скриньки» не альтернативний принципу «білої скриньки». Скоріше це доповнює підхід, який виявляє інший клас помилок.

Тестування «чорної скриньки» забезпечує пошук наступних категорій помилок: Некоректних чи відсутніх функцій; Помилок інтерфейсу; Помилок у зовнішніх структурах даних або в доступі до зовнішньої бази даних; Помилок характеристик (необхідна ємність пам'яті і т.д.); Помилок ініціалізації та завершення.

Обрано умови розповсюдження – proprietary software.

Програмне забезпечення, на яке зберігаються як немайнові, так і майнові авторські права. Отримавши або придбавши таке програмне забезпечення, користувач отримує обмежені права користування ним: може бути заборонено або закрито доступ до коду (вивчення), внесення змін, тиражування, розповсюдження та перепродаж. Програмне забезпечення вважається власницьким, якщо наявне хоча б одне з перелічених обмежень. Найчастіше основним методом захисту майнових прав на власницьке ПЗ, поза ліцензійною угодою, власник обирає закриття сирцевого коду, захищаючи свій продукт від модифікації і вбудовуючи системи обмеження користування через авторизацію. Таке програмне забезпечення називається закритим. Проте, код власницького продукту може бути і відкритим, але власник може обмежити права користувача умовами користувацької ліцензії. Власницьке програмне забезпечення та комерційне програмне забезпечення не є синонімами – власницьким може бути і безплатне (тобто, некомерційне) програмне забезпечення. На противагу власницькому ПЗ існує вільне програмне забезпечення, автори і власники якого дозволяють вивчати, модифікувати і поширювати свій продукт. Саме визначення власницького програмного забезпечення виникло в результаті діяльності громадського руху вільного програмного забезпечення (представленого Фондом вільного програмного забезпечення та іншими організаціями) і осмислення умов свободи користування програмами. Визначенням власницького програмного забезпечення є не невідповідність хоча б одній з базових умов вільного програмного забезпечення. Сама назва власницьке ПЗ підкреслює визначальне значення власника у способі використання і можливостях розвитку цього програмного забезпечення.

6 НАУКОВА НОВИЗНА

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти розроблено програмне забезпечення, яке призначено для системи інтелектуального керування цифровими правами на основі DRM.

Метою розробки є дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

Об'єктом дослідження є процес інтелектуального керування цифровими правами на основі DRM.

Предметом дослідження є методи інтелектуального керування цифровими правами на основі DRM.

Методи дослідження базуються на методах захисту інтелектуальної власності, методах математичної статистики, методах розробки програмного забезпечення.

Наукова новизна отриманих результатів. У процесі рішення завдань, обумовлених цілями дослідження, отримані наступні результати:

– Удосконалено метод інтелектуального керування цифровими правами на основі DRM.

– Розроблено вітчизняний продукт інтелектуального керування цифровими правами на основі DRM, який має більш широкі можливості, на відміну від існуючих аналогів.

7 МАРКЕТИНГОВЕ ТА ЕКОНОМІЧНЕ ОБГРУНТУВАННЯ ІТ-ПРОЄКТУ

7.1 Визначення цільової аудиторії кінцевого готового продукту

Результати цього дослідження можуть бути цікавими насамперед компаніям, які займаються створенням, розповсюдженням або захистом цифрового контенту – від музичних лейблів і кінокомпаній до онлайн-видавництв і розробників програмного забезпечення. Для них DRM є дієвим способом захисту авторських прав і збереження прибутку від легального використання продукції. У сучасних умовах цифрової економіки, де дані та контент мають ключову цінність, захист інтелектуальної власності стає не лише технічним, а й стратегічним питанням.

Ця тема також може зацікавити IT-компанії, які спеціалізуються на розробці хмарних сервісів або платформ потокового контенту. Для них інтеграція DRM – це не просто засіб контролю доступу, а конкурентна перевага, що дозволяє пропонувати користувачам легальний контент без ризику втрати репутації чи порушення прав третіх сторін.

Окрім того, дослідження є актуальним для освітніх і наукових установ, які працюють з великими обсягами цифрових матеріалів – навчальних курсів, лекцій, досліджень, публікацій. Для таких організацій впровадження DRM дозволяє захистити результати інтелектуальної праці викладачів і дослідників, а також контролювати використання матеріалів у межах ліцензійних угод.

7.2 Оцінка привабливості шляхом застосування методів експертних оцінок

Для визначення привабливості впровадження DRM-системи можна провести експертну оцінку, залучивши фахівців у сфері IT-безпеки, медіаіндустрії, права та електронної комерції. Експерти оцінюють проєкт за критеріями: технічна доцільність, рівень інноваційності, економічна вигода, складність інтеграції та потенціал для масштабування. Кожен із параметрів отримує бальну оцінку від 1 до 10, після чого обчислюється середнє значення.

Припустимо, що експерти оцінили систему на 9 балів за ефективність захисту контенту, 8 – за економічну вигоду, 7 – за складність впровадження, 9 – за потенціал масштабування. Середній бал 8,25 свідчить про високу привабливість розробки як з технологічного, так і з комерційного погляду.

Крім кількісних оцінок, експерти можуть надати якісні висновки – наприклад, що система має значні перспективи у сфері стрімінгових платформ або навчальних сервісів. Такі дані допомагають керівництву компанії ухвалити зважене рішення про інвестиції та визначити пріоритетні напрями розвитку продукту.

7.3 Вибір методу оцінки вартості ПЗ

Найдоцільніше використовувати метод оцінки вартості, що базується на концепції повної вартості володіння (Total Cost of Ownership, TCO). Цей метод дозволяє врахувати не лише початкові витрати на впровадження DRM-системи, але й усі супутні витрати протягом життєвого циклу – технічну підтримку, оновлення, навчання персоналу, інтеграцію з іншими платформами.

Також важливо провести аналіз вигід (Cost-Benefit Analysis), щоб зрозуміти, наскільки інвестиції в DRM забезпечують реальне зростання доходів. Наприклад, зниження кількості нелегальних копій на 20–30% може принести

компанії суттєве збільшення прибутку вже протягом перших місяців експлуатації.

Комбінація цих методів дає змогу не лише оцінити фінансову доцільність, але й визначити точку беззбитковості – момент, коли інвестиції починають приносити чистий прибуток. У випадку DRM, враховуючи її швидкий ефект у боротьбі з піратством, термін окупності зазвичай не перевищує одного року, що робить цю технологію привабливою для впровадження навіть у середніх компаніях.

7.4 Розрахунок економічної ефективності від впровадження реалізованого ПЗ як фактору його привабливості

Економічна ефективність від впровадження системи інтелектуального керування цифровими правами (Digital Rights Management, DRM) у компанії, що займається розповсюдженням цифрового контенту – відео, музики, електронних книг або програмного забезпечення розраховується наступним чином. Вхідні дані зафіксовано в таблиці 7.1.

Розрахунок економічного ефекту демонструє наступне: зменшення втрат від піратства – 1 050 000грн, збільшення доходів від легальних продажів – 900 000грн, економія від автоматизації управління правами (зменшення потреби у ручній перевірці) – 216 000грн, загальний річний економічний ефект – 2 166 000 грн, витрати на впровадження та підтримку DRM – 500 000 грн, чистий річний економічний ефект – 1 666 000 грн, термін окупності (Payback Period) – 0,24 року (~3 місяці), ROI (окупність інвестицій) – 416,5%.

Додаткові нефінансові вигоди: підвищення правової безпеки компанії – дотримання міжнародних стандартів ліцензування контенту (CISAC, WIPO), можливість виходу на нові ринки – DRM є обов'язковою умовою співпраці з глобальними платформами (Google Play, Apple iTunes, Netflix), захист бренду – зменшення випадків репутаційних ризиків, пов'язаних із нелегальним

поширенням контенту, аналітика використання – система дозволяє отримувати дані про споживчу поведінку користувачів і будувати ефективні маркетингові стратегії, гнучкість бізнес-моделей – DRM дає змогу впроваджувати pay-per-use або time-limited доступ, що підвищує прибутковість продукту.

Таблиця 7.1 – Вихідні дані для розрахунку

Показник	**До впровадження**	**Після впровадження**
Кількість нелегальних копій контенту	близько 25%	<3%
Річні втрати від піратства	1 200 000 грн	150 000 грн
Витрати на підтримку авторських прав	200 000 грн	250 000 грн
Вартість ліцензії та впровадження DRM-системи	—	400 000 грн
Річна підтримка системи DRM	—	100 000 грн
Дохід від легальних продажів контенту	3 000 000 грн	3 900 000 грн
Річна кількість користувачів, що сплачують	20 000	27 000

Впровадження системи інтелектуального керування цифровими правами на основі DRM не лише знижує рівень піратства, але й відкриває нові можливості для монетизації контенту. Уже в перший рік експлуатації система окуповується менш ніж за 4 місяці, а річний економічний ефект перевищує 1,6 млн грн.

Таке рішення робить бізнес більш конкурентоспроможним і захищеним, створює умови для співпраці з міжнародними партнерами та підвищує довіру з боку правовласників. У довгостроковій перспективі впровадження DRM стає стратегічним кроком у розвитку цифрових сервісів, сприяє легалізації ринку контенту та забезпечує стабільне зростання доходів компанії.

7.5 Пропозиція алгоритму просування проєкту розробки ПЗ

Просування проєкту системи DRM доцільно розпочати з формування чіткої стратегії позиціонування. Важливо підкреслити, що це не просто технічний інструмент, а бізнес-рішення, яке дозволяє захистити дохід і зберегти репутацію бренду. На першому етапі слід створити демонстраційний прототип, який би показував, як система контролює доступ і запобігає несанкціонованому копіюванню.

Далі варто залучати цільову аудиторію через тематичні конференції, виставки, IT-форуми та галузеві публікації. Особливо ефективним є співробітництво з медіа-платформами, дистриб'юторами програмного забезпечення та видавництвами, яким DRM може допомогти зменшити витрати на боротьбу з піратством.

Завершальним етапом має стати партнерство з компаніями, що надають хмарні сервіси або рішення з кібербезпеки. Інтеграція DRM у їхні продукти дозволить створити єдину екосистему захисту контенту, що значно розширить ринок і зробить систему привабливою для міжнародних клієнтів.

7.6 Оптимізація каналів збуту та шляхів реалізації ПЗ

Для ефективної реалізації проєкту варто використовувати комбінований підхід – одночасно працювати із великими корпораціями, медіа-холдингами та малими бізнесами через гнучкі моделі ліцензування. Наприклад, компанії можуть придбавати повну ліцензію, тоді як менші користувачі – працювати за моделлю SaaS, отримуючи доступ до функцій DRM у хмарі без необхідності встановлення складного програмного забезпечення.

Важливу роль відіграє партнерський маркетинг. Співпраця з IT-дистриб'юторами та консультантами з інформаційної безпеки допоможе швидше охопити корпоративний сегмент ринку. Крім того, просування через онлайн-

платформи з відкритими API дозволяє легко інтегрувати DRM у вже існуючі сервіси, що стимулює попит серед розробників і стартапів.

Додатковим фактором оптимізації може стати створення навчальних матеріалів та онлайн-вебінарів для клієнтів, які демонструватимуть, як DRM знижує ризики піратства і підвищує прибутковість. Таким чином, компанія не просто продає продукт, а формує культуру свідомого ставлення до захисту контенту.

7.7 Визначення ключових факторів успіху конкретного проєкту

Ключовими факторами успіху цього проєкту є технологічна надійність, простота інтеграції та ефективна стратегія монетизації. Система повинна гарантувати високий рівень захисту контенту, але водночас залишатися зручною для легальних користувачів. Баланс між безпекою і комфортом користування є вирішальним – надто суворий контроль може відштовхнути клієнтів, а надто м’який не забезпечить потрібного рівня захисту.

Велике значення має також підтримка актуальності системи. Постійні оновлення алгоритмів шифрування, адаптація до нових форматів контенту та платформ – це запорука конкурентоспроможності продукту.

Ще один важливий аспект – репутація компанії-розробника. Якщо клієнти відчувають, що отримують не просто інструмент, а комплексне рішення з надійною технічною підтримкою, довіра до бренду зростає. Саме поєднання технічної досконалості, зручності впровадження та відкритої комунікації з клієнтами визначає успіх будь-якого DRM-рішення у сучасному цифровому світі.

8 ЗАХОДИ З ОХОРОНИ ПРАЦІ ТА ТЕХНІКИ БЕЗПЕКИ

8.1 Вступ

В кожній ІТ-компанії є трудові відносини з працівниками. Згідно закону України “Про охорону праці” [3] кожна компанія впроваджує заходи з охорони праці. Реалізується трудові відносини з вживанням необхідних засобів з охорони праці та розробки відповідних документів:

– Інструкцій з охорони праці по кожній професії і загальні.

– Положення про охорону праці.

– Накази з охорони праці.

– Журнали реєстрації та інструктажу.

Роботодавець створює відділ який працює відповідно до типового положення, яке затверджується центральним органом виконавчої влади і забезпечує виконання вимог державної політики у сфері охорони праці.

За недотримання вимог, керівники ІТ-компаній можуть бути притягнуті до відповідальності, яка тягне накладання штрафу. Якщо в результаті порушення умов охорони праці є постраждалі працівники, то керівні особи ІТ-компаній притягуються до кримінальної відповідальності.

Законом України “Про охорону праці” [3] регламентуються загальні положення державної політики в галузі охорони праці, а конкретизуються ці положення нормативно-правовими актами про охорону праці, зокрема Наказом Міністерства соціальної політики України 14.02.2018 № 207, який зареєстровано в Міністерстві юстиції України 25 квітня 2018 р. за №508/31960 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями» [5], яким затверджено нормативно-правовий акт з охорони праці НПАОП 0.00-7.15-18, «Правила охорони праці під час експлуатації електронно-обчислювальних машин», та «Державні санітарні правила і норми

роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» ДСанПіН 3.3.2-007-98.

Програмісти у процесі роботи мають негативний вплив на органи зору, а також мають значну розумову напругою і нервово-емоційне навантаження. Руки (суглоби пальців та м'язи рук) при роботі з клавіатурою мають теж істотне навантаженням. До шкідливих факторів, які впливають на робітників галузі інформаційних технологій (ІТ) спеціалісти відносять високочастотні електромагнітні коливання (випромінювання) роботи апаратної частини ЕОМ та виділення шкідливих газів.

Ці шкідливі фактори можуть привести до професійних захворювань.

Розглянемо шкідливі чинники роботи програмістів керуючись наступними нормативно-правовими актами: «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно- обчислювальних машин» ДСанПіН 3.3.2-007-98 [5], та «Вимоги щодо безпеки та захисту здоров'я працівників під час роботи з екранними пристроями» НПАОП 0.00-7.15-18.

Умови праці програміста включають наступні фактори:

– параметри повітряного середовища в приміщенні;

– вентиляція приміщення;

– освітлення приміщення;

– параметри повітряного середовища в приміщенні, тощо.

Щоб запропонувати заходи щодо зменшення негативного впливу комп'ютера на організм людини визначимо фактори, які можуть викликати професійне захворювання і впливають на працездатність програміста.

8.2 Шкідливі і небезпечні фактори при роботі з комп'ютером

Електронно-обчислювальна машин (ЕОМ) та інше обладнання є джерелами небезпеки ураження електричним струмом. Оскільки робота програміста характеризується істотним зоровим навантаженням, то вимагає

належного освітлення. У приміщенні, в якому працюють програмісти, необхідно створити належний мікроклімат, параметри якого регламентуються Державними санітарними правилами і нормами, зокрема ДСанПіН 3.3.2.007-98.

При роботі з використанням ЕОМ відзначають наступні небезпечні та шкідливі фактори:

– ризик виникнення надзвичайних ситуацій природного або штучного характеру на об’єкті або території.

– ризик виникнення пожежі;

– негативний вплив на органи зору людини;

– ризики ураження електричним струмом;

– недостатня, або надмірна освітленість робочого місця;

– електромагнітні (у тому числі високочастотні) випромінювання (коливання);

– несприятливі мікрокліматичні умови;

– нервово-емоційна напруженість праці;

– інтелектуальні навантаження;

– монотонність праці;

– невідповідність ергономічних показників робочого місця діючим вимогам;

– шум;

– статичні навантаження на кістково-м’язовий апарат.

Працю користувачів ЕОМ відносять до психічних форм праці з високим ступенем навантаження. Ця діяльність пов'язана зі сприйняттям зображення на екрані, постійним стеженням за його динамікою, розрізненням картин, схем, читанням тексту рукописних та друкованих матеріалів, введенням інформації з клавіатури, необхідністю підтримувати активну увагу високою ціною помилки. Будь-яка діяльність із застосуванням ЕОМ супроводжується необхідністю активації уваги та інших вищих психічних функцій, а організм людини, крім того, піддається впливу кількох десятків різноманітних факторів.

8.3 Аналіз санітарно-гігієнічних умов праці на робочому місці програміста

Розглянемо умови праці у приміщенні, в якому працюють програмісти. Геометричні розміри приміщення наведено у таблиці 8.1.

Таблиця 8.1 – Розміри приміщення

Найменування	Значення, м
Ширина	5
Довжина	6,2
Висота	3,4

Таблиця 8.2 – Площа та обсяг приміщення, на одного працюючого*

Геометрична характеристика	Одиниця виміру	Нормативне значення*	Фактичне значення
Площа, S	$м^2$	не менше 6.0	6,2
Об'єм, V	$м^3$	не менше 20.0	21,8

* Згідно ДСанПіН 3.3.2.007-98 (Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин).

У зазначеному приміщенні працюють двоє людей. За даними, які наведено у таблицях 8.1-8.2, можна зробити висновок, що площа та об’єм приміщення у розрахунку на одне робоче місце програміста не відповідають нормативним вимогам ДСанПіН 3.3.2-007-98 «Державні санітарні правила і норми роботи з візуальними дисплейними терміналами електронно-обчислювальних машин» [5], але відповідають нормативним вимогам Наказу Міністерства соціальної політики України № 207, від 14.02.2018 «Про затвердження Вимог щодо безпеки та захисту здоров’я працівників під час роботи з екранними пристроями» [5] та НПАОП 0.00-1.28-10 «Правила охорони праці під час експлуатації електронно-

обчислювальних машин»). Таким чином, можна зробити висновок, що санітарно-гігієнічні умови праці на робочому місці програміста відповідають вимогам.

Температура повітря в приміщенні визначається впливом температури зовнішнього повітря і тепловою енергією, яка виділяється всередині приміщення. Джерелами виділення теплоти в даному приміщенні є електроустаткування, освітлювальні прилади, а також люди. У світлий час доби джерелом надлишкового тепла є сонячна радіація. Згідно Постанови №42 від 01.12.1999 Головного державного санітарного лікаря України, робота, виконувана в даному приміщенні, відноситься до категорії Іа. В цьому випадку людина витрачає енергії до 120 ккал у годину. Вологість повітря в приміщенні визначається впливом багатьох факторів, серед яких: вологість атмосферного повітря, виділення вологи людьми (при диханні та випарами з поверхні шкіри).

Мікроклімат повітряного середовища в приміщенні характеризується запиленістю та загазованістю повітря. Мікроклімат приміщення визначається діючим на організм людини поєднанням, вологості, температури, швидкості руху повітря та інтенсивності теплового випромінювання. Аналіз мікроклімату складається з визначення зазначених вище факторів і порівняння результатів із встановленими нормами.

У таблиці 8.3 наведено оптимальні та фактичні значення параметрів мікроклімату як для категорії ваги робіт Іа, так і розглянутого приміщення. У приміщеннях, де встановлено ЕОМ, рекомендується дотримування тільки оптимальних значень показників мікроклімату.

Проведений аналіз показує, що показники мікроклімату в приміщенні відповідають установленим нормам. Штучне опалення застосовується у холодний період року.

В літню пору застосовується кондиціонер.

Для боротьби з пилом робляться регулярні провітрювання та вологі прибирання приміщенні.

Таблиця 8.3 – Оптимальні і фактичні значення параметрів мікроклімату

Пора року	Оптимальні для Іа			Фактичні		
	Температура, °С	Воло-гість, %	Швидкість повітря, м/с	Температура,° С	Воло-гість, %	Швидкість повітря, м/с
Холодна	22-24	40-60	0,1	22-23	40-58	0,11
Тепла	23-25	50-70	0,1	24-25	50-65	0,2

У приміщенні знаходяться наступні джерела шуму: принтер Epson L132 (C11CE58403), електродвигуни вентиляторів ЕОМ.

Одним з найважливіших факторів, які впливають на ефективність трудової діяльності людини та попереджають травматизм і професійні захворювання програмістів, є освітлення на робочому місці.

З 2019 року діють Державні будівельні норми України “Природне і штучне освітлення” – ДБН В.2.5-28:2018 [1], у яких прописані вимоги до використання всіх освітлювальних приладів, у тому числі світлодіодних.

Роботу працівника, який постійно працює за комп’ютером, згідно ДБН В.2.5-28:2018 [1], можна віднести до роботи з малою точністю (найменший розмір об'єкта розрізнення від 1 до 5 мм) V-го розряду зорової роботи, з великою контрастністю об'єкта розрізнення (символів на екрані дисплея), з темним тлом (під розряд зорової роботи В).

Приміщення можна віднести до 1-ої групи приміщень, у яких проводиться розрізнення об'єктів зорової роботи при фіксованому напрямку лінії зору того, хто працює, на робочу поверхню. Для такого типу приміщень і розряду зорової роботи нормоване значення коефіцієнта природної освітленості (КПО) робочої поверхні (при поєднаному, спільному освітленні), повинен становити не більше 1,5%, освітленість при штучному освітленні повинна становити 300 Лк. [1], Крім того, все поле зору повинне бути освітлено достатньо рівномірно – це основна гігієнічна вимога. Оскільки яскраве світло на ділянці периферийного зору значно

збільшує напруженість очей і, як наслідок, призводить до їх швидкої стомлюваності, ступінь освітлення приміщення і яскравість екрану комп'ютера повинні бути приблизно однаковими.

8.4 Розробка заходів з умов поліпшення охорони праці

Згідно аналізу умов праці в розглянутому приміщенні, ми одержали наступні результати:

– розмірі приміщення, у розрахунку на одному працюючого, відповідають нормативам;

– мікроклімат відповідає нормативному значенню;

– акустичні умови роботи не перевищують нормативних значень;

Таким чином можна припустити, що основною причиною можливого зниження працездатності програміста є психофізіологічний фактор, тому основна пропозиція буде така: дотримання позитивної психологічної атмосфери в колективі та регламентованого режиму праці та відпочинку, організація робочого місця з урахуванням ергономічних вимог. Рекомендовані заходи: регулярні періодичні наочні огляди персоналом шляхів для евакуації людей із приміщення, відповідно до плану евакуації (який повинен розташовуватись на видному місці у приміщенні), включення до колективного договору мінімально можливого вмісту аптечок з обов'язковою наявністю масок-клапанів, або іншого спорядження для штучного дихання. Регулярна періодична перевірка параметрів заземлення та занулення (вимірювання опору ланцюга). Регулярна наочне знайомство персоналу із шляхами для евакуації людей із приміщення відповідно до плану евакуації, забезпечення розподільних щитів спеціальними розетками з заземлюючими контактами; організація заземлення всіх приладів і пристроїв, які працюють при напрузі вище 36 В. Так як при ураженні електричним струмом у людини може статися фібриляція шлуночків серця, в організації бажано мати дефібрилятор і підготовлений персонал для роботи з ним.

8.5 Розрахункова частина

Проведемо розрахунок штучного освітлення за методом коефіцієнту використання світлового потоку для приміщення ширина якого складає 5 м, довжина – 6,2 м, висота – 3,4 м.

У зазначеному приміщенні працює 5 людей.

Для того, щоб визначити потрібну кількість світильників, які повинні забезпечити нормований рівень освітленості, визначимо світловий потік, що падає на робочу поверхню за формулою:

$$F = E \cdot S \cdot K \cdot Z / n,$$

де

F – світловий потік, що розраховується, Лм;

E – нормована мінімальна освітленість, Лк; E = 300 Лк;

S – площа освітлюваного приміщення (у нашому випадку S = 5x6,2 = 31 $м^2$);

K – коефіцієнт запасу, що враховує зменшення світлового потоку лампи в результаті забруднення світильників в процесі експлуатації (його значення залежить від типу приміщення і характеру робіт, що проводяться в ньому, в нашому випадку K = 1,5);

Z – відношення середньої освітленості до мінімальної (зазвичай приймається рівним 1.1... 1.2, в нашому випадку Z = 1,1);

n – коефіцієнт використання світлового потоку, (відношення світлового потоку, що падає на розрахункову поверхню, до сумарного потоку всіх ламп, обчислюється в долях одиниці; залежить від характеристик світильника, розмірів приміщення, забарвлення стін і стелі, що характеризуються коефіцієнтами відбиття від стін ($\rho_{стін.}$) і стелі ($\rho_{стелі}$), значення коефіцієнтів дорівнюють $\rho_{стін}$ = 50% і $\rho_{стелі}$ = 50%.

Обчислимо індекс приміщення за формулою:

$$i = S/(h \cdot (A+B)),$$

де:

S – площа приміщення, S = 31 м2;

h – розрахункова висота підвісу, h = 3 м (співпадає з висотою стелі, оскільки лампи освітлення закріплюються на стелі);

A – ширина приміщення, A = 5 м;

B – довжина приміщення, B = 6,2 м.

Підставимо всі значення у формулу та визначимо індекс приміщення: i=0,43.

Знаючи індекс приміщення, за знаходимо n = 0,23 (з табличних даних коефіцієнтів використання світлового потоку (n) світильників з відповідним типом лампам). Підставимо всі значення у формулу, визначимо світловий потік: F=66717 Лм.

Для розрахунку будемо використовувати світлодіодні панелі LED панель PL PFM 600 30W/3000K, світловий потік яких $F_л$ = 3000 Лм.

Число ламп визначається за формулою:

$$N=F/F_л$$

де

F – світловий потік,

$F_л$ – світловий потік однієї лампи.

Підставимо всі значення у формулу та визначимо потрібну кількість ламп:

$$N= 66717/3000 = 22,18 \text{ шт.}$$

Приймаємо необхідну кількість світлодіодних світильників 23 шт.

Висновки до розділу

Дотримання всіх необхідних умов праці не лише сприяє збереженню здоров’я працівників, а також підвищує ефективність виробництва в цілому.

З цих міркувань було здійснено аналіз умов праці, призначеного для праці програмістів, проведено розгляд небезпечних та шкідливих факторів, що негативно впливають на програмістів під час роботи. Виконано розрахунок штучного освітлення, як одного з ключових факторів впливу на працездатність та здоров’я програміста. Розроблено заходи з охорони праці.

9 ОСНОВНІ ВИСНОВКИ

Програмне забезпечення, створене в результаті виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти, призначено для системи інтелектуального керування цифровими правами на основі DRM.

В межах України в недостатній мірі представлені вітчизняні розробки в цій області.

У випускній кваліфікаційній роботі за другим (магістерським) рівнем вищої освіти наведені теоретичне узагальнення й рішення наукового завдання дослідження методів інтелектуального керування цифровими правами на основі DRM.

Рішення даного завдання полягало у вирішенні наступних задач:

– Був проведений огляд існуючих систем інтелектуального керування цифровими правами на основі DRM.

– Досліджена система інтелектуального керування цифровими правами на основі DRM.

– На основі отриманих результатів досліджень створена програмна реалізація системи інтелектуального керування цифровими правами на основі DRM.

Розроблені під час виконання випускної кваліфікаційної роботи за другим (магістерським) рівнем вищої освіти алгоритми дозволяють успішно вирішувати завдання інтелектуального керування цифровими правами на основі DRM.

Проведено аналіз предметної галузі в ході якого були виявлені об'єкти, взаємодія яких носить істотний характер для функціональної діяльності предметної галузі, і їхні основні характеристики; побудована алгоритм і вибраний середовище розробки.

Розроблене програмне забезпечення має простий, дружній та зручний інтерфейс користувача, що забезпечує легкість у освоєнні роботи програмного продукту, зручність у використанні, і не потребує особливих спеціальних знань.

При створені програмного забезпечення було використано об'єктно-орієнтований підхід, що відповідає сучасним тенденціям у галузі розробки комерційних програмних систем.

Програма реалізована на мові високого рівня Python. Дана мова програмування дозволяє найбільш ефективно обробляти дані. Це дозволило мінімізувати строк розробки програмного забезпечення, і, як слід, зменшити витрати на його розробку. Запропоноване програмне забезпечення ділиться на загальне програмне забезпечення, що поставляється із засобами обчислювальної техніки й спеціальне програмне забезпечення, що спеціально розроблене для даної конкретної системи й включає програми, що реалізують її функції.

Програма призначена для виконання під управлінням багатозадачної операційної системи Windows 10/11.

Даються необхідні рекомендації з установки розробленого програмного забезпечення.

Для підвищення рівня безпеки запропоновано застосовувати алгоритм FEAL.

В цілому створене програмне забезпечення підтверджує правильність використаних проектних рішень та повністю відповідає вимогам технічного завдання. Створене програмне забезпечення має потенційну можливість для подальшого вдосконалення і застосування у різних галузях.

Проведено маркетингове та економічне обґрунтування ІТ-проєкту, що дозволило визначити ключові фактори успіху даного проекту.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. Скрипка М.А. Дослідження та програмна реалізація системи інтелектуального керування цифровими правами на основі DRM // Збірник праць молодих науковців ЦНТУ. – Вип. 15. – Кропивницький: ЦНТУ, 2025.

2. Awais Rashid, Howard Chivers, George Danezis, Emil Lupu, Andrew Martin. CyBOK The Cyber Security Body of Knowledge. The National Cyber Security Centre. 2019. 854 p.

3. Loren Kohnfelder. Designing Secure Software. No Starch Press. 2022. 332 p.

4. Samir Kumar Rakshit. Ethical Hacker's Penetration Testing Guide. BPB Online. 2022. 509 p.

5. Corey J. Ball. Hacking APIs. No Starch Press. 2022. 353 p.

6. Kevin Beaver. Hacking for Dummies. John Wiley & Sons. 2022. 419 p.

7. Mark S. Merkow. Practical Security for Agile and DevOps. CRC Press. 2022. 236 p

8. Derek Fisher. Application Security Program Handbook. Manning Publications. 2021. 155 p.

9. Cameron Wyatt PH.D. Kali Linux Tutorial. Independently published. 2021. 60 p.

10. Alex Matrosov, Eugene Rodionov, Sergey Bratus. Rootkits and Bootkits. No Starch Press. 2019. 450 p.

11. Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед та ін. «Системи інформаційної зброї та технології інформаційної війни»: підручник / Петрик В.М., Присяжнюк М.М., Аль-Файюмі Халед, Жарков Я.М., Смірнов О.А, Буравченко К.О., Давидюк А.В., Кононович В.Г., Корчинский В.В., Кудирко В.М., Фесенко А.О.; за заг. ред. В.М. Петрика, М.М. Присяжнюка.– К.: Видавничий центр “Кафедра”, 2025.– 320 с.

12. Усік, П.С., Смірнова, Т.В., Буравченко, К.О., Смірнов, О.А., Улічев, О.С., Смірнов, С.А. «Дослідження технологій забезпечення кібербезпеки банківських систем з використанням штучного інтелекту». *Кібербезпека: освіта, наука, техніка.* 2025. Том 1 № 29. С.704–716, 2025.

13. Kuznetsov, O., Frontoni, E., Kuznetsova, K., Arnesano, M., Smirnov, O. «A secure biometric authentication architecture for blockchain-driven cyber-physical systems». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 193–224.

14. Kuznetsov, O., Atzeni, G., Arnesano, M., Randieri, C., Smirnov, O. «Secure IoT-based smart wheelchair system: From implementation to security enhancement strategy». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 225–257.

15. Kuznetsov, O., Smirnov, O., Kuznetsova, T., Shaikhanova, A., Svatowsky, I. «Privacy-utility trade-offs in IoT networks: A comparative analysis of differential privacy mechanisms for sensor data aggregation». *Security and Privacy of Cyber Physical Systems Emerging Trends Technologies and Applications*, 2025, pp. 589–622.

16. Kuznetsov, O., Poluyanenko, N., Smirnov, O., Kozhakhmetova, D. «Optimized Simulated Annealing for Efficient Generation of Highly Nonlinear S-Boxes». *Advancements in Cybersecurity Next Generation Systems and Applications,* 2025. 146-174

17. Kuznetsov, O., Poluyanenko, N., Smirnov, O., Bekeshova, G. «Enhanced Cryptographic Security through Advanced S-Box Optimization: A Hybrid Heuristic Approach». *Advancements in Cybersecurity Next Generation Systems and Applications*, 2025. pp. 56-78.

18. Kuznetsov, O., Poluyanenko, N., Smirnov, O., Abduraimova, B «Enhancing Cryptographic Strength: A Novel Approach to S-Box Generation Using Modified Simulated Annealing». *Advancements in Cybersecurity Next Generation Systems and Applications*, 2025. pp. 79-116.

19. Kuznetsov, O., Derevianko, Y., Frontoni, E., Arnesano, M., Smirnov, O. «Factorial Representation of S-Boxes: A Novel Approach to Cryptographic Analysis and Optimization». *Advancements in Cybersecurity Next Generation Systems and Applications,* 2025. pp. 117-145.

20. Kuznetsov, O., Poluyanenko, N., Smirnov, O., Shaikhanova, A., Khruskov, B. «Innovative Cost Functions for Optimizing Cryptographic S-Box Generation». *Advancements in Cybersecurity Next Generation Systems and Applications*, 2025. pp. 29-55.

21. Kuznetsov, O., Smirnov, O., Akhmetov, B., Alimseitova, Z., Imoize, A.L. «Deep Learning Frontiers in Copy-Move Forgery Detection: Advances, Challenges, and Future Directions». *Advancements in Cybersecurity Next Generation Systems and Applications,* 2025. 202-229.

22. Lakhno, V., Malyukov, V., Smirnov, O., Bebeshko, B., Chubaievskiy, V., Zhumadilova, M., Malyukova, I., Smirnov, S. «Multifactorial Model for Targeted Attacks Counteracting Within the Framework of a Multi-Step Quality Game with Fuzzy Information». *8th International Symposium on Intelligent Informatics, ISI 2023*, 2025. vol 389. pp 377-389. Springer, Singapore.

23. Kuznetsov, O., Smirnov, O., Mormul, M., Kotukh, Y., Zvieriev, V. «Comparative Research on Cryptocurrency Efficiency: An Objective Analysis of Key Metrics». *International Journal of Computing* 23(4), 2024. pp. 563-573.

24. Kuznetsov O., Frontoni E., Kuznetsova K., Smirnov O., Kostenko V. «Blockchain applications in metaverse environments: new horizons». *Advanced Metaverse Wireless Communication Systems.* pp. 255-293. 2024.

25. Kuznetsov, O., Frontoni, E., Chevardin, V., Smirnov, O., Imoize, A.L. «Advancing metaverse security with cryptographic innovations». *Advanced Metaverse Wireless Communication Systems.* pp 351-386. 2024.

26. Kuznetsov O., Frontoni E., Kuznetsova Y., Smirnov O., Moskovchenko I. «Trust-Based Security Architecture for Edge Computing: A Simulation Study of

Dynamic Trust Evolution and Attack Detection». *CEUR Workshop Proceedings,* 2024, 3909, pp. 227–241.

27. Kuznetsov, O., Frontoni, E., Kryvinska, N., Chevardin, V., Smirnov, O. «Wireless Network Encryption Stream Ciphers, Computational Modeling, and Security Analysis». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 379–402.

28. Kuznetsov, O., Frontoni, E., Kryvinska, N., Smirnov, O., Imoize, G.L. «Computational Modeling of Enhanced Spread Spectrum Codes for Asynchronous Wireless Communication». *Computational Modeling and Simulation of Advanced Wireless Communication Systems*, 2024, pp. 403–447.

29. Ткаченко, О., Ільєнко, А., Улічев, О., Мелешко, Є., Смірнов, О. «Правові засади поширення інформаційних впливів в соціальних мережах». *Електронне фахове наукове видання «Кібербезпека: освіта, наука, техніка»,* 2024. № 2(26), С. 170–188.

30. Смірнова Т.В., Конопліцька-Слободенюк О.К., Буравченко К.О., Смірнов С.А., Кравчук О.В., Козірова Н.Л., Смірнов О.А. «Дослідження технологій забезпечення кібербезпеки хмарних сервісів IaaS, PaaS та SaaS». *Кібербезпека: освіта, наука, техніка.* 2024. №4(24), С. 6-27.

31. Вінтенко, Б., Миронець, І., Смірнов, О., Кравчук, О., Козірова, Н., Савеленко, Г., Коваленко, А. «Дослідження вимог та аналіз кібербезпеки програмного забезпечення інформаційно-керуючих систем АЕС, важливих для безпеки». *Кібербезпека: освіта, наука, техніка.* 2024. №3(23), С. 111-131.

32. Kuznetsov, O., Frontoni, E., Kandiy, S., Smirnova, T., Prokopov, S., Bilanovych, A. «New Cost Function for S-boxes Generation by Simulated Annealing Algorithm». Lecture Notes on Data Engineering and Communications Technologies, 2023. vol 180. pp. 310-320. Springer, Cham.

33. Kuznetsov, O., Frontoni, E., Kandiy, S., Smirnov, O., Ulianovska, Y., Kobylianska, O. «Heuristic Search for Nonlinear Substitutions for Cryptographic

Applications». *Lecture Notes on Data Engineering and Communications Technologies,* 2023. vol 180. Springer, Cham. pp. 288-298.

34. Kuznetsov, O., Kuznetsova, Y., Smirnov, O., Kostenko, O., Zvieriev, V. «Evaluating Hashing Algorithms in the Age of ASIC Resistance». *CEUR Workshop Proceedings*, 2023, 3628, pp. 93-105.

35. Kuznetsov O., Frontoni E., Kuznetsova Ye., Smirnov O., Chevardin V. «Achieving Enhanced Security in Biometric Authentication: A Rigorous Analysis of Code-Based Fuzzy Extractor». *CEUR Workshop Proceedings,* Volume 3624, 2023, pp. 330-339.

36. Smirnov, O., Sydorenko, V., Aleksander, M., Zhyharevych, O., Yenchev, S. «Simulation of the cloud IoT-based monitoring system for critical infrastructures». *CEUR Workshop Proceedings,* Volume 3530, 2023, pp. 256-265.

37. Kuznetsov, O., Kandiy, S., Frontoni, E., Smirnov, O. «Trade-offs in Post-Quantum Cryptography: A Comparative Assessment of BIKE, HQC, and Classic McEliece». *CEUR Workshop Proceedings,* Volume 3504, 2023, pp. 1-11.

38. Смірнов О.А. Козлов Я.О., Смірнова Т.В. «Дослідження застосування SIEM-систем для забезпечення кібербезпеки та захисту інформації». *II Міжнародна науково-практична Інтернет-конференція «Інновації та перспективні шляхи hозвитку інформаційних технологій (ІПШРІТ-2023)»* м.Черкаси 6 грудня 2023 року – Черкаси: ЧДТУ.– 2023. – С.251-252.

39. Козлов Я.О., Смірнова Т.В., Смірнов О.А. «Дослідження SIEM-систем для забезпечення кібербезпеки». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя кафедри кібербезпеки та програмного забезпечення,* м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 26.

40. Козлов Я.О., Козірова Н.Л., Смірнов О.А. «Дослідження структури та принципу роботи SIEM-системи». *VII міжнародна науково-практична конференція "Інформаційна безпека та комп'ютерні технології" до 30-ти річчя*

кафедри кібербезпеки та програмного забезпечення, м. Кропивницький. 1 листопада 2023 р. – Кропивницький: ЦНТУ. – 2023. – С. 59.

41. Вінтенко Б.Ю., Смірнов О.А., Коваленко О.В., Смірнов С.А., Коваленко А.С. «Дослідження нормативних документів та галузевих стандартів розробки програмного забезпечення комп'ютерних систем управління АЕС, важливих для безпеки». *Системи управління, навігації та зв'язку*, 2023, вип. 2(72), С. 170-178.

42. Смірнова Т.В., Гнатюк С.О., Бердибаєв Р.Ш., Сидоренко В.М., Жигаревич О.К., «Система корелювання подій та управління інцидентами кібербезпеки на об'єктах критичної інфраструктури». *Кібербезпека: освіта, наука, техніка*, №*3*(19), 2023, С. 176-196.

43. Smirnov, O., Neskorodieva, T., Fedorov, E., Rudakov, K., Neskorodieva, A. «Method Detection Audit Data Anomalies on Basis Restricted Cauchy Machine» *CEUR Workshop Proceedings,* Volume 3187, 2022,

44. Smirnov, O., Lakhno, V., Akhmetov, B., Chubaievskyi, V., Khorolska, K., Bebeshko, B. «Selection of a Rational Composition of İnformation Protection Means Using a Genetic Algorithm». *In: Rajakumar, G., Du, KL., Vuppalapati, C., Beligiannis, G.N. (eds) Intelligent Communication Technologies and Virtual Mobile Networks. Lecture Notes on Data Engineering and Communications Technologies*, vol 131. 2023. Springer, Singapore. pp. 21-34.

45. Smirnov O.A., Al-Oraiqat A.M., Ulichev O.S., Meleshko Ye.V., Al-Rawashdeh H.S., Polishchuk L.I. «Modeling strategies for information influence dissemination in social networks». *Journal of Ambient Intelligence and Humanized Computing* Volume 13, Issue 5. Springer, Cham. 2022, pp. 2463-2477.

46. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Смірнов С.А., Поліщук Л.І., «Дослідження стійкості до диференціального криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку,* 2022, № 3(69). С. 93-98.

47. Смірнов О.А., Смірнова Т.В., Якименко Н.М., Поліщук Л.І., Смірнов С.А. «Дослідження статистичної стійкості та швидкісних характеристик запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Вісник Хмельницького національного університету. Серія: «Технічні науки»,* № 2 (307). С. 46-52. 2022.

48. Смірнов О.А., Смірнова Т.В., Константинова Л.В., Смірнов С.А., Якименко Н.М., «Дослідження стійкості до лінійного криптоаналізу запропонованої функції гешування удосконаленого модуля криптографічного захисту в інформаційно-комунікаційних системах» *Системи управління, навігації та зв'язку*, 2022, № 1(67). С. 84-89.

49. Смірнов О.А., Конопліцька-Слободенюк О.К., Смірнов С.А., Буравченко К.О., Смірнова Т.В., Книшук А.В. «Вступ до кібербезпеки»: навчальний посібник – Кропивницький: ЦНТУ – 2022. – 968 с.

50. Теорія та практика сучасного інформаційно-психологічного протиборства: навчальний посібник / [В.М. Петрик, С.О. Гнатюк, М.М. Присяжнюк та ін.]; за заг. ред. С.О. Гнатюка, В.М. Петрика та О.А Смірнова. – Полтава, 2022. – 334 с.

51. Smirnov O., Kuznetsov A., Zhora V., Onikiychuk A., Pieshkova O. «Hiding Messages in Audio Files Using Direct Spread Spectrum». *11th IEEE International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications, IDAACS 2021,* Cracow, Poland, 22-25 September 2021. P. 414-418

52. Smirnov O., Kuznetsov A., Lokotkova I., Kuznetsova T., Florov S., Lebid O. «Using Orthogonal Signals to Hide Information in Images». *4 IEEE International Conference on Advanced Information and Communication Technologies (AICT) – 2021*, Lviv, Ukraine, September 21-25, 2021. P. 255-260.

53. Smirnov O., Kuznetsov A., Girzheva O., Kiian A., Nakisko O., Kuznetsova T. «Advanced Code-Based Electronic Digital Signature Scheme». *2020*

IEEE International Conference on Problems of Infocommunications Science and Technology, PIC S and T 2020, Kharkiv, 6 October 2020-9 October 2020, P. 358-362.

54. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova K. «Data hiding scheme based on spread sequence addressing». *CEUR Workshop Proceedings* Volume 2805, 2020, Pages *44-58.*

55. Smirnov, O., Kuznetsov, A., Potii, O., Poluyanenko, N., Stelnyk, I., Mialkovsky, D. «Combining and filtering functions in the framework of nonlinear-feedback shift register». *International Journal of Computing*; 2020, Volume 19, Issue 2 – Research Institute for Intelligent Computer Systems – 2020. – P. 247-256.

56. Smirnov O., Kuznetsov A., Kiian A., Kuznetsova T. «Non-binary constant weight coding technique». *CEUR Workshop Proceedings*. Volume 2740, 2020, Pages 102-114.

57. Smirnov O., Alimseitova Zh., Adranova A., Akhmetov B., Lakhno V., Zhilkishbayeva G. «Models and algorithms for ensuring functional stability and cybersecurity of virtual cloud resources». *Journal of theoretical and applied information technology* Vol.98. No 21, 2020, P. 3334-3346.