УДК 004.056.55

О. М. Дзюбенко
Г. О. Черноволик

ВЕБСИСТЕМА ДЛЯ ЗАБЕЗПЕЧЕННЯ КРИПТОГРАФІЧНОЇ БЕЗПЕКИ ФАЙЛІВ

Вінницький національний технічний університет, м. Вінниця

Анотація *Розроблено програмний застосунок вебсистеми для криптографічної безпеки файлів. Система дозволяє користувачам завантажувати файли, обирати метод шифрування, задавати ключі, виконувати шифрування або розшифрування даних, а також перевіряти цілісність інформації за допомогою геш-функцій. Передбачено підтримку як симетричних, так і асиметричних алгоритмів шифрування.*

Систему розроблено з використанням мови програмування JavaScript, середовища Visual Code, бібліотеки NodeJS та інтерфейсних технологій HTML, CSS, JavaScript, TypeScript.

Ключові слова: веб-застосунок, шифрування, криптографія, PBKDF2, захист файлів, бінарний код, конфіденційність, пароль, інтерфейс, диск.

Abstract *Software application of a web system for cryptographic file protection have been developed. The system allows users to upload files, select an encryption method, set keys, perform encryption or decryption of data, and verify the integrity of information using hash functions. Support for both symmetric and asymmetric encryption algorithms is provided.*

The system was developed using the JavaScript programming language, the Visual Studio Code environment, the Node.js library, and front-end technologies such as HTML, CSS, and JavaScript, TypeScript.

Keywords: web application, encryption, cryptography, PBKDF2, file protection, binary code, confidentiality, password, interface, disk.

Вступ

Сучасний етап розвитку інформаційних технологій характеризується стрімким зростанням обсягів цифрової інформації, яка передається, обробляється та зберігається в глобальних комп'ютерних мережах. З появою нових форм зберігання і обробки даних, таких як хмарні технології, вебзастосунки та великі дані, виникає безліч проблем, пов'язаних з безпекою цієї інформації. Зокрема, однією з основних загроз є несанкціонований доступ до інформації, її зміна або крадіжка. В умовах постійного зростання цифрових даних і все більш активного використання хмарних сервісів для зберігання та передачі файлів необхідно забезпечити високий рівень захисту цієї інформації. Веб-сервіси, що надають доступ до файлів, можуть бути піддані різноманітним загрозам, що ставить питання про необхідність реалізації ефективних методів захисту даних, таких як криптографія [1].

Криптографія — це наука про методи захисту інформації від несанкціонованого доступу, змін і крадіжки через її перетворення в зашифровану форму. Цей процес дозволяє забезпечити конфіденційність, цілісність, автентичність та невідмовність переданої інформації. Розвиток криптографічних алгоритмів надає можливість створювати високонадійні системи захисту даних, що мають важливе значення для збереження безпеки в умовах зростаючої кількості кіберзагроз.

До недавнього часу криптографічний захист даних в основному використовувався великими компаніями та урядовими структурами, і доступ до подібних рішень був обмежений. Однак з розвитком веб-технологій, зокрема в напрямку надання користувачам можливості шифрування та дешифрування даних без необхідності встановлення додаткового програмного забезпечення, виникла потреба в створенні нових інтуїтивно зрозумілих та доступних криптографічних веб-сервісів. Це дозволяє простим користувачам швидко і безпечно виконувати операції з даними, захищаючи їх від несанкціонованого доступу або змін.

Загрозливі тенденції щодо безпеки даних вимагають високого рівня захисту навіть для звичайних користувачів. Поява таких технологій, як хмарні сервіси, призводить до нових викликів для забезпечення конфіденційності. Більшість традиційних методів захисту, зокрема паролі або обмежений доступ до серверів, є недостатньо ефективними для захисту файлів у таких середовищах. Тому все

більшу значущість набувають рішення, що забезпечують криптографічний захист, дозволяючи користувачам самостійно шифрувати і дешифрувати свої дані.

Актуальність цієї теми також підкреслюється зростанням популярності використання веб-застосунків для виконання різноманітних задач, що включають зберігання, обробку та передачу даних. Веб-технології забезпечують високу доступність сервісів, але з цим виникає проблема забезпечення належного рівня захисту. Тому існує необхідність у створенні зручних і безпечних веб-застосунків, які дозволяють здійснювати криптографічний захист даних без потреби в додатковому програмному забезпеченні та забезпечують доступ до функцій шифрування для широкого кола користувачів.

Результати дослідження

Розроблено вебзастосунок для криптографічного захисту файлів, реалізований із використанням сучасних вебтехнологій React, TypeScript, HTML, CSS та Web Crypto API. Система дозволяє користувачу завантажувати файл через інтерфейс браузера, вводити код доступу, виконувати шифрування та дешифрування даних із використанням алгоритму AES-256-GCM, а також отримувати результат у вигляді зашифрованого або відновленого файлу. Формування криптографічного ключа здійснюється на основі введеного пароля за допомогою алгоритму PBKDF2, що підвищує стійкість системи до атак перебору.

Інтерфейс вебзастосунку побудовано за компонентним принципом. Основна робоча область відповідає за вибір файлу, введення коду доступу, вибір режиму роботи та запуск операцій шифрування або дешифрування. Додатково передбачено інформаційні блоки, які пояснюють користувачу принцип роботи системи та особливості криптографічної обробки файлів. Усі операції виконуються безпосередньо у браузері користувача, що дозволяє уникнути передавання відкритих даних на сервер і підвищує рівень конфіденційності.

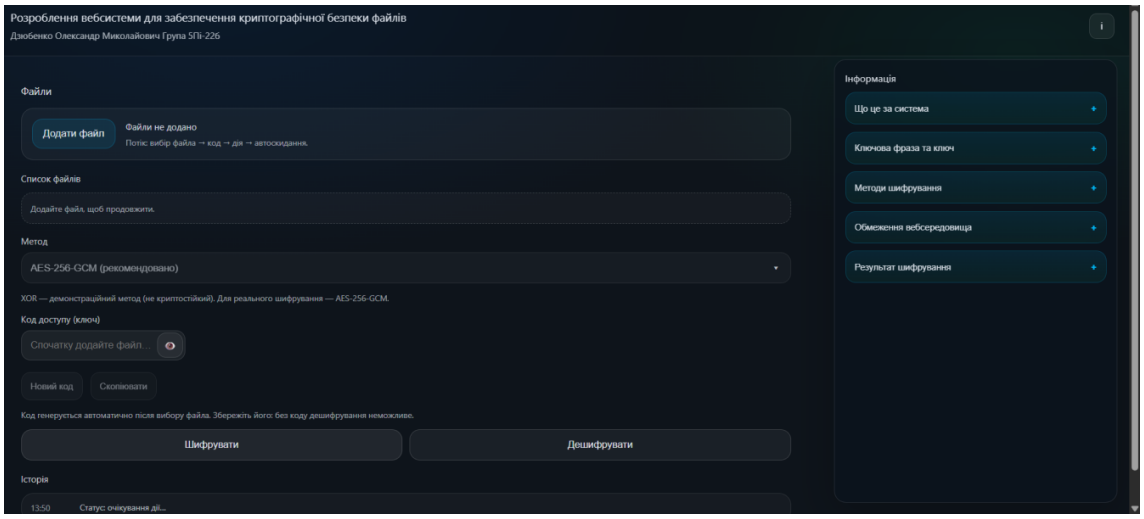

Рисунок 1 – Інтерфейс вебзастосунку у браузері

Криптографічна логіка винесена в окремий програмний модуль, який відповідає за формування ключа, генерацію випадкової сольової послідовності, створення вектора ініціалізації, виконання шифрування та дешифрування файлів. Після шифрування система формує захищений контейнер, до складу якого входять службові параметри, необхідні для подальшого коректного дешифрування: сіль, вектор ініціалізації та зашифрований вміст файлу. Такий підхід забезпечує структуроване збереження даних і можливість їх відновлення лише за умови введення правильного коду доступу.

Модуль роботи з файлами забезпечує зчитування вхідних даних у вигляді байтового масиву, передавання їх до криптографічного модуля та формування результату для завантаження користувачем. Для цього використовується механізм File API браузера та об’єкт Blob, за допомогою якого створюється вихідний файл після завершення операції. У разі введення неправильного пароля або пошкодження захищеного контейнера система припиняє дешифрування та повідомляє користувача про помилку.

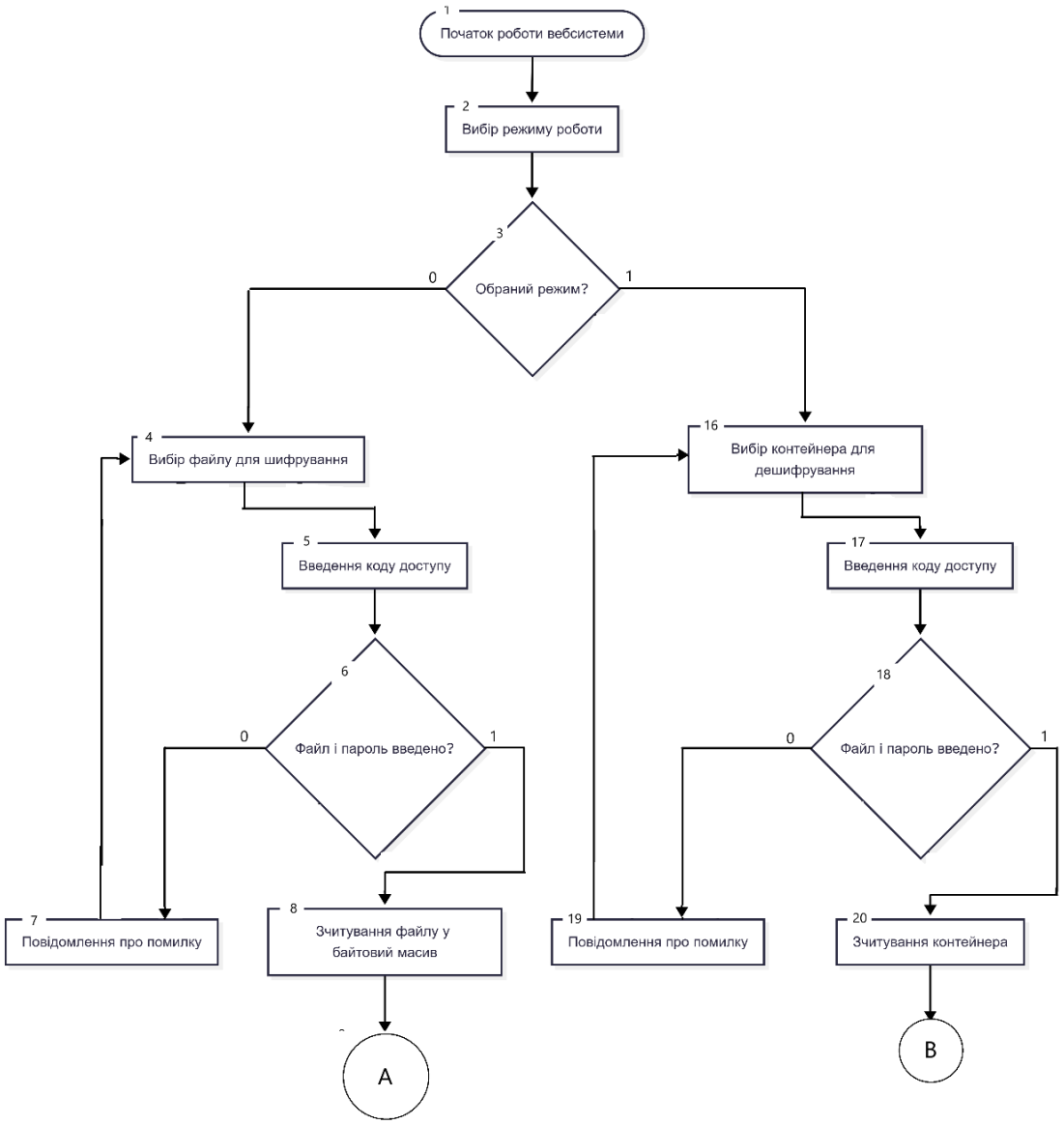

Рисунок 2 – Схема роботи модуля шифрування файлів у вебзастосунку

Продовження діаграми діяльності для гілки шифрування (позначеної як точка переходу A) наведено на рисунку 3. У даній частині деталізовано внутрішні етапи криптографічної обробки даних, зокрема генерацію випадкового значення salt, формування криптографічного ключа за алгоритмом PBKDF2, створення вектора ініціалізації (IV) та безпосереднє виконання шифрування за алгоритмом AES-256-GCM. Після цього здійснюється формування захищеного контейнера, який містить усі необхідні параметри для подальшого дешифрування, а також завантаження зашифрованого файлу користувачем.

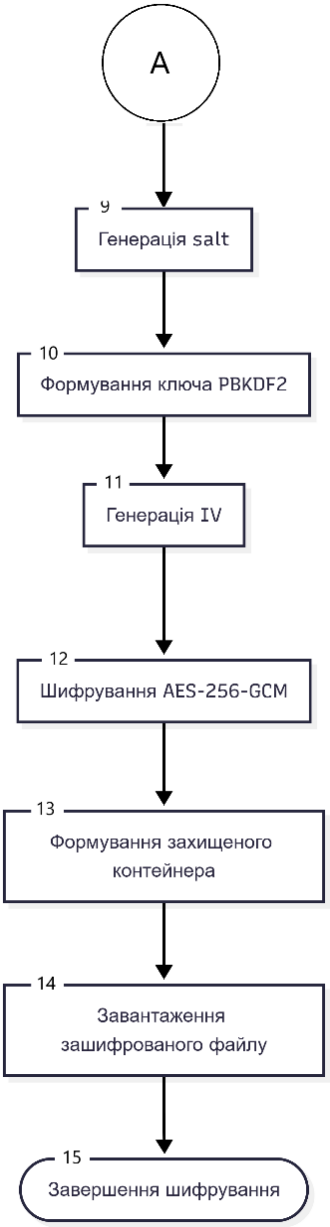

Рисунок 3 – Діаграма діяльності алгоритму роботи вебсистеми, продовження гілки A

Продовження діаграми для гілки дешифрування (позначеної як точка переходу B) представлено на рисунку 4. У цьому фрагменті відображено процес розбору захищеного контейнера з виділенням компонентів salt, IV та ciphertext, повторне формування ключа за допомогою PBKDF2 та виконання дешифрування алгоритмом AES-256-GCM. Окрему увагу приділено перевірці автентичності даних, яка дозволяє виявити помилки, пов’язані з неправильним паролем або пошкодженим файлом. У разі успішної перевірки відбувається формування та завантаження відновленого файлу, що завершує процес дешифрування.

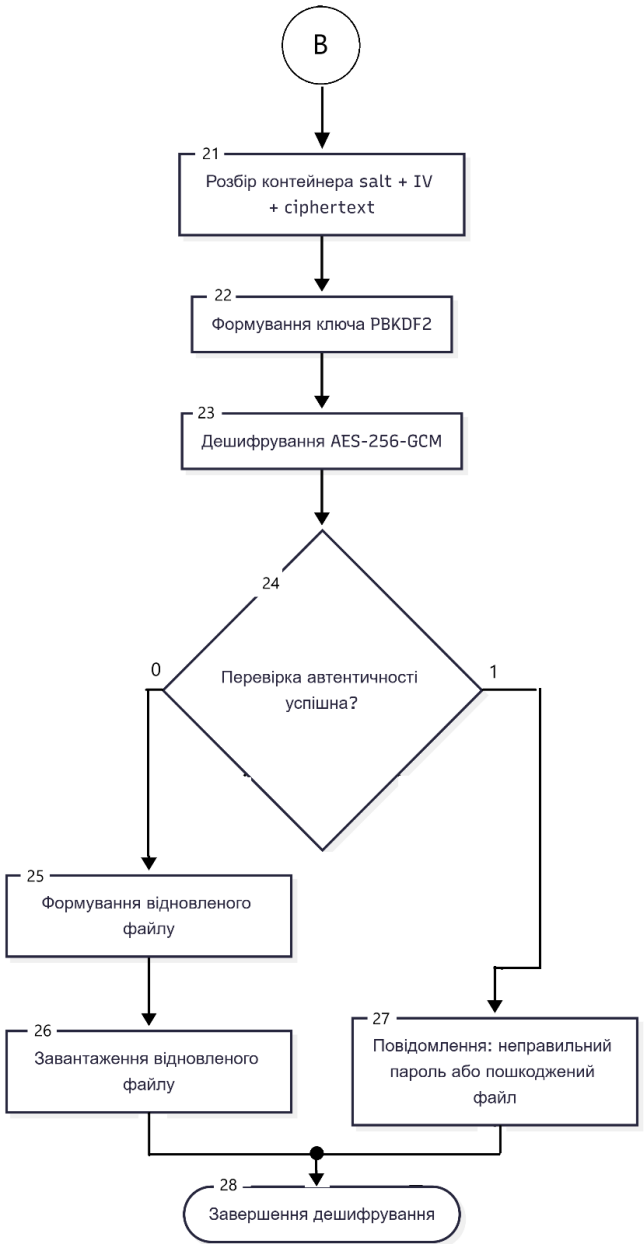

Рисунок 4 – Діаграма діяльності алгоритму роботи вебсистеми, продовження гілки B

Тестування проведено в середовищі сучасних браузерів Google Chrome, Mozilla Firefox та Microsoft Edge на ОС Windows 10/11. У процесі перевірки було протестовано вибір файлів різних форматів, шифрування, дешифрування, обробку неправильного пароля, формування захищеного контейнера та завантаження результату. Застосунок показав стабільну роботу, коректну обробку даних і зручність використання для кінцевого користувача. Особлива увага приділена забезпеченню конфіденційності файлів, локальності виконання криптографічних операцій та простоті інтерфейсу.

Висновки

Розглянуто підходи до створення вебзастосунку для криптографічного захисту файлів із використанням сучасних вебтехнологій та клієнтської обробки даних у браузері. Запропоновано рішення, що поєднує зручний користувацький інтерфейс із механізмами безпечного шифрування та дешифрування файлів без передавання відкритих даних на сервер. Реалізовано модулі для вибору та обробки файлів, формування криптографічного ключа на основі пароля за допомогою PBKDF2, шифрування і дешифрування з використанням алгоритму AES-256-GCM, а також створення та розбору захищеного контейнера файлу. Інтерфейс вебзастосунку розроблено з використанням React та TypeScript, що забезпечує зручність взаємодії користувача із системою. Результати тестування

підтвердили коректність роботи основних функцій, стабільність обробки файлів, правильність перевірки коду доступу та можливість відновлення даних лише за умови введення правильного пароля. Розроблене рішення може бути використане як практичний інструмент для захисту конфіденційних файлів, а також як основа для подальшого розширення функціональності вебсистеми криптографічної безпеки.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Romaniuk O. N., Zaharchuk M. D., Korobeinikova T. I. Utilizing Three-Dimensional Graphics in the Medical Field. Youth in Science: Research, Problems, Perspectives (MN-2020) : materials of the Youth Scientific and Practical Internet Conference of Students, Postgraduates, and Young Scientists. Vinnytsia : VNTU, 2021. P. 3.
2. Сучасні алгоритми шифрування : вебсайт. URL: https://www.kingston.com/ua/blog/data-security/what-is-encryption (дата звернення: 27.03.2025).

Дзюбенко Олександр Миколайович – студент групи 5ПІ-22Б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: shyrik3x@gmail.com

Черноволик Галина Олександрівна – доцент кафедри програмного забезпечення, Вінницький національний технічний університет, м. Вінниця

Dziubenko Oleksandr Mykolaiovych – student of group 5PI-22B, Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: shyrik3x@gmail.com

Chernovolyk Halyna Oleksandrivna – Associate Professor of the Department of Software, Vinnytsia National Technical University, Vinnytsia