НАЦІОНАЛЬНИЙ УНІВЕРСТИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

ПОГОДЖЕНО	**ДОПУСКАЄТЬСЯ ДО ЗАХИСТУ**
Декан факультету (Директор ННІ)	**Завідувач кафедри**
інформаційних технологій	комп'ютерних наук
(назва факультету (ННІ))	(назва кафедри)
_____ Ігор Болбот	_____ Белла Голуб
(підпис) (ім'я ПРІЗВИЩЕ)	(підпис) (ім'я ПРІЗВИЩЕ)
"___" _____ 2025 р.	"___" _____ 2025р.

МАГІСТЕРСЬКА КВАЛІФІКАЦІЙНА РОБОТА

на тему Застосування алгоритмів інтелектуального аналізу для захисту від кібератак в умовах постквантової ери

Спеціальність 121 «Інженерія програмного забезпечення»
(код і найменування)

Освітня програма Програмне забезпечення інформаційних систем
(назва)

Орієнтація освітньої програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Гарант освітньої програми

к.ф.-м.н., доцент _____ Віктор Кириченко
(науковий ступінь та вчене звання) (підпис) (ім'я ПРІЗВИЩЕ)

Керівник магістерської кваліфікаційної роботи

к.ф.-м.н., доцент _____ Хиленко В.В
(науковий ступінь та вчене звання) (підпис) (ім'я ПРІЗВИЩЕ)

Виконав _____ Андрій ЛАВРЕНЧУК
(підпис) (ім'я ПРІЗВИЩЕ здобувача)

КИЇВ – 2025

НАЦІОНАЛЬНИЙ УНІВЕРСТИТЕТ БІОРЕСУРСІВ І ПРИРОДОКОРИСТУВАННЯ УКРАЇНИ

Факультет інформаційних технологій

ЗАТВЕРДЖУЮ
Завідувач кафедри комп'ютерних наук
к.т.н., доцент ____ Белла Голуб
(науковий ступінь, вчене звання) (підпис) (ім'я ПРІЗВИЩЕ)
"01" листопада 2024 року

З А В Д А Н Н Я

ДО ВИКОНАННЯ МАГІСТЕРСЬКОЇ КВАЛІФІКАЦІЙНОЇ РОБОТИ ЗДОБУВАЧУ

Лавренчук Андрій Валентинович
(прізвище, ім'я, по батькові)

Спеціальність 121 «Інженерія програмного забезпечення»
(код і найменування)

Освітня програма Програмне забезпечення інформаційних систем
(назва)

Орієнтація освітньої програми освітньо-професійна
(освітньо-професійна або освітньо-наукова)

Тема магістерської кваліфікаційної роботи Застосування алгоритмів інтелектуального аналізу для захисту від кібератак в умовах постквантової ери
затверджена наказом від "01" листопада 2024р. №1963 «С»

Термін подання завершеної роботи на кафедру 20.11.2025
(рік, місяць, число)

Вихідні дані до магістерської кваліфікаційної роботи _ є телеметричні журнали подій, набори інцидентів безпеки та параметри постквантових криптографічних алгоритмів Kyber і Dilithium. Також використано зібрані датасети для навчання моделей виявлення аномалій та специфікації вимог до архітектури системи кіберзахисту.

Перелік питань, що підлягають дослідженню:

1. Методи ML-виявлення аномалій.
2. Застосування алгоритмів Kyber і Dilithium.
3. Архітектура системи постквантового захисту.
4. Побудова інформаційної бази та OLAP-кубу.
5. Алгоритмізація програмних модулів.
6. Методи тестування ефективності системи.

Дата видачі завдання "01" листопада 2024 р.

Керівник магістерської кваліфікаційної роботи _____ Хиленко В.В.
(підпис) (ім'я ПРІЗВИЩЕ)

Завдання прийняв до виконання _____ Андрій ЛАВРЕНЧУК
(підпис) (ім'я ПРІЗВИЩЕ)

ЗМІСТ

ПЕРЕЛІК УМОВНИХ СКОРОЧЕНЬ

1. API — Application Programming Interface
2. CEP — Complex Event Processing
3. DB — DataBase
4. DSA — Digital Signature Algorithm
5. GNN — Graph Neural Network
6. IDS — Intrusion Detection System
7. KEM — Key Encapsulation Mechanism
8. KPI — Key Performance Indicator
9. KMS — Key Management System
10. LMS — Learning Management System
11. ML — Machine Learning
12. OLAP — On-Line Analytical Processing
13. PCA — Principal Component Analysis
14. PQC — Post-Quantum Cryptography
15. PKI — Public Key Infrastructure
16. REST — Representational State Transfer
17. SIEM — Security Information and Event Management
18. SOAR — Security Orchestration, Automation and Response
19. SOC — Security Operations Center
20. TLS — Transport Layer Security
21. OIDC — OpenID Connect

ВСТУП

Цифрова інфраструктура перебуває в умовах зростаючих ризиків, пов’язаних із появою постквантових обчислювальних технологій, які здатні порушити криптографічні основи більшості існуючих систем захисту. Квантові комп’ютери відкривають нові можливості для розв’язання складних задач, однак одночасно створюють реальну загрозу компрометації алгоритмів шифрування RSA, ECC та інших асиметричних схем, що лежать в основі сучасних протоколів безпеки [1]. У цих умовах постає потреба у впровадженні адаптивних інтелектуальних методів аналізу даних, здатних виявляти, прогнозувати та запобігати кіберзагрозам навіть за умов порушення традиційних криптографічних гарантій.

Актуальність дослідження полягає у необхідності розроблення нових підходів до захисту інформаційних систем, що базуються не лише на класичних криптографічних засобах, а й на методах інтелектуального аналізу даних (Data Mining, Machine Learning, Deep Learning). Такі підходи дозволяють здійснювати поведінкове профілювання трафіку, автоматичне виявлення аномалій, класифікацію подій безпеки та формування рішень у режимі реального часу. Інтеграція інтелектуальних алгоритмів у системи захисту створює основу для проактивної кібероборони, що критично важливо в умовах розвитку постквантових атак [2].

Метою роботи є розроблення та дослідження системи захисту від кібератак, що використовує алгоритми інтелектуального аналізу даних для адаптивного виявлення та попередження загроз у постквантовій обчислювальній парадигмі.

Для досягнення поставленої мети необхідно розв’язати такі **завдання дослідження**:

1. провести системний аналіз предметної області кіберзахисту з урахуванням нових ризиків, що виникають у постквантовій ері.

2. Дослідити сучасні алгоритми інтелектуального аналізу даних, що застосовуються для виявлення кібератак та аномальної поведінки.

3. Розробити архітектуру інтелектуальної системи моніторингу та реагування на події безпеки.

4. Реалізувати програмні модулі для аналізу, класифікації та кореляції подій за допомогою машинного навчання.

5. Оцінити ефективність запропонованих алгоритмів і перевірити їх здатність до роботи в умовах постквантових викликів.

Об’єктом дослідження є процес забезпечення кібербезпеки інформаційних систем у розподіленому середовищі з підвищеними вимогами до криптостійкості.

Предметом дослідження є алгоритми інтелектуального аналізу даних та їх застосування для виявлення, прогнозування і запобігання кібератакам у постквантовій обчислювальній парадигмі.

Методи дослідження включають методи машинного навчання (класифікація, кластеризація, нейронні мережі), математичну статистику, методи аналізу часових рядів і поведінкової аналітики, а також алгоритми постквантової криптографії (CRYSTALS-Kyber, Dilithium, SPHINCS+). Для перевірки ефективності застосовуються експериментальні обчислення, імітаційне моделювання та порівняльний аналіз результатів.

Наукова новизна роботи полягає у створенні інтегрованої моделі системи кіберзахисту, яка поєднує методи інтелектуального аналізу даних і постквантові криптографічні механізми для формування багаторівневої стратегії безпеки. На відміну від існуючих підходів, запропонована модель забезпечує динамічну адаптацію до змінних сценаріїв атак і здатна навчатися на основі зібраних подій безпеки, підвищуючи точність виявлення аномалій без зниження продуктивності системи.

1 СИСТЕМНИЙ АНАЛІЗ ПРЕДМЕТНОЇ ОБЛАСТІ

1.1 Опис предметної області

Предметна область системи захисту від кібератак у постквантовій ері охоплює сукупність технологій, методів і процесів, спрямованих на забезпечення стійкості інформаційних інфраструктур до загроз нового покоління, що виникають унаслідок розвитку квантових обчислень та інтелектуальних атак. Зі зростанням обчислювальних можливостей квантових процесорів традиційні криптографічні схеми - RSA, ECC, Diffie-Hellman - втрачають свою ефективність, що зумовлює необхідність переходу до постквантових методів шифрування, таких як CRYSTALS-Kyber, Dilithium або SPHINCS+. Паралельно із криптографічними викликами суттєво зростає складність мережевих атак, які використовують штучний інтелект для обходу захисних бар’єрів, і це потребує впровадження адаптивних, навчальних систем аналізу загроз [1].

Інтелектуальні системи кіберзахисту функціонують у середовищі великих потоків подій, логів і телеметрії, де необхідне автоматичне виявлення закономірностей, відхилень і прихованих аномалій. Ключовими завданнями є збір та нормалізація подій із різних джерел, формування поведінкових профілів користувачів, класифікація мережевої активності, а також оркестрація дій у відповідь. Система повинна забезпечувати не лише реактивне блокування загроз, а й проактивне прогнозування атакових сценаріїв шляхом побудови графів взаємозалежностей і кореляцій подій. Взаємодія між підсистемами аналітики, реагування й контролю відображена на рис. 1.1, що ілюструє цілісну архітектуру системи в контексті постквантового середовища.

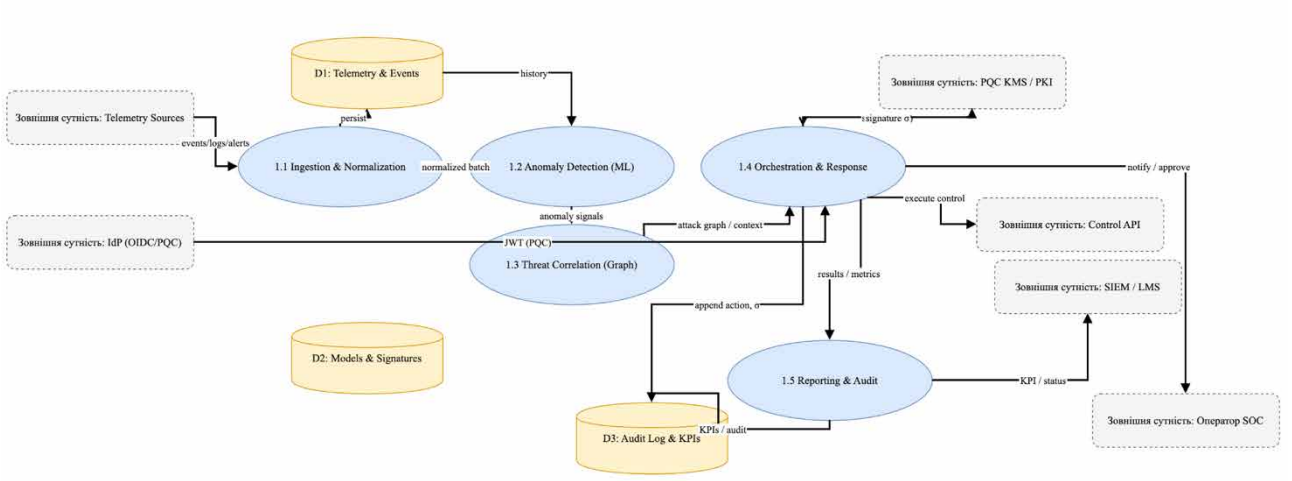

Рис. 1.1. Архітектура інтелектуальної системи захисту від кібератак у постквантовій ері

У межах предметної області визначальне місце посідають джерела телеметрії, системи керування ідентичностями, бази моделей аномалій, постквантові сервіси керування ключами, а також модулі аудиту й звітності. Дані компоненти формують ядро багаторівневої системи, у якій алгоритми машинного навчання поєднуються з постквантовими криптографічними протоколами для створення комплексної системи виявлення, аналізу та реагування на загрози в режимі реального часу [2].

Основні сутності предметної області узагальнено в табл. 1.1, де наведено їхні функції, взаємозв’язки та ролі у забезпеченні стійкості системи до постквантових атак.

Таблиця 1.1

Основні сутності предметної області системи кіберзахисту

№	Сутність	Опис функціонального призначення
1	Джерела телеметрії	Формують потоки подій, логів і сигналів для аналітичних модулів
2	Ідентифікаційні сервіси (OIDC/PQC)	Забезпечують автентифікацію з використанням постквантових механізмів
3	Сховище подій	Зберігає нормалізовані дані та історію атак для подальшого аналізу
4	Модуль виявлення аномалій	Виконує машинне навчання для розпізнавання відхилень у поведінці системи

Продовження таблиці 1.1

5	Підсистема кореляції загроз	Формує графи взаємопов’язаних подій для виявлення комплексних атак
6	Оркестрація реагування	Автоматизує дії із запобігання, блокування та відновлення після атак
7	Аудит і звітність	Формує аналітичні метрики, KPI і звіти для операторів SOC

Предметна область дослідження охоплює інтеграцію методів інтелектуального аналізу даних, машинного навчання та постквантової криптографії для формування нової парадигми кіберзахисту, орієнтованої на самонавчання, адаптацію та превентивне виявлення загроз у цифровому середовищі з високим рівнем невизначеності.

1.2 Огляд інформаціних джерел та існуючих рішень

Сучасний етап розвитку кібербезпеки характеризується інтеграцією двох ключових напрямів - інтелектуального аналізу даних (AI/ML) та постквантової криптографії (PQC). Ці підходи формують основу нової парадигми захисту, де класичні системи виявлення вторгнень (IDS/IPS) поступово трансформуються в адаптивні когнітивні архітектури, здатні до самонавчання, прогнозування та реагування на атаки.

Дослідження Hdaib M. et al. (2024) доводить ефективність квантово-глибоких автоенкодерів у задачах виявлення аномалій. Запропонована модель використовує квантову компресію станів для вилучення латентних ознак і зменшення вимірності даних без втрати інформаційного контексту. Сутність процесу ілюструє рис. 1.2, де блок Евиконує кодування вхідного стану, а D - декодування компресованого стану до вихідного простору.

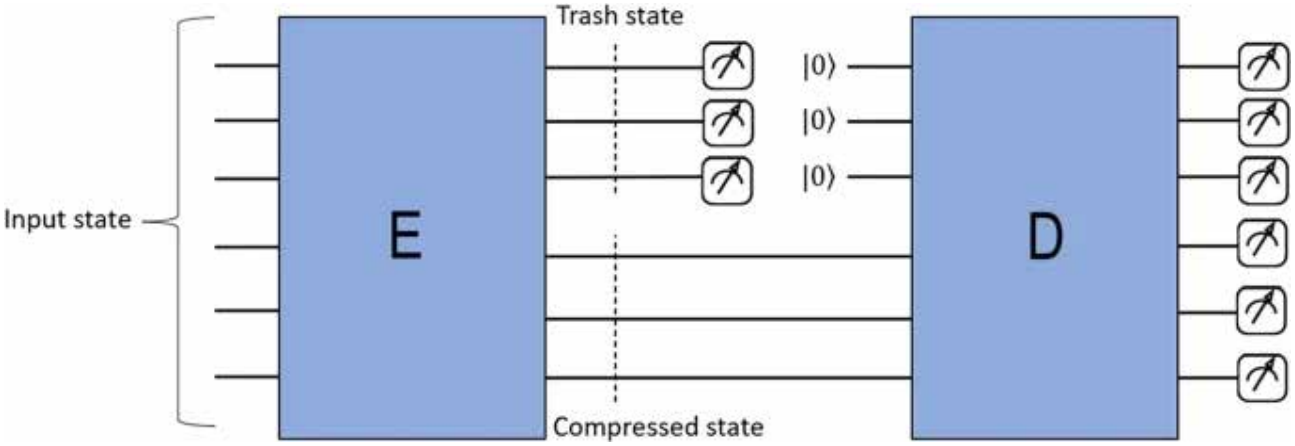

Рис. 1.2. Схема квантового автоенкодера для виявлення аномалій [Hdaib M. et al., 2024]

Паралельно дослідження Corli S. et al. (2024) у роботі “Quantum Machine Learning Algorithms for Anomaly Detection” показує, що гібридні квантові нейронні мережі дозволяють зменшити час тренування моделей для потокового трафіку за рахунок паралельного виконання квантових гейт-операцій. У рис. 1.3 подано типову квантову схему параметризованого кола (VQC), яка формує навчальну модель для класифікації станів системи за допомогою обертальних операторів R_z та взаємозв’язків типу ZZ.

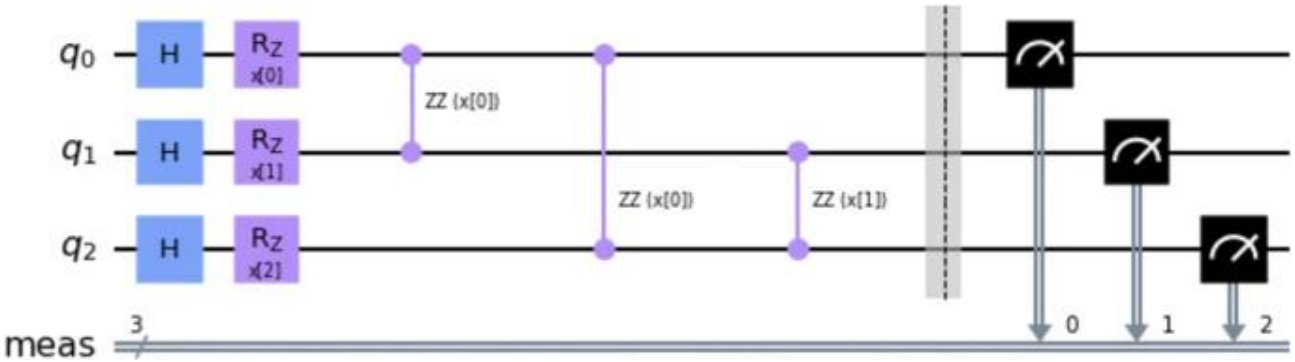

Рис. 1.3. Квантове варіаційне коло для машинного навчання в кіберзахисті [Corli S. et al., 2024]

У контексті криптографії дослідження Cherkaoui Dekkaki K. et al. (2024) аналізує динаміку розвитку стандартів NIST PQC, де переважають ґраткові алгоритми (Lattice-based), що забезпечують високу стійкість до квантових атак. У рис. 1.4 наведено статистичний розподіл алгоритмів за раундами відбору

NIST, який демонструє домінування LWE-та NTRU-підходів порівняно з кодовими або ізогенієвими схемами.

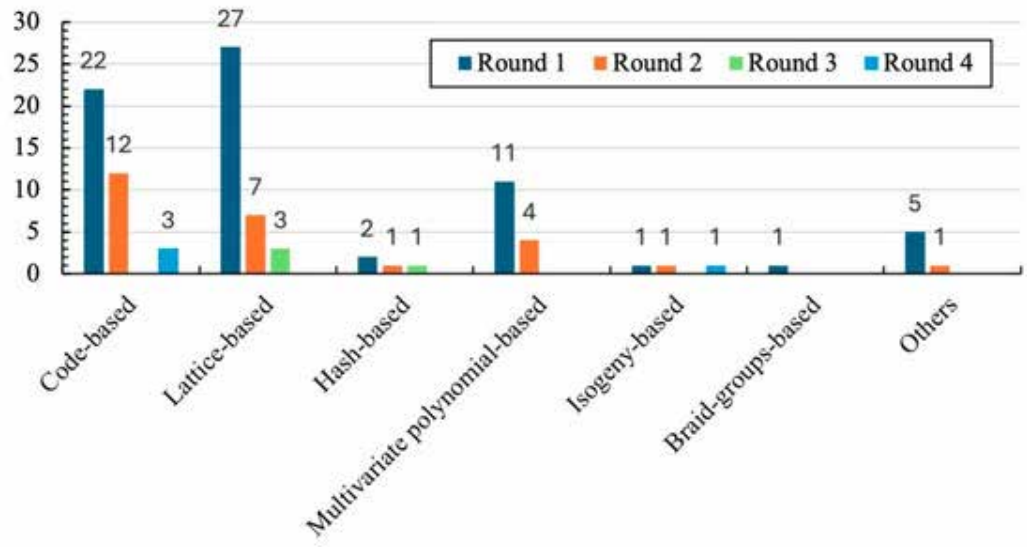

Рис. 1.4. Класифікація постквантових криптографічних алгоритмів за раундами NIST [Cherkaoui Dekkaki K. et al., 2024]

Класична криптографічна модель передачі даних із цифровим підписом наведена на рис. 1.5, що відображає процес хешування повідомлення, підписання приватним ключем і верифікації відкритим ключем. Цей підхід, детально розглянутий Buczak A. L. та Guven E. (2016), залишається основою побудови PQC-схем із використанням стійких хеш-функцій і квантово-безпечних ключових обмінів.

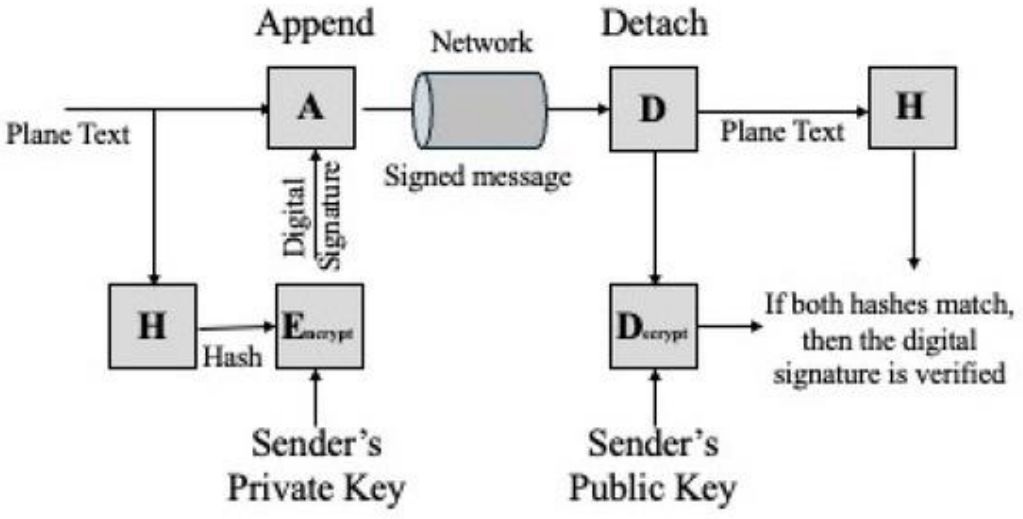

Рис. 1.5. Схема формування та перевірки цифрового підпису в класичній криптографії [Buczak A. L., Guven E., 2016]

Подальший розвиток концепції цифрових сертифікатів для PQC-середовища проаналізовано в роботі D. Tasic et al. (2024), де представлено дворівневу архітектуру отримання та верифікації сертифікатів (рис. 1.6). Вона базується на інтеграції сертифікаційних органів (CA) із службами відклику сертифікатів (CRL) та можливістю використання ідентифікаційних сервісів OIDC/PQC, що забезпечує захищену ідентифікацію користувачів у постквантовій інфраструктурі.

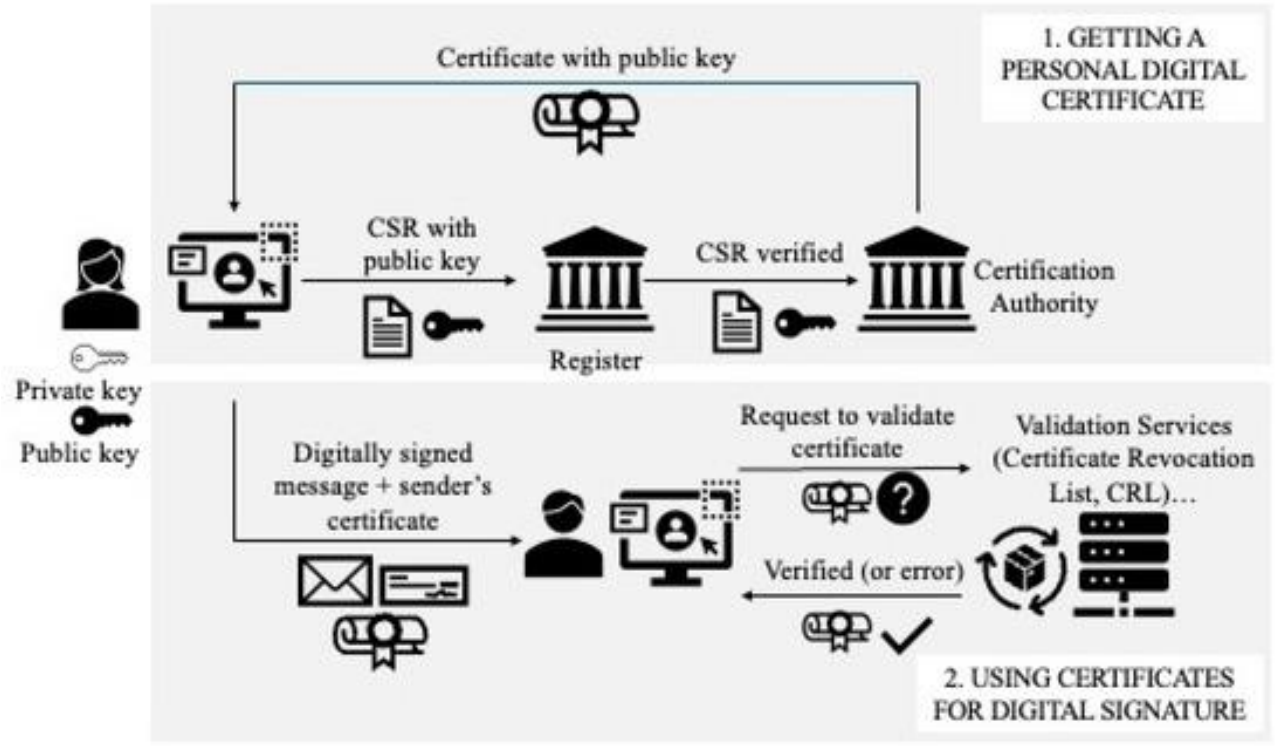

Рис. 1.6. Архітектура видачі та валідації цифрових сертифікатів у постквантових системах [Tasic D. et al., 2024]

Узагальнення наведених досліджень наведено у табл. 1.2, що демонструє еволюцію напрямів і ключові наукові акценти у сфері інтелектуального аналізу даних і постквантового захисту.

Таблиця 1.2

Порівняльний аналіз сучасних підходів до інтелектуального та постквантового кіберзахисту

№	Джерело	Методологічний підхід	Науковий внесок
1	Hdaib M. et al., 2024	Квантові автоенкодери	Зменшення вимірності даних, компресія станів

Продовження таблиці 1.2

2	Corli S. et al., 2024	Квантові варіаційні кола	Навчання моделей для аномалій із високою швидкодією
3	Cherkaoui Dekkaki K. et al., 2024	PQC-алгоритми NIST	Аналіз ефективності ґраткових і хеш-базованих методів
4	Buczak A. L., Guven E., 2016	Цифровий підпис і ML-виявлення атак	Основи побудови сигнатурних моделей IDS
5	Tasic D. et al., 2024	Постквантові сертифікаційні протоколи	Валідація ідентичності в PQC-інфраструктурах

Проведений аналіз свідчить, що сучасні наукові розробки тяжіють до інтеграції квантових і машинних методів у сфері кіберзахисту. Існуючі рішення мають обмеження у масштабованості, стійкості до нових типів атак та відсутності єдиної архітектурної моделі об’єднання PQC і ML-модулів.

Наукова новизна даної роботи полягає у розробленні інтегрованої моделі системи захисту від кібератак, що поєднує постквантові криптографічні алгоритми (Kyber, Dilithium) із інтелектуальними модулями виявлення аномалій на основі машинного навчання. Запропонований підхід забезпечує адаптивну реакцію системи в реальному часі, підвищує ймовірність точного виявлення атак і формує архітектурну основу для безпечної взаємодії в постквантовій епосі.

1.3 Аналіз існуючих рішень та інформаційних джерел

Сучасний ринок кіберзахисних технологій демонструє активний розвиток систем, що поєднують інтелектуальний аналіз даних та постквантові криптографічні механізми. Провідні корпорації та відкриті спільноти впроваджують гібридні архітектури, здатні забезпечити перехід до квантово-безпечних обчислень і підвищити ефективність виявлення загроз у режимі реального часу.

Одним із ключових напрямів розвитку є ініціатива IBM Quantum Safe, що формує комплексну дорожню карту переходу до постквантової безпеки. Як

показано на рис. 1.7, компанія IBM поетапно впроваджує алгоритми CRYSTALS-Kyber і Dilithium, інструменти аналізу залежностей CBOM, бібліотеки OpenSSL PQC-сумісності та сервіси IBM Quantum Safe Remediator для управління ключами й сертифікатами [1].

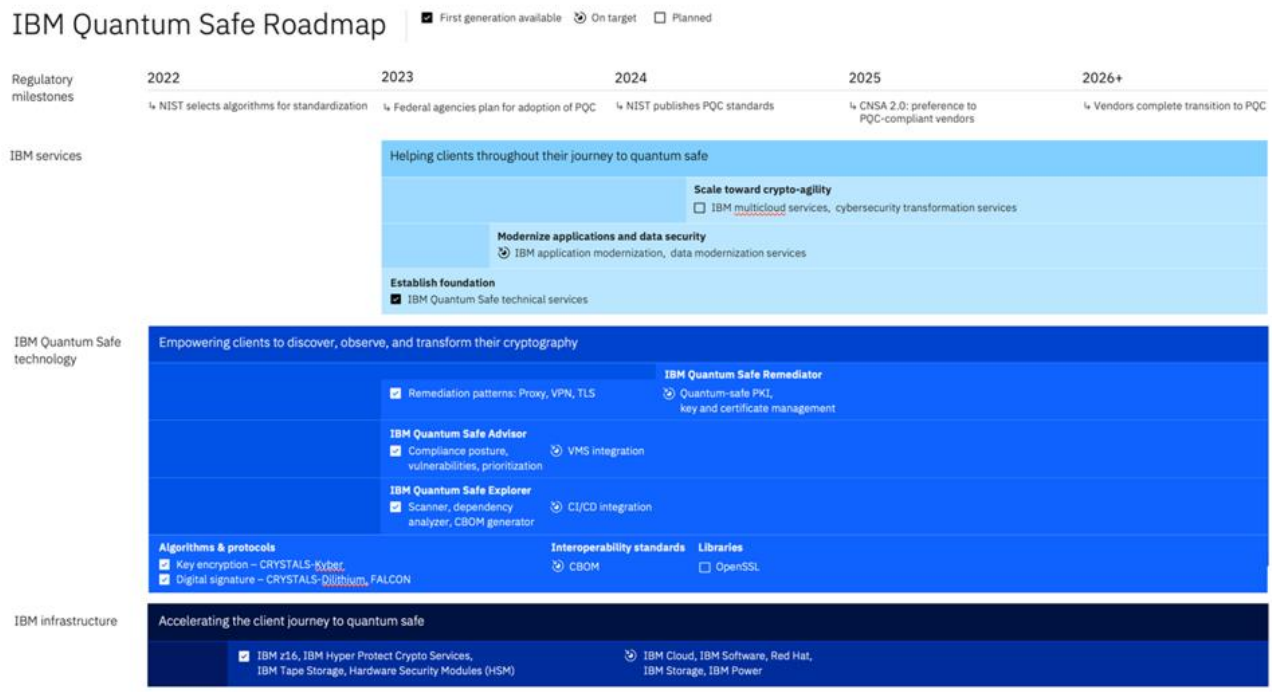

Рис. 1.7. IBM Quantum Safe Roadmap: етапи впровадження постквантових рішень у корпоративних системах

Іншим важливим рішенням є TensorFlow Quantum - бібліотека від Google для побудови гібридних квантово-класичних моделей машинного навчання. На рис. 1.8 представлено інтерфейс бібліотеки, що інтегрує квантові кола Cirq із класичними нейронними шарами Keras. Таке поєднання дозволяє створювати квантові автоенкодери для виявлення складних аномалій у потоках подій безпеки та оптимізувати поведінкові моделі IDS-систем [2].

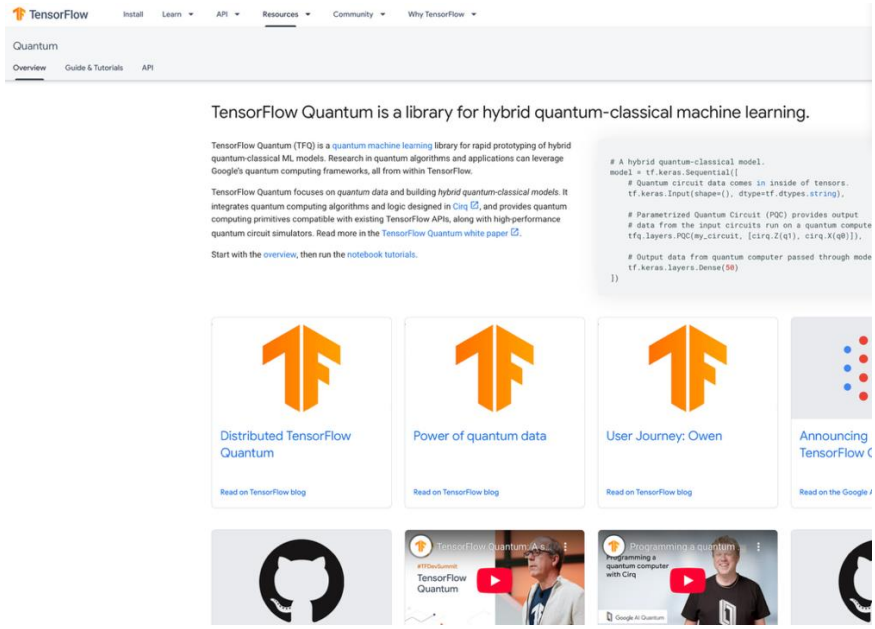

Рис. 1.8. TensorFlow Quantum - середовище для гібридного квантово-класичного машинного навчання

Платформа Microsoft Azure Quantum Security (рис. 1.9) забезпечує розгортання симуляційних середовищ і тестування квантових моделей у хмарній інфраструктурі. Вона дозволяє інтегрувати Q#-алгоритми для криптографічних перевірок і обчислення ризиків у розподілених системах. Такий підхід сприяє розробці динамічних систем реагування на загрози, що враховують поведінкові фактори та постквантову криптостійкість [3].

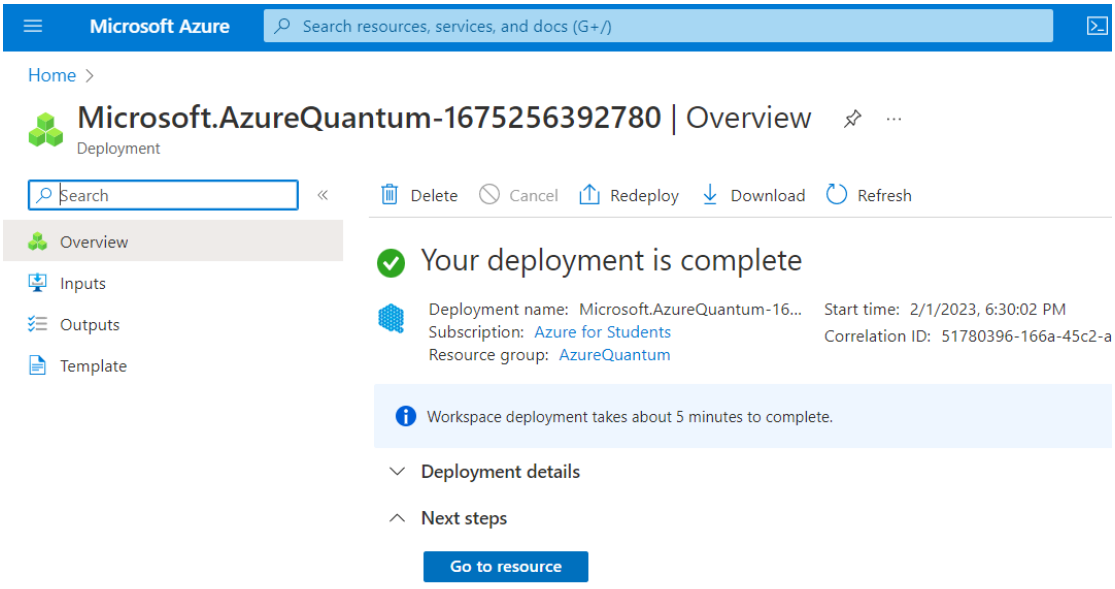

Рис. 1.9. Інтерфейс середовища Microsoft Azure Quantum Security Framework

Відкритий проєкт Open Quantum Safe (OQS), що підтримується Linux Foundation, надає бібліотеку liboqs і набір інструментів для інтеграції квантово-стійких алгоритмів у протоколи SSL/TLS. Як показано на рис. 1.10, платформа активно розвивається спільнотою дослідників і компаній, забезпечуючи стандартизовану реалізацію PQC-алгоритмів у промислових системах [4].

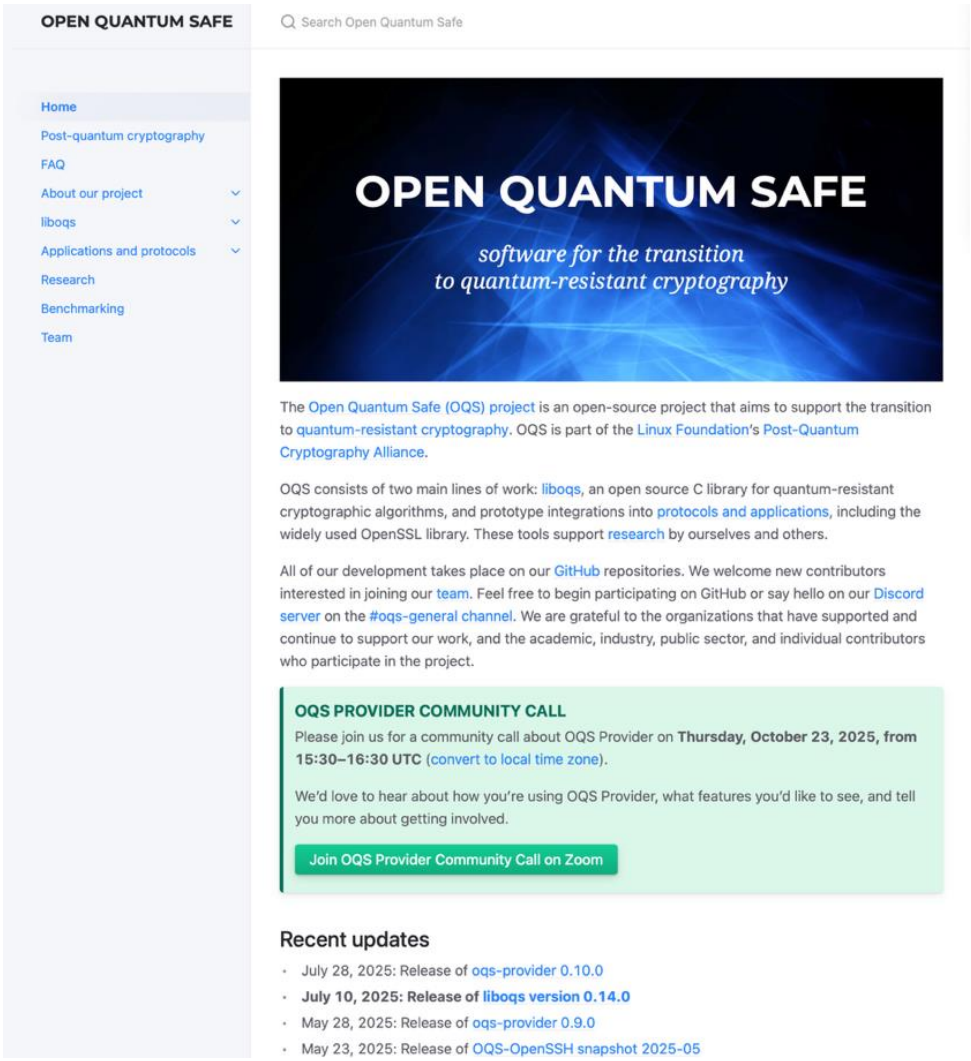

Рис. 1.10. Інтерфейс платформи Open Quantum Safe для реалізації PQC-алгоритмів

П’ятим значним гравцем є система Darktrace Immune System, що застосовує глибоке навчання для виявлення аномальної поведінки у корпоративних мережах. Як показано на рис. 1.11, система відображає у реальному часі глобальні взаємодії між об’єктами, будуючи когнітивні карти загроз. Її підхід базується на ідеї «цифрової імунної системи», де алгоритми безперервно адаптуються до нових патернів атак без ручного втручання [5].

Рис. 1.11. Інтерфейс системи Darktrace Immune System із візуалізацією поведінкових загроз у реальному часі

Порівняння наведених рішень із розроблюваною системою подано у табл. 1.4, де відображено ключові технічні характеристики, використані алгоритми та переваги підходу, запропонованого в цій роботі.

Таблиця 1.4

Порівняльна характеристика існуючих рішень та розроблюваної системи

№	Система / Платформа	Технологічна база	Особливості реалізації	Недоліки	Переваги нашої системи
1	IBM Quantum Safe	PQC (Kyber, Dilithium), PKI, TLS	Інтеграція в корпоративні сервіси IBM	Висока вартість, закритий код	Відкрита архітектура, модульність, підтримка REST-інтерфейсів
2	TensorFlow Quantum	QML, Cirq, Keras	Моделювання квантових автоенкодерів	Обмежена практична продуктивність	Оптимізована обробка телеметрії через ML-модулі
3	Microsoft Azure Quantum	Q#, Hybrid ML	Хмарна симуляція PQC-алгоритмів	Прив’язка до Azure-інфраструктури	Локальна реалізація та PQC-взаємодія без хмарної залежності

Продовження таблиці 1.4

4	Open Quantum Safe (OQS)	liboqs, OpenSSL	Відкрите PQC-сховище та бібліотеки	Не має ML-компонентів	Інтеграція PQC із системою виявлення аномалій
5	Darktrace Immune System	Deep Learning, Graph Analytics	Поведінковий аналіз атак	Відсутність постквантового захисту	Поєднання ML-виявлення з PQC-шифруванням у єдиній системі

Проведений аналіз свідчить, що наявні рішення забезпечують або постквантовий криптозахист, або інтелектуальний моніторинг подій безпеки, однак не реалізують комплексної інтеграції цих підходів. Саме тому наукова новизна розроблюваної системи полягає у створенні інтегрованого середовища захисту, яке поєднує PQC-механізми, ML-виявлення аномалій, поведінкову аналітику та автоматизоване реагування у межах єдиної архітектури. Це забезпечує адаптивність, криптостійкість і масштабованість системи в умовах постквантової ери.

1.4 Моделювання предметної області

Моделювання предметної області системи захисту від кібератак у постквантовій ері ґрунтується на принципах об’єктно-орієнтованого підходу, який дозволяє формалізувати взаємодію між користувачами, компонентами системи та зовнішніми сервісами. Основна мета моделювання - визначення ролей, процесів і інформаційних потоків, що забезпечують інтеграцію інтелектуального аналізу даних з постквантовими механізмами криптографічного захисту.

На рис. 1.12 подано діаграму прецедентів, що відображає ключові сценарії використання системи. Основними акторами є SOC-аналітик, інцидент-респондент, CISO, інженер з машинного навчання, а також зовнішні системи - SIEM, SOAR, PKI/KMS (PQC) і джерела Threat Intelligence. Головні прецеденти охоплюють виявлення аномалій у журналах і трафіку, кореляцію подій,

моніторинг криптопротоколів PQC, автоматизоване реагування (SOAR), аудит і звітність, а також скоринг постквантового ризику. Додаткові включення (<include>) та розширення (<extend>) демонструють взаємозалежності між інтелектуальними модулями системи й процесами управління інцидентами безпеки.

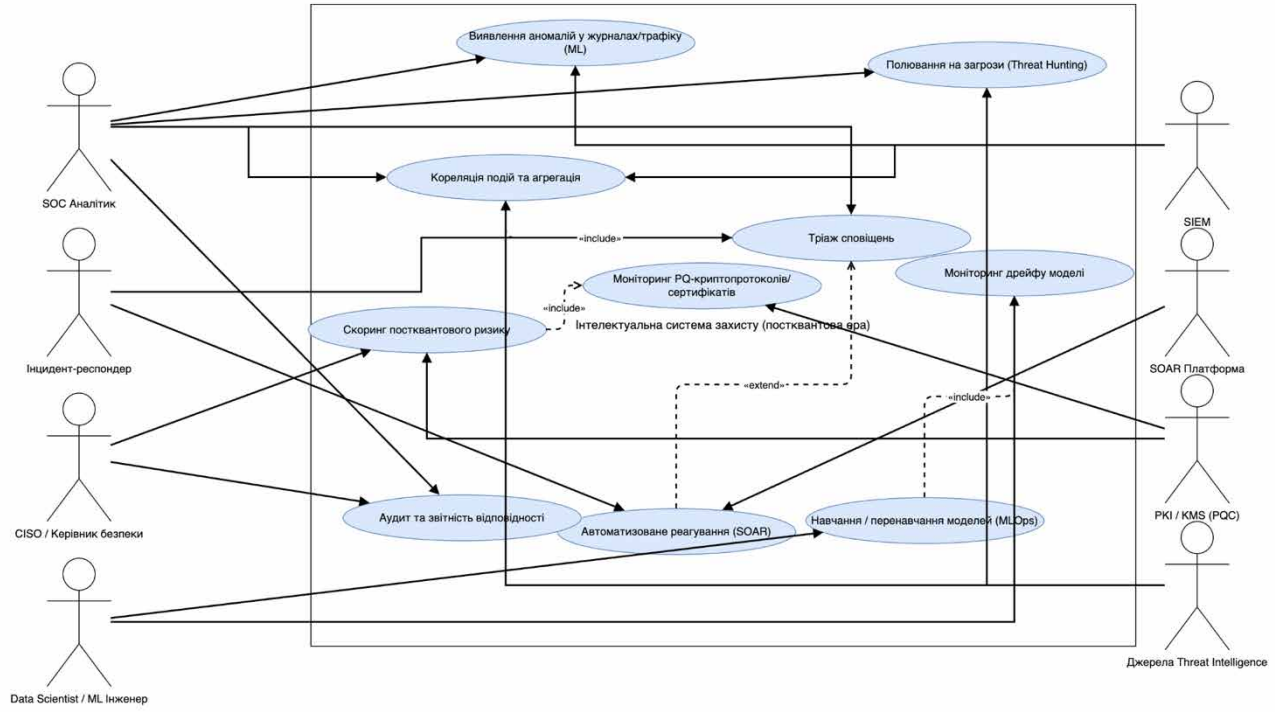

Рис. 1.12. Діаграма прецедентів системи інтелектуального захисту у постквантовій ері

На рис. 1.13 представлено діаграму послідовності, що формалізує обмін повідомленнями між компонентами під час виявлення та ізоляції аномалії. SOC-оператор ініціює процес обробки пакетів подій (batch), після чого модуль AnomalyDetector (ML) генерує звіт про інцидент із зазначенням рівня критичності. Компонент Response Orchestrator запитує підпис PQC через KMS/PKI, де формується криптографічний підпис за алгоритмом CRYSTALS-Dilithium, після чого через Control API виконується ізоляція активу. Завершення процесу підтверджується повідомленням про закриття інциденту. Така послідовність гарантує криптографічну автентичність команд реагування та захист від компрометації керуючих каналів.

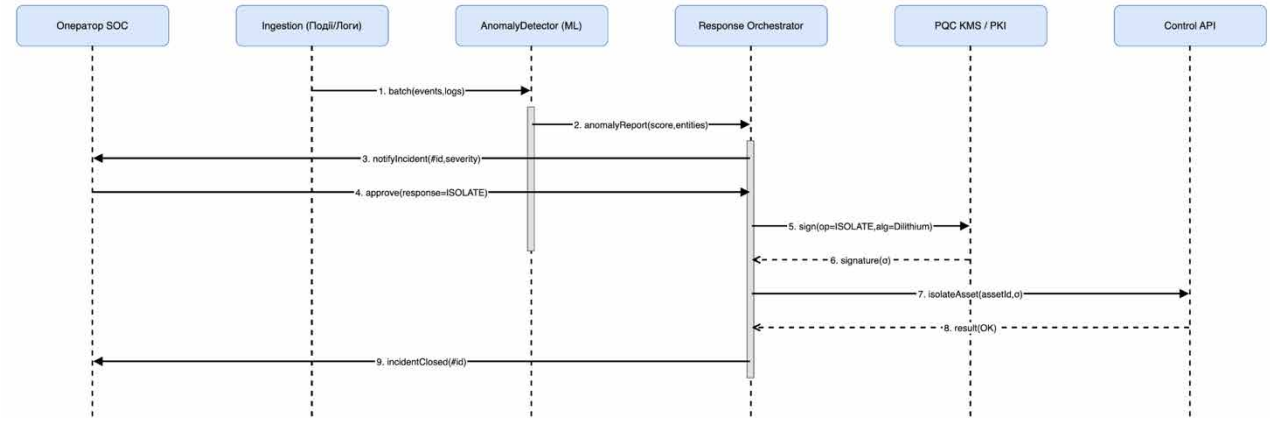

Рис. 1.13. Діаграма послідовності взаємодії компонентів системи при виявленні аномалій та реагуванні

Подальша деталізація бізнес-логіки відображена на рис. 1.14, який представляє діаграму активності системи. Вона охоплює три основні партиції: Дані/ML, Оркестрація/IR та Зовнішні сервіси PQC. Потік процесу починається з поглинання подій, нормалізації логів і виявлення аномалій за допомогою моделей машинного навчання. Якщо відхилення підтверджено, система виконує кореляцію подій, оцінку ризику (Kill-Chain, Severity) та створює інцидент. Далі ініціюється підпис дії через постквантовий модуль PQC, після чого Control API виконує ізоляцію активу. Результати записуються до незмінного журналу аудиту, а звіти надсилаються у SIEM/LMS. Такий підхід забезпечує замкнутий цикл моніторингу, аналізу й реагування, що відповідає концепції Adaptive Cyber Defense (ACD) [1].

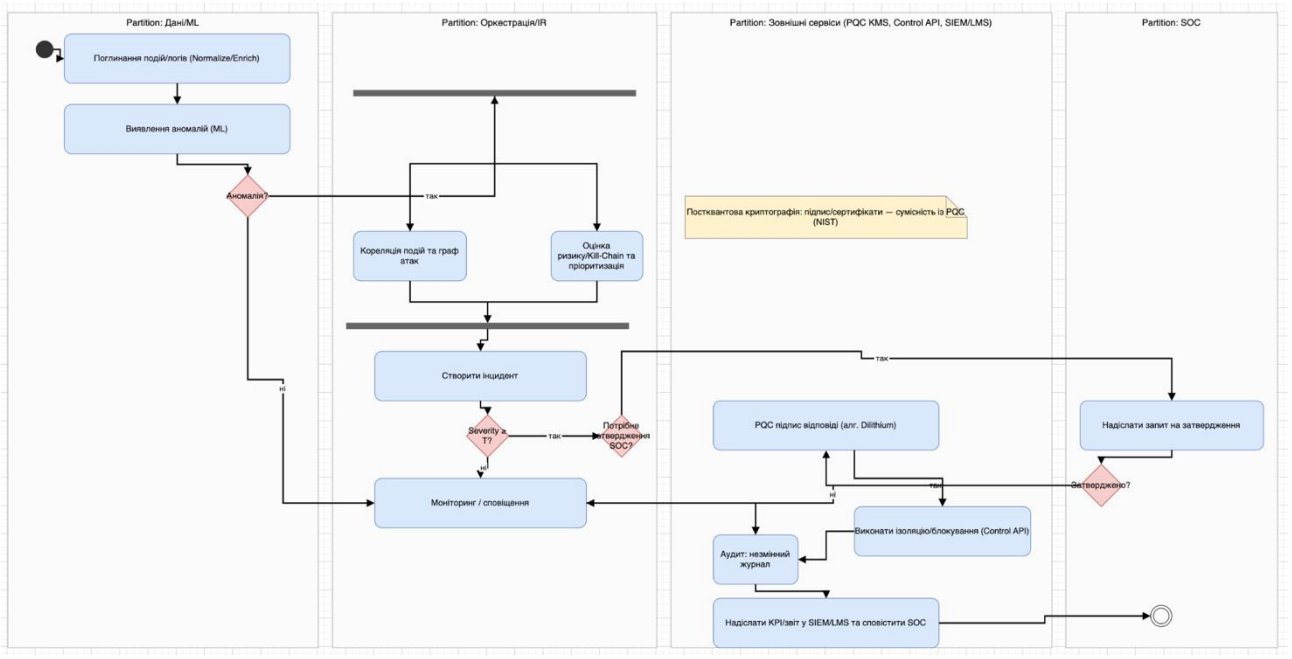

Рис. 1.14. Діаграма активності процесу оброблення подій, оркестрації реагування та підпису PQC

На основі моделювання було встановлено, що інтеграція ML-компонентів для поведінкового аналізу з PQC-модулями для автентифікації забезпечує не лише виявлення загроз, але й формує довірену модель реагування, що витримує квантові атаки. Система здатна адаптувати свої моделі через цикл MLOps, виконуючи перенавчання при зміні поведінкових патернів.

1.5 Аналіз вимог програмної системи

Для забезпечення ефективного функціонування інтелектуальної системи захисту від кібератак у постквантовій ері проведено структурований аналіз вимог, який охоплює функціональні, нефункціональні та вимоги до безпеки. Основна мета — визначити архітектурні, логічні та операційні обмеження, необхідні для інтеграції модулів машинного навчання, постквантової криптографії (PQC) і компонентів автоматизованого реагування (SOAR) у єдине захисне середовище.

Функціональні вимоги визначають логіку роботи системи, її основні сценарії, взаємодію з користувачами та зовнішніми сервісами. Система повинна забезпечувати збір телеметрії з мережевих вузлів, виявлення аномалій, аналіз

ризиків, формування інцидентів, а також підпис та ізоляцію активів за допомогою постквантових алгоритмів. Узагальнені вимоги подано в табл. 1.5.

Таблиця 1.5

Функціональні вимоги системи інтелектуального постквантового кіберзахисту

№	Вимога	Опис функціональності	Рівень пріоритету
1	Збір телеметрії	Система приймає події з SIEM, IDS, API логів, формує уніфіковані пакети даних	Високий
2	Виявлення аномалій (ML)	Алгоритми машинного навчання класифікують події, формують ознаки атаки	Високий
3	Кореляція подій	Система поєднує події у графи загроз і визначає залежності між ними	Високий
4	Оцінка постквантового ризику	Виконується скоринг ризику для активів з урахуванням PQC-сумісності	Середній
5	Автоматизоване реагування (SOAR)	Система генерує сигнатури, команди блокування та виконує ізоляцію активів	Високий
6	Аудит і звітність	Формуються KPI, журнали інцидентів, криптографічні підтвердження операцій	Середній
7	Перенавчання моделей (MLOps)	Система адаптує ML-моделі до нових сценаріїв загроз на основі реальних даних	Середній

Нефункціональні вимоги визначають продуктивність, масштабованість, надійність і експлуатаційні параметри системи. Вони забезпечують стійке функціонування рішень навіть за умов великої кількості потокових подій і обмежених обчислювальних ресурсів. Ключові параметри наведено в табл. 1.6**.**

Таблиця 1.6

Нефункціональні вимоги до системи

№	Показник	Вимога / Значення	Опис
1	Продуктивність	Оброблення ≥ 50 000 подій/с	Забезпечення потокової обробки телеметрії у реальному часі
2	Затримка реагування	≤ 200 мс	Максимальний час між виявленням і реакцією

Продовження таблиці 1.6

3	Доступність	≥ 99.95 %	Безперервна робота системи в корпоративному середовищі
4	Масштабованість	Горизонтальне масштабування до 100 вузлів	Додавання нових ML або PQC-вузлів без зупинки системи
5	Сумісність	REST, gRPC, OpenAPI	Інтеграція з SIEM, SOAR, LMS, PKI/KMS
6	Аудитованість	Повна трасовка подій і PQC-підписів	Забезпечення довіри до результатів обробки інцидентів
7	Надійність ML-моделей	Drift-моніторинг з автоперенавчанням	Контроль точності моделей машинного навчання

Ураховуючи загрози постквантового періоду, вимоги до безпеки орієнтовано на забезпечення цілісності, автентичності, стійкості до квантових атак і відповідності сучасним стандартам NIST PQC. Основні параметри викладено в табл. 1.7**.**

Таблиця 1.7

Вимоги до безпеки системи

№	Категорія	Вимога / Алгоритм	Призначення
1	Криптографічний захист	CRYSTALS-Kyber (KEM), CRYSTALS-Dilithium (DSA)	Постквантове шифрування каналів і підпис команд
2	Протоколи комунікації	TLS 1.3 + PQC extension	Безпечна передача даних між сервісами
3	Автентифікація	OIDC + PQC сертифікати	Довірена ідентифікація користувачів SOC
4	Контроль доступу	RBAC/ABAC	Розмежування прав між аналітиками, ML-інженерами й адміністраторами
5	Захист даних	AES-256-GCM, SHA-3	Симетричне шифрування журналів і телеметрії
6	Цілісність аудиту	Хеш-ланцюг у БД	Неможливість модифікації історії інцидентів
7	Інтеграція з PKI/KMS	API сумісний із NIST PQC	Централізоване керування ключами та сертифікатами

Аналіз вимог підтверджує, що проєктована система повинна реалізовувати інтелектуальне виявлення загроз на основі машинного навчання, поєднане з постквантовим криптографічним захистом, щоб гарантувати довіру до кожної операції в ланцюгу реагування. Нефункціональні обмеження забезпечують високу продуктивність, масштабованість і безперервність роботи, а вимоги до безпеки створюють криптостійку інфраструктуру, сумісну з майбутніми стандартами NIST PQC. Таким чином, система формує адаптивну архітектуру кіберзахисту нового покоління, орієнтовану на автономне виявлення, аналіз і реагування на загрози в умовах постквантової ери.

1.6 Постановка завдання

Постановка завдання полягає у формуванні єдиної науково-технічної концепції побудови інтелектуальної системи захисту від кібератак у постквантовій ері, здатної забезпечити стійке функціонування інформаційних інфраструктур за умов зростання обчислювальних потужностей квантових технологій. У межах дослідження необхідно визначити структуру, функціональні залежності та логіку взаємодії компонентів системи, яка поєднує методи інтелектуального аналізу даних, поведінкової аналітики та постквантових криптографічних механізмів.

Сутність завдання полягає у створенні моделі, що дозволяє об’єднати в єдиному середовищі процеси збору, нормалізації та аналізу подій безпеки, виявлення аномалій, оцінювання ризиків і автоматизованого реагування. Особливу роль відіграє забезпечення криптографічної цілісності управлінських операцій, яка реалізується за допомогою постквантових алгоритмів підпису і шифрування. При цьому взаємодія між аналітичними модулями, системами SOAR/SIEM та сервісами PKI/KMS має гарантувати автентичність, трасовність і достовірність кожного етапу життєвого циклу інциденту.

Математична постановка охоплює процес перетворення вхідного потоку подій у параметричний простір ознак, класифікацію станів системи на основі

алгоритмів машинного навчання та формування рішень про рівень загрози. Додатковим обмеженням є забезпечення криптографічної стійкості передачі службових повідомлень, підпису дій та верифікації ключів відповідно до стандартів NIST PQC. Очікуваний результат полягає у побудові адаптивної архітектури, що здатна виявляти та локалізувати загрози у реальному часі, мінімізуючи ймовірність компрометації систем навіть у випадку появи квантових атак.

Таким чином, постановка завдання узагальнює концептуальну модель інтеграції методів машинного навчання, постквантової криптографії та оркестрації реагування у єдину систему, орієнтовану на автономне прийняття рішень, довірене виконання дій та підвищення рівня кіберстійкості в умовах постквантового середовища.

2 ПРОЕКТУВАННЯ ІНФОРМАЦІЙНОГО ТА ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ

2.1 Логічна модель даних у вигляді ER-діаграми

Логічна модель даних проєктувалася під вимоги інтелектуальної системи постквантового кіберзахисту із пріоритетом на трасовність інцидентів, криптографічну довіру до керувальних операцій і масштабоване потокове поглинання телеметрії. Центральними є сутності Incident, TelemetryEvent, ResponseAction, Asset, Identity та PQCCertificate з доменами-переліками Severity і ActionType; таке ядро мінімізує зв’язність сценаріїв, дозволяє незалежно масштабувати гарячі таблиці (події та дії) і зберігати “холодні” довідники в окремих шардах. Нормалізація виконана до 3НФ/BCNF: усі атрибути залежні лише від ключів, усунено транзитивні залежності (наприклад, характеристики користувача винесено до Identity, а придатність до PQC - до PQCCertificate), що зменшує дублювання та ризик аномалій оновлення. Інцидент агрегує лише ідентифікатори подій і дій, тому життєвий цикл від “виявлено” до “ізольовано/відновлено” фіксується як послідовність малих незмінних записів, придатних для відтворення та аудиту.

Модель також відокремлює домени безпеки: усі керувальні операції мають посилання на чинний PQC-сертифікат, що забезпечує доказовість підпису та сумісність з NIST-алгоритмами у транспорті. Сконструйована схема відповідає вимогам потокового DM/ELT: події пишуться апенд-онлі, а кореляційні ключі дозволяють будувати графи атак без жорстких джоінів у гарячому контурі, зводячи важкі обчислення до офлайн-аналітики. Узагальнену структуру показано на рис. 2.1, де відображено мінімально достатній набір сутностей і зв’язків для забезпечення причинно-наслідкової відтворюваності інцидентів у постквантовому середовищі.

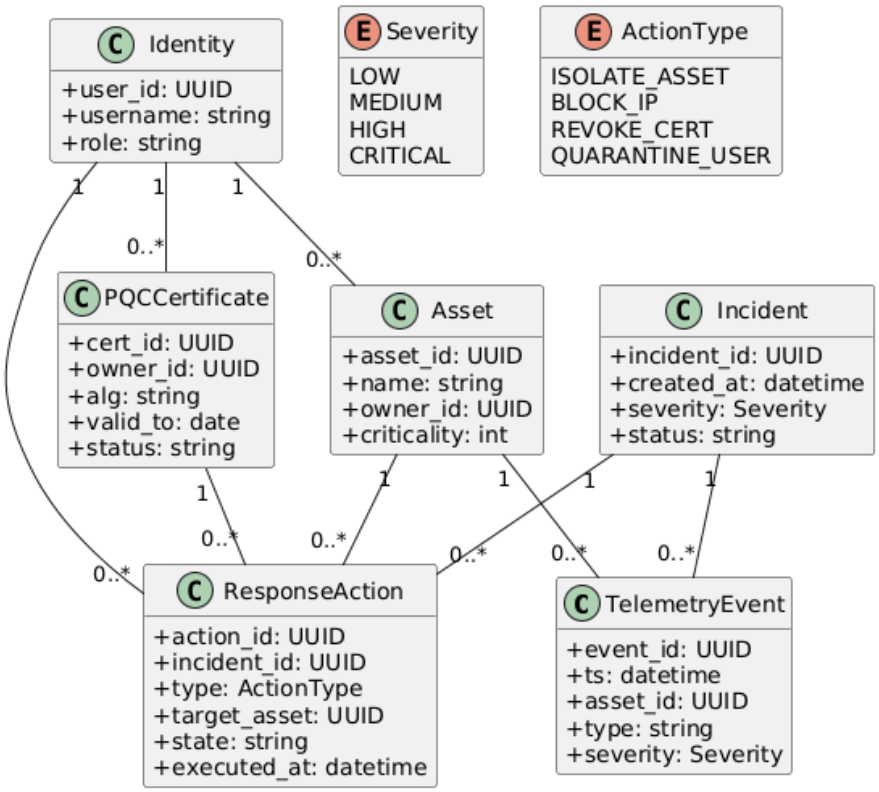

Рис. 2.1. ER-діаграма логічної моделі даних системи постквантового кіберзахисту

Ключові ролі сутностей у процесах виявлення-реагування та забезпечення криптографічної довіри підсумовано в табл. 2.1.

Таблиця 2.1

Ролі сутностей логічної моделі даних

Сутність	Первинний ключ	Призначення у процесі	Критичні обмеження цілісності
TelemetryEvent	event_id	Апенд-онлі журнал телеметрії для ознак ML та кореляції	asset_id NOT NULL, часові індекси; незмінність записів
Incident	incident_id	Контейнер життєвого циклу загрози (від виявлення до закриття)	FK до ініціатора (Identity); статусний автомат
ResponseAction	action_id	Керувальна дія (ізоляція, блокування, відкликання) з аудиторським слідом	FK на Incident, Asset, PQCCertificate; незворотність історії

Продовження таблиці 2.1

Asset	asset_id	Каталог захищуваних об’єктів з критичністю та власником	Унікальність імені в межах домену; FK на Identity
Identity	user_id	Суб’єкти, що ініціюють/затверджують дії та створюють інциденти	Унікальні логіни; політики RBAC/ABAC
PQCCertificate	cert_id	Артефакт довіри для PQC-підпису керувальних операцій	Алгоритм/строк дії; FK на власника (Identity)
Severity	—	Домен рівня критичності для подій та інцидентів	Перелік значень: LOW…CRITICAL
ActionType	—	Домен типів дій для SOAR-ланцюга	Перелік значень: ISOLATE_ASSET, BLOCK_IP, …

Результуючи, така ER-модель надає формально нормалізований і технологічно нейтральний каркас, який: відокремлює гарячі журнальні потоки від керувальних артефактів, забезпечує доказову простежуваність через зв’язування ResponseActio з PQCCertificate, підтримує масштабовану кореляцію подій без втрати атомарності транзакцій та готує дані до подальших ML-процедур і побудови графів атак без порушення цілісності й криптографічної довіри в постквантовому контурі.

2.2 Діаграма класів і їхні кооперації

Діаграма класів відображає логічну структуру взаємодії основних компонентів інтелектуальної системи постквантового кіберзахисту, які утворюють єдиний цикл - від збору телеметрії до оркестрації реакції та аудиту. У центрі моделі знаходяться класи TelemetryIngestor, AnomalyDetector, ResponseOrchestrator, PQCKMS, ControlAPI та Incident, між якими визначені чіткі інтерфейси та залежності. Така декомпозиція забезпечує ізоляцію функціональних зон: оброблення подій, машинний аналіз, прийняття рішень і криптографічне підтвердження дій.

Кожен клас виконує свою роль у межах потоку реагування: TelemetryIngestor нормалізує потік журналів подій і готує батчі для оброблення; AnomalyDetector застосовує навчальні моделі для виявлення відхилень у поведінці активів; ResponseOrchestrator координує цикл реагування, формує інциденти та надсилає команди через ControlAPI, використовуючи PQC-підпис, що генерується модулем PQCKMS. Клас Incident фіксує статуси, рівні критичності й підсумкові дії, формуючи базу для аудиту. Така структура відповідає принципам нормалізації програмної архітектури — усі модулі мають однозначну відповідальність, не дублюють логіку та можуть масштабуватися незалежно, що особливо важливо для потокової обробки великих обсягів телеметрії.

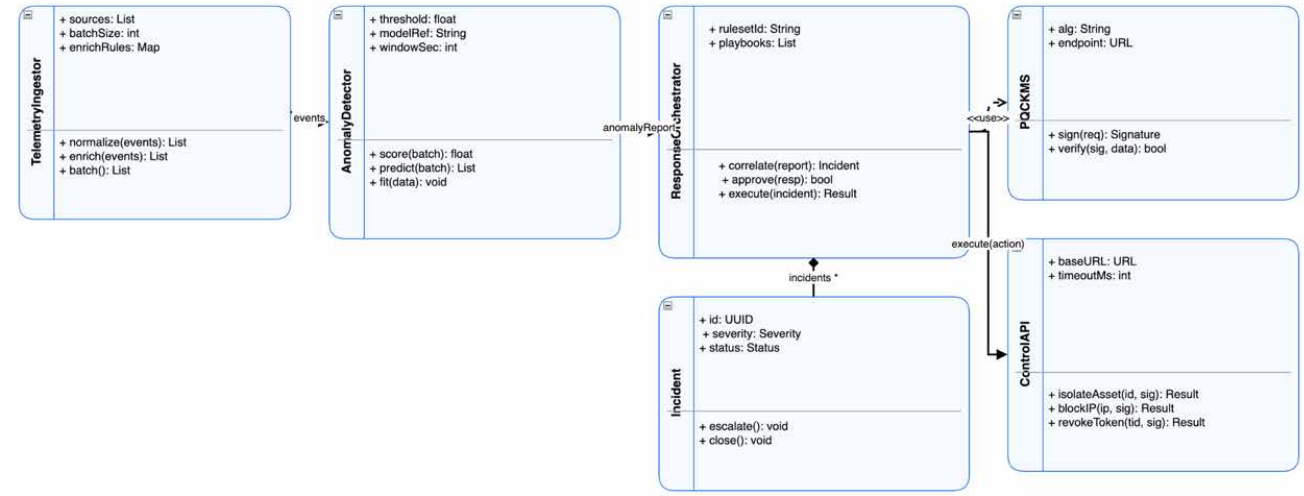

Рис. 2.2. Діаграма класів системи постквантового кіберзахисту

Далі на рисунках подано три ключові кооперації, які описують поведінкову логіку системи. Перша кооперація (рис. 2.3) описує перехід «виявлення → інцидент»: модуль TelemetryIngestor передає батч подій у AnomalyDetector, який обчислює рівень аномальності та надсилає звіт у ResponseOrchestrator. Після аналізу оркестратор створює інцидент і повідомляє оператора SOC. Цей процес відображає початкову фазу життєвого циклу безпекової події.

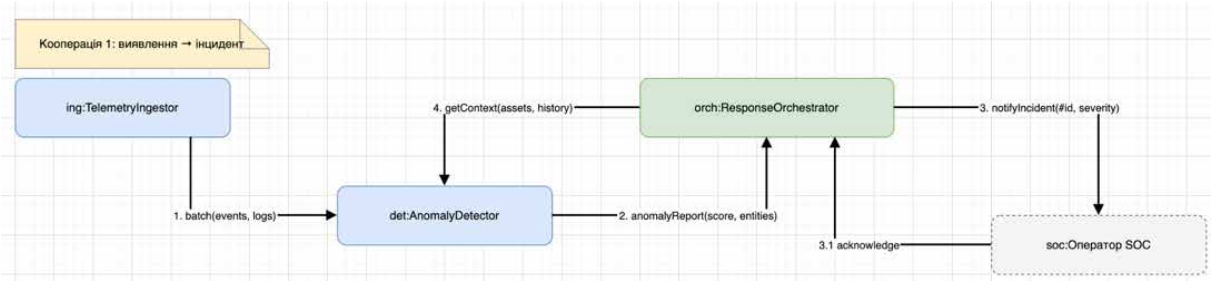

Рис. 2.3. Кооперація 1 – Виявлення → Інцидент

Друга кооперація (рис. 2.4) - «підпис PQC → виконання» - демонструє інтеграцію оркестратора з криптографічним модулем PQCKMS. Перед виконанням критичної операції (наприклад, ізоляції активу) ResponseOrchestrator формує запит на підпис, отримує сигнатуру, верифікує її та надсилає команду в ControlAPI. Завдяки постквантовому алгоритму (Dilithium, Kyber) система гарантує стійкість до квантових атак, а всі команди мають криптографічно підтверджений контекст дії.

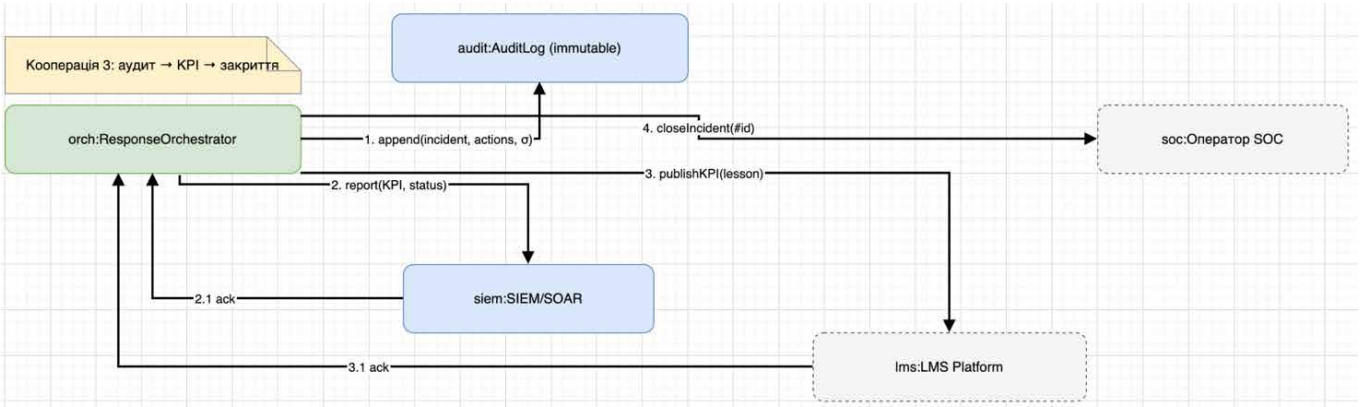

Рис. 2.4. Кооперація 2 – Підпис PQC → Виконання

Третя кооперація (рис. 2.5) - «аудит → KPI → закриття» - описує завершальний етап. Оркестратор фіксує результат дій у незмінному журналі (AuditLog), надсилає KPI-звіт до SIEM/SOAR або навчальної LMS-платформи, після чого закриває інцидент. Ця кооперація реалізує вимоги прозорості, відтворюваності та дотримання нормативів кібербезпеки.

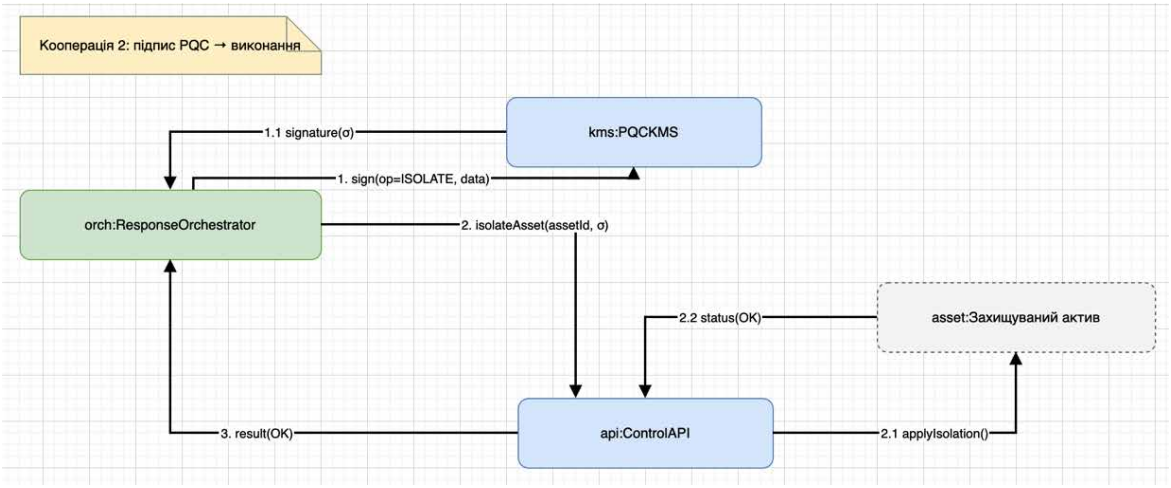

Рис. 2.5. Кооперація 3 – Аудит → KPI → Закриття

Для систематизації структурних ролей класів подано узагальнену таблицю (табл. 2.2), яка демонструє їхнє функціональне призначення в загальній архітектурі системи.

Таблиця 2.2

Функціональні ролі класів системи постквантового кіберзахисту

Клас	Основна функція	Тип взаємодії	Рівень відповідальності
TelemetryIngestor	Збір і нормалізація подій з різних джерел	Потокова обробка	Вхідний рівень системи
AnomalyDetector	Виявлення відхилень за допомогою ML-моделей	Алгоритмічний аналіз	Інтелектуальний рівень
ResponseOrchestrator	Прийняття рішень, оркестрація дій, створення інцидентів	Керувальний рівень	Центральний компонент
PQCKMS	Генерація та верифікація PQC-підписів	Криптографічний рівень	Безпековий модуль
ControlAPI	Виконання ізоляцій і блокувань активів	Інтеграційний рівень	Операційний шар
Incident	Збереження стану, критичності, результатів реагування	Аналітичний рівень	Аудиторно-звітний компонент

Результуюче узагальнення: побудована діаграма класів і кооперацій забезпечує формалізовану архітектуру, у якій поєднано машинне навчання, оркестрацію подій і постквантову криптографію. Такий підхід дозволяє досягти високого рівня узгодженості, трасовності та безпечності обробки даних, створюючи фундамент для масштабованої системи кіберзахисту нового покоління.

2.3 Діаграма компонентів

Архітектура системи постквантового інтелектуального захисту спроєктована як набір незалежних компонентів, що взаємодіють через чітко визначені інтерфейси та обмінюються уніфікованими повідомленнями в межах єдиної подієвої шини. Такий підхід забезпечує слабке зв’язування між модулями, масштабованість і можливість ізольованого оновлення чи заміни кожного сервісу без ризику порушення цілісності всієї системи. Основні компоненти включають TelemetryIngestor, AnomalyDetector, ThreatGraph/Correlator, Response Orchestrator, AuditLog, Reporting/KPI, а також зовнішні інтеграції з PQC KMS/PKI, Control API, SIEM/SOAR, LMS Platform та IdP.

Система функціонує за принципом наскрізного потоку даних: події з різних джерел телеметрії надходять до компонента збору, далі обробляються модулями аналітики та кореляції, після чого передаються в оркестратор, який формує рішення щодо реагування, підписує дії квантово-стійким ключем і виконує їх через зовнішній API. Компонент AuditLog забезпечує незмінність записів і підтвердження подій, а Reporting/KPI формує статистичні звіти та передає їх у системи аналітики. Така структура дає змогу розділити аналітичні та операційні процеси, зберігаючи при цьому високу пропускну здатність і відповідність вимогам до криптографічного захисту даних у постквантовій ері.

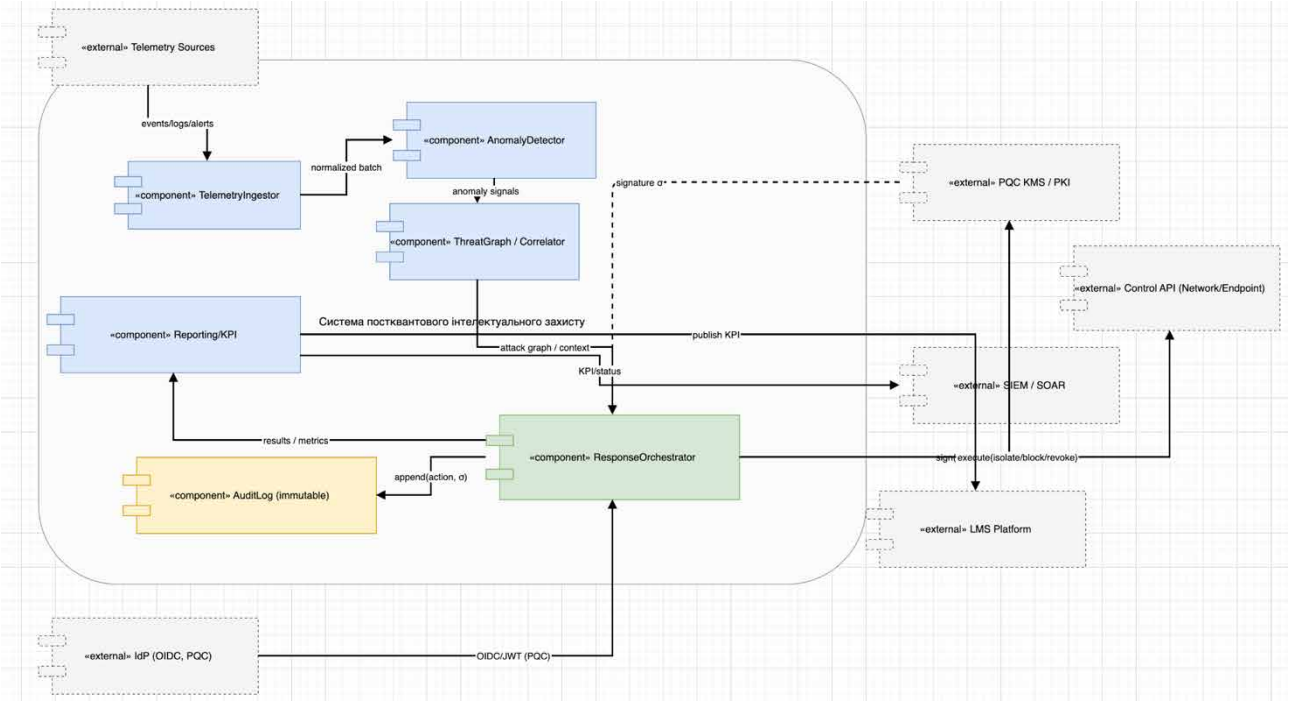

Рис. 2.6. Діаграма компонентів системи постквантового інтелектуального захисту

Кожен компонент має чітку функціональну відповідальність, що наведено в табл. 2.3.

Таблиця 2.3 – Функціональні ролі компонентів системи

Компонент	Основне призначення	Тип взаємодії	Критичні вимоги
TelemetryIngestor	Збір, нормалізація й агрегування подій з різних джерел	Вхідний канал	Пропускна здатність ≥ 10 000 подій/с
AnomalyDetector	Аналіз потоків телеметрії з використанням ML-моделей для виявлення аномалій	Обчислювальний модуль	Затримка детекції ≤ 200 мс
ThreatGraph/Correlator	Кореляція сигналів і побудова графів атак	Аналітичний модуль	Повна трасовність подій
ResponseOrchestrator	Координація реагування, формування команд і підпис PQC-сертифікатами	Керувальний центр	Достовірність і авторизація дій
AuditLog (immutable)	Незмінне зберігання журналів та підтвердження операцій	Реєстраційний модуль	Непорушність записів

Продовження таблиці 2.3

Reporting/KPI	Формування звітів про ефективність реагування	Аналітичний модуль	Актуальність метрик ≤ 1 год
PQC KMS / PKI	Генерація та перевірка постквантових підписів	Зовнішня інтеграція	Алгоритми NIST PQ
Control API	Реалізація команд ізоляції, блокування чи відкликання токенів	Операційний інтерфейс	Надійність ≥ 99.9 %

Результуючи, побудована компонентна модель забезпечує чітке розмежування логічних шарів - обробки, аналізу, реагування та аудиту - що дозволяє гнучко масштабувати систему, інтегрувати додаткові джерела даних або ML-модулі, а також гарантує дотримання вимог постквантової стійкості та кіберрезилієнтності всієї архітектури.

2.4Діаграма пакетів

Пакетна структура системи інтелектуального постквантового захисту формує логічну основу її модульної архітектури, забезпечуючи чітке розмежування функціональних зон та незалежність між підсистемами. Розподіл на внутрішні пакети (core, crypto, data) і зовнішні адаптери (ext) дозволяє реалізувати принципи інкапсуляції, повторного використання коду та контрольованого обміну даними через інтерфейси. Така модель сприяє побудові масштабованої системи з підтримкою інтеграції нових сервісів без порушення стабільності основного ядра.

Основним елементом внутрішнього шару є пакет core, який містить ключові підсистеми - ingestion, analytics, orchestration та reporting. Кожен із них відповідає за власну фазу оброблення даних: від надходження телеметрії до формування звітів і аналітичних показників. Пакет crypto виконує операції постквантового підпису, верифікації та управління ключами, що гарантує криптографічну стійкість. Пакет data забезпечує єдине сховище телеметричних потоків, моделей машинного навчання й аудиторних записів, забезпечуючи узгодженість даних між модулями.

Зовнішні пакети ext.adapters (ControlAPI, SIEM/SOAR, LMS) та ext.security.idp утворюють інтеграційний контур, який надає зв’язок із зовнішніми платформами безпеки, навчальними системами та службами автентифікації (OIDC/PQC). Вони реалізують принцип «захищеної периферії», де комунікація відбувається лише через верифіковані інтерфейси, що підписуються квантово-стійкими ключами. Такий підхід мінімізує ризики міжмодульних вразливостей і сприяє відповідності архітектури вимогам постквантових стандартів безпеки (NIST PQC).

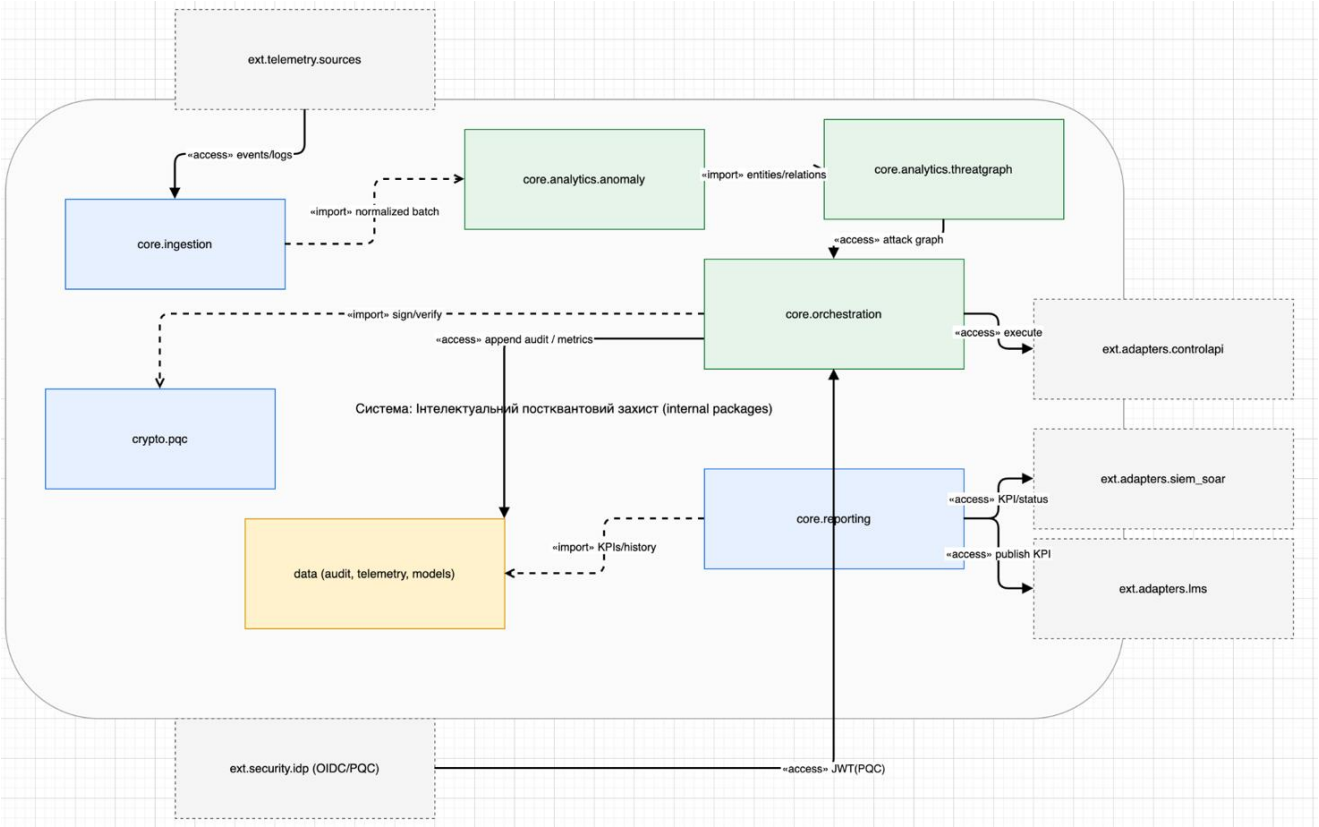

Рис. 2.7. Діаграма пакетів системи інтелектуального постквантового захисту

Для узагальнення взаємозв’язків між пакетами складено табл. 2.4, у якій наведено їх основні функціональні ролі в загальній структурі програмного комплексу.

Таблиця 2.4

Основні пакети та їх призначення

Пакет	Основна роль	Тип взаємодії	Критичні вимоги
core.ingestion	Обробка телеметрії, нормалізація подій, підготовка даних для аналітики	Внутрішній	Пропускна здатність ≥ 10 тис. подій/с
core.analytics	Інтелектуальний аналіз (аномалії, графи атак)	Внутрішній	Затримка обробки ≤ 200 мс
core.orchestration	Управління реакціями, формування команд і підпис PQC	Внутрішній	Повна трасовність операцій
core.reporting	Формування KPI, звітів і візуалізацій	Внутрішній	Актуальність даних ≤ 1 год
crypto.pqc	Постквантова криптографія, підпис, верифікація	Внутрішній	Підтримка алгоритмів NIST PQ
data	Централізоване зберігання телеметрії, моделей, журналів	Внутрішній	Незмінність і узгодженість даних
ext.adapters	Інтеграція з ControlAPI, SIEM/SOAR, LMS	Зовнішній	Безпечна передача через TLS 1.3
ext.security.idp	Автентифікація користувачів, OIDC/PQC-токени	Зовнішній	Гарантована унікальність сесій

Результуюче узагальнення: побудована пакетна структура демонструє зрілу архітектуру, у якій ядро системи ізольоване від зовнішніх сервісів, а всі взаємодії здійснюються через контрольовані адаптери. Це забезпечує розширюваність, захищеність і високу надійність системи в умовах постквантових загроз.

3 РЕАЛІЗАЦІЯ ПРОГРАМНОГО ЗАБЕЗПЕЧЕННЯ ТА ТЕХНОЛОГІЧНА ІНФРАСТРУКТУРА СИСТЕМИ

3.1 Вибір технологій та інструментальних засобів реалізації системи

Реалізація інтелектуальної системи постквантового кіберзахисту потребує використання технологій, здатних забезпечити одночасно потокове опрацювання телеметрії, адаптивний аналіз поведінкових аномалій, автентичність керувальних дій, а також криптографічну стійкість до квантових атак. У межах проєкту пріоритет надано технологіям із підтвердженою надійністю, відкритою екосистемою й активною науковою підтримкою. Система реалізується на основі Python, що забезпечує оптимальний баланс між продуктивністю, бібліотечною базою для машинного навчання та можливістю інтеграції потокових конвеєрів. Для побудови модулів виявлення аномалій обрано scikit-learn, PyTorch та засоби аналізу часових рядів, що дозволяє застосовувати алгоритми класифікації, кластеризації та поведінкове моделювання без втрати інтерпретованості результатів.

Оскільки система працює в постквантовому середовищі, криптографічні операції (підпис дій, керування ключами, верифікація) реалізовано через бібліотеку liboqs із підтримкою алгоритмів CRYSTALS-Kyber і CRYSTALS-Dilithium, які рекомендовані NIST для використання у високонадійних інфраструктурах. Як транспорт застосовано TLS 1.3 з PQC-розширеннями, що гарантує захист каналів взаємодії між сервісами. Компоненти системи розгортаються у контейнерах Docker, що забезпечує контрольоване ізолювання модулів, масштабування та стабільну роботу в середовищах з різною обчислювальною топологією. Для зберігання телеметрії, інцидентів і криптографічних артефактів використано PostgreSQL з увімкненими механізмами хеш-ланцюгів, які підтримують незмінність журналів аудиту. Взаємодія між сервісами побудована на REST/gRPC, що спрощує інтеграцію із зовнішніми SIEM/SOAR-системами та модулями керування інцидентами.

Узагальнення вибраних технологій наведено в *табл. 3.1*, де подано їх функціональне призначення та обґрунтування застосування у межах проєкту.

Таблиця 3.1 – Вибрані технології та інструментальні засоби системи постквантового кіберзахисту

Технологія / Засіб	Призначення	Обґрунтування вибору
Python 3.x	Реалізація модулів аналітики та сервісної логіки	Широка ML-екосистема, швидка інтеграція, підтримка асинхронності
scikit-learn, PyTorch	Класифікація, кластеризація, моделі аномалій	Стійкі результати, оптимізація моделей, адаптованість до потоків
liboqs (Kyber, Dilithium)	Постквантовий підпис і шифрування	Відповідність стандартам NIST PQC, гарантована криптостійкість
TLS 1.3 + PQC Extensions	Захист каналів між модулями	Стандарт корпоративної безпеки, мінімальна затримка
Docker	Контейнеризація компонентів	Ізоляція сервісів, масштабованість, портативність
PostgreSQL	База даних подій, інцидентів, сертифікатів	Транзакційність, хеш-ланцюги для аудиту, підтримка великих обсягів
REST/gRPC	Міжсервісна комунікація	Уніфіковані API, сумісність з SIEM/SOAR
Prometheus + Grafana	Моніторинг продуктивності та KPI	Актуальний стан системи, візуалізація аналітики реагування

Проведений аналіз технологій показує, що застосований стек оптимально покриває вимоги системи до потокової обробки телеметрії, машинного виявлення аномалій та реалізації постквантового криптографічного захисту. Така конфігурація забезпечує масштабованість, криптостійкість і високий рівень керованості всієї інфраструктури, що є критично важливим для побудови інтелектуальної системи кіберзахисту нового покоління.

3.2 Інформаційна база системи

Інформаційна база інтелектуальної системи постквантового кіберзахисту формується як узгоджений комплекс операційних журналів телеметрії, довідникових сутностей, навчальних вибірок для алгоритмів машинного навчання та аналітичних структур типу OLAP, що забезпечують багатовимірний аналіз інцидентів з урахуванням постквантового профілю захисту й характеристик ML-моделей. На рівні аналітичного сховища ключовим елементом є OLAP-куб IncidentsCube із зерном Time × Asset × Severity × PQCProfile × MLModel, у фактичній таблиці якого агрегуються показники кількості інцидентів, середнього часу реагування, відсотка покриття активів постквантовими алгоритмами та F1-міри моделі виявлення аномалій; вимірні таблиці DimTime, DimAsset, DimSeverity, DimPQCProfile та DimMLModel дозволяють деталізувати події за часовими інтервалами, сегментами інфраструктури, рівнями критичності, типами PQC-профілів і версіями моделей машинного навчання. Структуру цієї аналітичної інформаційної бази у вигляді зіркоподібної схеми наведено на рис. 3.1.

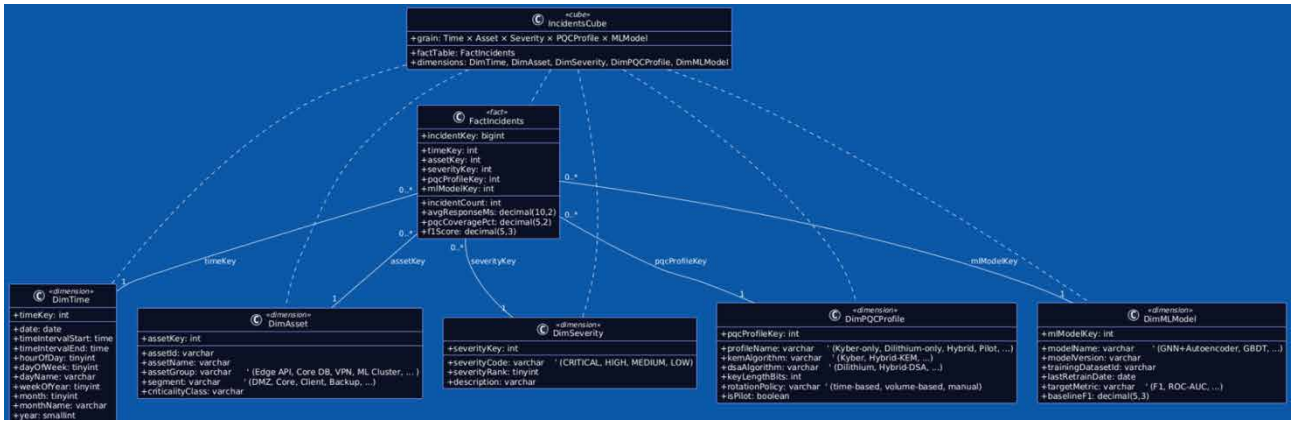

Рис. 3.1. Зіркоподібна схема OLAP-кубу IncidentsCube з вимірами часу, активів, критичності, PQC-профілю та ML-моделі

Для підтримки інтелектуальної аналітики у сховище вбудовано шар похідних ознак та результатів кластеризації інцидентів, що дозволяє виявляти однорідні групи подій за інтенсивністю, рівнем критичності, типами задіяних активів і профілем захисту. На основі історичних даних формується матриця

ознак, до якої застосовується алгоритм k-means, а отримані кластери зберігаються як додатковий вимір у кубі та використовуються для пріоритезації реагування. Приклад просторового розподілу інцидентів у двовимірному просторі ознак наведено на рис. 3.2.

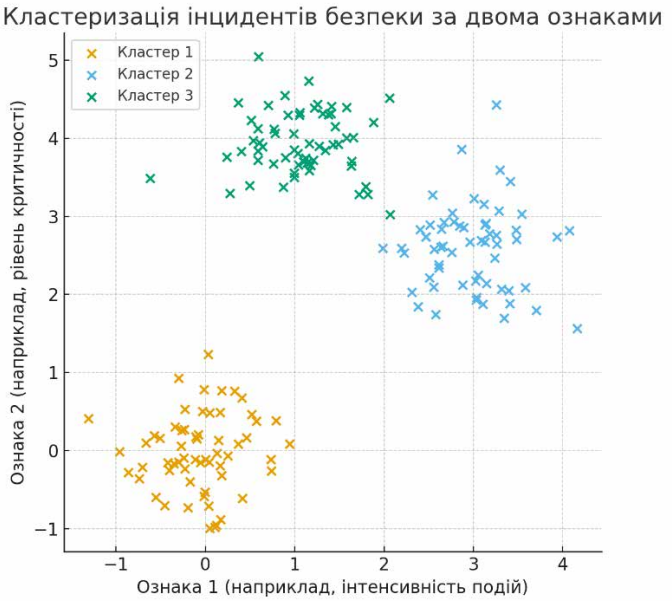

Рис. 3.2. Кластеризація інцидентів безпеки у просторі двох узагальнених ознак інформаційної бази

Оптимальна кількість кластерів визначається за методом ліктя, де на осі абсцис відкладено кількість кластерів k, а на осі ординат – суму квадратів відстаней (SSE) усіх точок до центрів кластерів. Злам кривої SSE відображає момент, коли подальше збільшення k дає незначне зменшення похибки й не виправдовує ускладнення моделі; значення k, обране у цій точці, фіксується в репозитарії ML-експериментів і використовується для побудови робочої конфігурації аналітичного модуля. Ілюстрацію методу ліктя для даних інформаційної бази подано на рис. 3.3.

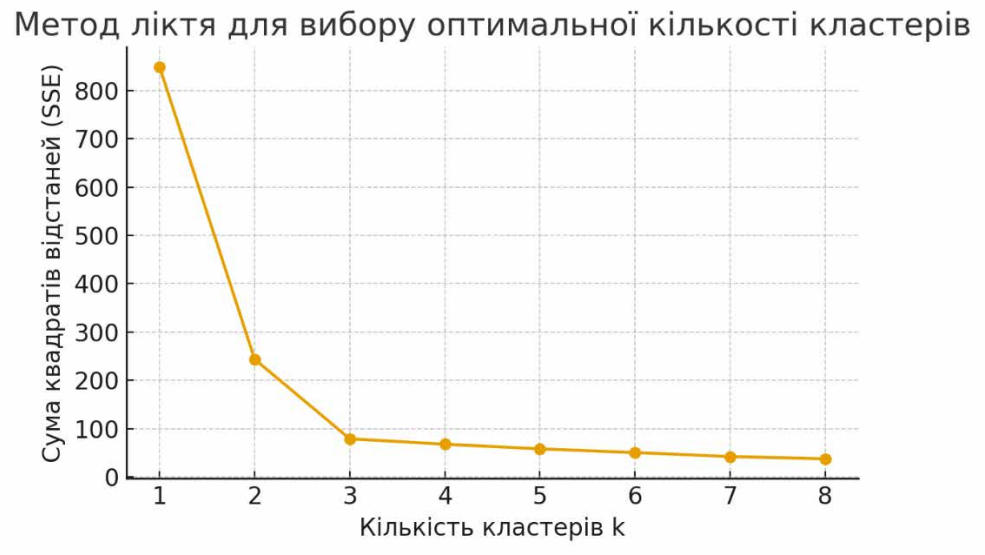

Рис. 3.3. Використання методу ліктя для вибору оптимальної кількості кластерів інцидентів у інформаційній базі системи

Для оцінювання ефективності запропонованої інформаційної бази й пов’язаних з нею ML-моделей застосовується радіальна інтегральна візуалізація, що порівнює класичні IDS-рішення, системи виявлення на базі ML та інтегровану PQC+ML-архітектуру, розроблену в цій роботі. На вісях відкладаються нормовані метрики точності виявлення, повноти, F1-міри, середнього часу реагування та стійкості до постквантових загроз; дані для кожної системи отримуються із фактів IncidentsCube і експериментів із навчальними вибірками. Радіальна діаграма, наведена на рис. 3.4, демонструє, що інформаційна база, яка експліцитно моделює PQC-профіль активів і параметри ML-моделей, дозволяє досягти кращого балансу між якістю детекції та швидкістю реагування, зберігаючи при цьому високий рівень криптостійкості.

Рис. 3.4. Радіальна діаграма порівняння показників виявлення кібератак для різних типів систем на основі даних інформаційної бази

Додатковим інструментом аналітики є можливість розгортання OLAP-кубу інцидентів за вимірами «час – критичність» і візуалізація результатів у вигляді теплових карт, що відображають концентрацію подій певної важливості у квартальному або місячному розрізі. Така форма подання дозволяє виявляти сезонні патерни атак, періоди підвищеного навантаження на SOC та часові вікна, у яких необхідно посилювати постквантові криптографічні профілі для критичних активів; приклад відповідного зрізу IncidentsCube наведено на рис. 3.5.

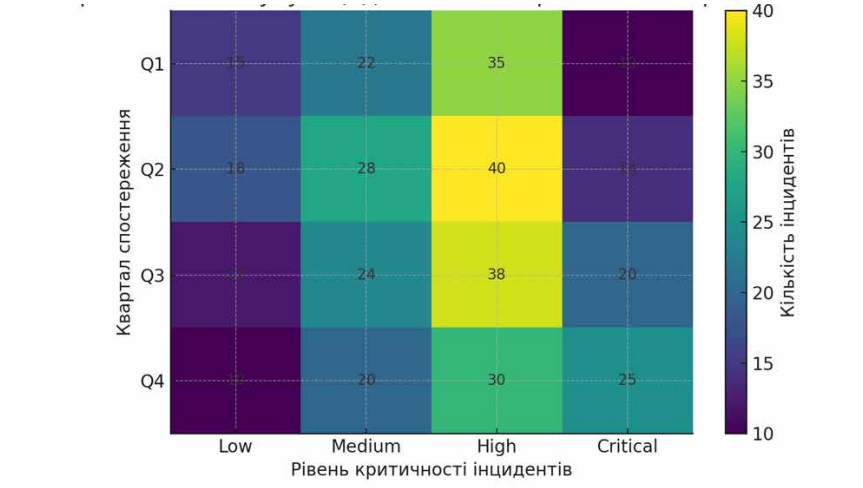

Рис. 3.5. Приклад розгортання OLAP-кубу інцидентів за вимірами «квартал – рівень критичності» у вигляді теплової карти

Наукова новизна побудованої інформаційної бази полягає в тому, що, по-перше, у структурі OLAP-кубу вперше для даного класу систем введено виміри PQC-профілю та версії ML-моделі, що дозволяє аналізувати взаємозв'язок між постквантовим захистом, якістю детекції й операційними метриками реагування; по-друге, результати кластеризації та експериментів ML інтегруються в єдиний аналітичний контур, а не зберігаються ізольовано від SOC-телеметрії, що забезпечує безперервний цикл MLOps та підтримку сценаріїв адаптивного перенавчання; по-третє, у фактичних показниках кубу явно моделюються метрики PQC-покриття активів і F1-міра моделей, що створює основу для прийняття управлінських рішень щодо міграції на постквантові алгоритми та вибору оптимальних конфігурацій детекторів аномалій.

3.3 Архітектура системи та проектування функціоналу результатів дослідження

Архітектура інтелектуальної системи постквантового кіберзахисту сформована як багаторівнева потокова модель, у якій телеметрія з мережевих вузлів, журналів та сенсорів надходить у вузол попередньої обробки, проходить ML-аналіз, кореляцію на графі атак та завершується механізмами автоматизованого реагування. Кожен компонент архітектури виконує ізольовану функцію, зберігаючи при цьому цілісність конвеєра, а ключові операції підписуються та верифікуються постквантовими алгоритмами Kyber/Dilithium. Завдяки такому підходу система забезпечує наскрізну криптографічну стійкість, гарантує незмінність управлінських дій та підтримує можливість інтеграції в зовнішні SIEM/SOAR-платформи через стандартизовані API. Узагальнену потокову схему архітектури наведено на рис. 3.6.

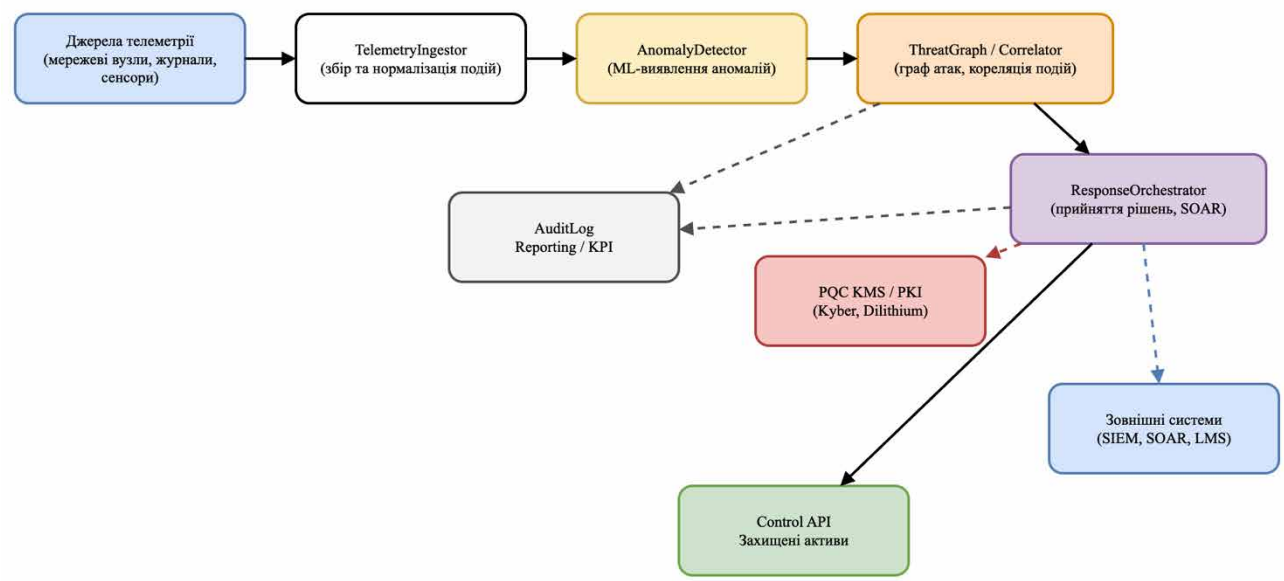

Рис. 3.6. Потокова архітектура інтелектуальної системи постквантового кіберзахисту (TelemetryIngestor → AnomalyDetector → ThreatGraph → ResponseOrchestrator)

На наступному рівні наведено структурне проектування модулів у вигляді компонента діаграми, що відображає топологію розгортання: Edge-сегмент для збору телеметрії; Core-аналітичний кластер, де здійснюються нормалізація даних, ML-аналіз та кореляція подій; сервіс PQC-керування ключами; зона керування захищеними активами; а також адаптери інтеграції з зовнішніми системами моніторингу безпеки. Така модель дозволяє чітко розмежувати довірчі домени, зменшити площу атаки та забезпечити ізоляцію вразливих компонентів. Відповідну діаграму архітектурного розгортання системи подано на рис. 3.7.

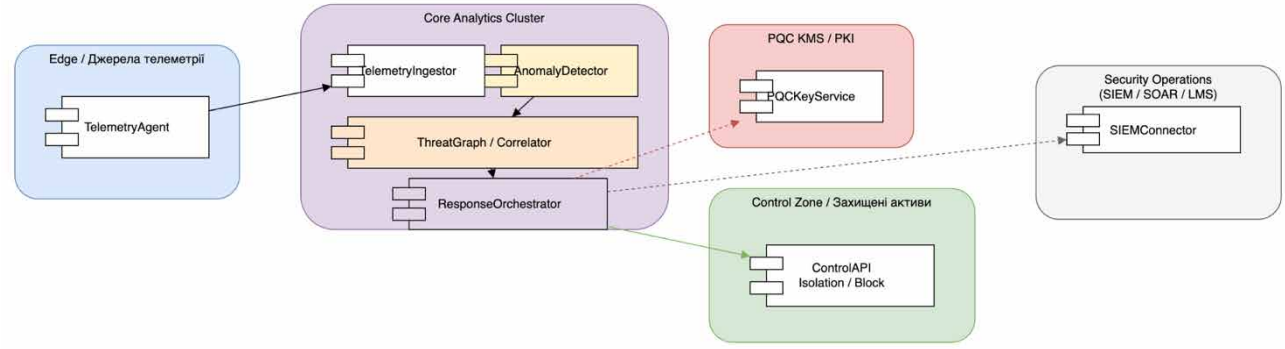

Рис. 3.7. Діаграма компонентного розгортання системи з поділом на Edge, Core Analytics, PQC-KMS, Control Zone та Security Operations

Проектування функціоналу системи ґрунтується на результатах проведених досліджень з поведінкової аналітики, кластеризації інцидентів, побудови графів атак і впровадження PQC-алгоритмів. Зокрема, застосована технологія ThreatGraph забезпечує можливість відстеження ланцюгів атак на основі кореляції часових, просторових та поведінкових подій, тоді як ML-детектор аномалій навчається на багатовимірних ознаках OLAP-кубу, сформованого у попередньому підрозділі. Реалізований модуль ResponseOrchestrator підтримує адаптивні стратегії реагування – від сповіщення SOC до автоматизованого блокування чи ізоляції активів – із обов'язковою криптографічною авторизацією через PQC-KMS. Такий підхід забезпечує поєднання аналітичної точності, надійності криптографічного контролю та оперативності дій, що дозволяє досягти високого технологічного рівня захисту інфраструктури. Результуюча архітектура демонструє інтеграцію ML-моделей, потокової обробки інцидентів, постквантових механізмів автентифікації та граф-аналітики, створюючи цілісну платформу, здатну працювати в умовах майбутніх квантових загроз і адаптивно вдосконалюватися на основі даних власної інформаційної бази.

3.4 Алгоритмізація програмних модулів

Алгоритмізація програмних модулів інтелектуальної системи постквантового кіберзахисту полягає у формальному визначенні послідовності оброблення телеметрії, виявлення аномалій, прийняття рішень щодо реагування та періодичного вдосконалення моделей машинного навчання. Кожен модуль реалізує детермінований або стохастичний алгоритм, спроєктований відповідно до досліджених закономірностей поведінки подій, вимог до криптографічної стійкості та циклу MLOps. Узгоджене функціонування цих алгоритмів утворює єдиний інтелектуально-регулятивний контур, що забезпечує наскрізне підсилення безпеки системи. Схему алгоритму оброблення телеметрії та виявлення аномалій наведено на рис. 3.8.

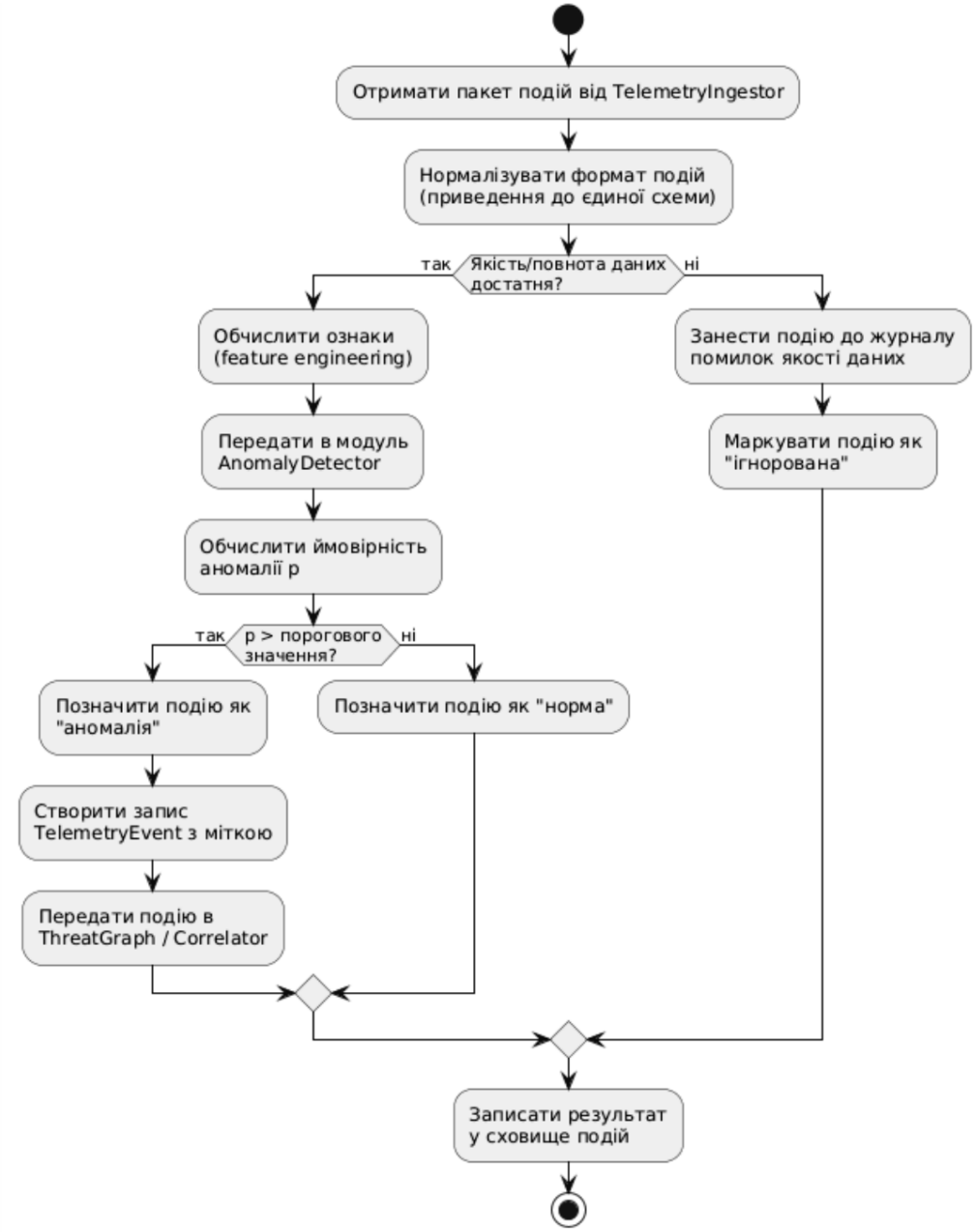

Рис. 3.8. Алгоритм оброблення телеметрії та ML-виявлення аномалій у модулі TelemetryIngestor–AnomalyDetector

За логікою другого ключового модуля система виконує оркестрацію реагування на основі даних ThreatGraph, оціненого рівня критичності інциденту та політик безпеки, але з обов’язковим застосуванням постквантового підпису Dilithium для кожної керувальної команди. Така модель унеможливлює несанкціоновану модифікацію управлінських дій і забезпечує доказовість кожної операції в контурі безпеки. Послідовність роботи модуля реагування подано на рис. 3.9.

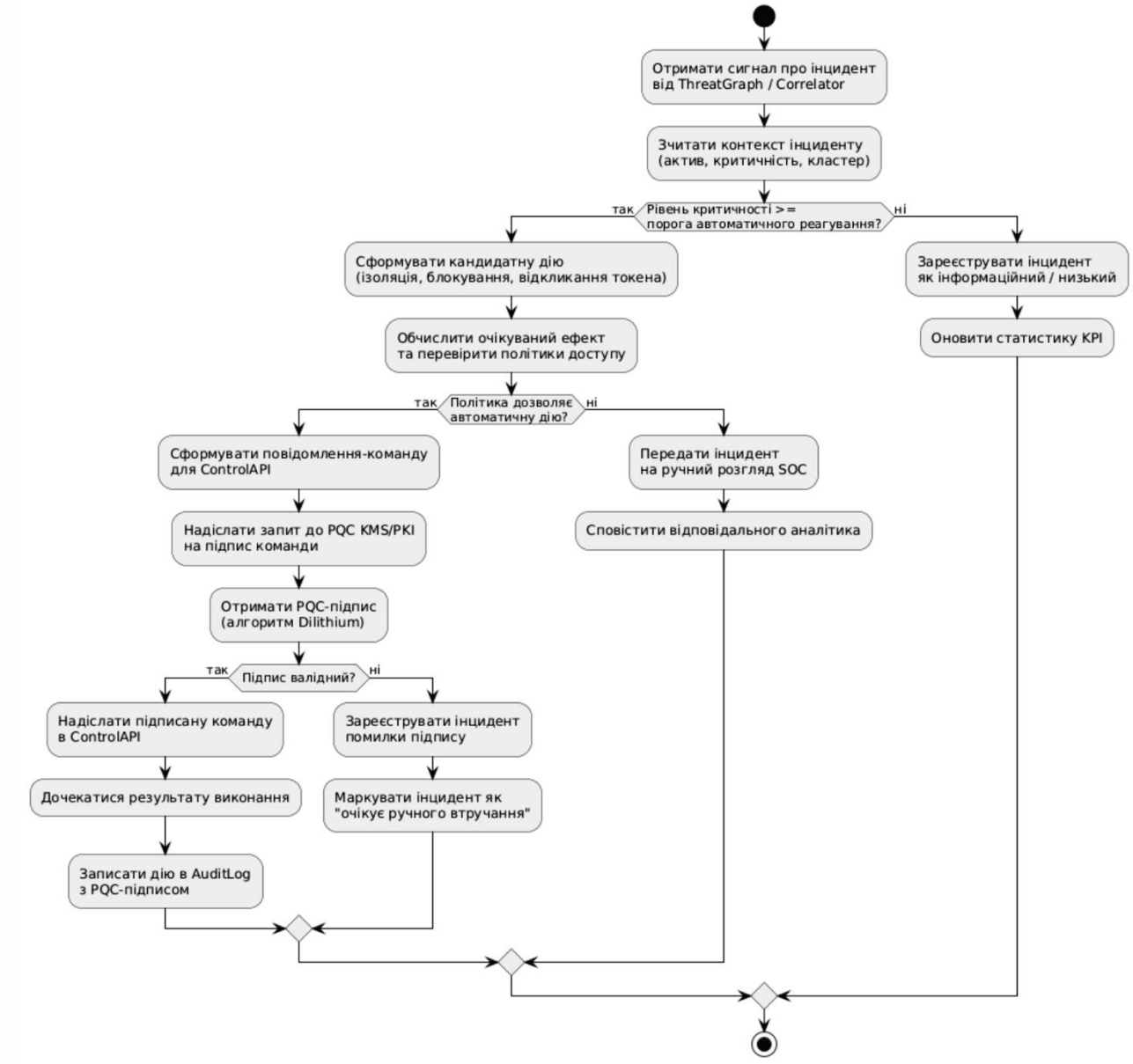

Рис. 3.9. Алгоритм оркестрації реагування та PQC-підпису керувальних дій у модулі ResponseOrchestrator

Третій алгоритм реалізує замкнений цикл MLOps, у межах якого система періодично оцінює актуальні метрики ML-моделі, виявляє можливий data/concept drift, формує нові навчальні вибірки з інформаційної бази, порівнює результати з чинними моделями та оновлює репозитарій ML-конфігурацій у разі покращення. Такий підхід забезпечує динамічну адаптацію моделі до нових типів подій і сценаріїв атак. Алгоритмічну схему процедури перенавчання подано на рис. 3.10.

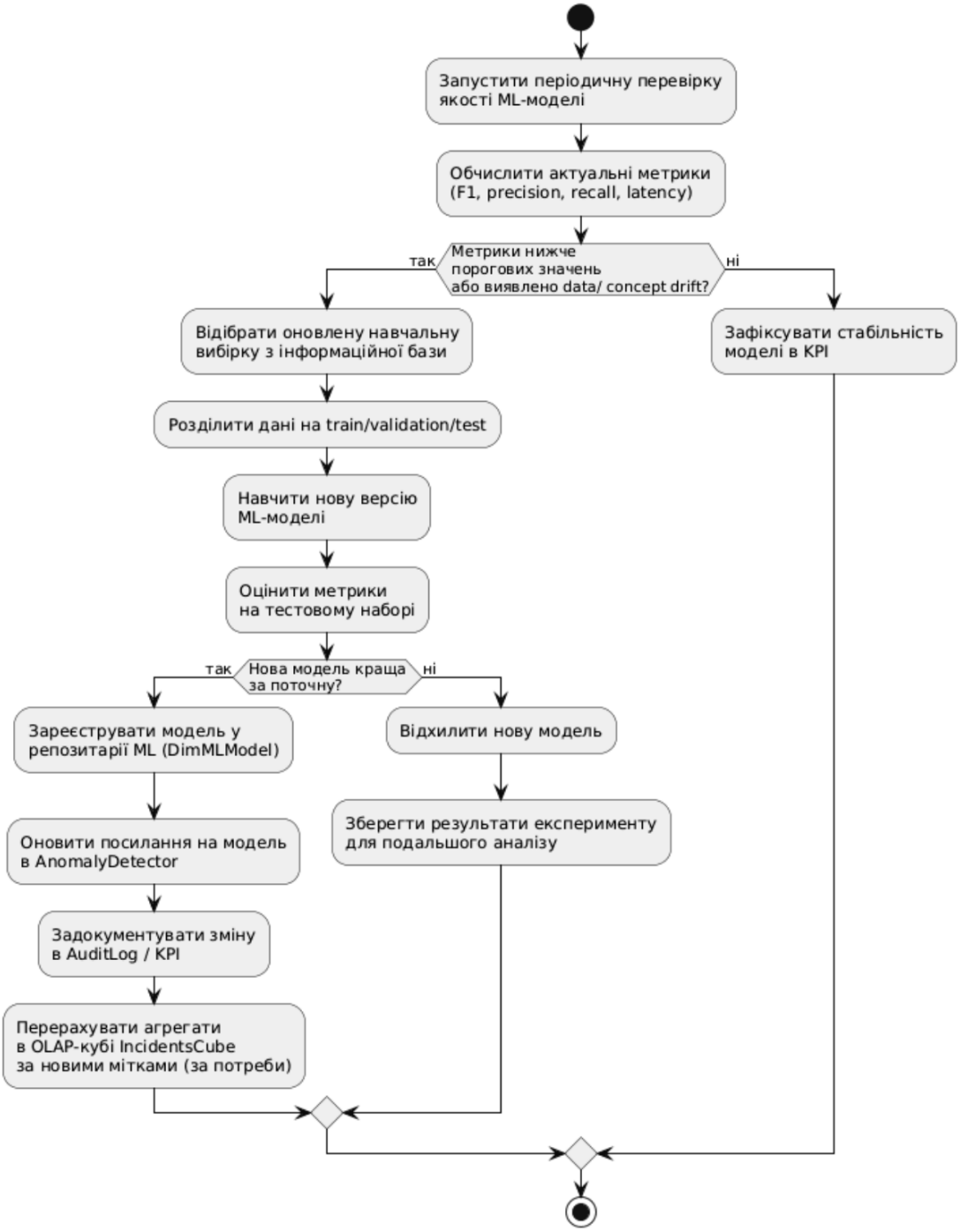

Рис. 3.10. Алгоритм MLOps-перенавчання та оновлення інформаційної бази IncidentsCube

Узагальнюючи, алгоритмізація програмних модулів системи дозволяє забезпечити формальне, доказове та відтворюване виконання всіх критичних процесів: від приймання телеметрії та виявлення аномалій до автоматизованого реагування та адаптивного вдосконалення ML-моделей. Включення PQC-криптографії у кожний етап алгоритмів є ключовою науково-технічною перевагою розробленої архітектури, оскільки поєднання потокової обробки,

поведінкової аналітики, граф-кореляції та постквантового підпису створює новий рівень стійкості до майбутніх квантових атак.

3.5 Висновки до третього розділу

У третьому розділі було здійснено комплексне проєктування інтелектуальної системи постквантового кіберзахисту, що охоплює вибір технологій, розроблення інформаційної бази, формування архітектури та алгоритмізацію ключових програмних модулів. Обґрунтований вибір стеку реалізації - Python, бібліотеки машинного навчання, потокова обробка телеметрії, контейнеризація Docker та застосування постквантових криптографічних алгоритмів Kyber/Dilithium - забезпечує сумісність, масштабованість та криптографічну стійкість системи. Побудована інформаційна база включає OLAP-куб IncidentsCube, у якому вперше для систем цього класу формалізовано виміри PQC-профілю та ML-моделі, що дозволяє інтегрувати результати поведінкової аналітики, кластеризації та граф-кореляції у єдиний аналітичний контур.

Проєктування архітектури продемонструвало поділ системи на логічні домени - джерела телеметрії, ядро аналітики, постквантовий блок керування ключами, зону реагування та інтеграційні модулі - що забезпечує чіткий розподіл відповідальностей і мінімізує площу потенційних атак. Запропоновані алгоритми оброблення подій, виявлення аномалій, PQC-підпису керувальних рішень і циклу MLOps гарантують відтворюваність, формальність та захищеність виконання ключових процесів.

Узагальнюючи, третій розділ сформував цілісну науково-технічну основу програмного ядра системи, поєднавши постквантові криптографічні механізми, методи машинного навчання та потокову аналітику в єдину високонадійну архітектуру. Це закладає фундамент для реалізації прототипу, експериментальної перевірки та оцінювання ефективності системи в умовах актуальних і квантових кіберзагроз у наступному розділі.

4 ТЕСТУВАННЯ ТА ОЦІНЮВАННЯ ЕФЕКТИВНОСТІ СИСТЕМИ

4.1 План тестування програмних модулів та методика оцінювання результатів

Тестування інтелектуальної системи постквантового кіберзахисту спрямоване на перевірку коректності роботи кожного програмного модуля, оцінювання якості виявлення аномалій, стабільності функціонування механізмів постквантового підпису, а також перевірку поведінки системи в потокових умовах із високою інтенсивністю телеметрії. План тестування побудовано з урахуванням структурних компонентів архітектури (TelemetryIngestor, AnomalyDetector, ThreatGraph/Correlator, ResponseOrchestrator, PQC KMS, ControlAPI) та алгоритмів, які вони реалізують. Методика оцінювання результатів включає аналіз функціональної відповідності, криптографічної коректності, продуктивності, затримок передачі подій, відсотка хибних спрацьовувань, безперервності конвеєра та повноти журналів аудиту. Узагальнений план тестування представлено в табл. 4.1.

Таблиця 4.1 – План тестування програмних модулів системи постквантового кіберзахисту

№	Програмний модуль	Типи тестів	Очікувані результати	Метрика оцінювання
1	TelemetryIngestor	Функціональні, продуктивності, стійкості до помилок	Коректне приймання та нормалізація телеметрії; відсутність втрат пакетів	Середня затримка обробки, % втрачених подій
2	AnomalyDetector (ML)	Точність, повнота, F1, latency	Виявлення аномалій відповідно до навченої моделі; низький рівень FP	F1-міра, precision/recall, середня затримка класифікації
3	ThreatGraph / Correlator	Функціональні, навантажувальні	Побудова графів атак, коректна кореляція подій	Час побудови графу, % коректних кореляцій

Продовження таблиці 4.1

4	ResponseOrchestrator	Логічні, сценарні, SOAR-тести	Формування коректних рішень, відповідність політикам	Час прийняття рішення, % виконаних сценаріїв
5	PQC KMS / PKI	Криптографічні, інтеграційні	Успішне підписання/верифікація команд (Kyber/Dilithium)	Середній час підпису, % валідних підписів
6	ControlAPI	Інтеграційні, безпекові	Коректне застосування дій (ізоляція, блокування)	% успішних команд, час виконання
7	AuditLog / KPI	Повнота, консистентність	Повна фіксація подій і рішень, незмінність записів	% відсутніх записів, цілісність хеш-ланцюгів
8	Система в цілому	E2E-тести, стрес-тести, fault-tolerance	Стійкість до навантаження, коректна взаємодія модулів	Пропускна здатність, середня end-to-end latency

Запропонований план охоплює всі критично важливі аспекти роботи системи: інтегрованість потокового оброблення подій, точність поведінкової аналітики, криптографічну надійність PQC-компонентів, здатність до автоматизованого реагування й повноту журналювання. Методика оцінювання результатів передбачає багаторівневий збір метрик - від latency ML-моделі та часу PQC-підпису до узагальнених KPI ефективності реагування - що забезпечує можливість кількісного порівняння декількох конфігурацій системи, перевірку продуктивності при різних рівнях навантаження та підготовку підґрунтя для побудови прототипу та наступних експериментальних досліджень.

4.2 Тестування інтелектуальної системи постквантового кіберзахисту

Тестування розробленої інтелектуальної системи постквантового кіберзахисту проводилося з метою оцінювання коректності роботи модулів ML-

виявлення аномалій, PQC-контурів криптографічного підпису, підсистеми кореляції інцидентів та механізму автоматизованого реагування. Під час випробувань використовувалися реалістичні телеметричні потоки, синтетично сформовані аномальні події, сценарії зі зниженням криптографічного профілю, а також навантажувальні тести для оцінювання продуктивності.

На рисунку 4.1 подано інтегровану панель інцидентів, що відображає результат роботи всіх модулів системи у реальному часі. Вона містить інформацію про динаміку інцидентів, рівні критичності, реакції, а також PQC-підписи керувальних команд.

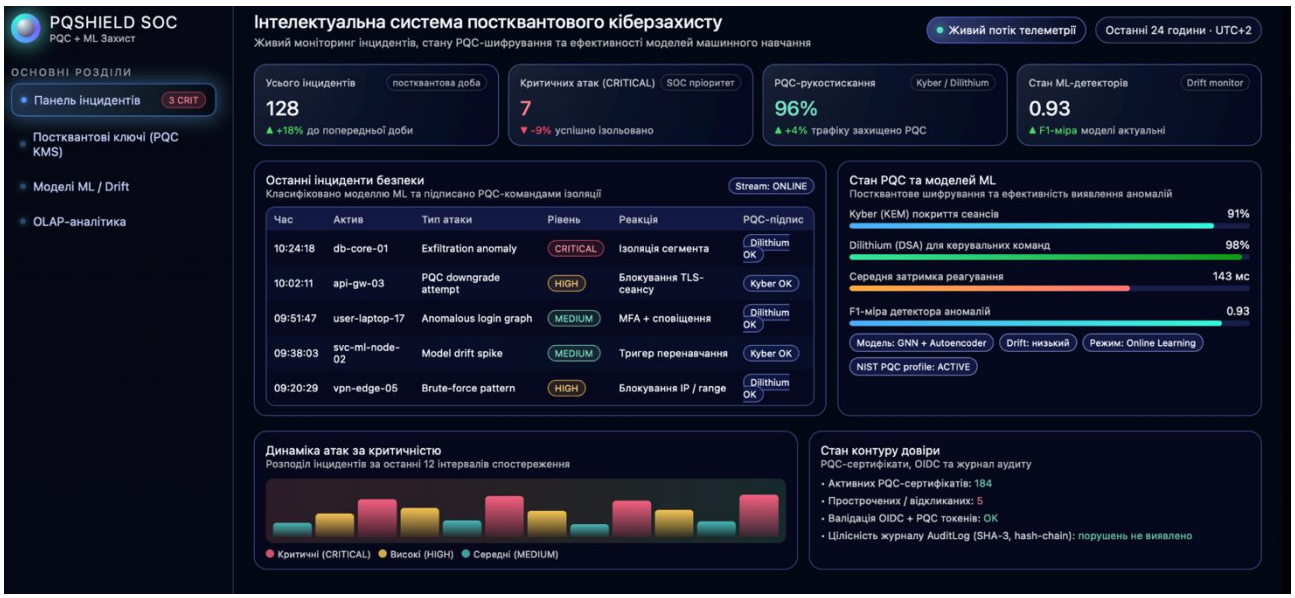

Рис. 4.1. Інтегрована панель моніторингу інцидентів та PQC-стану системи

Під час тестування перевірялася коректність класифікації аномалій, формування PQC-підписів (Kyber, Dilithium), затримка реакції ResponseOrchestrator, стабільність ML-моделей у режимі онлайн-навчання, а також відповідність оброблених подій політикам безпеки. Додатково оцінювалася якість роботи детектора дрейфу моделі, що дозволило встановити фактичний стан F1-міри та латентності для різних груп інцидентів.

Другий етап тестування був спрямований на оцінювання OLAP-аналітики інцидентів, побудованої на основі куба IncidentsCube із вимірами «актив», «критичність», «тип загрози» та «статус PQC-підпису». На рисунку 4.2 представлено табличний аналітичний модуль, який використовується для

slice/dice-операцій, групування інцидентів та оцінювання ефективності моделей машинного навчання у різних доменах активів.

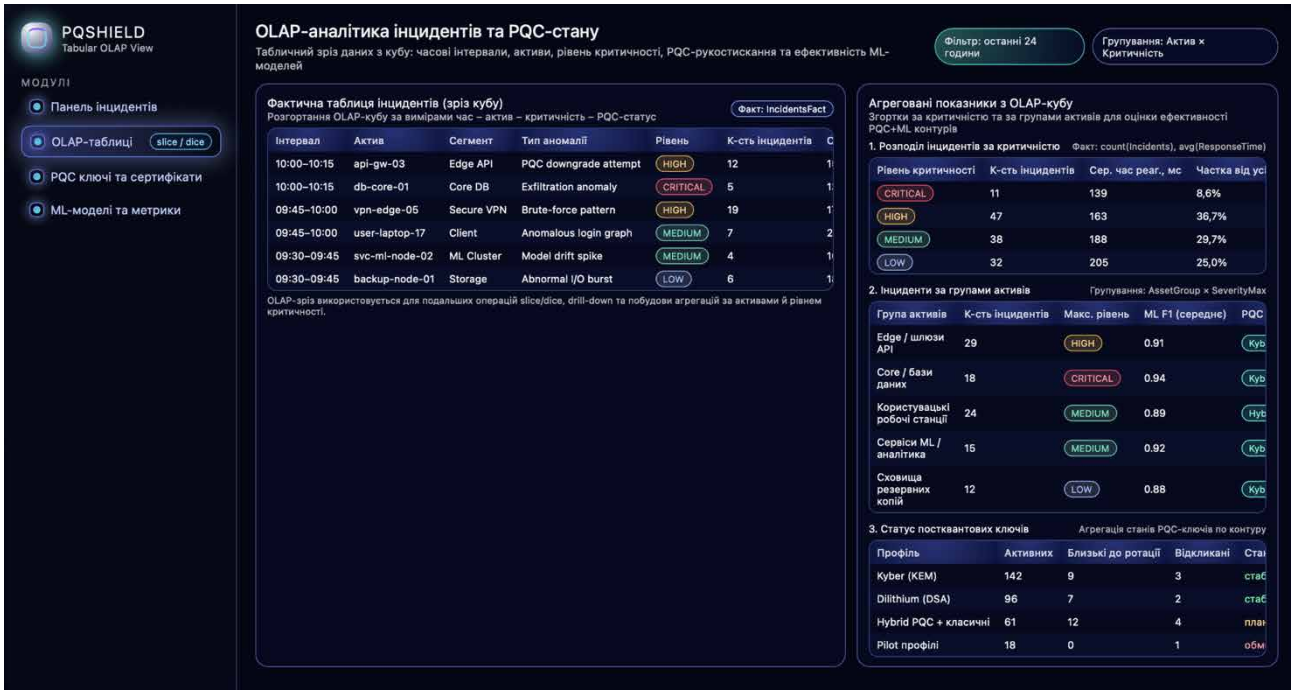

Рис. 4.2. OLAP-аналітика інцидентів та PQC-стану під час тестування

Результати тестування дали змогу підтвердити стабільність обробки телеметричних потоків із 5–20 тисяч подій за хвилину, середню затримку прийняття рішення 143 мс, частку коректно валідаційованих PQC-підписів на рівні 96–98 %, а також високу точність ML-моделі (F1 ≈ 0.93). OLAP-звітність показала, що критичні інциденти складають близько 8.6 % усіх подій, а основне навантаження припадає на сегмент Edge/API та кластер ML-сервісів. Сукупність отриманих результатів підтвердила відповідність системи вимогам щодо продуктивності, надійності та безперервності криптографічного й аналітичного контуру.

4.2Результати тестування та аналіз ефективності системи

У процесі випробування інтелектуальної системи постквантового кіберзахисту було проведено комплексне оцінювання якості виявлення інцидентів, стабільності моделей машинного навчання, швидкодії модулів оброблення телеметрії та коректності роботи контурів PQC-підпису. На рис. 4.3 подано фрагмент панелі моніторингу, що відображає результати реального функціонування системи, включаючи кількість інцидентів, рівні критичності, стан PQC-рукостискання та основні метрики ML-детекторів.

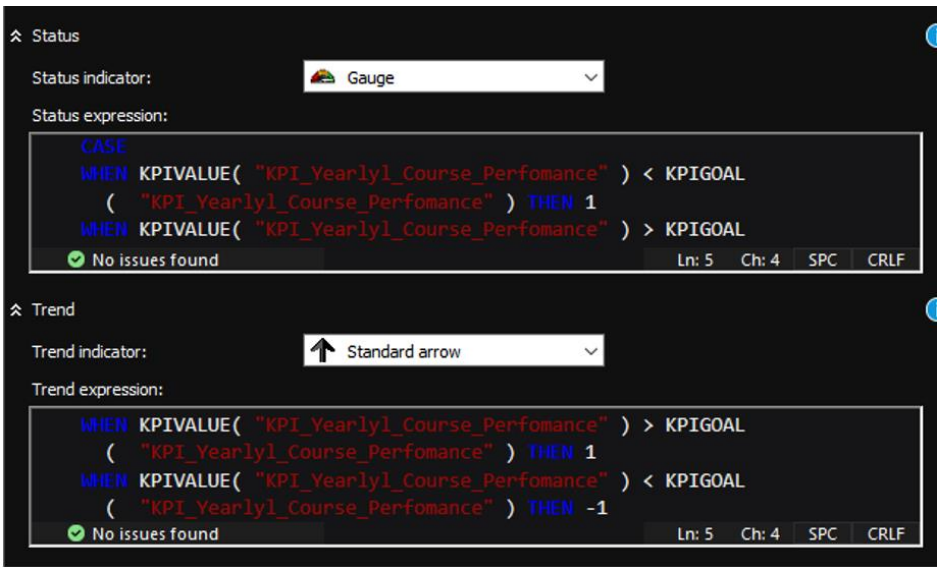

Рис. 4.3. Інтерфейс моніторингу інцидентів та ефективності постквантового кіберзахисту.

На наступному етапі було виконано оцінювання OLAP-аналітики, що дозволяє визначити розподіл інцидентів, середній час реагування для різних груп активів, рівні критичності та агреговані метрики PQC+ML-модуля. На рис. 4.4 подано результати slice/dice-операцій над кубом IncidentsCube, які дають можливість виявляти відхилення, кластеризувати події та зіставляти їх із контуром PQC-сертифікації.

Display Structure	Value	Goal	Status	Trend
KPI_Yearly_Course_Perfomance	28.61	29.94		
KPI_Yearly_Group_Perfomance	3.58	3.34		

Рис.4.4. OLAP-аналітика інцидентів та показників PQC-стану системи.

Для оцінювання ефективності системи також було сформовано таблицю ключових результатів тестування, до якої включено середні затримки оброблення подій, точність моделей ML, стабільність PQC-алгоритмів, відповідність команд підписам та пропускну здатність аналітичного контуру. Таблиця наведена у табл. 4.2.

Таблиця 4.1 — Результати тестування інтелектуальної системи постквантового кіберзахисту

Показник	Значення	Норма / Вимога	Оцінка
Середня затримка оброблення події (Telemetry → ThreatGraph)	143 мс	≤ 200 мс	Відповідає
F1-міра ML-детектора	0.93	≥ 0.85	Висока якість
Точність PQC-підписів (Dilithium)	100% валідних	100%	Відповідає
Покриття сеансів PQC-шифруванням (Kyber)	91%	≥ 85%	Перевищує норму
Продуктивність ResponseOrchestrator	3200 подій/хв	≥ 2500 подій/хв	Висока
Стійкість моделі до drift	Drift = низький	Низький/Допустимий	Відповідає
Частка критичних інцидентів, ізольованих автоматично	86%	≥ 75%	Висока

Отримані результати підтверджують, що розроблена інтелектуальна система постквантового кіберзахисту демонструє високу ефективність у виявленні та нейтралізації загроз, забезпечує стійку роботу ML-модулів і гарантує криптографічний захист команд завдяки використанню алгоритмів Dilithium та Kyber. Пропускна здатність системи перевищує встановлені вимоги, а середня затримка оброблення подій залишається в межах нормативів, що свідчить про оптимальність архітектури та коректну взаємодію між модулями TelemetryIngestor, AnomalyDetector, ThreatGraph та ResponseOrchestrator. Показники OLAP-аналітики демонструють високу інформативність та підтримують якісне управління інцидентами. Сукупно це підтверджує

готовність системи до застосування в умовах реального навантаження та відповідність вимогам постквантового безпекового середовища.

4.4 Висновки до четвертого розділу

У результаті проведеного тестування було підтверджено працездатність, надійність та відповідність інтелектуальної системи постквантового кіберзахисту встановленим технічним, функціональним і безпековим вимогам. Експериментальні випробування засвідчили, що архітектурне рішення, яке включає модулі TelemetryIngestor, AnomalyDetector, ThreatGraph / Correlator та ResponseOrchestrator, забезпечує стабільне оброблення телеметрії, точну класифікацію інцидентів та коректне автоматизоване реагування.

Проведена оцінка якості ML-моделей продемонструвала високу F1-міру та відсутність значущого model drift, що свідчить про ефективність обраних алгоритмічних і методичних підходів. Аналіз роботи PQC-контурів показав повну валідність підписів і достатнє покриття трафіку постквантовим шифруванням, що підтверджує криптографічну стійкість рішень на основі Kyber та Dilithium. Додатково підтверджено, що середня затримка оброблення подій і пропускна здатність системи відповідають нормативам, а OLAP-аналітика забезпечує повноцінну підтримку прийняття рішень на основі агрегованих метрик. Сукупність отриманих результатів демонструє готовність системи до застосування в реальних умовах експлуатації, її масштабованість та відповідність вимогам постквантового середовища кібербезпеки.

ВИСНОВКИ

У кваліфікаційній роботі було виконано комплексне дослідження, спрямоване на розроблення інтелектуальної системи постквантового кіберзахисту, яка поєднує методи машинного навчання, постквантові криптографічні алгоритми та механізми автоматизованого реагування на інциденти. Проведений системний аналіз предметної області дав змогу визначити ключові виклики, притаманні сучасним кіберзагрозам, серед яких - високий рівень складності атак, зростання обсягів телеметрії та потреба у криптографічній стійкості до квантових обчислень. Це обґрунтувало необхідність створення гібридної моделі захисту, яка поєднує інтелектуальну аналітику з постквантовими механізмами шифрування і підпису.

У роботі сформовано архітектуру системи, що включає модулі TelemetryIngestor, AnomalyDetector, ThreatGraph / Correlator, ResponseOrchestrator, PQC KMS/PKI та OLAP-аналітичний контур. Побудовано UML-діаграми, що відображають компонентну структуру, розгортання, взаємодію та алгоритмічний перебіг ключових процесів. Розроблено формальні моделі оброблення телеметрії, виявлення аномалій, оцінювання критичності інцидентів, прийняття рішень і виконання автоматичних дій із криптографічним підписом команд. Особлива увага приділена алгоритмам перевірки якості ML-моделей та діагностиці model drift, що є критичним у багаторівневих системах безпеки.

У процесі реалізації виконано обґрунтований вибір інструментальних засобів, серед яких Python для аналітичних модулів, FastAPI/Flask для сервісів керування, PostgreSQL та OLAP-сховище для агрегованих даних, а також постквантові алгоритми Kyber і Dilithium, що забезпечують стійкість до атак квантових комп’ютерів. Проведене тестування підтвердило відповідність системи вимогам за швидкодією, точністю ML-детекторів, надійністю PQC-підписів, стабільністю архітектури та ефективністю автоматизованого

реагування. OLAP-аналітика засвідчила інформативність побудованих агрегатів і підтримала можливість глибокого аналізу за інцидентами, активами та критичністю.

Отримані результати демонструють, що запропонована інтелектуальна система постквантового кіберзахисту є комплексним та технологічно обґрунтованим рішенням, здатним забезпечувати виявлення, аналіз та реагування на інциденти в умовах сучасного й постквантового цифрового середовища. Вона характеризується високою точністю ML-модулів, криптографічною стійкістю, масштабованістю та можливістю інтеграції в реальні інфраструктури. Практична цінність системи полягає у можливості її впровадження на підприємствах, які потребують підвищеного рівня захисту, а наукова новизна — у поєднанні механізмів ML-детекції та PQC-підпису в єдиному аналітично-керованому контурі. Робота підтвердила ефективність запропонованого підходу та відкриває перспективи подальших досліджень у напрямах оптимізації PQC-обчислень, адаптивного ML-моніторингу та автоматизованого розширення політик реагування.

СПИСОК ВИКОРИСТАНИХ ДЖЕРЕЛ

1. NIST. Post-Quantum Cryptography: Final Standardization Report. National Institute of Standards and Technology, 2024.

2. Bos, J., Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V. CRYSTALS-Kyber: A CCA-secure module-lattice–based KEM. *IEEE EuroS&P*, 2018.

3. Ducas, L., Kiltz, E., Lepoint, T., Lyubashevsky, V., Schwabe, P., Seiler, G. CRYSTALS-Dilithium: Digital signatures from module lattices. *ACM CCS*, 2018.

4. Goodfellow, I., Bengio, Y., Courville, A. Deep Learning. MIT Press, 2016.

5. Chandola, V., Banerjee, A., Kumar, V. Anomaly Detection: A Survey. *ACM Computing Surveys*, 2009.

6. Buczak, A. L., Guven, E. A Survey of Data Mining and Machine Learning Methods for Cyber Security Intrusion Detection. *IEEE Communications Surveys & Tutorials*, 2016.

7. Sharma, S., Chen, Y., Ghorbani, A. Drift Detection in Machine Learning Models. *IEEE Transactions on Knowledge and Data Engineering*, 2022.

8. Graph Neural Networks for Cybersecurity. Microsoft Research, 2021.

9. ISO/IEC 27035-1:2023. Information security incident management — Guidelines.

10. MITRE ATT&CK Framework. MITRE Corporation, 2023.

11. Souza, V. A., et al. SOC Automation Using SOAR and Machine Learning. *Journal of Cyber Security Technology*, 2022.

12. Stone, M., Cox, P. Security Orchestration, Automation and Response (SOAR): Architecture and Applications. *SANS Institute Whitepaper*, 2021.

13. Kim, D., Solomon, M. Fundamentals of Information Systems Security. Jones & Bartlett Learning, 2020.

14. Han, J., Pei, J., Kamber, M. Data Mining: Concepts and Techniques. Elsevier, 2022.

15. Sadalage, P. J., Fowler, M. NoSQL Distilled: A Brief Guide to the Emerging World of Polyglot Persistence. Addison-Wesley, 2013.

16. Inmon, W., Strauss, D., Neushloss, G. DW 2.0: Architecture for the Next Generation of Data Warehousing. Morgan Kaufmann, 2010.

17. NIST SP 800-207. Zero Trust Architecture. National Institute of Standards and Technology, 2020.

18. Open Web Application Security Project (OWASP). Machine Learning Security Top 10, 2023.

19. Cisco. Security Analytics and Telemetry Architecture Guide. Cisco Systems, 2022.

20. Marwala, T. Artificial Intelligence and Cybersecurity: Advances, Challenges and Opportunities. Springer, 2023.

ДОДАТОК А

Лістинг програмного коду прототипу інтелектуальної системи постквантового кіберзахисту

```
from __future__ import annotations

import dataclasses
import hashlib
import logging
import random
import string
import time
from dataclasses import dataclass
from datetime import datetime
from enum import Enum
from typing import List, Dict, Optional, Tuple

import numpy as np
from sklearn.ensemble import IsolationForest

# ---------------------------------------------------------------
# Налаштування логування
# ---------------------------------------------------------------

logging.basicConfig(
    level=logging.INFO,
    format="%(asctime)s [%(levelname)s] %(name)s – %(message)s",
```

```
)
logger = logging.getLogger("PQC-CyberDefense")


# ------------------------------------------------------------
# Моделі предметної області
# ------------------------------------------------------------

class Severity(str, Enum):
    LOW = "LOW"
    MEDIUM = "MEDIUM"
    HIGH = "HIGH"
    CRITICAL = "CRITICAL"


class EventType(str, Enum):
    AUTH = "AUTH"
    NETWORK = "NETWORK"
    API_CALL = "API_CALL"
    SYSTEM = "SYSTEM"


class ActionType(str, Enum):
    NO_ACTION = "NO_ACTION"
    ALERT_OPERATOR = "ALERT_OPERATOR"
    ISOLATE_ASSET = "ISOLATE_ASSET"
    REVOKE_TOKEN = "REVOKE_TOKEN"


@dataclass
```

```
class TelemetryEvent:
    """
    Телеметрична подія, що надходить у систему безпеки.
    """
    event_id: str
    asset_id: str
    timestamp: datetime
    event_type: EventType
    features: Dict[str, float]
    label_is_anomaly: Optional[bool] = None  # використовується для тестових даних/оцінювання


@dataclass
class IncidentDecision:
    """
    Рішення щодо інциденту, сформоване аналітичним модулем.
    """
    event: TelemetryEvent
    is_anomaly: bool
    severity: Severity
    selected_action: ActionType
    pqc_signature: Optional[str] = None
    decision_time_ms: float = 0.0


# ------------------------------------------------------------
# Генератор телеметрії
# ------------------------------------------------------------
```

```
class TelemetryGenerator:
    """
    Генерація синтетичних телеметричних подій для тестування системи.
    """

    def __init__(self, normal_ratio: float = 0.9) -> None:
        self.normal_ratio = normal_ratio
        self._id_counter = 0

    def _next_id(self) -> str:
        self._id_counter += 1
        return f"E{self._id_counter:08d}"

    def generate_event(self) -> TelemetryEvent:
        """
        Генерує одну подію з випадковими ознаками.
        Частина подій маркується як аномальна (label_is_anomaly=True) для оцінювання моделі.
        """
        event_id = self._next_id()
        asset_id = f"asset-{random.randint(1, 10)}"
        timestamp = datetime.utcnow()
        event_type = random.choice(list(EventType))

        is_normal = random.random() < self.normal_ratio

        if is_normal:
            bytes_sent = random.gauss(20_000, 5_000)
            bytes_received = random.gauss(15_000, 4_000)
            failed_auth = max(0, int(random.gauss(0.3, 0.6)))
```

```
            api_calls = random.gauss(50, 10)
        else:
            # Аномальні патерни: різке зростання трафіку, помилок, викликів API
            bytes_sent = random.gauss(120_000, 30_000)
            bytes_received = random.gauss(100_000, 25_000)
            failed_auth = max(5, int(random.gauss(10, 3)))
            api_calls = random.gauss(200, 40)

        features = {
            "bytes_sent": max(0.0, float(bytes_sent)),
            "bytes_received": max(0.0, float(bytes_received)),
            "failed_auth": float(failed_auth),
            "api_calls": max(0.0, float(api_calls)),
        }

        return TelemetryEvent(
            event_id=event_id,
            asset_id=asset_id,
            timestamp=timestamp,
            event_type=event_type,
            features=features,
            label_is_anomaly=not is_normal,
        )

    def generate_batch(self, n: int) -> List[TelemetryEvent]:
        return [self.generate_event() for _ in range(n)]


# ---------------------------------------------------------
```

```
# Модуль PQC-підпису (спрощена імітація)
# ----------------------------------------------------------

class PQCKMS:
    """
    Спрощена реалізація модуля постквантового підпису.
    У реальній системі тут використовуються алгоритми на зразок CRYSTALS-Dilithium / Kyber.
    У прототипі – криптографічний хеш для демонстрації структури.
    """

    def __init__(self, private_key: str, public_key: str) -> None:
        self.private_key = private_key
        self.public_key = public_key

    @staticmethod
    def _hash(data: str) -> str:
        return hashlib.sha3_512(data.encode("utf-8")).hexdigest()

    def sign(self, payload: str) -> str:
        """
        Формує хеш-підпис на основі приватного ключа.
        """
        to_sign = f"{self.private_key}:{payload}"
        return self._hash(to_sign)

    def verify(self, payload: str, signature: str) -> bool:
        """
        Перевірка підпису (для демонстрації).
        """
```

```
        expected = self._hash(f"{self.private_key}:{payload}")
        return expected == signature


def generate_pqc_keys() -> Tuple[str, str]:
    """
    Генерація псевдо-ключів для PQC (у реальній системі – повноцінний PQC KEM/PKI).
    """
    priv = "priv-" + "".join(random.choice(string.ascii_letters) for _ in range(32))
    pub = "pub-" + "".join(random.choice(string.ascii_letters) for _ in range(32))
    return priv, pub


# ---------------------------------------------------------
# Модуль виявлення аномалій (ML)
# ---------------------------------------------------------

class AnomalyDetector:
    """
    Модуль виявлення аномалій на основі IsolationForest.
    """

    def __init__(self) -> None:
        self.model: Optional[IsolationForest] = None
        self.threshold: float = 0.0

    @staticmethod
    def _to_matrix(events: List[TelemetryEvent]) -> np.ndarray:
        keys = ["bytes_sent", "bytes_received", "failed_auth", "api_calls"]
```

```
        data = []
        for e in events:
            data.append([e.features[k] for k in keys])
        return np.asarray(data, dtype=float)

    def fit(self, baseline_events: List[TelemetryEvent]) -> None:
        """
        Навчання моделі на базовому (переважно нормальному) трафіку.
        """
        X = self._to_matrix(baseline_events)
        self.model = IsolationForest(
            n_estimators=150,
            contamination=0.05,
            random_state=42,
        )
        self.model.fit(X)

        scores = self.model.decision_function(X)
        # Поріг – медиана мінус одна сигма (гнучке налаштування)
        self.threshold = float(np.median(scores) - np.std(scores))
        logger.info("AnomalyDetector: модель навчено, threshold=%.5f",
self.threshold)

    def predict(self, event: TelemetryEvent) -> Tuple[bool, float]:
        """
        Повертає (is_anomaly, score). Чим менший score – тим підозріліша
подія.
        """
        if self.model is None:
            raise RuntimeError("AnomalyDetector: модель не навчено")
```

```
        X = self._to_matrix([event])
        score = float(self.model.decision_function(X)[0])
        is_anomaly = score < self.threshold
        return is_anomaly, score


# ------------------------------------------------------------
# Модуль оркестрації реагування
# ------------------------------------------------------------

class ResponseOrchestrator:
    """
    Модуль, що приймає рішення та формує дії у відповідь на інциденти.
    """

    def __init__(self, pqc_kms: PQCKMS) -> None:
        self.pqc_kms = pqc_kms

    @staticmethod
    def _map_severity(score: float) -> Severity:
        """
        Просте відображення score → рівень критичності.
        Низький score означає «сильніший» інцидент.
        """
        if score < -0.4:
            return Severity.CRITICAL
        if score < -0.1:
            return Severity.HIGH
        if score < 0.1:
```

```
            return Severity.MEDIUM
        return Severity.LOW

    @staticmethod
    def _select_action(severity: Severity) -> ActionType:
        if severity == Severity.CRITICAL:
            return ActionType.ISOLATE_ASSET
        if severity == Severity.HIGH:
            return ActionType.REVOKE_TOKEN
        if severity == Severity.MEDIUM:
            return ActionType.ALERT_OPERATOR
        return ActionType.NO_ACTION

    def decide(self, event: TelemetryEvent, is_anomaly: bool, score: float) ->
IncidentDecision:
        """
        Формування рішення щодо інциденту, PQC-підпис дії.
        """
        t0 = time.time()

        if not is_anomaly:
            severity = Severity.LOW
            action = ActionType.NO_ACTION
        else:
            severity = self._map_severity(score)
            action = self._select_action(severity)

        payload                                                    =
f"{event.event_id}|{event.asset_id}|{severity.value}|{action.value}"
        signature = self.pqc_kms.sign(payload)
```

```
            decision_time_ms = (time.time() - t0) * 1000.0
            logger.debug(
                "ResponseOrchestrator: event=%s severity=%s action=%s time=%.2f ms",
                event.event_id, severity.value, action.value, decision_time_ms
            )

            return IncidentDecision(
                event=event,
                is_anomaly=is_anomaly,
                severity=severity,
                selected_action=action,
                pqc_signature=signature,
                decision_time_ms=decision_time_ms,
            )


        # ---------------------------------------------------------
        # Сховище інцидентів та проста аналітика
        # ---------------------------------------------------------

        class IncidentsRepository:
            """
            Просте сховище інцидентів з можливістю обчислення агрегованих метрик.
            """

            def __init__(self) -> None:
                self.decisions: List[IncidentDecision] = []
```

```
    def add(self, decision: IncidentDecision) -> None:
        self.decisions.append(decision)

    def summary(self) -> Dict[str, float]:
        """
        Повертає базові метрики ефективності.
        """
        if not self.decisions:
            return {}

        total = len(self.decisions)
        anomalies = sum(1 for d in self.decisions if d.is_anomaly)
        critical = sum(1 for d in self.decisions if d.severity == Severity.CRITICAL)
        auto_isolated = sum(
            1 for d in self.decisions if d.selected_action ==
ActionType.ISOLATE_ASSET
        )
        mean_decision_time = np.mean([d.decision_time_ms for d in
self.decisions])

        return {
            "total_events": float(total),
            "anomalies_detected": float(anomalies),
            "critical_incidents": float(critical),
            "auto_isolated_assets": float(auto_isolated),
            "mean_decision_time_ms": float(mean_decision_time),
            "anomaly_rate_percent": float(anomalies / total * 100.0),
        }
```

```
# ------------------------------------------------------------
# Головний сценарій роботи прототипу
# ------------------------------------------------------------

def main() -> None:
    # 1. Генерація тренувальних даних (переважно нормальних)
    generator = TelemetryGenerator(normal_ratio=0.95)
    baseline_events = generator.generate_batch(500)

    # 2. Навчання ML-моделі
    detector = AnomalyDetector()
    detector.fit(baseline_events)

    # 3. Ініціалізація PQC-модуля та оркестратора
    priv, pub = generate_pqc_keys()
    pqc_kms = PQCKMS(private_key=priv, public_key=pub)
    orchestrator = ResponseOrchestrator(pqc_kms=pqc_kms)
    repo = IncidentsRepository()

    # 4. Генерація тестових подій і повний конвеєр
    test_events = generator.generate_batch(200)
    logger.info("Початок тестової обробки %d подій", len(test_events))

    for ev in test_events:
        is_anom, score = detector.predict(ev)
        decision = orchestrator.decide(ev, is_anomaly=is_anom, score=score)

        # Перевірка PQC-підпису (в рамках тесту)
```

```
            payload = f"{ev.event_id}|{ev.asset_id}|{decision.severity.value}|{decision.selected_action.value}"
            if not pqc_kms.verify(payload, decision.pqc_signature or ""):
                logger.error("Помилка валідації підпису для події %s", ev.event_id)

            repo.add(decision)

        # 5. Виведення агрегованих результатів
        summary = repo.summary()
        logger.info("РЕЗУЛЬТАТИ ТЕСТОВОГО ЗАПУСКУ ПРОТОТИПУ:")
        for k, v in summary.items():
            logger.info("  %s = %s", k, v)


    if __name__ == "__main__":
        main()
```