МІНІСТЕРСТВО ОБОРОНИ УКРАЇНИ

ВІЙСЬКОВИЙ ІНСТИТУТ ТЕЛЕКОМУНІКАЦІЙ ТА ІНФОРМАТИЗАЦІЇ

КАФЕДРА КІБЕРЗАХИСТУ

**ЗВІТ**

про виконання практичної роботи:

з дисципліни «Прикладна криптологія»

**КОМП’ЮТЕРНИЙ ПРАКТИКУМ №3**

**ПРОГРАМНА РЕАЛІЗАЦІЯ ШИФРУВАЛЬНИКА ФАЙЛІВ ШИФРОМ AES**

**Виконав:** курсант навчальної групи 314

солдат \_\_\_\_\_\_\_\_\_\_\_\_\_Петренко К.В.

28.06.2024

**Перевiрив:** викладач кафедри № 33

капітан \_\_\_\_\_\_\_\_\_\_\_\_Прийма О.О.

Київ – 2024

1.**Програмні коди**

Усі програмні коди розміщнні на GitHub: [KuMuLaTiV4iK/PK-3\_Stiller (github.com)](https://github.com/KuMuLaTiV4iK/PK-3_Stiller)

# **Виконані завдання**

Завдання 1. Написати програму, яка виконуює шифрування та розшифрування файлів шифром AES. Програма повинна шифрувати і розшифровувати файли вказані користувачем у консолі.

У класі FolderEncryptor було реалізовано шифрування і розшифрування файлів шифром AES. Інтерфейс програми представлений зручним меню у якому є всі необхідні функції для операцій над файлами. Всі результати шифрування відображаються в консолі.

Результат роботи:

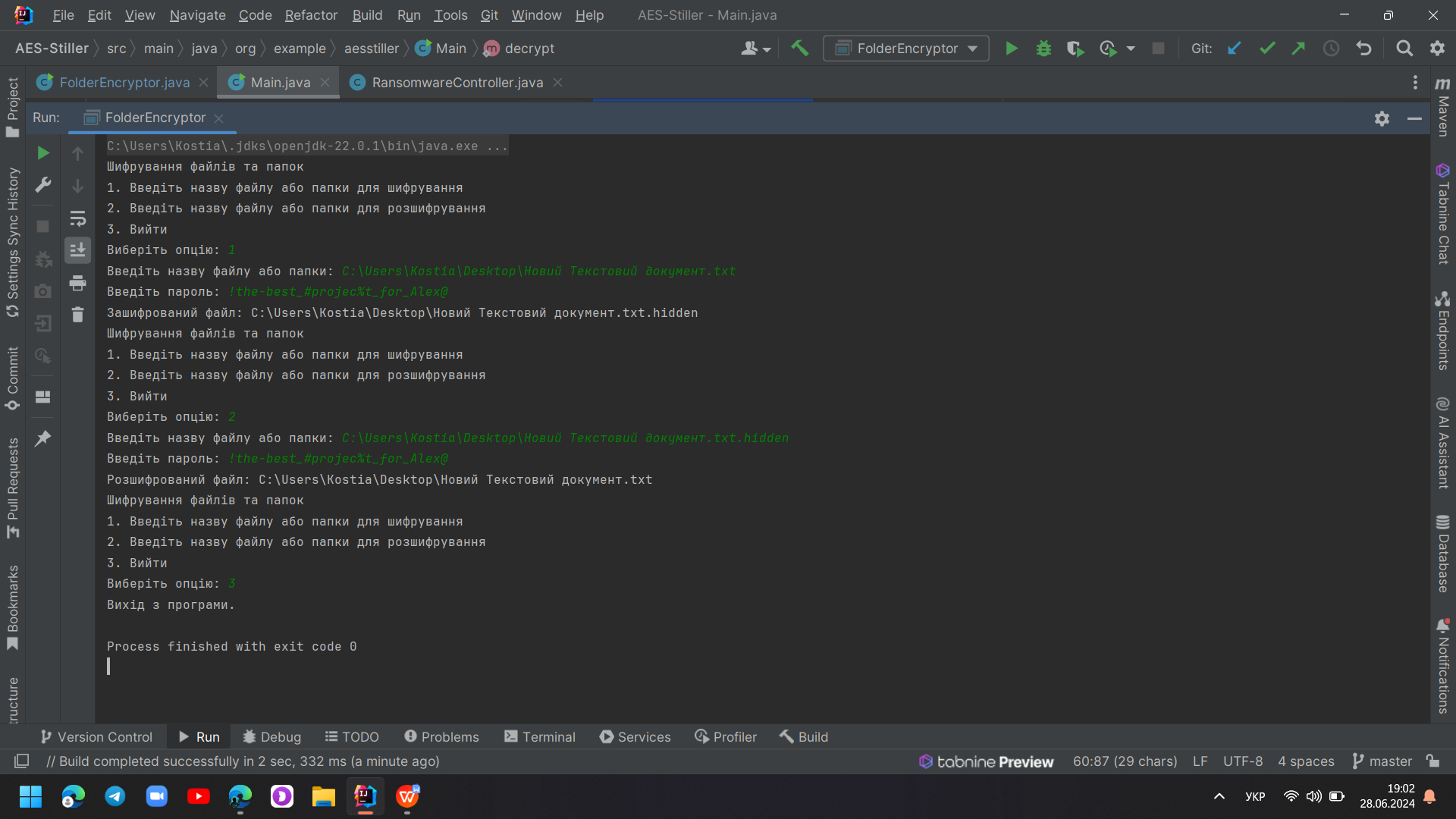


Рисунок 1

Завдання 2. Написати програму вимагач, яка замаскована під провідник і шифрує всі вибрані жертвою файли та видає вікно з функціоналом перевірки оплати за розшифрування файлів.

Проект реалізований з використанням платформи JavaFX. Функціонал програми реалізований у класі Main та RansomwareController, який керує файлом Ransomware.fxml.

У програмі релізовано шифрування фалів за допомогою алгоритму AES, які вибираються в фальшивому вікні провідника.

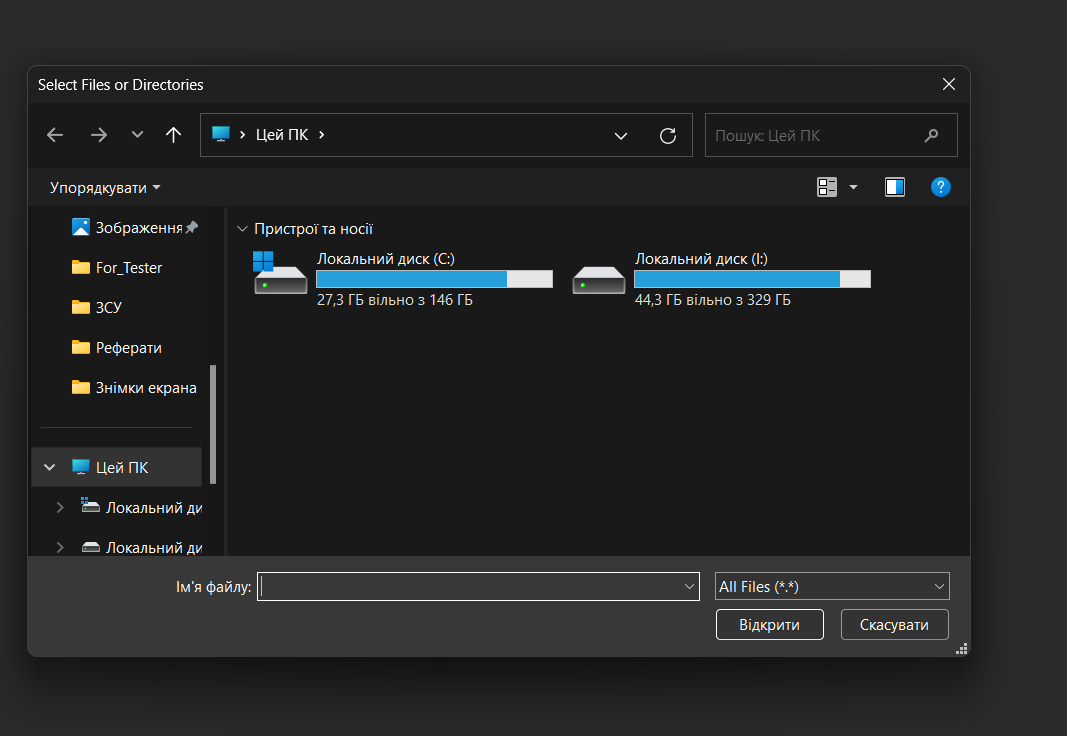


Рисунок 2. Вікно для вибору файлів

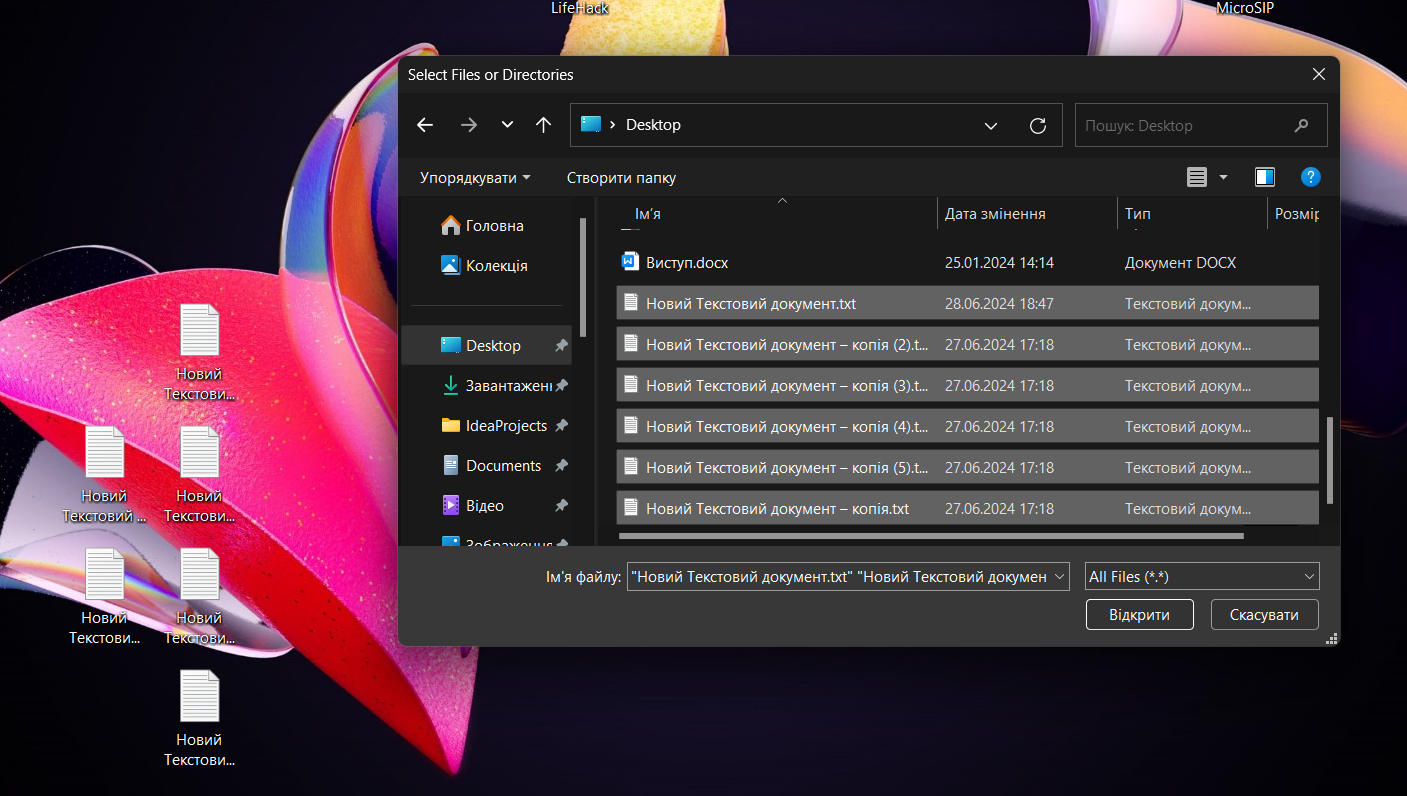


Рисунок 3. Вибираємо декілька файлів

Після натиснення на кнопку “Відкрити” відкривається вікно програми вимагача.

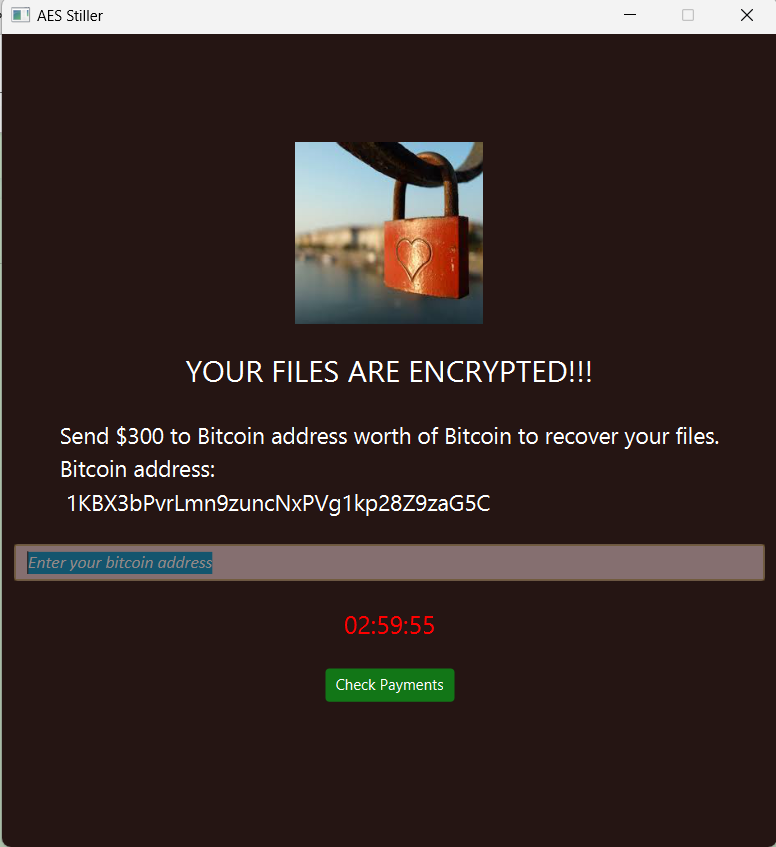


Рисунок 4. Вікно вимагача

Для того щоб розшифрувати вибрані файли жертва повинна відправити 300$ у Біткоіні на вказану вимагачем адресу.

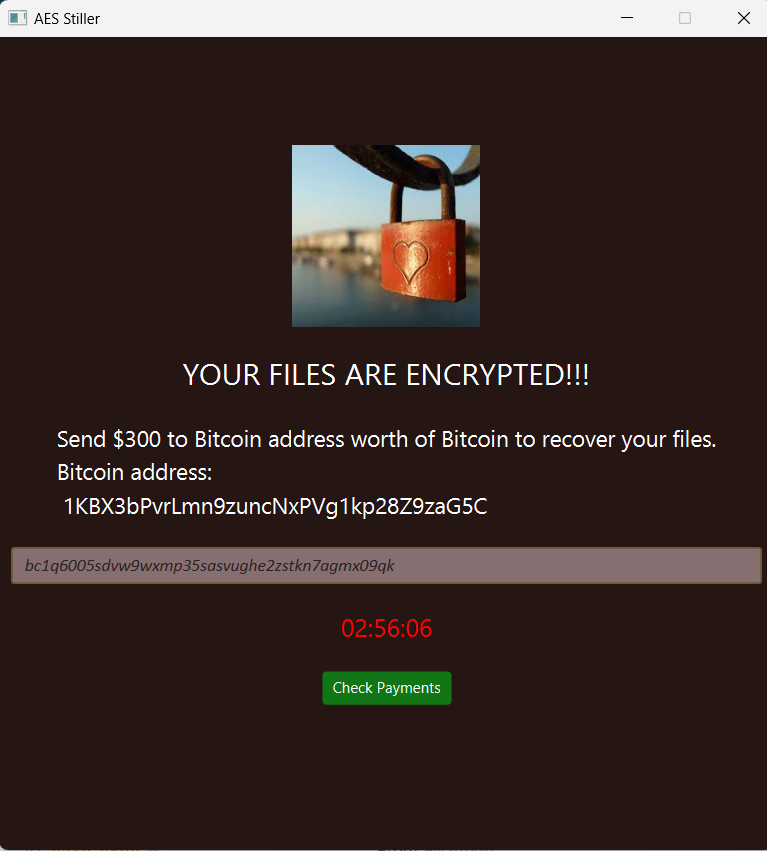


Рисунок 5. Вставляємо Біткоін адресу з якої відправили кошти

На платіж користувач має 7 спроб. При невірних діях та недостатній сумі переказу жертви йому про це повідомляється та зменшується кількість спроб. При коректній оплаті жертва отримує назад свої дані.

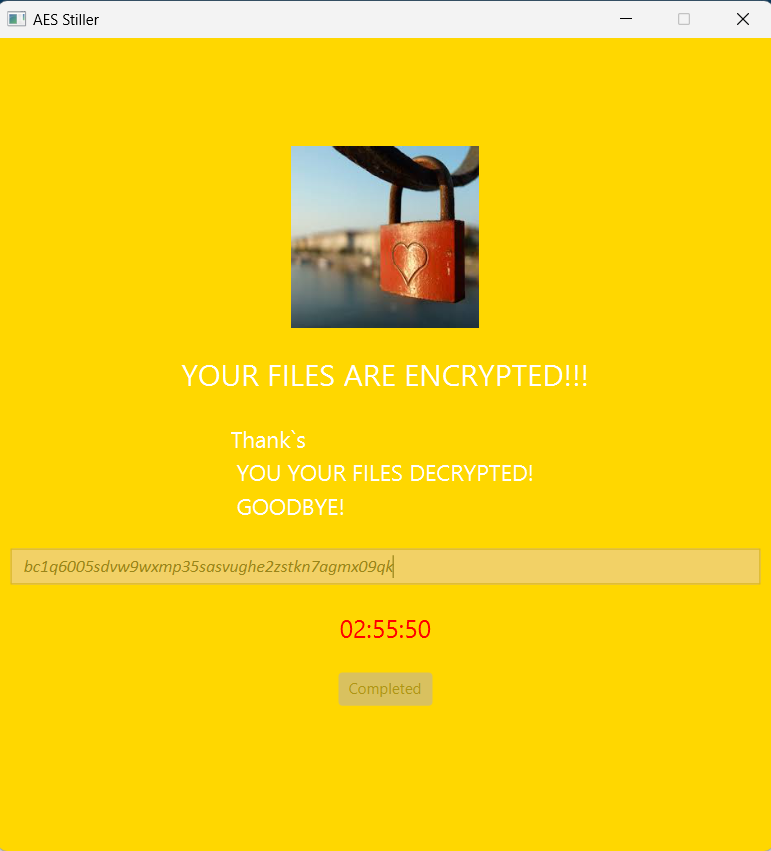


Рисунок 6. Успішна оплата

У класі Main реалізовано шифрування\розшифрування файлів з викликом методів з класу FolderEncryptor.

Ключові функції класу включають:

Вибір файлів для шифрування за допомогою діалогового вікна FileChooser, користувач може обирати файли або папки для шифрування.

Шифрування файлів - використовуючи клас FolderEncryptor, обрані файли шифруються з використанням заданого пароля, а результати зберігаються у списку encryptedFiles.

Ініціалізація інтерфейсу користувача - завантажує інтерфейс користувача з файлу FXML (Ransomware.fxml) та встановлює його в головному вікні додатка.

Блокування закриття додатка - додаток блокує можливість закриття головного вікна, запобігаючи втраті шифрованих файлів.

Розшифрування файлів: Метод decrypt дозволяє розшифрувати файли, які були зашифровані раніше, використовуючи той самий пароль.

Клас RansomwareController є контролером для JavaFX додатка, що реалізує функціонал програми-вимагача. Основні функції класу:

Ініціалізація UI компонентів - завантаження зображення замка, відображення повідомлення з Bitcoin адресою, та налаштування дій для кнопки перевірки платежу.

Відлік часу - запуск таймера, який відраховує три години, після чого повідомлення змінюється на "Time's up! YOU LOST YOUR FILES!" і заблоковує кнопку перевірки платежу.

Перевірка платежу - перевірка, чи надійшов необхідний платіж на вказану Bitcoin адресу, та оновлення повідомлення відповідно до результату перевірки.

Шифрування та розшифрування файлів - виклик методів для шифрування або розшифрування файлів після успішної перевірки платежу.

Затримка перед виходом - затримка перед закриттям програми для надання користувачеві можливості побачити кінцеве повідомлення.

Клас FolderEncryptor забезпечує шифрування та розшифрування файлів та папок з використанням алгоритму AES. Основні функції класу:

Шифрування файлів та папок:

Метод encryptFile(File file, String password) - шифрує вказаний файл з використанням заданого паролю та зберігає зашифрований файл з розширенням .hidden.

Метод encryptFolder(File folder, String password): Рекурсивно шифрує всі файли у вказаній папці.

Розшифрування файлів та папок:

Метод decryptFile(File file, String password): Розшифровує вказаний файл з використанням заданого паролю та відновлює його оригінальну назву, видаляючи розширення .hidden.

Метод decryptFolder(File folder, String password): Рекурсивно розшифровує всі файли у вказаній папці.

Генерація ключа шифрування:

Метод generateKey(String password): Генерує ключ шифрування з використанням заданого паролю, конвертуючи його у 32-байтовий масив для AES-256.

Шифрування та розшифрування байтових масивів:

Метод encrypt(byte[] input, String password, byte[] iv): Шифрує вхідний байтовий масив з використанням заданого паролю та вектора ініціалізації (IV).

Метод decrypt(byte[] input, String password, byte[] iv): Розшифровує вхідний байтовий масив з використанням заданого паролю та вектора ініціалізації (IV).

Таймер, реалізований через Timeline, відраховує час до виконання вимог та змінює інтерфейс у випадку завершення таймера.

Кнопка для перевірки транзакцій інтерактивна та дозволяє користувачам вводити Bitcoin-адресу для перевірки наявності платежу.

Перевірка Bitcoin-транзакцій:

Метод checkBitcoinTransaction у класі RansomwareController здійснює запит до API для перевірки транзакцій на вказану Bitcoin-адресу.

Функція getBitcoinPrice отримує поточний курс Bitcoin для конвертації отриманих коштів у долари США.

Якщо сума транзакцій перевищує вказану суму, програма відображає повідомлення про успішне виконання вимог та розшифровує файли.

Обробка виключень та повідомлення про помилки роблять використання програми більш надійним та інформативним.

**Висновок:** за результатами виконаних робіт було успішно реалізовано Проект, що представляє собою комплексне авторське рішення програми вимагача і є потужним інструментом для демонстрації шифрування та розшифрування файлів та папок з використанням алгоритму AES. Надає приклад створення простого, але функціонального графічного інтерфейсу реалізований на платформі JavaFX для взаємодії з користувачами. Додатково, можливість перевірки Bitcoin-транзакцій додає реалізму до імітації програм-вимагачів, що може бути використано в освітніх цілях для демонстрації небезпеки таких програм.