УДК 004.056

Г. Е. Голуб

А.О. Єфімченко

Ю.В. Баришев

БІБЛІОТЕКА ДЛЯ РЕЄСТРУВАННЯ КОРИСТУВАЧІВ

Вінницький національний технічний університет

Анотація

Створено бібліотеку для реєстрації користувачів з функцією гешування паролів за допомогою алгоритму SHA-256, реалізація якої виконана з використанням мови програмування Java. Проведено аналіз масштабованості бібліотеки з одночасним використанням інших алгоритмів гешування.

Ключові слова: *алгоритми гешування, програмна бібліотека, паролі, автентифікація користувачів, кібербезпека*

Abstract

A library for user registration with the function of password hashing using the SHA-256 algorithm has been created, which was implemented using the Java programming language. The scalability of the library is analyzed with the simultaneous use of other hashing algorithms.

Keywords: *hashing algorithms, software library, passwords, user authentication, cybersecurity*

Вступ

З метою автентифікації користувачів використовуються фактори автентифікації, найпопулярнішим з яких є фактори на основі знання. Найбільш поширеним фактором автентифікації на основі знання багатокористувацьких застосунків, як-то веб-застосунки, є паролі. Водночас паролі зберігати у відкритому вигляді небезпечно, тому потрібно їх захищати. Одним з методів такого захисту паролів є гешування. Загалом, гешування — це процес односпрямованого перетворення даних довільної довжини в дані фіксованої довжини, які називають геш-значенням повідомлення [1]. При цьому в багатокористувацьких застосунках виникає актуальне завдання реєстрування користувачів, в межах якої відбувається зчитування еталонних значень факторів автентифікації та зберігання їх у захищеному вигляді.

Метою дослідження є покращення захисту автентичності користувачів шляхом розроблення програмної бібліотеки, яку можливо використовувати при розробці застосунків.

Завданнями дослідження є такі: проєктування архітектури бібліотеки; її розробка та тестування.

Результати дослідження

Утворена програмна бібліотека зосереджена на безпеці обробки вхідних даних користувачів. Основні функціональні можливості бібліотеки описуються низкою методів: перетворення вхідних даних в геш-значення; експортування вихідних даних у текстовий файл; імпортування вхідних даних для подальшого редагування. Для розробки бібліотеки було обрано мову об’єктно-орієнтованого програмування Java та вбудованого API — Java Cryptography API [2], а саме класу MessageDigest.

В межах доведення концепції в бібліотеці було реалізоване використання алгоритму гешування SHA-256, проте структура файлів для зберігання (рис. 1) та інтерфейс обраної бібліотеки Java Cryptography API надає можливість масштабувати застосування алгоритмів гешування для утворення можливості одночасного використання декількох алгоритмів гешування.

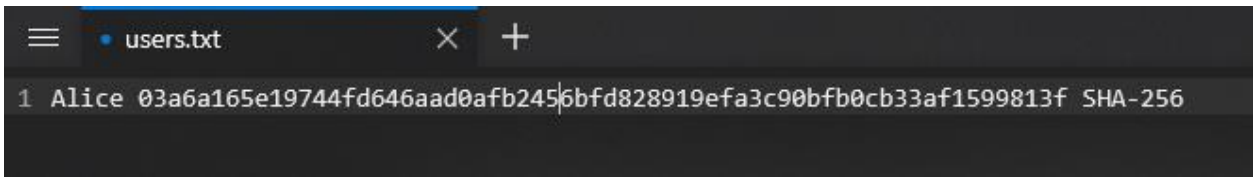

Рисунок 1 – Вигляд файлу зі збереженим геш-значенням пароля

Метод, який реалізує процес гешування називається hashPassword. У якості аргументів, метод приймає об’єкти типу String, які означають інформацію, над якої необхідно виконати операцію гешування, а також алгоритм гешування. За замовчуванням, значення алгоритму гешування дорівнює “SHA-256”. Метод hashPassword відповідає за роботу з паролем та збереження його у захищеному вигляді перед зберіганням чи передаванням.
Нижче описано ключові кроки реалізації.

1. Ініціалізація об'єкта дайджесту повідомлень. Спочатку створюється об'єкт, який призначений для імплементування алгоритму гешування (за замовчуванням SHA256)
2. Обробка вхідного рядка. Вхідний рядок (пароль) перетворюється у масив байтів. Це необхідно, оскільки гешування відбувається над байтами даних.
3. Гешування даних. Викликається метод, який виконує операцію гешування та повертає геш-значення.
4. Форматування геш-значення. Створюється об'єкт, який використовується для перетворення масиву байтів геш-значення в рядок шістнадцяткових чисел. В циклі кожен байт форматується в двозначне шістнадцяткове число (з використанням %02x) і додається до об’єкту.
5. Повернення отриманого рядка. Після завершення перетворення, сформатоване шістнадцяткове преставлення конвертується в об’єкт типу String та повертається як результат методу
6. Обробка виняткових ситуацій. У випадку виникнення помилок обробки або виключних ситуацій, помилка відображається через функцію, що повертає рядок помилки.

Цей метод реалізації гешування даних забезпечує перетворення вхідного пароля в його геш-значення, використовуючи за замовчуванням алгоритм SHA-256, який є стандартним вибором для багатьох застосунків з огляду на його криптографічну стійкість та швидкість реалізації. Таким чином розроблена бібліотека може бути підключена до проектів та використана для реалізації реєстрування користувачів для формування бази геш-значень користувачів. При цьому масштабованість дозволяє змінювати алгоритми гешування під потреби конкретного проєкту.

Інтерфейси, що вбудовані у бібліотеку, пройшли низку тестів за методикою Black-box [3] — техніки тестування, за використанням якої внутрішня робота програмного забезпечення не відома тестувальнику. Тестувальник зосереджується лише на вхідних та вихідних даних програми.

Проведені тести з використанням з використанням вбудованого класу “System” у Java надали змогу оцінити швидкість роботи методу hashPassword. При виконанні одиничного синхронного запиту над вхідною строкою довжиною у 25 символів та алгоритмом гешування “SHA-256” — метод надає відповідь за 30 мс.

Висновки

Результатом розробки стала бібліотека для покращення захисту автентичності користувачів. Розроблену бібліотеку можливо налаштувати під умови кожного проєкту та впровадити як компонент сервісу, наприклад на основі мікросервісної архітектури або як загального компоненту реєстрації користувачів. Варто зазначити, що бібліотека має шляхи в удосконаленні розробки: одночасне використання різних алгоритмів гешування, наприклад — bcrypt, SHA-3, Argon2; розширення функціоналу із експортуванням утворених гешованих даних; оптимізація програмної структури бібліотеки.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. ДСТУ 7564:2014. Інформаційні технології. Криптографічний захист інформації. Функція гешування. З поправкою. – Чинний від 2015-04-01. Вид. офіц. – Київ : Ін-т інформ. технологій, 2015. – 39 с.

2. Jenkov J. Java Cryptography [Electronic resource] / Jakob Jenkov // Jenkov.com Tech & Media Labs - Resources for Developers, IT Architects and Technopreneurs. – Mode of access: https://jenkov.com/tutorials/java-cryptography/index.html (date of access: 01.05.2024). – Title from screen.
3. Black box testing - Software Engineering - GeeksforGeeks [Electronic resource] // GeeksforGeeks. – Mode of access: https://www.geeksforgeeks.org/software-engineering-black-box-testing/ (date of access: 01.05.2024). – Title from screen.

Голуб Георгій Едуардович — студент групи 1БС-22Б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: holub.heorhii@gmail.com.

***Єфімченко Анастасія Олексіївна* —** студентка групи 1БС-22Б, факультет інформаційних технологій та комп'ютерної інженерії, Вінницький національний технічний університет, м. Вінниця, e-mail: 14estasyf09@gmail.com.

Баришев Юрій Володимирович - к. т. н., доцент, доцент кафедри захисту інформації, Вінницький національний технічний університет, м. Вінниця, e-mail: yuriy.baryshev@vntu.edu.ua

Heorhii Holub — Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: holub.heorhii@gmail.com.

Anastasiia Yefimchenko — Faculty of Information Technologies and Computer Engineering, Vinnytsia National Technical University, Vinnytsia, e-mail: 14estasyf09@gmail.com

Yurii Baryshev — associate professor of the Information Protection Department, Vinnytsia National Technical University, Vinnytsia, e-mail: yuriy.baryshev@vntu.edu.ua