УДК 004.056.55, 004.414.2

**І. В. Осипенко**
**І. В. Богач**

# ЗАБЕЗПЕЧЕННЯ ІНФОРМАЦІЙНОЇ БЕЗПЕКИ МОБІЛЬНИХ ДОДАТКІВ НА FLUTTER

Вінницький національний технічний університет

***Анотація***

*У роботі висвітлено підходи до підвищення інформаційної безпеки мобільних застосунків, створених на платформі Flutter. Здійснено аналіз актуальних векторів атак, таких як несанкціонований доступ до коду, компрометація облікових даних та втручання у бізнес-логіку. Окрему увагу приділено практичним методам протидії цим загрозам: мінімізації впливу реверс-інжинірингу, шифруванню даних, забезпеченню цілісності програми та контролю середовища виконання.*

**Ключові слова:** Flutter, безпека мобільних додатків, зворотний інжиніринг, офускація, шифрування, антидебаг.

***Abstract***

*This work focuses on enhancing the information security of mobile applications developed with Flutter. It analyzes current attack vectors such as unauthorized code access, data breaches, and manipulation of business logic. Practical countermeasures are highlighted, including reverse engineering mitigation, data encryption, application integrity verification, and runtime environment control.*

**Keywords**: Flutter, mobile application security, reverse engineering, obfuscation, encryption, anti-debugging.

## Вступ

Інформаційна безпека мобільних застосунків є ключовим елементом сучасної цифрової екосистеми. Flutter як інструмент кросплатформної розробки здобув широку популярність, однак його особливості створюють нові виклики в контексті захисту від потенційних атак. Зокрема, об’єднання логіки додатку в одному шарі створює привабливу мішень для зловмисників.

Метою даної роботи є систематизація знань щодо актуальних кіберзагроз для Flutter-застосунків та висвітлення найефективніших методів їхньої нейтралізації.

## Результати дослідження

У контексті стрімкого розвитку мобільних технологій зростає і кількість кіберзагроз, що прямо впливають на конфіденційність, цілісність та доступність даних користувача. Flutter, як популярний фреймворк для кросплатформної розробки, потребує додаткових заходів захисту, зважаючи на його відкриту архітектуру та спільне використання єдиного коду для Android та iOS. Сучасний підхід до безпеки мобільних застосунків повинен передбачати комплексне впровадження як програмних, так і організаційних рішень.

Модель комплексного пакету безпеки

Згідно з дослідженням [1], доцільно реалізовувати багаторівневу систему захисту, яка базується на чотирьох функціональних компонентах:

Безпечна архітектура додатку

Flutter-додатки повинні розроблятись із врахуванням безпечної архітектури, яка передбачає чітке розділення відповідальностей між фронтендом та бекендом, шифрування критичних даних при передачі та зберіганні, а також суворе обмеження доступу до локальних ресурсів.

Наприклад, обробка платіжних даних, аутентифікація, генерація токенів повинні бути повністю перенесені на сервер, щоб уникнути компрометації бізнес-логіки на клієнтському боці.

Принципи CIA (Confidentiality, Integrity, Availability)

Конфіденційність: реалізується через використання алгоритмів шифрування (AES, RSA), Flutter Secure Storage, а також ізоляцію даних у sandbox-середовищі.

Цілісність: досягається шляхом хешування важливих об'єктів, перевірки контрольних сум ресурсів, а також автентифікації запитів за допомогою JWT.

Доступність: забезпечується реалізацією fail-safe механізмів, використанням кешування, стійких до збоїв, та безпечного збереження конфігурацій в offline-режимі.

Інструменти виявлення загроз у реальному часі

Моніторинг спроб дебагу, запуску в емуляторах, змін коду чи ін'єкції нових інструкцій дозволяє миттєво припинити роботу застосунку. Зокрема, виявлення бібліотек Frida, Xposed, Magisk є обов’язковим заходом на етапі production.

Запропоновано інтеграцію механізмів контролю середовища, які здатні аналізувати статус root-доступу, наявність емуляторів, порушення політик безпеки та запуск у підозрілих умовах.

Організаційна безпека та документування

Впровадження процедур перевірки коду (code review), автоматичного тестування безпеки (SAST, DAST), CI/CD-інтеграцій з політиками блокування небезпечного коду — є частиною загальної стратегії DevSecOps.

Додатково розробникам рекомендується використовувати безпечні бібліотеки з відкритим кодом, проходити регулярні тренінги, дотримуватись інструкцій з OWASP Mobile Top 10 [2].

Інтеграція з корпоративними системами

У рамках побудови масштабованих Flutter-рішень особливої уваги потребує інтеграція з системами управління користувачами (IAM), політиками доступу (RBAC) та журналами аудиту. Це дозволяє забезпечити:

1. контроль над розмежуванням прав доступу;
2. централізовану реєстрацію дій користувача;
3. автоматичне блокування облікових записів при виявленні підозрілої активності.

Ці практики мають бути впроваджені не тільки на бекенд-рівні, а й продубльовані у Flutter-застосунку через застосування відповідних middleware-рішень.

Перевірка безпеки на етапі розробки

Автоматизоване сканування коду з використанням інструментів, таких як SonarQube, Fortify або MobSF, дозволяє виявити вразливості на ранніх етапах. Прототипування також має включати створення загрозових моделей (threat modeling) та ризик-аналіз, що дозволяє адаптувати архітектуру під реальні сценарії загроз.

Крім основних рівнів захисту, доцільно впроваджувати політику мінімального доступу до апаратних ресурсів пристрою, обмежуючи використання небезпечних дозволів, таких як доступ до геолокації, камери або контактів [3]. Особливу увагу варто приділяти контролю залежностей сторонніх пакетів у Flutter-проєкті, адже уразливі бібліотеки можуть стати точкою компрометації застосунку. У зв’язку з розвитком квантових обчислень зростає актуальність постквантових криптографічних алгоритмів, які мають бути враховані при проєктуванні довгострокових рішень у сфері безпеки. Також слід реалізовувати анонімізацію та псевдонімізацію персональних даних відповідно до вимог міжнародних стандартів, таких як GDPR або ISO/IEC 27001. Проведення регулярного етичного хакінгу (пентестів) допомагає своєчасно виявляти приховані вектори атак і забезпечити актуальність впроваджених заходів кіберзахисту.

## Висновки

Комплексне впровадження заходів захисту дозволяє підвищити стійкість Flutter-додатків до кіберзагроз. Особливу увагу слід приділяти контролю середовища, перевірці цілісності та правильній реалізації криптографічного захисту. Безпека застосунків повинна розглядатися як інтегрована частина всього життєвого циклу розробки.

СПИСОК ВИКОРИСТАНОЇ ЛІТЕРАТУРИ

1. Enhancing Mobile Security through a Comprehensive Development Package [Електронний ресурс] – Режим доступу: https://www.researchgate.net/publication/375456009 (дата звернення: 06.06.2025).

2.OWASP Mobile Top Ten 2023 [Електронний ресурс] – Режим доступу: https://owasp.org/www-project-mobile-top-10/ (дата звернення: 06.06.2025).

3.MobSF (Mobile Security Framework) Documentation [Електронний ресурс] – Режим доступу: https://mobsf.github.io/docs/(дата звернення: 06.06.2025).

***Осипенко Ірина Віталіївна*** – здобувач вищої освіти, кафедра автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: student.osypenko@gmail.com.

***Богач Ілона Віталіївна*** – к.т.н., доцент кафедри автоматизації та інтелектуальних інформаційних технологій, факультет інтелектуальних інформаційних технологій та автоматизації, Вінницький національний технічний університет, м.Вінниця, e-mail: ilona.bogach@gmail.com.

***Osypenko Iryna Vitaliivna*** – higher education student, Department of Automation and Intelligent Information Technologies, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: student.osypenko@gmail.com.

***Bogach Ilona Vitaliivna*** – Associate Professor of Automation and Intelligent Information Technologies Department, Faculty of Intelligent Information Technology and Automation, Vinnytsia National Technical University, Vinnytsia, e-mail: ilona.bogach@gmail.com.