УДК 004.9:004.8
DOI https://doi.org/10.32689/maup.it.2025.3.7

Остап ГЕТЬМАН
аспірант кафедри комп'ютерних наук та програмної інженерії,
Приватний вищий навчальний заклад «Європейський університет»
***ORCID:** 0009-0003-6726-9418*

Роман ЯРОВИЙ
кандидат технічних наук, доцент,
декан факультету інформаційних систем та технологій,
Приватний вищий навчальний заклад «Європейський університет»
***ORCID:** 0000-0001-8978-8137*

АДАПТИВНІ СТРАТЕГІЇ ЗАБЕЗПЕЧЕННЯ ЗАХИСТУ API МОБІЛЬНИХ ДОДАТКІВ НА ОСНОВІ МАШИННОГО НАВЧАННЯ В УМОВАХ ОБМЕЖЕНИХ РЕСУРСІВ

Анотація. *У статті проведено огляд технічних обмежень, характерних для апаратно-програмного середовища мобільних додатків, що використовують API-інтерфейси для взаємодії з мережевими сервісами. Продовжено розробку проблеми забезпечення кіберзахисту мобільних API в умовах обмежених ресурсів, де ключовими факторами виступають пропускна здатність каналів мобільного зв'язку, обсяг оперативної пам'яті та рівень доступного обчислювального ресурсу.*

Мета статті *полягає у формуванні комплексної методики побудови адаптивної системи захисту API мобільного додатку на основі машинного навчання із урахуванням обмежень пристрою, варіативності запитів, сценаріїв загроз та вимог до продуктивності.*

Методологія*. Використано систематизацію векторів атак на API та впроваджено багаторівневу структуру методів захисту, яка включає аутентифікацію, шифрування, контроль доступу, виявлення аномалій, захист інформаційного сховища та оновлення компонентів. Проведено класифікацію моделей машинного навчання за придатністю до реалізації у мобільному середовищі. Показано ефективність застосування ансамблевих методів та SVM у режимі локального використання. Запропоновано гібридну архітектуру, що поєднує локальний фільтр запитів із хмарною нейромережею для виявлення складних та нетипових патернів.*

Наукова новизна *полягає у розробці адаптивної архітектури системи захисту мобільних API, яка інтегрує локальні модулі з хмарними сервісами та забезпечує баланс між продуктивністю і рівнем безпеки. Запропоновано використання легковагових моделей машинного навчання у мобільному середовищі та поведінкового аналізу API-запитів як ключового елементу адаптивного реагування на нові типи атак.*

Висновки. *Основний акцент було зроблено на створенні гібридної системи кіберзахисту API мобільних додатків, що поєднує переваги локальної та хмарної обробки. Проаналізовано особливості застосування методів машинного навчання для виявлення кіберзагроз, що супроводжують використання API. Запропоновано методику побудови комплексної системи захисту, яка охоплює модулі аутентифікації, шифрування трафіку, обфускації коду, контейнеризації, фіксації подій, реагування на інциденти та оновлення політик безпеки.*

Ключові слова: *захист API, мобільні додатки, обмеження ресурсів, машинне навчання, хмарні обчислення, гібридна система безпеки, поведінковий аналіз.*

Ostap HETMAN, Roman YAROVYI. ADAPTIVE STRATEGIES FOR API SECURITY IN MOBILE APPLICATIONS BASED ON MACHINE LEARNING UNDER RESOURCE CONSTRAINTS

Abstract. *The article reviews the technical limitations characteristic of the hardware and software environment of mobile applications that use API interfaces to interact with network services. The development of the problem of ensuring cyber protection of mobile APIs in conditions of limited resources is continued, where the key factors are the bandwidth of mobile communication channels, the amount of RAM and the level of available computing resources.*

The purpose of the article *is to form a comprehensive methodology for building an adaptive mobile application API protection system based on machine learning, taking into account device limitations, query variability, threat scenarios and performance requirements.*

Methodology. *The systematization of attack vectors on APIs is used and a multi-level structure of protection methods is implemented, which includes authentication, encryption, access control, anomaly detection, information storage protection and component updates. A classification of machine learning models is carried out according to their suitability for implementation in a mobile environment. The effectiveness of the use of ensemble methods and SVM in local use mode is shown. A hybrid architecture is proposed that combines a local query filter with a cloud neural network to detect complex and atypical patterns.*

The scientific novelty *lies in the development of an adaptive architecture for the mobile API protection system, which integrates local modules with cloud services and provides a balance between performance and security level. The use of lightweight machine learning models in the mobile environment and behavioral analysis of API requests as a key element of adaptive response to new types of attacks is proposed.*

© О. Гетьман, Р. Яровий, 2025

Стаття поширюється на умовах ліцензії CC BY 4.0

Conclusions. *The main emphasis was placed on creating a hybrid mobile application API cyber protection system that combines the advantages of local and cloud processing. The features of the application of machine learning methods to detect cyber threats accompanying the use of APIs are analyzed. A methodology for building a comprehensive protection system is proposed, which includes authentication modules, traffic encryption, code obfuscation, containerization, event capture, incident response, and security policy updates.*

Key words: *API security, mobile applications, resource constraints, machine learning, cloud computing, hybrid security system, behavioral analysis.*

Вступ. Стрімкий розвиток інформаційних технологій у галузі мережевих сервісів впродовж останнього десятиріччя супроводжується цифровізацією усіх сфер суспільної діяльності, як то проведення банківських операцій [1], організації медичних сервісів [26], налаштування систем дистанційного навчання [11], тощо. Одним із ключових напрямів цього процесу стало зростання ролі мобільних платформ, що забезпечують користувачам постійний доступ до цифрового контенту та сервісів у режимі реального часу [25; 27]. При цьому спостерігається не лише кількісне збільшення мобільних додатків, але й ускладнення архітектури, що передбачає інтеграцію з хмарними обчисленнями [10], а також використання моделей машинного навчання [19] для персоналізації контенту й аналізу поведінки користувачів. Взаємозалежність між компонентами таких систем значно підвищує вимоги до їхньої захищеності, особливо з огляду на обробку чутливої інформації, як то фінансових, медичних і персональних даних. У цьому контексті особливої актуальності набуває проблема забезпечення безпеки прикладних програмних інтерфейсів (Application Programming Interface, API), які виступають основним каналом взаємодії між мобільними додатками, серверною інфраструктурою та зовнішніми сервісами [2; 8; 16]. Саме API як «точка входу» кінцевого користувача до функціоналу мобільного додатка, і його компрометація може призвести до значних наслідків, від витоку даних до повного блокування сервісу та отримання зловмисником повного контролю над сервісом. З огляду на відкритість API та високу інтенсивність обробки запитів, даний компонент стає найбільш вразливим елементом сучасної мобільної інфраструктури. Таким чином, дослідження методів оптимізації системи безпеки API у мобільному середовищі з урахуванням ресурсних обмежень та складності патернів кібер-атак, є ***актуальним завданням*** як у науковій, так і в прикладній площині.

Аналіз останніх наукових досліджень. Аналіз наукових досліджень присвячених проблемам захисту API у мобільному середовищі, надав можливість вказати на необхідність урахування ресурсних обмежень мобільних пристроїв при проектуванні стратегій кібербезпеки, особливо при впровадженні сервісів на основі алгоритмів машинного навчання [2; 8; 16; 19]. У більшості мобільних пристроїв ці ресурси розраховані на обслуговування користувацьких задач із низьким рівнем складності, тому впровадження складних процедур безпеки без адаптації призводить до зростання затримок і зниження стабільності системи [4]. Це особливо актуально у випадках, коли безпекові механізми мають працювати у режимі реального часу, наприклад, при обробці потокових API-запитів або моніторингу трафіку [6]. При цьому зазначається, що API є ключовими точками взаємодії між мобільними додатками, серверною інфраструктурою та зовнішніми сервісами, що робить їх пріоритетною ціллю для зловмисників [9]. У гібридному середовищі, яке поєднує локальне виконання частини функцій із делегуванням складніших обчислень на хмару, виникають нові виклики: необхідність безпечної передачі даних, синхронізації станів сесій, перевірки автентичності міжконтекстних запитів [18]. Вразливості можуть виникати як на стороні клієнта, так і на рівні серверної логіки, що обробляє запити без належної перевірки [7]. У рамках забезпечення адаптивного захисту API широко застосовуються алгоритми машинного навчання, які дозволяють виявляти аномалії, формувати поведінкові профілі та класифікувати запити за рівнем ризику [29]. Ключовим є питання щодо місця розташування алгоритмів машинного навчання, що надає можливість виділити дві категорії [12; 13]:

1. Локальне розташування моделі на мобільному пристрої. Моделі машинного навчання відповідної категорії мають перевагу в швидкодії та автономності, проте їх функціонал значним чином обмежений доступним обчислювальним ресурсом.

2. Мережеве розташування моделі на хмарному сервері. Моделі машинного навчання відповідної категорії забезпечують високу точність за рахунок доступу до потужніших обчислювальних ресурсів і великих обсягів даних, але вимагають стабільного каналу зв'язку та характеризуються високим рівнем латентності.

Для врахування ресурсних обмежень мобільного середовища активно розробляються легковагові моделі машинного навчання (Lightweight Machine Learning Models, LWM-LM), зокрема на основі технологій стиснення моделі (Model Compression, MC), дистиляції знань (Knowledge Distillation, KD), проріджування моделі (Model Pruning, MP) і квантизації моделі (Model Quantization). Такі моделі здатні функціонувати на пристроях із обмеженим обчислювальним ресурсом, забезпечуючи базовий рівень

аналізу без постійного доступу до хмарного сервісу [5; 14; 23]. При цьому зберігається можливість динамічного оновлення моделей або делегування більш складних задач до серверної частини системи за умов збільшення мережевої доступності. При цьому відсутність узагальненої методологічної бази, яка дозволяє співвіднести рівень ефективності захисту API з обраною конфігурацією машинного навчання та складністю її інтеграції у гібридне мобільно-хмарне середовище, розглядається як ***невирішений аспект загального підходу*** до побудови адаптивних систем безпеки. Найбільшою мірою стосується завдань, де захист API не є ізольованим компонентом, а функціонує у зв'язку з іншими елементами багаторівневої безпекової архітектури.

Таким чином, ***метою роботи*** стало формування комплексної методології адаптації машинного навчання для оцінки й забезпечення безпеки API у мобільних додатках. Відповідний підхід має передбачити врахування впливу архітектурних і алгоритмічних рішень на навантаження обчислювального середовища, а також розробку критеріїв оцінки ефективності інтеграції легковагових моделей машинного навчання у загальну інфраструктуру безпекового контролю з урахуванням ресурсних обмежень.

Виклад основного матеріалу. *Постановка задачі забезпечення захисту API мобільних додатків.* Як показав проведений аналіз, зростання складності мобільних сервісів, інтегрованих у гібридну архітектуру клієнт-сервер, супроводжується підвищенням навантаження на API, які виступають основною точкою взаємодії між користувацькими додатками, серверною логікою та зовнішніми сервісами. За умов обмежених ресурсів мобільного середовища, зокрема пропускної здатності мережі, обсягу оперативної пам'яті та обчислювальних можливостей, виникає потреба у спеціальних стратегіях забезпечення безпеки, орієнтованих на адаптивне використання ресурсів. Побудова ефективної системи захисту API потребує врахування відповідних обмежень на ранньому етапі розробки, а отже, дослідження включає у себе послідовне виконання наступних етапів:

– аналіз технічних характеристик мобільної платформи (пропускна здатність каналів передачі даних, обсяг доступної пам'яті та рівень процесорного навантаження), на основі якого формуються початкові стратегії адаптації, які дозволяють визначити оптимальні сценарії реалізації захисних модулів;

– вибір способу розташування функціональних компонентів системи кіберзахисту як комбінації локальної обробки запитів на апаратній платформі мобільного пристрою та впровадження хмарних сервісів для виконання найбільш складних обчислення, що вимагає врахування топології мережевого розташування і цільових показників ефективності аналізу запитів;

– адаптація алгоритмів машинного навчання до ресурсних умов мобільної платформи, через побудову легковагових моделей машинного аналізу за допомогою технологій стискання і квантизації моделі, а також дистиляції знань, що дозволяє реалізувати базові сценарії аналізу загроз без необхідності постійного доступу до віддалених обчислювальних ресурсів;

– побудова комплексної багаторівневої структури захисту, що включає: аутентифікацію, авторизацію, шифрування, захист API, моніторинг, контейнеризацію та оновлення мобільного додатку, що дозволяє забезпечити цілісну модель безпеки, яка адаптується до змін ресурсного профілю пристрою та мережевих умов.

На основі зазначених підходів має бути сформульовано методологію дослідження, яка дозволяє співвіднести архітектурні та алгоритмічні рішення з рівнем обчислювального навантаження та ефективністю захисту API у мобільних додатках (рис. 1).

Представлена структура виконує функцію концептуальної моделі, що забезпечує методологічну основу для подальшого обґрунтування адаптивного підходу до оцінювання та впровадження алгоритмів машинного навчання з урахуванням обмежень апаратно-програмної платформи мобільного додатку. Такий підхід дозволяє не лише забезпечити цілісність системи кіберзахисту API, але й адаптувати її до змінних умов експлуатації у рамках мобільного середовища та динаміки зовнішніх загроз.

Побудова моделі загроз API відповідно до міжнародних стандартів. Розглянута у попередньому розділі постановка задачі окреслила необхідність комплексного підходу до забезпечення захисту API мобільних додатків в умовах обмежених ресурсів. Однак для формалізації подальшого аналізу доцільним є застосування моделей загроз, що відповідають міжнародним стандартам кібербезпеки. Це дозволяє забезпечити системність і відтворюваність оцінки ризиків, а також узгодити пропоновані механізми захисту із загальноприйнятими практиками проєктування безпечних інформаційних систем. Відповідно до рекомендацій «NIST SP 800–154» [22] та «OWASP API Security Top – 10» [17], одним з ефективних методів класифікації загроз є використання моделі «STRIDE» [20], що у контексті мобільних API набуває наступної інтерпретації:

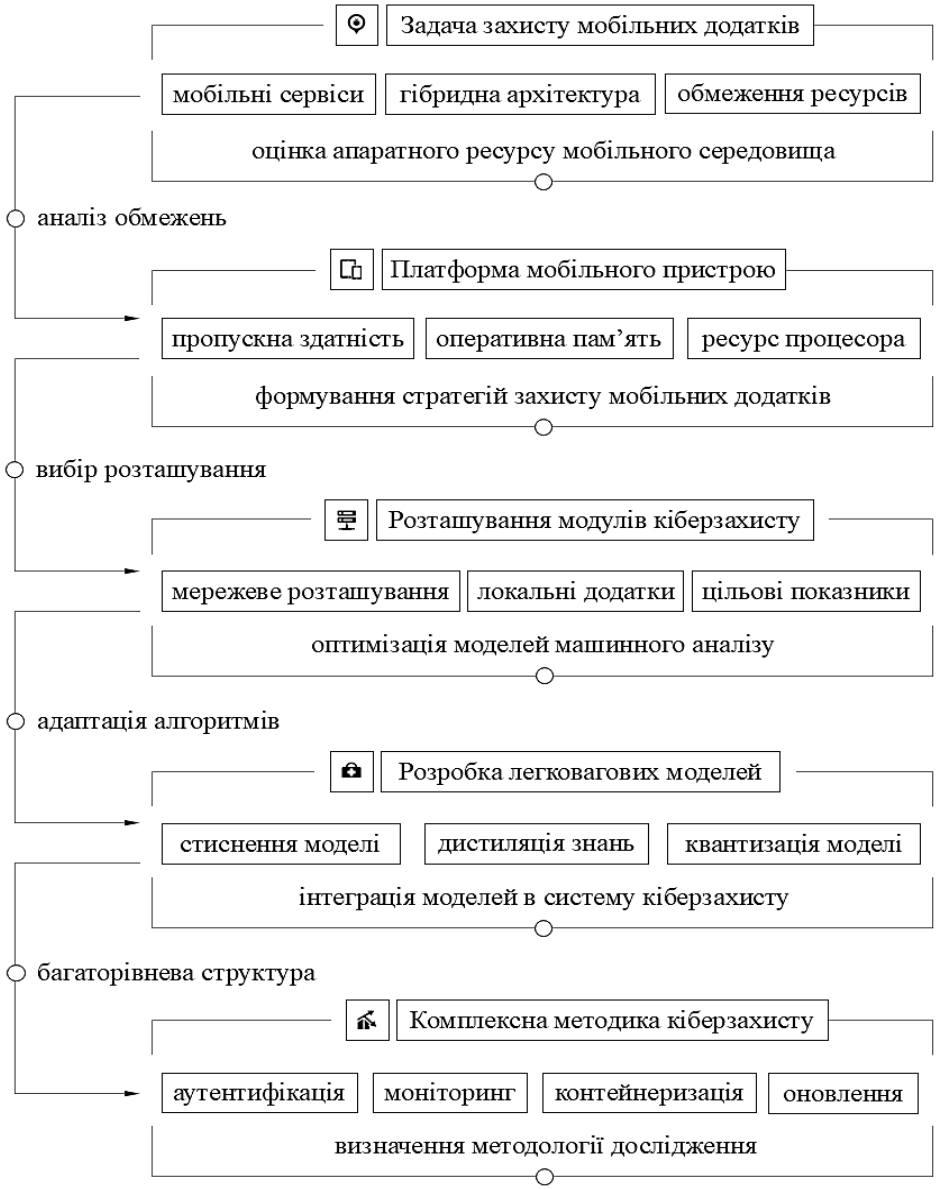

Рис. 1. Логіко-функціональна схема побудови багаторівневої системи безпеки API для мобільних додатків

1. Підробка засобів автентифікації (Spoofing, S) як компрометація токенів доступу, використання вкрадених облікових даних та підроблених сертифікатів для доступу до API.

2. Нелегальна модифікація даних сервісу (Tampering, T) через зміну параметрів у запитах, втручання у трафік, ін'єкції коду на рівні API-викликів.

3. Відсутність доказовості дій (Repudiation, R) через недостатній аудит операцій, що дозволяє зловмиснику уникати відповідальності.

4. Розголошення інформації (Information Disclosure, I) як витік персональних чи фінансових даних через некоректну обробку запитів або помилки шифрування.

5. Відмова в обслуговуванні (Denial of Service, D): перевантаження API великою кількістю запитів, що блокує доступ легальних користувачів.

6. Підвищення привілеїв (Elevation of Privilege, E) отримання доступу до адміністративних функцій API шляхом експлуатації логічних вразливостей.

Застосування цієї класифікації дозволяє прямо співвіднести вектори атак з конкретними контрзаходами, як то багатофакторну автентифікацію, контроль швидкості надходження запитів, обов'язковий аудит операцій, використання протоколів TLS 1.3 з підтримкою процедури «Certificate Pinning», а також динамічна ротація ключів і регулярне оновлення політик доступу [17; 20; 22].

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

Водночас важливо зазначити, що у рамках дослідження наведена модель загроз не розглядається як статична. У динамічному мобільному середовищі вона повинна виконувати функцію адаптивного каркасу, який дозволяє корелювати специфіку API-інтерфейсу з наявними ресурсними обмеженнями та сценаріями атак. На відміну від традиційного застосування STRIDE як інструменту аудиту, у даному випадку модель розглядається як інтегрований елемент архітектури гібридного захисту. Це забезпечує можливість динамічного віднесення загроз до класів, що підлягають обробці локальними або хмарними компонентами системи, і формує підґрунтя для розробки адаптивних стратегій кіберзахисту, орієнтованих на ресурсні обмеження мобільного середовища.

Таким чином, відповідно задачі дослідження необхідно вказати, що окрему групу загроз формують ризики, пов'язані з процесом внесення змін до мобільного додатку та API. Уразливості можуть виникати як у момент оновлення компонентів, так і при взаємодії різних версій програмного забезпечення. Згідно з підходами «OWASP Mobile Security Testing Guide» та рекомендаціями «NIST» [17; 20; 22], безпечний життєвий цикл оновлень включає такі механізми:

– введення цифрового підпису оновлень для перевірки криптографічної цілісності пакета перед інсталяцією;

– ротація ключів як регулярне оновлення криптографічних ключів для зменшення ризику компрометації;

– передача оновлень лише через захищені канали (TLS 1.3 / mTLS) для запобігання атакам «Man in the Middle»;

– rollback-захист як алгоритм блокування інсталяції застарілих версій, що містять відомі вразливості.

– A / B-розгортання як поетапне розповсюдження оновлень із можливістю повернення до попередньої версії без компрометації системи;

– оцінка сумісності версій API через контроль відповідності клієнтських і серверних версій інтерфейсу для уникнення експлуатації логічних розривів.

У відповідності до проведеного аналізу можна вказати, що захист процесу оновлення виступає невід'ємною частиною моделі загроз API, оскільки дозволяє знизити ризик інжекції шкідливих змін, забезпечити контроль цілісності середовища та підтримувати стабільність у багатоверсійних конфігураціях мобільних додатків. У межах даного дослідження цей аспект інтегрується в архітектуру гібридного захисту як адаптивний модуль управління життєвим циклом оновлень, що поєднує криптографічні гарантії з методами поведінкового аналізу для виявлення нетипових сценаріїв у процесі розгортання.

Наведені вище механізми захисту життєвого циклу оновлень формують базовий рівень стійкості мобільного API, проте в умовах зростаючої складності атак додатково необхідно впроваджувати спеціалізовані контрзаходи, які враховують специфіку мобільного середовища, гібридної архітектури та сценаріїв реального використання додатків. Відповідні заходи здатні мінімізувати залишкові ризики після впровадження класичних методів автентифікації та шифрування й забезпечити більш глибокий рівень довіри до екосистеми мобільного додатку [17; 20; 22].

1. Прив'язка сертифікатів (certificate pinning) як механізм, який полягає у жорсткій фіксації конкретного SSL / TLS-сертифіката на стороні клієнтського додатку. Це унеможливлює використання підроблених сертифікатів навіть у випадку компрометації центру сертифікації, значно знижуючи ризик атак типу «людина посередині».

2. Двостороння автентифікація на основі протоколу «Transport Layer Security» (Mutual TLS, mTLS): передбачає, що як сервер, так і клієнт зобов'язані підтвердити власну автентичність за допомогою сертифікатів. Такий підхід дозволяє запобігти доступу до API з неперевірених мобільних клієнтів і підвищує рівень довіри між сторонами.

3. Сервіси атестації середовища виконання (Attestation Services, AS), що включає у себе набір спеціальних інфраструктурнмх сервісів («Google Play Integrity API», «SafetyNet», тощо), які підтверджують, що мобільний додаток виконується у незміненому середовищі, без ознак рутування, модифікацій чи запуску в емуляторі. Це унеможливлює експлуатацію API у неконтрольованих умовах.

4. Виявлення емуляторів і спеціалізованого середовища (Emulator and Root / Jailbreak Detection E&RJD) шляхом впровадження технічних механізмів, що дозволяють ідентифікувати запуск додатку на пристроях із модифікованою операційною системою або в емуляторі. Це запобігає проведенню аналізу або експлуатації API в умовах, де відсутній контроль виробника чи розробника.

5. Захист токенів доступу (Token Protection, TP) через використання токенів із коротким часом життя (Time-To-Live, TTL), їх прив'язка до конкретного пристрою або сесії, а також застосування

одноразових токенів (One-Time Tokens, OTT). Це мінімізує ризик повторного використання викрадених облікових даних і ускладнює їх підробку.

Завдяки цим контрзаходам формується додатковий рівень захисту, який не лише доповнює традиційні механізми безпеки, але й враховує особливості мобільного середовища, де загрози пов'язані не лише з мережевими комунікаціями, а й з фізичним контролем над пристроєм, станом операційної системи та специфікою виконання додатку.

Таким чином, модель загроз API, побудована на основі міжнародних стандартів і доповнена механізмами безпечного життєвого циклу оновлень та специфічними контрзаходами, не лише систематизує ризики, але й формує підґрунтя для адаптивної архітектури гібридного захисту. Поєднання формальної моделі STRIDE з динамічними механізмами оновлення та спеціалізованими контрзаходами дозволяє забезпечити відповідність стандартам і водночас врахувати ресурсні обмеження мобільного середовища. Це створює умови для практичної інтеграції моделі у високоефективні системи захисту, орієнтовані на динамічну адаптацію до змін середовища та сценаріїв реалізації атак.

Системний аналіз обмежень мобільного середовища в контексті захисту API. Зростання функціональних можливостей апаратної платформи мобільного пристрою відбувається одночасно з ускладненням архітектури сервісів, що надаються користувачам через мобільні додатки. Попри те, що обчислювальні характеристики таких пристроїв демонструють стабільну позитивну динаміку, темпи розвитку алгоритмів машинного навчання, механізмів шифрування та кіберзахисту значно випереджають відповідне зростання ресурсів. Це зумовлює необхідність формалізованого підходу до врахування обмежень мобільного середовища при проєктуванні захисних механізмів API. З метою забезпечення системності у подальшому аналізі, дослідження має бути поділено на дві взаємопов'язані частини, що включають у себе дослідження моделі ресурсних обмежень мобільного середовища та вплив зазначених обмежень на вибір архітектурної стратегії захисту API, включаючи розподіл навантаження між локальними та мережевими компонентами, а також визначення критичних факторів, що обумовлюють доцільність впровадження легковагових моделей аналізу в межах мобільного середовища.

Протягом останніх десятиліть розвиток обчислювальних ресурсів мобільних пристроїв демонструє стійку тенденцію до зростання, що наближається до експоненційної (рис. 2-а). У логарифмічному масштабі відображено еволюцію швидкодії: від мегафлопсів на початку 2000-х років до тера- та ексафлопсного діапазону в 2020-х [13; 28]. Таке зростання стало можливим завдяки послідовному вдосконаленню процесорних архітектур, впровадженню багатоядерних процесорів (Central Processing Unit, CPU), спеціалізованих графічних прискорювачів (Graphics Processing Unit, GPU), а також процесорів спеціалізованих для роботи з нейромережевими алгоритмами (Neural Processing Unit, NPU). Розширення обчислювального потенціалу забезпечило передумови для реалізації складних аналітичних і захисних сценаріїв безпосередньо на пристрої, включно з підтримкою шифрування в реальному часі, автентифікації, моделювання поведінкових аномалій та використання алгоритмів машинного навчання. Оперативна пам'ять мобільного пристрою також відіграє критичну роль у підтримці функціонування модулів системи захисту, що працюють у реальному часі. Як засвідчено на рис. 2-б, обсяг доступної пам'яті демонструє стійке зростання у логарифмічному. Така динаміка визначає можливості щодо реалізації більш складних захисних механізмів, розширеної багатозадачності та обробки великих масивів даних на пристрої. На ранніх етапах розвитку обмеження обсягу оперативної пам'яті обумовлювали низьку продуктивність і відсутність повноцінних механізмів шифрування чи моніторингу загроз. Із поступовим переходом до DDR SDRAM, а згодом до DDR4 / DDR5, відкрилися можливості для впровадження поведінкового аналізу, апаратного шифрування, динамічного сканування пам'яті й виявлення аномалій [5; 15].

Ефективність архітектури системи захисту, зокрема тих, що передбачають використання хмарних або гібридних компонентів для моніторингу, автентифікації та аналізу кібер-загроз, істотно залежить від параметрів пропускної здатності мобільного середовища. Як показано на рис. 2-в, за останні два десятиліття відбулося багаторазове зростання швидкості передачі даних, що дозволяє реалізовувати обчислення на стороні сервера в режимі, близькому до реального часу. Зазначений графік також представлено у логарифмічному масштабі для наочної демонстрації темпів зростання. Впровадження стандартів зв'язку 3G у 2000-х роках стало відправною точкою для розвитку мобільного інтернету, але водночас актуалізувало проблеми перехоплення даних та експлуатації незахищених каналів. З переходом до 4G LTE у 2010-х роках з'явилися умови для широкого використання ресурсомістких сервісів, що супроводжувалося зростанням масштабних атак типу DDoS. Із приходом технології 5G у 2020-х роках були створені передумови для повноцінного функціонування розподілених

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

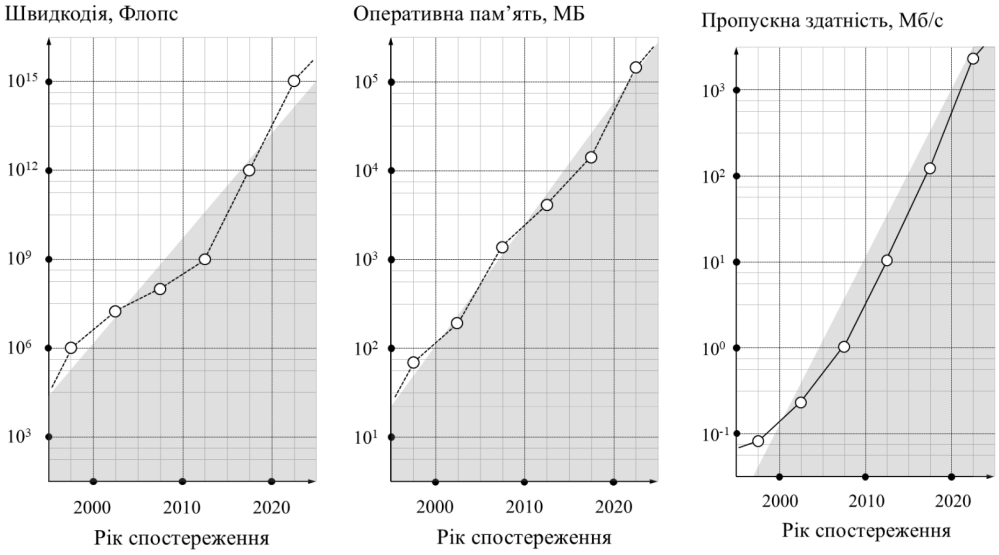

Рис. 2. Ріст параметрів мобільної платформи: (а) обчислювальна швидкодія, (б) оперативна пам'ять, (в) пропускна здатність каналів [13; 15; 21; 28]

систем захисту API, зокрема тих, що базуються на глибинному навчанні та адаптивному аналізі сценаріїв [12; 21].

Аналіз еволюції апаратної платформи мобільного середовища демонструє суттєве зростання середніх показників обчислювальної потужності, обсягу оперативної пам'яті та пропускної здатності каналів зв'язку. Кожен із цих аспектів відіграє критичну роль у формуванні функціонального простору для реалізації механізмів захисту API, від базових локальних модулів до складних розподілених систем. Поступове зростання обчислювального ресурсу апаратної платформи відкриває можливості для інтеграції алгоритмів машинного навчання, багаторівневого шифрування та поведінкового аналізу загроз безпосередньо на пристрої. Водночас високошвидкісні мережеві інтерфейси дозволяють перенести частину обчислювального навантаження на хмарні сервіси, що забезпечують масштабованість, контекстну адаптацію та оперативне оновлення моделей захисту. Сучасна система безпеки мобільного середовища має будуватися як комплексна гібридна архітектура, яка динамічно балансує між локальним реагуванням в умовах обмежених ресурсів та централізованим аналізом у розподілених хмарних середовищах. Такий підхід дозволяє враховувати як технічні обмеження платформи, так і зростаючу складність кібератак, забезпечуючи стійкість, масштабованість та гнучкість систем захисту API.

Стратегії розміщення моделей машинного навчання у систем захисту API. У контексті динамічного розвитку сучасних мобільних інформаційних систем саме API виступають ключовими точками взаємодії та, водночас, надзвичайно вразливими елементами архітектури. Стандартизовані методи захисту на основі сигнатур або статичних правил доступу дедалі частіше демонструють обмежену ефективність у протидії динамічно змінюваним шаблонам кібератак. Це зумовлює необхідність впровадження інтелектуальних систем виявлення загроз, здатних адаптуватися до нових сценаріїв, навчатися на нових даних і забезпечувати безпеку в умовах обмежених ресурсів мобільного середовища. Ключову роль у цьому відіграють методи машинного навчання, що забезпечують автоматизовану обробку API-запитів, виявлення аномалій і класифікацію загроз. У рамках дослідження було проведено класифікацію таких моделей за принципом їх придатності до виконання в локальному середовищі мобільного пристрою або необхідності делегування обчислень до сервісів хмарної інфраструктури. До категорії локально реалізованих моделей, що пропонується розглянути у рамках дослідження, належать [5; 12; 13; 14; 23]:

1. Метод опорних векторів (Support Vector Machine, SVM) є ефективним для швидкої бінарної класифікації запитів з метою виявлення потенційних загроз. Перевагою зазначеного підходу є можливість виконання на пристрої за умови обмеженої розмірності ознакового простору.

2. Ансамблеві методи надають можливість для побудови багатокласових класифікаторів зі стійкістю до перенавчання. Зазначений підхід використовується для виявлення типових атак, як то SQL-ін'єкції, порушення авторизації, тощо.

3. Градієнтні дерева є доцільними для обробки наборів нерівномірно розподілених даних, у тому числі атак нульового дня, що становлять найбільшу загрозу для складових мобільного середовища. Обмежене застосування зазначеного підходу на апаратній платформі мобільного пристрою можливе лише за умови спрощеної конфігурації моделі.

До категорії моделей, що при обробці набору вхідних даних вимагають делегування на запитів сервер або обчислювальний кластер хмарної інфраструктури, належать [7; 18; 29]:

1. Нейромережеві алгоритми глибинного навчання, як то моделі на основі DNN, RNN, BERT і трансформери, призначені для виявлення складних нелінійних залежностей у запитах, обробки послідовностей, семантичного аналізу тіла запиту, а також інтеграції з SIEM-системами. Висока обчислювальна складність передбачає застосування хмарних сервісів або гібридних архітектур із попередньою фільтрацією.

2. Контекстно-адаптивні трансформери розглядаються як перспективні в задачах класифікації запитів із прихованими залежностями, але потребують значного обсягу оперативної пам'яті й засобів паралельної обробки, що доступні у рамках хмарної інфраструктури.

Розподіл між локальним виконанням та хмарною обробкою формується на основі таких критеріїв:

– складність та розмірність ознакового простору;
– наявність часової залежності у наборі даних;
– критичність до затримки у відповіді на загрозу;
– обчислювальні та енергетичні обмеження мобільного пристрою.

З метою формалізації підходу до вибору алгоритмів машинного навчання залежно від обчислювального середовища проведено оцінювання їхньої придатності до виконання на мобільному пристрої та у хмарній інфраструктурі. Результати систематизації наведено в (табл. 1), яка узагальнює типові моделі кіберзахисту API, класифікує їх за параметрами можливого розміщення, а також окреслює характерні особливості реалізації кожної моделі з урахуванням вимог до показників ресурсомісткості.

Таблиця 1

Оцінка придатності моделей до виконання у локальному середовищі мобільного пристрою та інфраструктурі хмарного сервісу [5; 7; 12; 13; 14; 18; 23; 29]

Тип моделі кіберзахисту	Пам'ять	Швидкодія і затримка	Енергоспоживання	Особливості використання
Метод опорних векторів	низьке	висока	низьке	Найбільш ефективно на етапі бінарної класифікації запитів у локальному середовищі.
Ансамблеві моделі	середнє	середня	середнє-високе	Забезпечують стійкість до перенавчання і при цьому придатні як для локального виконання, так і для хмарної аналітики у складніших конфігураціях.
Градієнтні дерева	середнє	середня	середнє	Дає хороші результати на нерівномірно розподілених даних, використовуються у мобільному середовищі у спрощеній формі, а також масштабуються для хмарної інфраструктури.
Глибинні нейромережі (DNN / RNN)	високе	низька	високе	Необхідні великі обсяги даних і GPU / TPU, тому найбільш ефективні у хмарній інфраструктурі.
Трансформери та BERT	дуже високе	низька	дуже високе	Забезпечують контекстний аналіз, потребують спеціалізованих прискорювачів (GPU / NPU) і достатніх обчислювальних ресурсів.

Таким чином, побудова ефективної системи захисту API вимагає формування гібридної архітектури [12; 13], що поєднує переваги обох типів моделей, де локальні модулі використовуються для оперативного реагування, а хмарні сервісу виконують задачі глибинного аналізу та виявлення складних загроз.

Оптимізація моделей машинного навчання для мобільного середовища. Як було зазначено вище, забезпечення безперервного моніторингу та виявлення загроз у мобільному середовищі потребує використання моделей машинного навчання, які можуть функціонувати в умовах обмежених обчислювальних ресурсів, мінімального енергоспоживання та нестабільної пропускної здатності каналів зв'язку. У цьому контексті перспективним напрямом є впровадження легковагових нейромережевих архітектур та методів оптимізації моделей, таких як квантизація моделі,

ISSN 2786-5460 (Print), ISSN 2786-5479 (Online)

проріджування моделі і дистиляція знань. На загальному рівні можна виокремити дві архітектурні концепції впровадження легковагових нейромережевих систем:

1. Концепція TinyML передбачає виконання інтелектуальних обчислень безпосередньо на пристроях із обмеженими апаратними ресурсами, як то на мікроконтролерах із мінімальним обсягом оперативної пам'яті та низьким енергоспоживанням [24]. Відповідний підхід застосовується у задачах попереднього виявлення аномалій у потоках сенсорних даних, класифікації коротких послідовностей API-запитів, а також у попередній обробці даних безпосередньо на мобільному пристрої перед їх передачею у хмарне середовище.

2. Архітектурна модель «Edge AI» орієнтована на реалізацію гібридних рішень, у яких первинна обробка, фільтрація та виявлення аномалій відбувається локально (на рівні мобільного пристрою або периферійного вузла), тоді як більш складна аналітика, пов'язана з нейромережевим алгоритмом глибокого навчання, делегується до хмарних сервісів [3]. Такий підхід дозволяє зменшити затримки при реагуванні, знизити навантаження на канали зв'язку та забезпечити адаптивність до контексту середовища.

Таблиця 2

Порівняльна характеристика методів оптимізації моделей машинного навчання для мобільного середовища [3; 24]

Метод оптимізації	Цільовий ефект	Вплив на точність	Повторне навчання
дистиляція знань	менша складність	низький	потребує
квантування моделі	менший розмір	помірний	не обов'язково
проріджения моделі	менше параметрів	помірно високий	потребує
архітектурна оптимізація	проста структура	помірно низький	не потребує

У свою чергу, методологічні рішення щодо оптимізації моделей ґрунтуються на наступних підходах:

– процедура дистиляції знань передбачає тренування спрощеної студент-моделі на основі прогнозів попередньо навченої великої моделі-наставника, що дозволяє зберегти високу якість при суттєвому зменшенні обчислювального навантаження;

– процедура квантування моделі передбачає зменшення точності числового представлення параметрів моделі, що зменшує обсяг оперативної пам'яті, що використовується алгоритмом, а також пришвидшує обробку набору вхідних даних;

– процедура проріджения моделі полягає у видаленні найменш значимих параметрів а також нейронів, що дозволяє зменшити загальний розмір нейромережевої архітектури без суттєвого зниження якості обробки набору вхідних даних.

Ці методи дозволяють реалізовувати адаптивну конфігурацію моделей відповідно до характеристик апаратної платформи мобільного пристрою. Для формалізації вибору конкретної техніки оптимізації у заданих обмеженнях запропоновано класифікацію (див. табл. 2), що враховує цільовий ефект, рівень втрати точності, а також необхідність повторного навчання.

Архітектурна модель гібридного захисту API мобільного додатку. Забезпечення ефективного та стійкого захисту API мобільних додатків, таким чином, вимагає впровадження багаторівневої моделі безпеки, яка охоплює всі критичні етапи обробки та взаємодії даних. Така модель має бути не лише комплексною за структурою, а й адаптивною до характеристик мобільного середовища, зокрема обмежених обчислювальних ресурсів, нестабільної пропускної здатності каналів зв'язку, енергоспоживання та типу платформи. У межах запропонованої архітектурної моделі гібридного захисту API доцільно виокремити такі ключові рівні:

1. Аутентифікація та авторизація, що реалізується переважно локально з використанням токенів, біометричних даних та системних засобів ідентифікації. При цьому критично важливо забезпечити швидкий доступ без передачі чутливих даних через мережу.

2. Шифрування трафіку здійснюється через протоколи TLS / SSL, з можливістю адаптивного вибору криптографічних параметрів залежно від поточних параметрів апаратної платформи мобільного пристрою.

3. Захист інформаційного сховища даних через шифрування на рівні файлової системи або використання захищених контейнерів.

4. Фільтрація API-запитів засобами базової евристичної перевірки та легковагових класифікаторів, що реалізуються локально, у той час як глибока перевірка виконується на сервері або через хмарні SIEM-системи.

5. Контейнеризація та ізоляція через впровадження ізольованого середовища виконання, що обмежує вплив потенційно скомпрометованих модулів.

6. Моніторинг та виявлення загроз реалізується у гібридному режимі, де первинний контроль поведінки та виявлення відхилень виконується локально, а централізований аналіз проводиться у середовищі хмарного сервісу із залученням нейромережевих моделей.

7. Тестування та оновлення включають у себе перевірку цілісності компонентів і своєчасне отримання оновлень є критично важливими для протидії новим вразливостям.

Адаптивність цієї моделі досягається за рахунок динамічного розподілу функцій між локальним середовищем і хмарною інфраструктурою, що дозволяє зменшити затримку у реагуванні на загрози, забезпечити функціонування навіть при тимчасовій втраті мережевого підключення та підвищити масштабованість і здатність до самооновлення моделей. Таким чином, побудова комплексної моделі гібридного захисту дозволяє сформувати баланс між ефективністю, гнучкістю та продуктивністю, що є критичним у контексті динамічно змінюваних умов мобільного середовища та зростаючої складності кібер-атак на API-додатки.

Висновки. У результаті проведеного дослідження було проаналізовано особливості захисту API мобільних додатків в умовах обмеженої обчислювальної інфраструктури. Розроблено комплексну методику, що враховує апаратні характеристики мобільних пристроїв, сценарії розміщення компонентів безпеки, а також адаптивне використання моделей машинного навчання відповідно до доступних ресурсів. Розглянуто основні стратегії розміщення інтелектуальних компонентів безпеки у рамках локального та хмарного виконання, включаючи можливості застосування легковагових архітектур і технік стиснення моделей. Запропоновано критерії вибору між локальним аналізом та делегуванням задач на хмарні платформи з урахуванням параметрів затримки, енергоспоживання, пропускної здатності та рівня загроз. Сформовано архітектурну модель гібридного захисту API, яка передбачає багаторівневу систему, що складається з процедур аутентифікації, авторизації, шифрування, моніторингу, фільтрації запитів, а також модульну інтеграцію алгоритмів виявлення загроз.

Таким чином, результати дослідження можуть бути покладені в основу створення адаптивних систем безпеки API мобільних додатків, здатних до ефективної роботи навіть в умовах обмежених ресурсів, забезпечуючи при цьому гнучкість і масштабованість відповідно до змін середовища та характеристик загроз.

Список використаних джерел:

1. Acosta-Prado J. C., Rojas J. Rincón S., Mejía A. Martínez M., Riveros A. Tarazona R. Trends in the literature about the adoption of digital banking in emerging economies: A bibliometric analysis. *Journal of Risk and Financial Management*. 2024. No 17(12). DOI: https://doi.org/10.3390/jrfm17120545

2. Alshamrani A., Myneni S., Chowdhary A., Huang D. A survey on advanced persistent threats: Techniques, solutions, challenges, and research opportunities. *IEEE Communications Surveys & Tutorials*. 2019. No 21(2). P. 1851–1877. DOI: https://doi.org/10.1109/COMST.2018.2869441

3. Alzubaidi A., Kalutarage H., Wills G. B. Edge AI architectures for Internet of Things applications: A survey. *Smart Systems and Resilient Technologies*. 2023. No 5. DOI: https://doi.org/10.1016/j.ssrt.2023.100038

4. Beldachi R., Sallabi F., El Khatib H. Lightweight security solutions for resource-constrained mobile devices. *International Journal of Network Security & Its Applications (IJNSA)*. 2018. No 10(3). P. 11–25.

5. Dantas P. V., da Silva W. Jr S., Cordeiro L. C., Carvalho C. B. A comprehensive review of model compression techniques in machine learning. *Applied Intelligence*. 2024. Vol. 54. P. 11804–11844. DOI: https://doi.org/10.1007/s10489-024-05747-w

6. Enck W., Gilbert P., Chun B.-G., Cox L. P., Jung J., McDaniel P., Sheth Taint A. Droid: An information-flow tracking system for realtime privacy monitoring on smartphones. In: *Proceedings of the 9th USENIX Symposium on Operating Systems Design and Implementation (OSDI '10)*. Berkeley: USENIX Association, 2010. P. 1–16.

7. Gupta A., Lee S. Client-side versus server-side vulnerabilities in mobile APIs: A comparative study. *Journal of Systems Architecture*. 2021. Vol. 115. DOI: https://doi.org/10.1016/j.sysarc.2021.102061

8. Gupta P., Sandhu A. A review on API security challenges and solutions in modern web applications. *Journal of Network and Computer Applications*. 2023. Vol. 213. DOI: https://doi.org/10.1016/j.jnca.2022.103504

9. Haris N., Chen K., Song A., Pou B. Finding vulnerabilities in mobile application APIs: A modular programmatic approach. *arXiv preprint* : website. 2023. DOI: https://doi.org/10.48550/arXiv.2310.14137

10. Khan R., Othman M., Madani S. A., Khan S. U. A survey of mobile cloud computing application models. *IEEE Communications Surveys & Tutorials*. 2014. Vol. 16(1). P. 393–413. DOI: https://doi.org/10.1109/SURV.2013.052313.00134

11. Kumar A., Sethi N. Digital transformation trends in service industries: A systematic review. *International Journal of Service Science, Management, Engineering and Technology*. 2022. Vol. 13(1). P. 45–60.

12. Kumar P., Singh R. Mobile-Edge and Cloud-Based M. Hybrid L. Models for Secure API Ecosystems. *International Journal of Network Security*. 2021. No 23(4). P. 667–680. DOI: https://doi.org/10.6633/IJNS.202104_23(4).01

13. Li X., Zhao J. Edge-based versus cloud-based ML for real-time anomaly detection in mobile services. *ACM Transactions on Internet Technology*. 2019. No 19(1). DOI: https://doi.org/10.1145/3311699

14. Liu D., Zhu Y., Liu Z., Liu Y., Han C., Tian J., Li R., Yi W. A survey of model compression techniques: past, present, and future. *Frontiers in Robotics and AI*. 2025. No 12. DOI: https://doi.org/10.3389/frobt.2025.1518965

15. Liu D., Zhu Y., Zhang Z. et al. A survey of model compression techniques: past, present, and future. *Frontiers in Robotics and AI*. 2025. No 12.

16. Meddeb A. API security: Why it's more important than ever. *Computer Fraud & Security*. 2020. No 5. P. 8–11. DOI: https://doi.org/10.1016/S1361-3723(20)30057-7

17. OWASP Foundation. OWASP Top 10 API Security Risks – 2023. OWASP Foundation, 2023. 50 p.

18. Pal S., Misra S. Security challenges in mobile–cloud integrated systems: A survey. *IEEE Communications Surveys & Tutorials*. 2022. No 24(3). P. 1873–1897. DOI: https://doi.org/10.1109/COMST.2021.3124843

19. Paul C. Mobile app personalization using machine learning algorithms. *International Journal of Advanced Computer Science & Applications (IJACSA)*. 2023. No 14(7). P. 205–218.

20. Shostack A. Threat Modeling: Designing for Security. Hoboken: Wiley, 2014. 624 p.

21. Skosana S., Mlambo S., Madiope T., Thango B. Evaluating wireless network technologies (3G, 4G, 5G) and their infrastructure: A systematic review. *SSRN Electronic Journal*. 2024. https://doi.org/10.2139/ssrn.4992432

22. Souppaya M., Scarfone K. Guide to Data-Centric System Threat Modeling (NIST SP 800-154, Initial Public Draft). Gaithersburg: National Institute of Standards and Technology, 2016. 65 p.

23. Suwannaphong T., Jovan F., Craddock I., McConville R. Optimising TinyML with quantization and distillation of transformer and Mamba models for indoor localisation on edge devices. *arXiv preprint* : website. 2024. DOI: https://doi.org/10.48550/arXiv.2412.09289

24. Suwannaphong T., Jovan F., Craddock I., McConville R. Optimising TinyML with quantization and distillation of transformer and Mamba models for indoor localisation on edge devices. *Internet of Things and Cyber-Physical Systems*. 2024. No 4. DOI: https://doi.org/10.1016/j.iotcps.2023.100086

25. Teodorescu C. A., Durnoi A. N., Vargas V. M. The rise of the mobile Internet: Tracing the evolution of portable devices. *Proceedings of the International Conference on Business Excellence*. 2023. No 17(1). P. 1645–1654. DOI: https://doi.org/10.2478/picbe-2023-0147

26. World Health Organization, European Commission. Assessing the impact of digital transformation of health services. *Expert Panel Opinion. Luxembourg: Publications Office of the European Union*, 2019. 120 p.

27. Zhang C., Patras P. Long-term mobile traffic forecasting using deep spatio-temporal neural networks. *arXiv preprint* : website. 2017. URL: https://arxiv.org/abs/1712.08083 (last accessed: 18.09.2025).

28. Zhang H., Huang J. Challenging GPU dominance: When CPUs outperform for on-device LLM inference. *arXiv* : website. 2025. DOI: https://doi.org/10.48550/arXiv.2505.06461

29. Zhang Y., Wang L. Machine learning–driven API threat detection: Methods and opportunities. *Journal of Computer Security*. 2020. No 28(6). P. 773–795. DOI: https://doi.org/10.3233/JCS-200457

Дата надходження статті: 18.09.2025
Дата прийняття статті: 20.10.2025
Опубліковано: 04.12.2025